

Сигнальные микропроцессоры и их применение в системах телекоммуникации и электроники

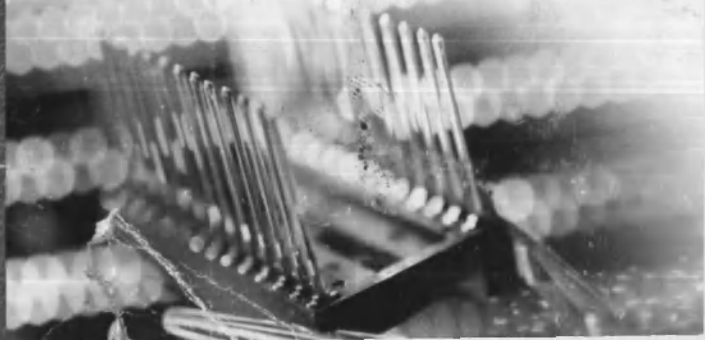


СОВРЕМЕННАЯ  
ЭЛЕКТРОНИКА



*В. С. Сперанский*

# Сигнальные микропроцессоры и их применение в системах телекоммуникаций и электроники

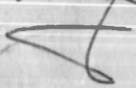


10508 6244

14.09 560



191003 7925



100210 8686



**В. С. Сперанский**

# **СИГНАЛЬНЫЕ МИКРОПРОЦЕССОРЫ И ИХ ПРИМЕНЕНИЕ В СИСТЕМАХ ТЕЛЕКОММУНИКАЦИЙ И ЭЛЕКТРОНИКИ**

*Рекомендовано УМО по образованию  
в области телекоммуникаций в качестве учебного пособия  
для студентов высших учебных заведений, обучающихся  
по специальностям: «Радиосвязь, радиовещание и телевидение»;  
«Средства связи с подвижными объектами»;  
«Многоканальные телекоммуникационные системы»*

Б

**Библиотека  
МТУСИ**

Москва  
Горячая линия – Телеком  
2008

УДК 681.325.5

ББК 32.84

С71

**Сперанский В. С.**

**С71** Сигнальные микропроцессоры и их применение в системах телекоммуникаций и электроники. Учебное пособие для вузов. – М.: Горячая линия – Телеком, 2008. – 168 с.: ил.

**ISBN 978-5-9912-0035-6.**

Рассмотрены необходимые теоретические вопросы и практика использования цифровых микропроцессоров обработки сигналов. Кратко рассмотрены основные операции цифровой обработки сигналов, архитектура, классификация и характеристики современных сигнальных процессоров. Существенное внимание уделено процессорам фирмы Analog Devices, приведены алгоритмы реализации типовых устройств. Описаны процессоры фирм Motorola, Texas Instruments, векторный процессор НТЦ Модуль и микроконтроллер Мультикор (НПЦ Элвис). Отдельный раздел посвящен программируемым логическим интегральным схемам и системе на кристалле. В приложениях содержатся программы реализации нерекурсивного фильтра, формирователя гармонического колебания, быстрого преобразования Фурье, а также описания отладочной платы процессора ADSP2181 и его симулятора.

Для студентов, обучающихся по специальностям «Радиосвязь, радиовещание и телевидение», «Средства связи с подвижными объектами», «Многоканальные телекоммуникационные системы», будет полезна специалистам.

**ББК 32.84**

*Адрес издательства в Интернет WWW.TECHBOOK.RU*

Учебное издание

**Сперанский Валентин Сергеевич**

**Сигнальные микропроцессоры и их применение  
в системах телекоммуникаций и электроники**

Учебное пособие для вузов

Обложка художника В. Г. Ситникова

Подписано в печать 22.12.07. Формат 60×90/16. Гарнитура Times. Усл. печ. л. 9.

ООО «Научно-техническое издательство «Горячая линия–Телеком»

Отпечатано в ООО «Типография Полимаг» 127242. Москва, Дмитровское шоссе, 107. Заказ № 332

**ISBN 978-5-9912-0035-6**

© Сперанский В. С., 2008

© Оформление издательства  
«Горячая линия–Телеком», 2008

## Предисловие

Сигнальные процессоры представляют собой современную базу и имеют различные области применения в первую очередь в телекоммуникациях, в том числе в системах подвижной связи, системах радиодоступа, радиовещания, телевидения; в аудио и видеотехнике, в модемах по проводным и физическим линиям, в радиолокации и радионавигации, в медицине, научных исследованиях и в промышленности.

Основная цель данного учебного пособия – помочь студентам и инженерам изучить теорию и технику цифровых сигнальных процессоров, освоить основы программирования на примерах реализации типовых устройств. Для освоения материала необходимы знания теории цифровой обработки сигналов (ЦОС) и радиотехники. Поэтому в первой главе рассмотрены основные операции ЦОС: аналого-цифровое преобразование сигналов, дискретное преобразование Фурье, Z-преобразование, характеристики цифровых фильтров и форматы представления чисел при обработке. Во второй главе содержатся классификация сигнальных процессоров, особенности гарвардской архитектуры, типовая схема цифрового сигнального микропроцессора и основные характеристики.

В третьей главе описан базовый процессор ADSP2181 фирмы Analog Devices, его структура, характеристики, применение. Далее подробно рассмотрены вычислительные блоки, формирователи адресов, устройство управления программой, периферия и система команд. Отдельная глава содержит реализации на указанном процессоре типовых устройств: фильтров, формирователей гармонического колебания и вычислителя быстрого преобразования Фурье. Программы на языке Макроассемблера ADSP2181 приведены в приложениях. Столь большое внимание к процессору объясняется тем, что данный ЦСП методически один из наиболее подходящих для изучения и освоения. Кратко описано семейство процессоров с фиксированной точкой нового поколения ADSP2153\* Blackfin.

В следующей главе представлены высокопроизводительные процессоры с плавающей точкой типа SHARC. Отдельный раздел пособия посвящен процессорам и микроконтроллерам фирмы «Мо-

torola», их классификации, особенностям и характеристикам. В частности, рассмотрен процессор известный DSP 56300, его ядро и его система команд. Описаны структура и назначение микроконтроллеров «Motorola», и применение в системах телекоммуникаций.

Процессоры разных поколений фирмы Texas Instruments описаны в следующем разделе. Особое внимание уделено платформам C200, C5000 и C6000, применяющимся в системах телекоммуникаций. Большое внимание уделено процессорам C54 и C55, предназначенных для подвижной связи и систем радиодоступа. Представлены высокопроизводительные процессоры платформы C6000 с параллельными вычислениями и «длинной» командой. Особое место занимает мультимедиа-видео ЦСП C80, предназначенный для обработки изображений и трехмерной графики.

Далее описаны микроконтроллеры Мультикор фирмы «Элвис» и процессор с векторным сопроцессором фирмы «Модуль» (Москва), разработанные российскими инженерами. Сигнальный процессор Мультикор представляет собой двухпроцессорную систему, содержащую центральный процессор и цифровой сигнальный процессор, решающую одновременно задачи управления и цифровой обработки. На процессоре фирмы «Модуль» реализованы ряд устройств, в частности, рассмотрена обработка в нейросети. Заключительный раздел содержит классификацию, характеристики, порядок проектирования и примеры применения программируемых логических интегральных схем (ПЛИС): реализация вычислителя БПФ и нерекурсивных фильтров.

Также рассмотрены основные направления развития цифровых сигнальных процессоров «система на кристалле» и перспективные архитектуры ЦОС, реализуемые на них.

При написании пособия использован опыт преподавания данной дисциплины в МТУСИ в течение ряда лет.

Автор выражает признательность за труд по ознакомлению с рукописью и ценные замечания рецензентам проф. Шемаханову М.М., проф. Витязеву В.В., инж. Жвания А.Г., благодарит директора фирмы «АУТЕХ» Кирпичникова А.П. за поддержку и помощь в организации лаборатории цифровой обработки сигналов на кафедре радиотехнических систем МТУСИ, а также доц. Косичкину Т.П. за помощь в подготовке рукописи и обсуждении материалов. Пособие может быть полезно студентам радиотехнических специальностей, радиоинженерам и преподавателям.

## **Глава 1. Основы цифровой обработки сигналов**

В настоящее время широкое применение находят цифровые системы связи и электроники, в которых обработка сигналов осуществляется на основе цифровых вычислительных устройств. Цифровые системы обладают рядом преимуществ перед аналоговыми системами [2, 3]. В частности следует выделить высокую помехозащищенность, высокую точность и стабильность характеристик, высокую степень микроминиатюризации аппаратуры. С помощью цифровых устройств можно реализовать практически любые алгоритмы, в том числе те, которые сложно реализовать с помощью аналоговых устройств.

К примеру, помехоустойчивое кодирование, сжатие речи, изображения и видео, алгоритмы шифрования могут быть реализованы только с помощью цифровой обработки сигналов. До того, как перейти к изучению сигнальных микропроцессоров, автор посчитал необходимым привести в книге основные сведения, касающиеся цифровой обработки сигналов. Изложение данного материала, в силу разных обстоятельств, носит конспективный, скорее справочный характер.

### **1.1. Основные операции ЦОС**

Как правило, на вход устройства цифровой обработки (УЦО) поступают аналоговые сигналы, поэтому необходимо произвести аналого-цифровое преобразование (АЦП) и после обработки, если необходимо, обратное – цифро-аналоговое преобразование (ЦАП). На рис. 1.1 показана обобщенная структурная схема устройства, в котором обработка сигналов производится цифровыми методами.

Аналого-цифровое преобразование сигнала происходит в три этапа (рис. 1.2): временная дискретизация сигнала, квантование амплитуды и преобразование выборок в многоуровневые (в частном случае – одноуровневые) двоичные коды.

Фильтр нижних частот необходим ввиду того, что реальные сигналы имеют бесконечный спектр. Во избежание наложения спектров при дискретизации входной сигнал ограничивают по полосе.

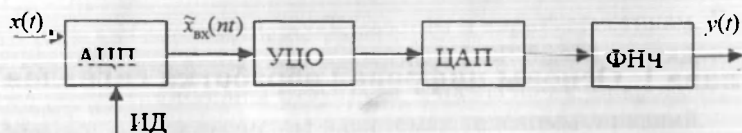


Рис. 1.1. Структурная схема устройства с ЦОС:

ИД – импульсы дискретизации; АЦП – аналого-цифровой преобразователь; УЦО – устройство цифровой обработки сигналов; ЦАП – цифро-аналоговый преобразователь; ФНЧ – фильтр нижних частот, выделяющий спектр сигнала

Согласно теореме о выборках, сигнал с ограниченной полосой, не имеющий спектральных компонент с частотами выше  $f_b$ , однозначно определяется отсчетами, следующими через равные промежутки времени:

$$T_d \leq \frac{1}{2f_b} = \frac{\pi}{\omega_b}, \quad (1.1)$$

таким образом,  $f_d = 1/T_d$  – частота дискретизации.

Увеличение частоты дискретизации упрощает реализацию входных и выходных фильтров.

Вид спектра аналогового и соответствующего ему дискретного сигналов показан на рис.1.3,а,б. Спектр дискретного сигнала периодический, он повторяется в точках, кратных  $f_d$ .

Если условие (1.1) не выполняется, то спектры перекрываются и возникают искажения сигнала рис 1.3,в. Полностью восстановить аналоговый сигнал можно, используя ряд В.А. Котельникова [1].

$$S(t) = \sum_{n=-\infty}^{\infty} S(nT_d) \frac{\sin \pi \left[ \frac{t - kT_d}{T_d} \right]}{\pi \left[ \frac{t - kT_d}{T_d} \right]}, \quad (1.2)$$

где  $T_d$  – период дискретизации.

Операция квантования заключается в замене всех возможных значений сигнала набором разрешенных значений – уровней квантования. Операция квантования, в отличие от дискретизации, приводит к появлению искажений.



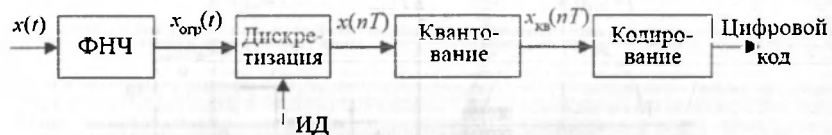


Рис. 1.2. Аналого-цифровое преобразование

Процесс квантования можно представить как прохождение сигнала через устройство со ступенчатой амплитудной характеристикой (рис. 1.4), в результате к сигналу добавляется шум квантования. Шкалы квантования могут быть равномерные и неравномерные, отличаться по положению относительно начала координат. На рисунке  $\Delta$  – шаг квантования,  $U_{\text{вых}}$  – выходной цифровой сигнал.

Кодирование – закон, устанавливающий соответствие между номером уровня квантования и структурой кодовой группы. Наиболее простым методом кодирования является импульсно-кодовая модуляция – каждый уровень кодируется двоичным кодом (подробнее о форматах представления чисел см. разд. 1.2.)

### Преобразования Фурье, Лапласа, Z-преобразование

Преобразование Фурье – непрерывного сигнала:

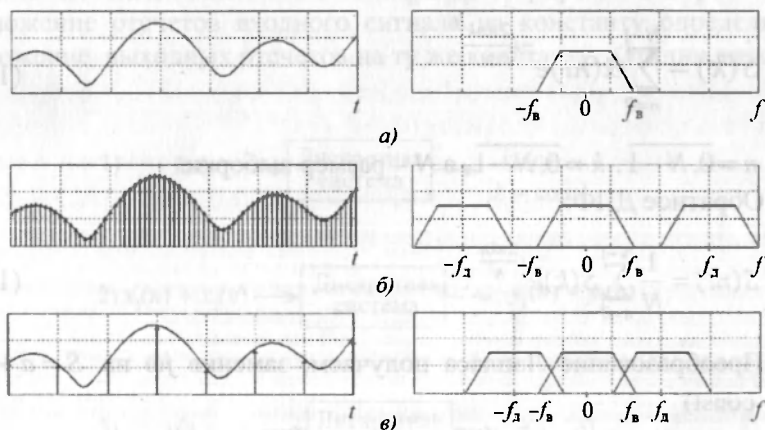


Рис. 1.3. Процесс дискретизации. На левых рисунках сигналы представлены во временной области, справа – в частотной

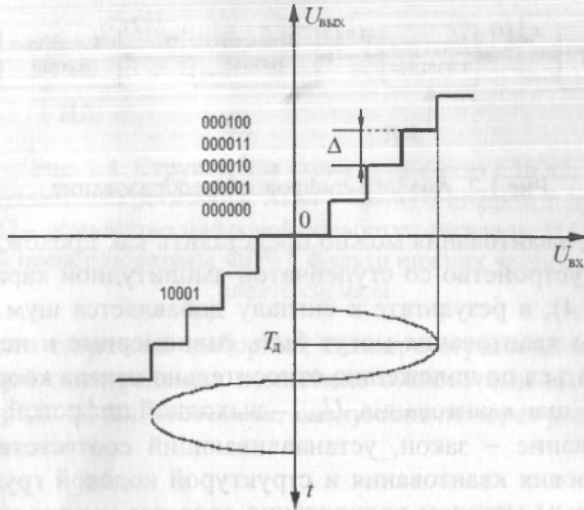


Рис. 1.4. Амплитудная характеристика АЦП

$$S(j\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt. \quad (1.3)$$

Дискретное преобразование Фурье периодического сигнала:

$$S(k) = \sum_{n=0}^{N-1} x(nt)e^{-i\frac{2\pi nk}{N}}, \quad (1.4)$$

где  $n = \overline{0, N-1}$ ,  $k = \overline{0, N-1}$ , а  $N$  – размер выборки.

Обратное ДПФ:

$$S(nt) = \frac{1}{N} \sum_{n=0}^{N-1} S(k)e^{j\frac{2\pi kn}{N}}. \quad (1.5)$$

Преобразование Лапласа получаем, заменив  $j\omega$  на  $S = a + j\omega$  ( $a - \text{const}$ )

$$X(s) = \int_{-\infty}^{\infty} x(t)e^{-st} dt, \quad (1.6)$$

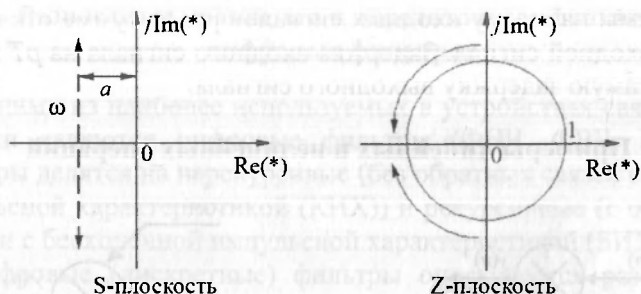


Рис. 1.5. S- и Z-плоскости

где  $x(t)$  – оригинал,  $X(s)$  – изображение.

Для дискретных систем используется дискретное Z-преобразование

$$X(z) = \sum_{n=-\infty}^{n=\infty} x(n)Z^{-n}, \quad (1.7)$$

где  $Z = e^{sT}$ ,  $Z^{-1} = e^{-sT}$  – единичная задержка.

При преобразовании Лапласа  $x(t)$  проектируется на S плоскость, а при Z-преобразовании – на Z-плоскость.

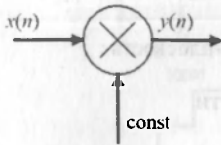
Линейная дискретная система обладает всеми свойствами линейной системы, основные из них проиллюстрированы на рис. 1.6. Умножение отсчетов входного сигнала на константу определяет умножение выходных отсчетов на ту же константу. Отклик линей-



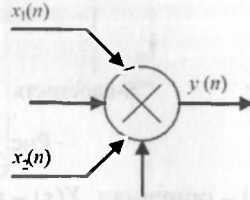
Рис.1.6. Свойства линейной системы

ной системы на сумму входных сигналов равен сумме откликов на каждый входной сигнал. Задержка входного сигнала на  $pT$  вызывает ту же самую задержку выходного сигнала.

**Примеры линейных и нелинейных операций**



Линейная операция



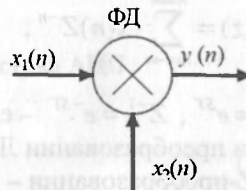
Нелинейная операция

Пример

$$x_1(nT) = A(t) \cos \omega_0 nT$$

$$x_2(nT) = B(t) \cos \omega_0 nT$$

ФД – фазовый детектор



$$x_1(t)x_2(t) = \frac{1}{2} A(t)B(t) + \frac{1}{2} A(t)B(t) \cos 2\omega_0 t$$

Первое слагаемое является полезной составляющей, содержащей информацию, а второе, нелинейная составляющая, относится к помехам и далее фильтруется (ФНЧ).

Для непрерывной системы импульсная характеристика является откликом на  $\delta$ -функцию. Импульсной характеристикой линейной дискретной системы является отклик на единичный импульс (рис. 1.7).



Рис. 1.7. Импульсная характеристика дискретной системы

Разностные уравнения и передаточные функции цифровых фильтров

Одними из наиболее используемых в устройствах связи и электроники являются цифровые фильтры (ФНЧ, ФВЧ, ПФ, РФ). Фильтры делятся на нерекурсивные (без обратных связей с конечной импульсной характеристикой (КИХ)) и рекурсивные (с обратными связями с бесконечной импульсной характеристикой (БИХ)).

Цифровые (дискретные) фильтры описываются разностными уравнениями, либо передаточными функциями. Нерекурсивный фильтр имеет разностное уравнение вида:

$$y(n) = a_0 + a_1x(n) + a_2x(n-1) + \dots + a_{N-1}x(n-N-1), \quad (1.8)$$

где  $a_n$  – коэффициенты фильтра, равные отсчетам импульсной характеристики,  $a_n = h_n$ .

Передаточная функция КИХ фильтра:

$$H(z) = a_0 + a_1z^{-1} + a_2z^{-2} + \dots + a_{N-1}z^{-N-1}. \quad (1.9)$$

Схема такого фильтра показана на рис. 1.8

Рекурсивный фильтр описывается разностным уравнением

$$y(n) = a_0x(n) + a_2x(n-1) + \dots + b_0y(n-1) + b_2y(n-2) + \dots, \quad (1.10)$$

где  $b_n$  – коэффициент обратной связи.

Передаточная функция рекурсивного фильтра может быть представлена несколькими способами, в том числе с помощью пря-

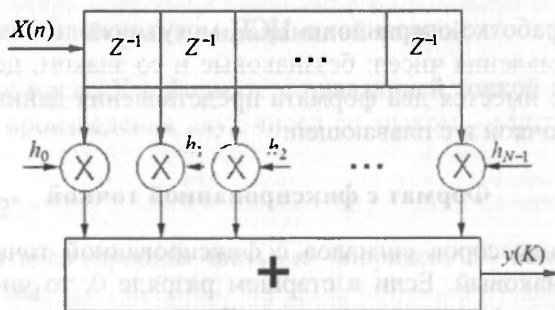


Рис. 1.8. Схема нерекурсивного фильтра

мого деления числителя на знаменатель [2]. Для примера, передаточная функция рекурсивного фильтра второго порядка (биквадратная секция) имеет вид:

$$H(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - b_0 z^{-1} - b_0 z^{-2}}. \quad (1.11)$$

Схема фильтра, реализованного по этому уравнению показана на рис 1.9.

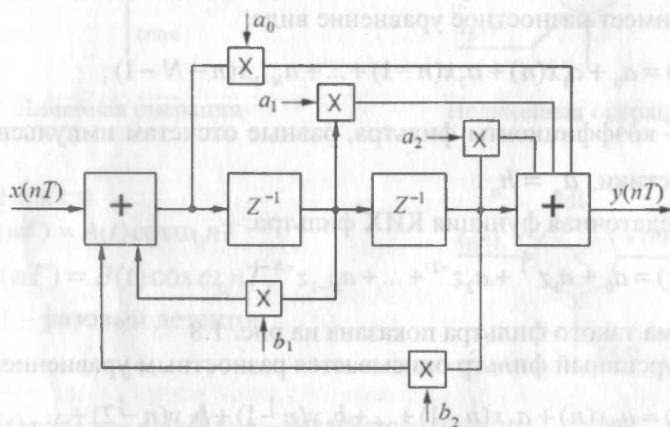


Рис. 1.9. Схема рекурсивного фильтра второго порядка

## 1.2. Форматы представления данных при обработке

При обработке операндов в ЦСП могут использоваться различные представления чисел: беззнаковые и со знаком, целые и дробные. Также имеется два формата представления данных: с фиксированной точкой и с плавающей.

### Формат с фиксированной точкой

Для процессоров сигналов с фиксированной точкой старший разряд – знаковый. Если в старшем разряде 0, то число положительное, если 1 – отрицательное. Далее следуют разряды целой части числа и за ними после точки – дробной.

Например,  $11010.10\dots0 \rightarrow -10,5$ . В частном случае данные могут быть целочисленными, тогда формат будет 16.0. Если процессор 16 разрядный, то диапазон знаковых 16 битных значений составляет  $-32768$  до  $32767$ . Например,  $4 + 12 = 16$ ,  $20000 - 30000 = -10000$ ,  $20000 + 30000 = 50000$ . В последнем случае появляется переполнение, процессор должен произвести операцию ограничения результата.

В процессорах удобно чисто дробное представление чисел  $1,15$  – один разряд знаковый, затем точка и 15 разрядов мантисса (значение чисел  $\pm 1$ ).

Такое представление удобно для уменьшения вероятности переполнения при вычислениях. Смещение разрядов при вычислениях устраняется операцией сдвига. Вычисления со знаковыми операндами проводятся в виде дополнения до 2-х (в дополнительном коде).

Пр и м е р. Число  $4 - 0110$ . Первая операция преобразования – инверсия  $0110 \rightarrow 1001$ , далее добавляется 1 к младшему разряду –  $1010$ , что и дает дополнение 2.

Положительные числа в форме дополнения до 2 не отличаются от обычного двоичного представления.

При умножении с фиксированной точкой надо учитывать, что произведение двух дробных чисел также является дробным числом, а произведение двух целых чисел – это целое число. Число разрядов после умножения удваивается. Для удаления дополнительного бита производится сдвиг влево на один разряд. При сложении дробей с фиксированной точкой операнды должны представляться в одном формате, также возможно переполнение.

### Формат с плавающей точкой

Двоичное число  $X$  в формате с плавающей точкой представляется в виде произведения двух чисел со знаком – мантиссы и экспоненты

$$X = M \times 2^n,$$

где  $M$  – мантисса (дробная часть),  $n$  – порядок, 2 – основание двоичной системы.

Мантисса не должна превышать 1. Экспонента определяет диапазон, а мантисса точность чисел. Отрицательные числа имеют ман-

тиссу со знаком (старший разряд). Числа представляются в нормализованном виде, например, 60 как  $0,6 \times 10^2$ ,  $0,0015 - 0,15 \times 10^{-2}$ . В памяти данных хранятся мантисса со знаком и порядок со знаком. При 32 разрядном процессоре число занимает 4 байта. Старший разряд определяет знак мантиссы, 23 следующих разряда занимает мантисса и 8 младших разрядов содержат знак экспоненты и показатель (7 разрядов). Числа имеют диапазон от  $-2 \times N^{127}$  до  $+ (2^{23-1}) \times N^{127}$ , а дроби  $+1 \times N^{-127}$ , где  $N = 2$ .

При перемножении

$$C = A \times B = M_1 \times M_2 \times 2^{n_1+n_2}.$$

При сложении, если  $A > B$ , то

$$C = A + B = (M_1 + M_2 \times 2^{n-n_2}) \times 2^{n_1},$$

если же  $B > A$ , то

$$C = A + B = (M_1 \times 2^{n_2-n_1} + M_2) \times 2^{n_2}.$$

При обработке сигналов реализация многих алгоритмов заключается в умножении-накоплении (типовая операция МАС – перемножить и сложить с предыдущим результатом).

### **Контрольные вопросы**

1. Каковы основные операции ЦОС?
2. Как выбирается частота дискретизации при АЦП?
3. Амплитудная характеристика амплитудного квантователя.
4. Преобразования Фурье и Лапласа. Области их применения.
5. Поясните смысл z-преобразования и почему оно удобно для описания цифровых систем?
6. Перечислите свойства дискретных линейных систем.
7. Что такое импульсная характеристика линейной и дискретной (цифровой) системы?
8. Поясните разницу нерекурсивной и рекурсивной фильтрации, разностные уравнения.
9. Форматы данных с фиксированной точкой. Дополнение до двух. Правила умножения.
10. Представление чисел с плавающей точкой.
11. Правила умножения и сложения чисел с плавающей точкой.



## Глава 2. Классификация, структура и характеристики сигнальных процессоров

### 2.1. Классификация процессоров обработки сигналов

Сигнальные процессоры можно разделить по виду сигналов на аналоговые и цифровые, как показано на рис. 2.1.

К аналоговым процессорам следует отнести устройства на поверхностных акустических волнах (ПАВ) [20], акустооптические процессоры [21, 22] и устройства функциональной электроники.

Цифровые процессоры, в свою очередь, делятся на программируемые и непрограммируемые с «жесткой логикой». Непрограммируемые процессоры представляют собой функционально законченное устройство, например, коррелятор, процессор преобразования Фурье и др. Программируемые процессоры также разделяются на

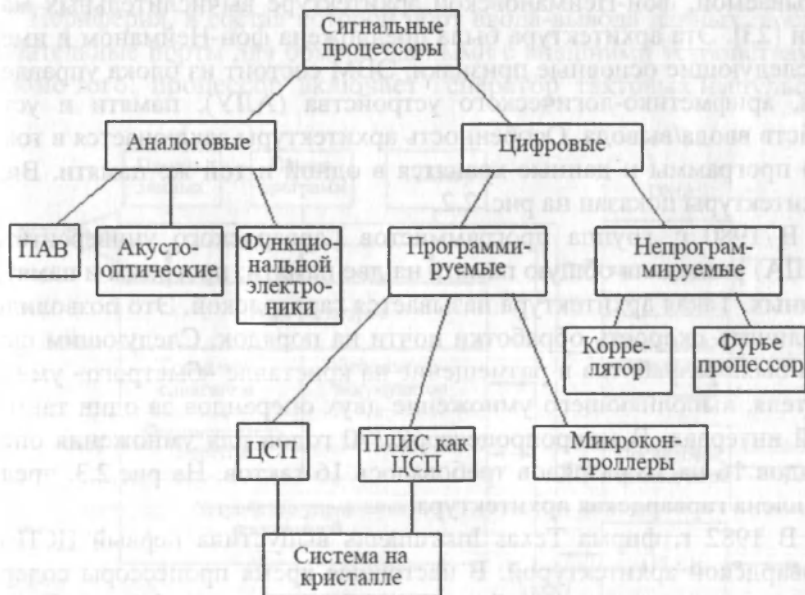


Рис. 2.1. Классификация сигнальных процессоров

два типа. К первому типу относятся процессоры, выполняющие программу, в процессе обработки сигнала. Среди них различают цифровые сигнальные процессоры (DSP) и микроконтроллеры. Первые производят обработку сигналов, вторые выполняют функции управления, коммутации и контроля [10]. Второй тип процессоров реализуется на программируемых логических интегральных схемах (ПЛИС). Их особенность заключается в том, что программирование ПЛИС происходит при проектировании, а готовое устройство имеет «жесткую логику» [14, 15]. На одном кристалле могут объединяться ЦСП и ПЛИС, образуя так называемую «систему на кристалле» [19].

Основное требование, предъявляемое к сигнальным процессорам, – обработка сигналов в реальном времени.

### 2.2. Архитектура и структура цифрового сигнального процессора

Ранее микропроцессоры в основном реализовывались по, так называемой, фон-Неймановской архитектуре вычислительных машин [23]. Эта архитектура была предложена фон-Нейманом и имеет следующие основные признаки: ЭВМ состоит из блока управления, арифметико-логического устройства (АЛУ), памяти и устройств ввода/вывода. Особенность архитектуры заключается в том, что программы и данные хранятся в одной и той же памяти. Вид архитектуры показан на рис. 2.2.

В 1980 г. группа программистов Гарвардского университета (США) разделила общую память на две памяти: программ и память данных. Такая архитектура называется гарвардской. Это позволило увеличить скорость обработки почти на порядок. Следующим шагом была разработка и размещение на кристалле «быстрого» умножителя, выполняющего умножение двух операндов за один тактовый интервал. В микропроцессорах 70 годов для умножения операндов 16 на 16 разрядов требовалось 16 тактов. На рис.2.3. представлена гарвардская архитектура.

В 1982 г. фирма Texas Instruments выпустила первый ЦСП с гарвардской архитектурой. В настоящее время процессоры содержат ряд шин связи памяти с вычислителями и периферией. Такая архитектура называется модифицированной гарвардской. Современ-

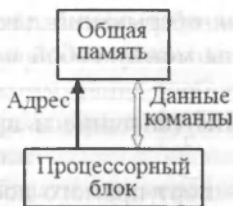


Рис. 2.2. Архитектура фон-Неймана



Рис. 2.3. Гарвардская архитектура

ные ЦСП содержат две или несколько памяти (данных и программ), собственно вычислитель (ядро-Core) и периферию для связи с внешними устройствами. На рис. 2.4 представлена обобщенная схема процессора.

Ядро ЦСП, производящее вычисления, включает в себя умножитель-аккумулятор, реализующий операцию MAC, арифметико-логическое устройство (АЛУ), кольцевой сдвигатель, устройство, контролирующее выполнение программы; формирователей адресов. Две памяти данных (DATA MEMORY – DM) и команд (PROGRAMM MEMORY – PM); содержат данные и команды.

Периферия, в состав которой порт ввода-вывода данных, последовательные порты для обмена данными с внешними устройствами. Кроме того, процессор включает генератор тактовых импульсов



Рис. 2.4. Структурная схема ЦСП

и таймер, необходимый для формирования прерываний для разделения операций. Все устройства соединены между собой шинами, число которых достигает 10.

Таймер формирует сигналы прерывания (внешние и внутренние)

ПДП (DMA – Direct Memory Access) – порт прямого доступа к памяти или обмена между внешней памятью данных и памятью программ не мешая работе ядра.

Два последовательных порта для ввода данных с АЦП и ЦАП и связи между процессорами.

Параллельный порт – к стандартному интерфейсу (RS-232).

ГТИ – генератор тактовых импульсов (определяет время инструкций).

Цифровые сигнальные процессоры в основном относятся к вычислительным устройствам с ограниченным числом команд RISC.

В ЦСП используется конвейерный режим обработки. При это одновременно выполняется несколько разных команд со сдвига на длительность инструкции. При трехступенчатом конвейере сначала следует вызов команды, затем ее дешифрирование и потом выполнение. Принцип работы трехступенчатого конвейера изображен на рис. 2.5.

Типовые операции, выполняемые ЦСП и микроконтроллерами:

- цифровая фильтрация;
- вычисление свертки;
- корреляция;
- вычисление БПФ;
- формирование сигналов;
- демодуляция;
- кодирование / декодирование;

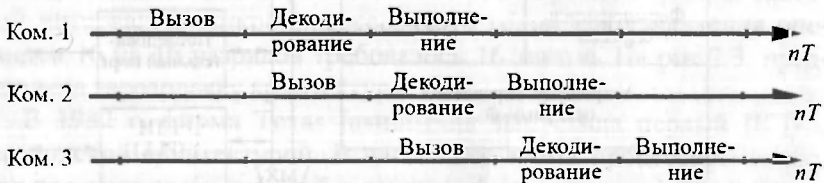


Рис. 2.5. Трехступенчатый конвейер

- обработка изображений;
- управление потоками данных;
- коммутация сообщений;
- реализация ФАПЧ, АПЧ, АРУ, регулировка мощности передатчика.

Области применения процессоров: обработка радиолокационных сигналов, радионавигация, радиоуправление, радиоразведка, обработка речи, синтез речи, модемы, цифровая телефония, передача данных, сотовая связь, цифровое радиовещание и телевидение, мультимедийные системы.

### **2.3. Характеристики цифровых сигнальных процессоров**

Сигнальные процессоры делятся на процессоры с фиксированной точкой и с плавающей точкой. [7]

К основным характеристикам процессоров относят:

**1. Производительность.** Для процессоров с фиксированной точкой производительность измеряется числом инструкций в секунду (MIPS – миллион операций с фиксированной точкой в секунду). Для ЦСП с плавающей точкой производительность измеряется в MFLOPS (миллион операций с плавающей точкой в секунду). Так же производительность оценивается в миллионах операций умножить-сложить в секунду – MMACS.

**2. Тактовая частота и длительность инструкции.** Тактовая частота определяется частотой внутреннего генератора тактовых импульсов с фазовой подстройкой частоты (PLL). Эта частота определяет быстродействие процессора. Обратная величина  $t_n = 1/F_T$  определяет длительность инструкции. Для выполнения команд требуется 2–3 времени инструкции. Следует отметить, что тактовая частота процессора должна превышать внешнюю частоту дискретизации обрабатываемого сигнала примерно на порядок, поскольку за время между двумя вводимыми отсчетами сигнала процессор должен выполнить несколько операций (команд) по обработке первого отсчета. Это обстоятельство нужно учитывать при выборе типа процессора.

**3. Число разрядов процессора** определяет точность вычислений и динамический диапазон чисел (сигналов).

$M = 16$  (с фиксированной точкой); 24 (Motorola) (с фиксированной точкой); 32 (с плавающей точкой); 64 (с плавающей точкой), где  $D = 20 * \lg(A_{\max}/A_{\min})$ , где  $A_{\max}$  – максимальное и  $A_{\min}$  – минимальное числа.

**4. Объем памяти.** Объем памяти как данных, так и программ составляет от нескольких килобит до Мбит. Как правило, необходимости к подключению внешней памяти нет.

**5. Технология.** КМОП (комплементарная металл-окисел-полупроводник) от 0,5 до 0,1 мкм (в последних поколениях ЦСП). Технология определяет производительность и быстродействие.

**6. Потребляемая мощность.** Процессор первого поколения TMS320C10 потреблял 2 Вт. Современные ЦСП 0,05 Вт и менее, что существенно, особенно для сотовых телефонов. Напряжение питания  $U_n = 5$  В; 3,3 В; 2,5; 1,8 В. Имеется режим малого потребления, включаемый специальной командой.

**7. Корпуса процессоров.** Корпуса ЦСП имеют от 40 до 300 отводов, а ПЛИС до 800.

**8. Язык программирования.** Язык программирования – Макроассемблер. К сожалению, для ЦСП разных фирм аббревиатуры команд разные. В настоящее время ведущие фирмы разработали трансляторы с языка высокого уровня  $C^+$  на языки конкретных процессоров, что существенно облегчает задачи программистов. Процессор с ограниченным набором команд называется RISC-процессором, если число команд неограничено, то это CISC-процессор.

**9. Стоимость.** Стоимость ЦСП составляет от одного до нескольких сотен долларов США в зависимости от производительности и быстродействия. Как правило, процессоры с плавающей точкой дороже, чем с фиксированной, однако, фирма Analog Devices выпускает процессоры типа SHARC с плавающей точкой минимальной стоимостью примерно 10 долларов при серийном производстве.

Повышение производительности ЦСП достигается за счет увеличения тактовой частоты путем улучшения технологии, применения конвейерной и параллельной обработки данных и уменьшения времени доступа к памяти. При параллельной обработке применяются (иногда совместно) три архитектуры: SIMD (single instruction, multiple data – одна команда, много данных), VLIW (very long

instruction ord) – очень длинное командное слово и суперскалярная обработка. При архитектуре SIMD одна команда передается нескольким вычислительным блокам при этом обработка идет одновременно. В процессорах с длинным командным словом имеется ряд вычислителей, работающих параллельно, причем каждому вычислителю параллельно передается своя команда. Длинная команда состоит из ряда коротких, вызываемых одновременно. Пример такого ЦП – платформа C 6000 фирмы Texas Instruments. При суперскалярной обработке система команд не содержит параллельности обработки, но несколько последовательных команд в процессоре выполняются параллельно.

### **Контрольные вопросы**

1. Перечислите основные признаки классификации сигнальных микропроцессоров.
2. В чем разница программирования ЦП и ПЛИС?
3. Чем отличаются архитектура МП фон-Неймана и Гарвардская? Что такое модифицированная Гарвардская архитектура?
4. Поясните смысл конвейерной обработки на примере трехступенчатого конвейера.
5. Что такое RISC архитектура?
6. Назовите основные блоки сигнального процессора.
7. Какие узлы включает ядро процессора?
8. Что входит в периферию сигнального процессора?
9. Перечислите типовые операции ЦОС, выполняемые процессорами.
10. Каковы основные характеристики сигнальных процессоров?
11. Как оценивается производительность ЦСП?
12. На какие характеристики ЦСП влияет число разрядов?
13. Что такое Макроассемблер ЦСП?

## Глава 3. Цифровой сигнальный процессор ADSP 2181 фирмы Analog Devices

### 3.1. Структурная схема и характеристики процессора

Процессор ADSP 2181 16-разрядный с фиксированной точкой [A1, A8]. Схема ЦСП показана на рис. 3.1. В состав процессора входят две памяти – данных (DATA MEMORY – DM) и программ (PROGRAM MEMORY – PM), ядро, состоящее из умножителя-аккумулятора, АЛУ, кольцевого сдвигателя, двух формирователей адресов данных и команд и устройства управления программой, и периферии, включающей два последовательных порта (SPORT0 и SPORT1), порт прямого доступа к памяти (ПДП – DMA), порты связи со стандартными интерфейсами, таймер, формирующий программные прерывания, и генератор тактовых импульсов.

Имеется 5 шин для пересылки данных и команд:

DMA – data memory address (шина адресов памяти данных);

PMA – program memory address (шина адресов памяти программ);

PMD – program memory data (шина данных памяти программ – шина команд);

DMD – data memory data (шина памяти данных – шина данных);

R – results (шина результатов). Наличие этой шины является отличием процессоров ADSP21\*\*. По ней результат вычислений может передаваться непосредственно на вход другого вычислителя, например, с выхода умножителя на сдвигатель, минуя шину данных, что повышает скорость обработки.

Характеристики процессора:

Производительность  $\Pi = 33 \text{ MIP S}$ .

Тактовая частота 33 МГц.

Память данных – 16 Кслов по 16 разрядов, программ – 16 К слов по 16 разрядов.

Технология КМОП 0,5 мкм.

Корпус имеет 128 выводов.

Питание 5 В или 3,3 В.



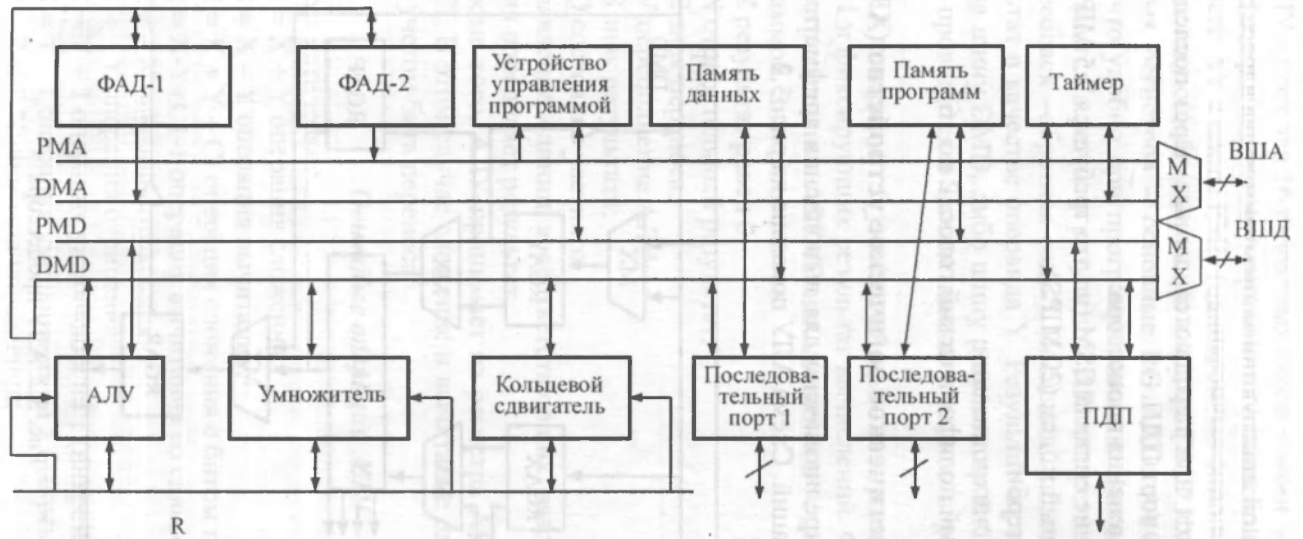


Рис. 3.1. Схема процессора ADSP 2181

Операции:

- за время одной инструкции записываются два адреса данных;
- выполняется одно вычисление;
- принимаются или передаются данные через последовательные порты, через порт ПДП.

Области применения процессора:

- формирование сигнала GSM (на это требуется 2.5 MIPS );
- универсальный модем (20 MIPS);
- декодер Витерби;
- устройства синхронизации;
- 24-канальный полифонический голос (2 по 16).

### 3.2. Арифметическое логическое устройство (АЛУ)

Устройство предназначено для выполнения арифметических и логических операций. Схема АЛУ показана на рис. 3.2.

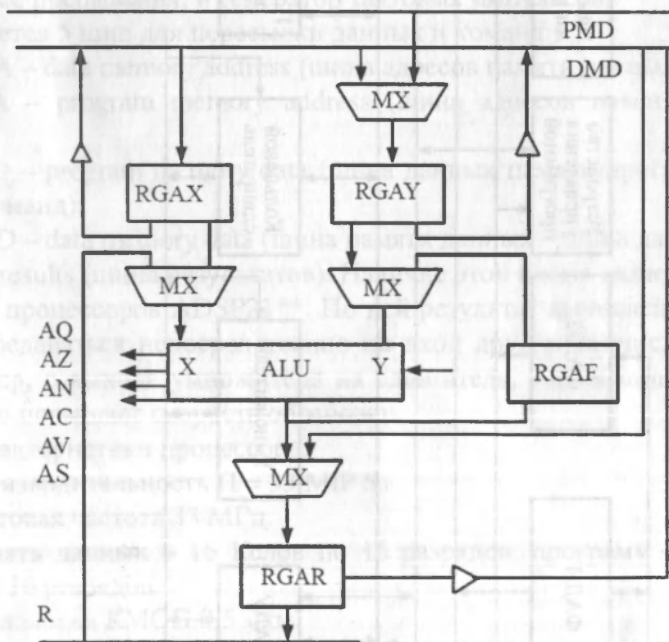


Рис. 3.2. АЛУ процессора

АЛУ имеет два 16 разрядных входа и один порт вывода результата. Входные операнды загружаются в регистр AX с шины DMD и в регистр AY с шины PMD (одновременно) или последовательно за время следующей инструкции с шины DMD. Имеются две группы входных регистров – основные RG AX0, AY0 и теньевые AX1, AY1, что удобно при программировании. Выходные данные АЛУ помещаются в регистр результата AR или регистр обратной связи AF (feedback – обратная связь), Последний позволяет использовать результат в качестве операнда Y. Результат с регистра AR передается на шину DMD, либо шину результатов R. Считывание с регистров проводится в начале цикла, а запись в регистры – в конце цикла.

АЛУ генерирует 6 указателей- флагов, принимающих значения 0 или 1 и фиксирующих результат вычислений. Флаги помещаются в регистр состояния ASTAT, находящийся в устройстве управления программой. Значения флагов:

- AZ результат равен 0;
- AN отрицательный результат;
- AC знак переноса;
- AV переполнение АЛУ;
- AS знак результата;
- AQ состояние частного.

Появление единиц в соответствующих разрядах регистра состояния определяет результат.

Кроме того, АЛУ принимает из регистра ASTAT сигнал переноса CI. В этом случае сложение и вычитание операндов выполняется с учетом бита переноса.

#### *Основные операции АЛУ*

Математические:

- $R = X + Y$  операция сложения;
- $R = X - Y$  операция вычитания;
- $R = X + Y + CI$  операция сложения с битом переноса;
- $R = X - Y + CI - 1$  операция вычитания со сдвигом и переносом;
- $R = -X$  операция отрицания;
- $R = -Y$  операция отрицания;
- $R = Y + 1$  операция увеличения на 1 (инкремент);
- $R = Y - 1$  операция уменьшения на 1 (декремент);
- $R = PASS X$  передача управления;

- $R = \text{PASS } Y$ ;
- $R=0$  результат равен 0;
- $R = \text{ABS } X$  абсолютное значение  $X$ ;

Логические:

- $R = X \text{ AND } Y$  логическое «И»;
- $R = X \text{ OR } Y$  логическое «ИЛИ»;
- $R = X \text{ XOR } Y$  исключающее «ИЛИ»;
- $R = \text{NOT } X$  логическое отрицание;
- $R = \text{NOT } Y$  логическое отрицание.

Источниками данных могут быть также регистры результатов умножителя или сдвигателя.

В регистре AR имеется режим насыщения для операций в дополнительном коде при появлении флага AV. При этом регистр AR устанавливается в максимальное положительное (7FFH) (если флаг переполнения  $AV = 1$ , а флаг переноса AC равен 0) или отрицательное число (8000H) ( $AV = 1, AC = 0$ ).

### **3.3. Умножитель-аккумулятор процессора (MAC)**

Умножитель-аккумулятор производит следующие операции:

- Умножение  $X$  и  $Y$ .
- Умножение  $X$  на  $Y$  и добавить результат к содержимому памяти MR.
- Умножение  $X$  на  $Y$  и вычесть результат из содержимого памяти MR.

MAC обеспечивает два стандартных режима умножения :

- Дробный режим в стандарте 1.15.
- Целый режим в стандарте 16.0.

На рис.3.3 представлена структурная схема умножителя.

Входные операнды по 16 разрядов поступают с шины DMD на регистр MX и одновременно с шины PMD на регистр MY или последовательно на него с шины DMD.

После умножения 32 разрядный результат добавляется или вычитается из содержимого аккумулятора и помещается в регистры результата MR2 (8p), MR1 (16p), MR0 (16p), либо в регистр обратной связи MF (16p) для использования в качестве операнда Y. Результат умножения – накопления возвращается на шину DMD или шину результата R.

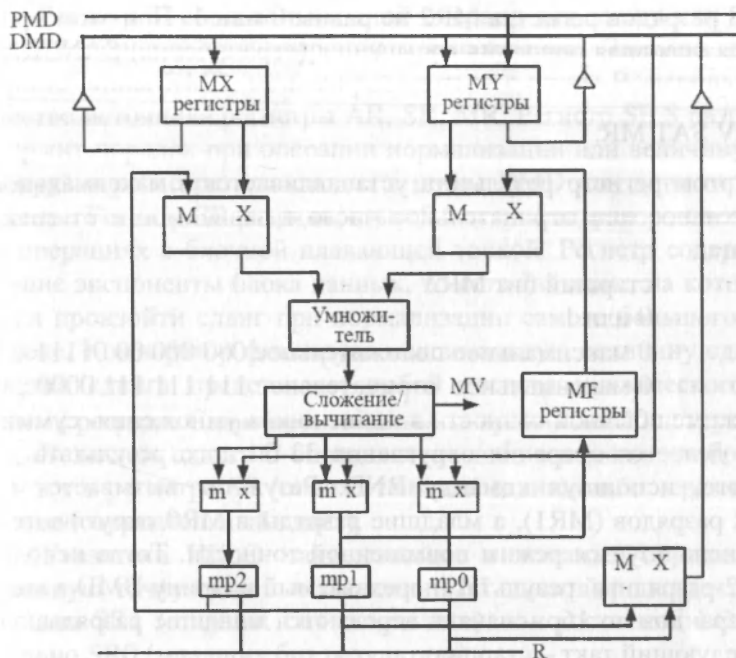


Рис.3.3. Умножитель-аккумулятор процессора

Источниками операнда X для умножения могут быть также регистры результата умножителя, сдвигателя и АЛУ.

### Основные операции умножения

$MR = MX * MY$  – умножение.

$MR = MR + MX * MY$  – умножение- сложение (MAC – операция).

$MR = MR - MX * MY$  – умножение – вычитание.

$MR = 0$  – очистить регистр MR.

(S,S) – вариант умножения – «знаковое на знаковое» (sign \* sign).

(US) – «незнаковое на знаковое» (unsign \* sign).

(SU) – «знаковое на незнаковое» (sign \* unsign).

(UU) – «незнаковое на незнаковое». (unsign \* unsign).

Форматы операндов записываются как часть инструкции.

При появлении единиц в регистре MR2 устанавливается флаг переполнения  $MV = 1$  (и записывается в регистр ASTAT), т. е. ко-

гда все 8 разрядов регистра MR2 не равны 0 или 1. При этом производится условная операция насыщения (ограничения) SANURATION

IF MV SAT MR.

При этом регистр результата устанавливается в максимальное положительное или отрицательное число в зависимости от знака результата.

MV	старший бит MR2
0	0 или 1
1	0 максимальное положительное 000 000 00 01111...1
1	0 максимальное отрицательное 111 111 111 0000...0

В режиме обычной точности в конце цикла умножения- суммирования делается операция округления 32-битного результата до 16-битного, используя команду RND. Результат вызывается из старших разрядов (MR1), а младшие разряды в MR0 округляются. Иногда используется режим повышенной точности. Тогда используется 32-разрядный результат, передаваемый на шину DMD в виде двух операндов по 16р: сначала передаются младшие разряды, затем в следующий такт – старшие.

Пример инструкции MAC с округлением результата

MR= MR + MX0×MY0 (RND).

### 3.4. Кольцевой сдвигатель (Barrel Shifter)

Сдвигатель процессора выполняет арифметические (ASIFT) и логические операции сдвигов (LSIFT) входного операнда, операцию нормализации – NORM преобразования числа с фиксированной точкой в число с плавающей точкой и денормализации – обратное преобразование и операцию с блочной плавающей точкой (EXPADJ).

Схема кольцевого сдвигателя представлена на рис 3.4.

Кольцевой сдвигатель включает входной регистр SI, решетку сдвига (SHIFTER ARRAY), два выходных регистра SR0, SR1 по 16 разрядов, логику OR/ PASS, детектор экспоненты, определяющий порядок при операции нормализации, цифровой компаратор и регистры SB и SE, а также устройство инвертирования знака (NEGATE).

### Глава 3. Цифровой сигнальный процессор ADSP

Регистр SI содержит входной операнд для сдвига или для установки порядка при операции нормализации. Загрузка регистра, кроме шины DMD, может проводиться с шины R, используя в качестве источника регистры AR, SR, MR. Регистр SE 8 разрядов содержит порядок при операции нормализации или величину сдвига. Загрузка регистра происходит с шины DMD или детектора экспоненты. Регистр SB разрядностью 5 бит и компаратор используются в операциях с блочной плавающей точкой. Регистр содержит значение экспоненты блока данных, то есть значение, на которое должен произойти сдвиг при нормализации самого большого числа в блоке. Компаратор фиксирует максимальную величину сдвига. Решетка сдвига представляет собой регистр циклического сдвига 16x32 разряда. Сдвиг производится за время одной инструкции.

Величина сдвига С определяется либо в команде, либо записывается в регистре SE (положительный сдвиг), либо через инвертор знака (отрицательный сдвиг).

Указатели HI/LO, записываемые в команде, определяют начальную точку сдвига: в состоянии HI сдвиги происходят относительно SR1 (старших бит результата), а в состоянии LO относительно SR0 (младших бит результата).

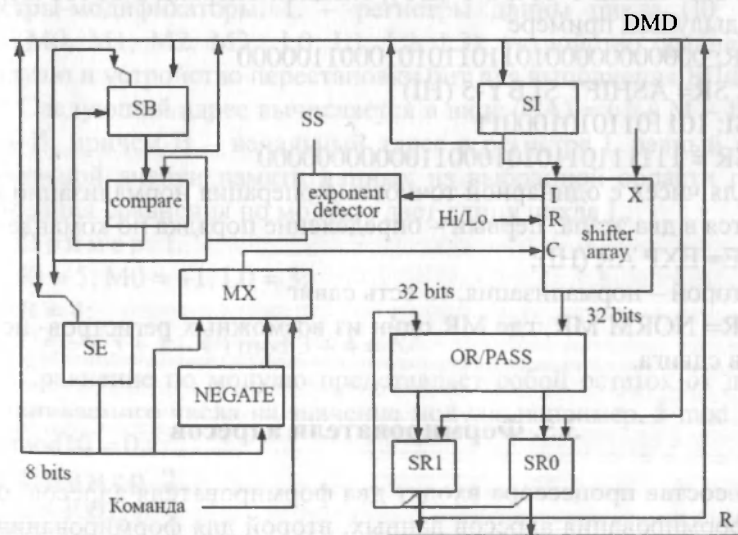


Рис. 3.4. Кольцевой сдвигатель

Все разряды справа от результата сдвига в регистрах SR1, SR0 заполняются нулями, а разряды слева нулями, если производился логический сдвиг, и единицами, если арифметический. Логика сдвига OR/PASS позволяет соединить части числа с повышенной точностью после их сдвига в одно число. В режиме PASS операнд после сдвига записывается в регистр SR без изменений. В режиме OR (ИЛИ) операнд перед записью в регистры результата подвергается логической операции ИЛИ с текущим содержимым регистра SR.

### Примеры команд

1. SR=LSHIFT SI BY-5 (HI) HI – сдвиг относительно 16 старших разрядов (SR1).

LO – сдвиг относительно 16 младших разрядов (SR0).

Произвести логический сдвиг операнда в SI на –5 разрядов (т.е. сдвиг входного значения относительно верхнего выходного регистра).

SI в двоичном коде 1011011010100011.

При логическом сдвиге команда L SHIFT. Освобождающийся раздел заполняется 0, а при команде A SHIFT – единицами.

SR: 0000010110110101

2. SR=LSHIFT SI BY-5 (LO), входной операнд в SI тот же, что в предыдущем примере

SR: 0000000000101101101010001100000

3. SR=ASHIFT SI BY-5 (HI)

SI: 1011011010100011

SR = 1111101101010001100000000000

Для чисел с одинарной точностью операция нормализации проводится в два этапа: первый – определение порядка по команде

SE= EXP AR (HI);

второй – нормализация, то есть сдвиг

SR= NORM MR, где MR один из возможных регистров-источников сдвига.

### 3.5. Формирователи адресов

В состав процессора входят два формирователя адресов: один для формирования адресов данных, второй для формирования адресов данных или команд, причем адреса вычисляются одновременно за время одной команды. Формирователи осуществляют



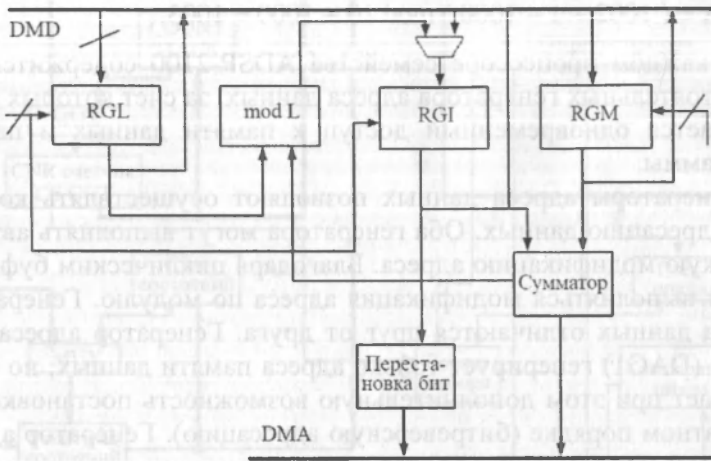


Рис. 3.5. Формирователь адресов данных

косвенную циклическую или линейную адресацию при выполнении программы. Схема первого формирователя показана на рис. 3.5. В него входят ряд регистров: I – индексные регистры, M – регистры-модификаторы, L – регистры длины цикла (I0, I1, I2, I3...M0, M1, M2, M3...L0, L1, L2, L3); устройство сравнения по модулю и устройство перестановки бит для выполнения БПФ.

Следующий адрес вычисляется в виде  $(CA) = (I + M - B) \bmod L + B$ , причем B – начальный адрес в регистре I, равный номеру начальной ячейки памяти данных из выбранной области памяти. Операция сравнения по модулю дает длину цикла L.

Пример 1.

$$I_0 = 5, M_0 = +1, L_0 = 3;$$

$$B = 4;$$

$$CA = (5 + 1 - 4) \bmod 3 + 4 = 6.$$

Сравнение по модулю представляет собой остаток от деления сравниваемого числа на значение модуля, например,  $5 \bmod 10 = 5$ ,  $10 \bmod 10 = 0$ .

Пример 2.

$$I = 1002;$$

$$M = 1;$$

$$B = 1000;$$

$L = 10;$

$CA = (1002 + 1 - 1000) \bmod 10 + 1000 = 1003.$

В каждом процессоре семейства ADSP-2100 содержится два самостоятельных генератора адреса данных, за счет которых обеспечивается одновременный доступ к памяти данных и памяти программы.

Генераторы адреса данных позволяют осуществлять косвенную адресацию данных. Оба генератора могут выполнять автоматическую модификацию адреса. Благодаря циклическим буферам, может выполняться модификация адреса по модулю. Генераторы адреса данных отличаются друг от друга. Генератор адреса данных 1 (DAG1) генерирует только адреса памяти данных, но обеспечивает при этом дополнительную возможность постановки бит в обратном порядке (битреверсную адресацию). Генератор адреса данных 2 (DAG2) генерирует как адреса памяти данных, так и адреса памяти программы, но не способен поддерживать битреверсную адресацию.

При ознакомлении с приводимым ниже описанием внутреннего устройства и работы генераторов адреса данных следует иметь в виду, что в результате развития программного обеспечения процессоров семейства ADSP-2100 появился прямой способ объявления буферов циклическими или линейными и управления положением буфера в памяти. Специальное программирование требуется только для инициализации регистров генераторов адреса данных

### **3.6. Устройство управления программой**

Устройство управления программой производит контроль за последовательностью выполнения команд, обработку прерываний и реализует команды условного и безусловного перехода. Выходом устройства является адрес следующей выполняемой команды. Схема устройства управления показана на рис. 3.6. Счетчик команд используется в качестве одного из источников следующей команды при последовательном выполнении программы. Стек счетчика команд используется как источник адреса следующей инструкции при возврате из подпрограммы или процесса отработки прерываний. Значение с вершины стека дает следующий адрес при возврате к началу цикла DO UNTIL.

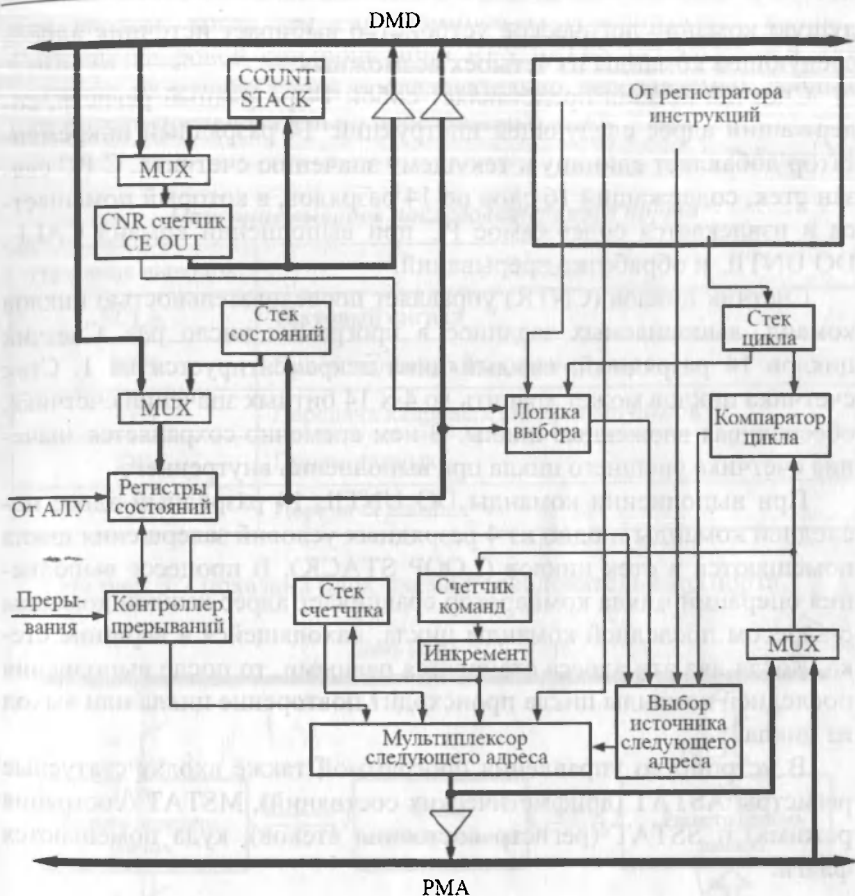


Рис. 3.6. Устройство управления программой

Регистр команд определяет следующий адрес в командах безусловного перехода. Адрес перехода 14 разрядов содержится в командном слове JAMP <addr>< CALL <addr>. Другой вариант адреса – индексные регистры в DAG2 при безусловном косвенном переходе по адресу, содержащемуся в индексном регистре, например, JAMP(I4). Счетчик PC загружается по шине RMA. При обработке прерываний по команде, поступающей на контроллер прерываний, на мультимплексор следующего адреса поступает адрес команды прерывания, расположенной в памяти команд по адресу соответствующего вектора прерываний. Пока процессор выполняет преды-

дущую команду логическое устройство выбирает источник адреса следующей команды из четырех возможных.

Счетчик команд представляет собой 14 разрядный регистр, содержащий адрес следующей инструкции. 14 разрядный инкрементатор добавляет единицу к текущему значению счетчика. С PC связан стек, содержащий 16 слов по 14 разрядов, в который помещается и извлекается содержимое PC при выполнении команд CALL, DO UNTIL и обработке прерываний.

Счетчик циклов (CNTR) управляет последовательностью циклов команд, выполняемых заданное в программе число раз. Счетчик циклов 14 разрядный, каждый цикл декрементируется на 1. Стек счетчика циклов может хранить до 4-х 14 битных значений счетчика, обеспечивая вложенные циклы. В нем временно сохраняется значение счетчика внешнего цикла при выполнении внутреннего.

При выполнении команды DO UNTIL 14 разрядный адрес последней команды и одно из 4 разрядных условий завершения цикла помещаются в стек циклов (LOOP STACK). В процессе выполнения операций цикла компаратор сравнивает адрес каждой команды с адресом последней команды цикла, находящейся в вершине стека. Когда два эти адреса становятся равными, то после выполнения последней команды цикла происходит повторение цикла или выход из цикла.

В устройство управления программой также входят статусные регистры ASTAT (арифметических состояний), MSTAT (состояния режима) и SSTAT (регистр состояния стеков), куда помещаются флаги.

### 3.7. Последовательные порты

Синхронные последовательные порты (SPORT0 и SPORT1), поддерживают множество протоколов обмена данными через последовательный порт (табл. 3.1) и могут обеспечить прямое соединение процессоров в системе, состоящей из нескольких процессоров.

Последовательный порт принимает последовательно передаваемые данные на выводе DR и последовательно передает данные через вывод DT. Работая в дуплексном режиме он может одновременно принимать и передавать данные. Эти биты данных синхронны с тактовыми синхроимпульсами генератора SCLK, которые являются выводом, когда процессор генерирует эти синхроимпульсы,

### Глава 3. Цифровой сигнальный процессор ADSP

или вводом, когда эти синхроимпульсы генерируются внешне. Сигналы кадровой синхронизации RFS и TFS используются для указания на начало слова последовательно передаваемых данных или потока последовательно передаваемых слов.

Таблица 3.1

Описание выводов последовательного порта

Название выводов	Функция
SCLK	Тактовый сигнал
RFS	Прием кадровых синхроимпульсов
TFS	Передача кадровых синхроимпульсов
DR	Прием данных
DT	Передача данных

На рис. 3.7 показана блок-схема последовательного порта.

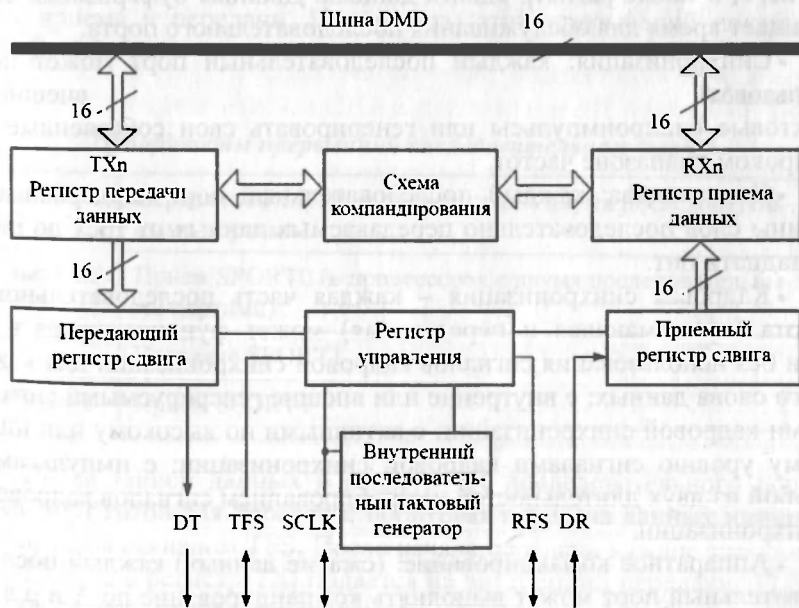


Рис.3.7. Схема последовательного порта

Данные, предназначенные для передачи, записываются через шину DMD из внутреннего регистра процессора в регистр передачи TX. Эти данные могут сжиматься в блоке компандирования, а затем автоматически передаются в передающий регистр сдвига. Из регистра сдвига биты передаются на вывод последовательного порта DT, начиная с самых старших бит, синхронно с тактовыми синхроимпульсами. Принимающая часть последовательного порта принимает данные с вывода DR, также синхронно с тактовыми синхроимпульсами. После приема одного слова данные расширяются в блоке компандирования, а затем передаются в регистр приема данных RX последовательного порта.

Далее рассмотрим свойства последовательного порта. Многие из них обладают перестраиваемой конфигурацией, за счет чего достигается гибкость в работе последовательного порта.

- Двухнаправленность: в каждом последовательном порту принимающая и передающая части независимы друг от друга.

- Двойная буферизация как передающей части, так и приемной: имеется регистр данных для передачи слов данных в процессор и из него, а также регистр сдвига данных. Двойная буферизация сокращает время для обслуживания последовательного порта.

- Синхронизация: каждый последовательный порт может использовать внешние тактовые синхроимпульсы или генерировать свои собственные в широком диапазоне частот.

- Длина слова: каждый последовательный порт поддерживает длины слов последовательно передаваемых данных от трех до шестнадцати бит.

- Кадровая синхронизация – каждая часть последовательного порта (принимающая и передающая) может функционировать с или без использования сигналов кадровой синхронизации для каждого слова данных; с внутренне или внешне генерируемыми сигналами кадровой синхронизации; с активными по высокому или низкому уровню сигналами кадровой синхронизации; с импульсами любой из двух длительностей и стробированием сигналов кадровой синхронизации.

- Аппаратное командирование: (сжатие данных) каждый последовательный порт может выполнять компандирование по А и  $\mu$  алгоритмам.

### Глава 3. Цифровой сигнальный процессор ADSP

- Автобуферизация с затратами не более, чем в один цикл: используя генераторы адресов данных, каждый последовательный порт может автоматически принимать и/или передавать целый циклический буфер данных с затратами в одно слово данных. Передачи между последовательным портом и циклическим буфером осуществляются в этом режиме автоматически и не требуют дополнительного программирования.

- Прерывания: каждая часть последовательного порта (принимающая и передающая) генерирует прерывания по завершению передачи слова данных или после передачи целого буфера при использовании автобуферизации.

- Многоканальность: SPORT0 может селективно принимать и передавать данные из потока последовательно передаваемых бит с использованием мультиплексирования 24 или 32 каналов с временным разделением информации. Это существенно важно для интерфейсов T1 или может использоваться для сетевой коммуникации нескольких процессоров.

- Прерывания. Каждый последовательный порт имеет прерывания приема и передачи. Приоритеты этих прерываний показаны в табл. 3.2

Таблица 3.2

#### *Приоритеты прерываний последовательного порта*

Высший	Передача SPORT0 (в процессорах с двумя последовательными портами)
	Прием SPORT0 (в процессорах с двумя последовательными портами)
	Передача SPORT1
Низший	Прием SPORT1

После записи данных в регистр TX последовательного порта этот порт готов для передачи; побитовая передача данных инициализируется сигналом TFS. После начала передачи каждое значение, записанное в регистр TX, подается на внутренний передающий регистр сдвига, откуда передается последовательно по битам, начиная с самого старшего бита. Каждый бит сдвигается по переднему фронту тактового синхроимпульса.



Рис. 3.8. Схема подключения АЦП и ЦАП к последовательному порту

После окончания передачи первого бита слова (самого старшего бита) последовательный порт генерирует прерывание передачи. Несмотря на то, что передача первого слова еще продолжается, становится возможной запись нового слова данных в регистр TX.

В принимающей части последовательного порта биты накапливаются в том порядке, как они поступают во внутренний регистр приема. По окончании приема одного целого слова оно записывается в регистр RX, и последовательный порт генерирует прерывание приема.

На рис 3.8. показано подключение АЦП и ЦАП к последовательному порту. В этом случае на АЦП и ЦАП подаются тактовые импульсы с внешнего генератора, они поступают также на последовательный порт для синхронизации и на специальные выходы процессора IRQ0 и IRQ1 в качестве сигналов прерываний.

### 3.8. Порт прямого доступа к памяти (ПДП- DMA)

Параллельный порт (ПДП) производит обмен данными и командами между внешними устройствами и внутренней памятью данных и команд, причем обмен может происходить в обе стороны. Работает под управлением процессора (host-processor) или контроллера ПДП. Достоинство данного порта заключается в том, что при обмене данными и инструкциями через него не требуется выключение ядра. Схема порта ПДП к внешнему интерфейсу представлена на рис. 3.9. На схеме также показан регистр управления портом.

По сигналам IS и IRD процессор выводит содержимое адресуемой ячейки внутренней памяти на шину данных порта. По сигналам



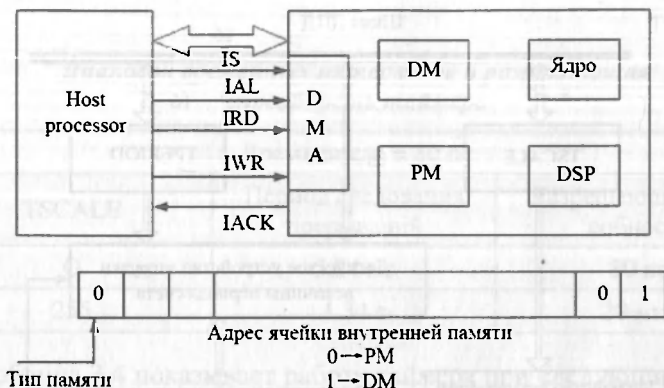


Рис. 3.9. Подключение порта ПДП к устройству управления DMA – верхняя шина передает адреса и данные; IS – выбор порта; IAL – управление записью адреса и ячеек памяти; IRD – чтение данных через порт; IWR – запись данных; IACK – операция чтения, записи произведена и порт готов к работе; RG – управление

лам IS и IWR входное слово с шины данных порта записывается в адресуемую ячейку внутренней памяти. Порт может использоваться для начальной загрузки памяти при перезагрузке процессора.

### 3.9. Таймер процессора

Таймер предназначен для формирования программных прерываний, необходимых для разделения подпрограмм. Схема таймера процессора изображена на рис.3.10. Таймер состоит из двух 16-разрядных регистров, TCOUNT и TPERIOD, и одного 8-разрядного регистра TSCALE.

Команда управления режимом инициализирует и блокирует таймер за счет установки и сброса бита 5 в регистре состояния режима MSTAT. Ниже показаны отображенные в карте памяти регистры таймера.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
					TPERIOD Регистр периода														0x3FFD	
					TCOUNT Регистр счетчика															0x3FFC
										TSCALE Регистр масштабирования									0x3FFB	

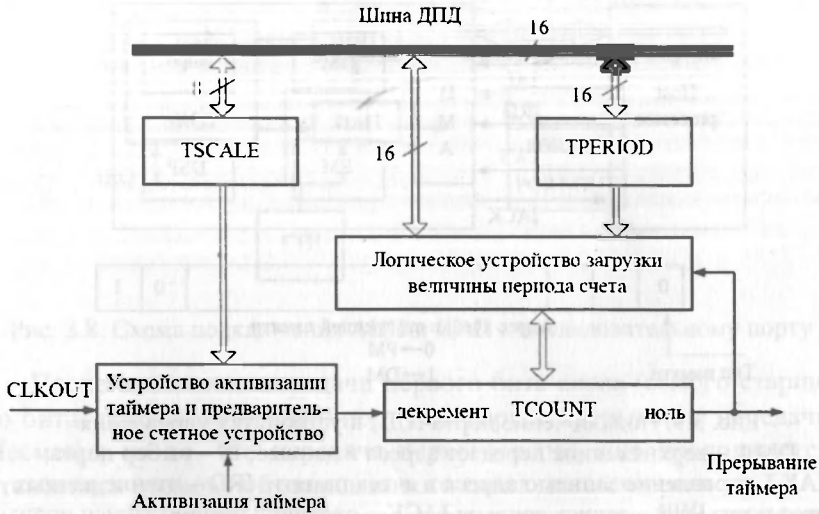


Рис. 3.10. Схема таймера

TCOUNT является регистром счетчика. Содержимое этого регистра подвергается после активизации таймера декрементированию в каждом командном цикле. Когда счетчик становится равным нулю, генерируется прерывание. Затем регистр TCOUNT перезагружается значением, хранящимся в регистре TPERIOD и счетчик снова начинает свою работу.

В регистре TSCALE содержится значение коэффициента масштабирования, которое всегда на единицу меньше, чем количество циклов между отрицательными приращениями (декрементами) значения в регистре TCOUNT. Например, когда значение в регистре TSCALE равно нулю, то значение в регистре счетчика декрементируется в каждом цикле. Когда TSCALE содержит 1, счетчик подвергается декрементированию каждые два цикла.

### Разрешающая способность

Регистр TSCALE позволяет задавать более длинные временные интервалы между прерываниями, расширяя, таким образом, возможности 16-разрядного регистра TCOUNT. В табл. 3.3 показан диапазон временных интервалов таймера и зависимость между периодом следования прерываний и разрешающей способностью при максимальном значении регистра TPERIOD.

Таблица 3.3

*Диапазон временных интервалов и разрешающая способность таймера*

Время цикла = 80 нс		
TSCALE	Период следования прерываний	Разрешающая способность
0	5,24 мс	80 нс
255	1,34 с	20 мкс

Таблица 3.4 показывает работу таймера при следующих состояниях регистров: TPERIOD = 5, TSCALE = 1 и TCOUNT = 5. После активизации таймера (цикл  $n - 1$ ) счетчик начинает свою работу. Так как TSCALE = 1, значение в регистре счетчика TCOUNT декрементируется каждый второй цикл. Перезагрузка регистра TCOUNT и продолжение работы счетчика происходит так, как это показано в таблице, во время выполнения подпрограммы обслуживания прерывания.

Таблица 3.4

*Пример работы таймера*

Цикл	TCOUNT	Действие
$n - 4$		В TPERIOD загружается значение 5
$n - 3$		В TSCALE загружается значение 1
$n - 2$		В TCOUNT загружается значение 5
$n - 1$	5	Выполняется команда активизации таймера ENA TIMER
$n$	5	Так как TSCALE = 1, TCOUNT не декрементируется
$n + 1$	5	TCOUNT декрементируется
$n + 2$	4	Нет декремента
$n + 3$	4	TCOUNT декрементируется
$n + 4$	3	Нет декремента
$n + 5$	3	TCOUNT декрементируется
$n + 6$	2	Нет декремента

## Сигнальные микропроцессоры

Цикл	TCOUNT	Действие
n + 7	2	TCOUNT декрементируется
n + 8	1	Нет декремента
n + 9	1	TCOUNT декрементируется
n+10	0	Нет декремента
n + 11	0	TCOUNT становится равным 0, происходит прерывание, TCOUNT загружается значением из TPERIOD
n+12	5	Нет декремента
n+13	5	TCOUNT декрементируется
n+14	4	Нет декремента
n + 15	4	TCOUNT декрементируется и т.д.

Прерывание происходит через каждые  $(TPERIOD + 1) * (TSCALE + 1)$  циклов. Для установки другого временного интервала в последующих прерываниях в регистр TCOUNT из регистра TPERIOD загружается требуемое значение. Интервал первого прерывания также вычисляется по формуле:  $(TCOUNT + 1) * (TSCALE + 1)$ .

Запись нового значения в регистр TSCALE или TCOUNT оказывает эффект незамедлительно. При записи нового значения в регистр TPERIOD никаких изменений не происходит до перезагрузки регистра TCOUNT.

### 3.10. Система команд процессора ADSP2181

Система команд (инструкций) процессора включает в себя команды управления программой, команды прерываний, команды пересылки и загрузки и команды вычислений (АЛУ, умножителя/аккумулятора и сдвигателя) и другие.

Система команд поддерживает прямую, косвенную и непосредственную адресацию.

Прямая адресация возможна только к памяти данных, при этом в команде обращения к памяти данных содержится адрес ячейки памяти. Косвенная адресация памяти данных и программ использует индексные (I) регистры формирователей адресов и регистры-модификации. В команде указываются номера соответствующих

регистров. При непосредственной адресации в команде содержится операнд, например, константа.

#### 1. Команды управления программой

К этим командам относятся команды перехода JUMP, вызова подпрограммы CALL, возврата из подпрограммы RTS и прерываний RTI, цикла DO UNTIL.

В команде JUMP записан 14 разрядный адрес перехода. Примеры: JAMP fir\_start – переход по адресу метки firststart. IF [cond] JUMP; CALL (16); если выполняется условие, то происходит передача перехода по адресу 16. При косвенной адресации адрес перехода находится в одном из адресных регистров DAG2 14, 15, 16, 17. Например, JUMP (14); – переход по адресу 14.

CALL (адрес) обращение к подпрограмме. Адрес находится в самой команде и загружается в следующем цикле в счетчик команд PC и в стек. Для возврата из подпрограммы стека в PC восстанавливается значение адреса. RTI – команда возврата из подпрограммы. CNTR – счетчик циклов. Например, CNTR (10) – число повторений цикла

Команда цикла DO UNTIL при использовании счетчика CNTR. При этом сначала устанавливается длина циклического буфера RGLO = 10; происходит загрузка указателя 1 элемента циклического буфера IO = ^data\_buffer; устанавливается регистр-модификатор MO = 1; устанавливается число повторений цикла CNTR = 10. Команда DO UNTIL GE – повторить цикл, пока не обнулится счетчик.

#### 2. Команды прерываний

IDLE – ожидание прерываний. При этой команде процессор входит в состояние пониженного потребления до получения сигнала прерывания.

INIT – инициализация прерывания.

IMASK – разрешение прерываний, например, IMASK = 0\*0018 (разрешение прерывания SPORT0).

RESET – сброс, восстановление исходных значений, регистры периферии устанавливаются в 0.

IRQ – прерывание по последовательному порту.

RTI – возврат из прерывания в главную программу.

NOP – пустой цикл.

### 3. Команды пересылок

Команды пересылок определяют обмен данными между памятью программ и данных и регистрами вычислителя, а также между памятью программ и данных и периферийными устройствами.

Пример.  $AXO = DM(12, MO)$ ,  $AYO = PM(14, M6)$ ; пересылка данных во входные регистры АЛУ из памяти данных и из памяти программ по указанным адресам. Возможны пересылки между регистрами, например,

$TXO = MR1$  – запись в регистр последовательного порта SPORTO операнда из регистра  $MR1$ .  $MRO = 0$  – обнуление регистра результата умножения.

Пример команды вычисления и записи в память  $DM(10, MO) = AR$ ,  $AR = AXO + AYO$ ; при этом сначала производится запись в память, а затем вычисление.

Вычисление с пересылкой регистр-регистр, например,  $AR = AXO + AYO$ ,  $AXO = MR1$ ; в начале цикла происходит сложение значений  $AXO$  и  $AYO$ , а в конце цикла загрузка нового операнда из  $AXO$  в  $MR1$ . Пересылка регистр-регистр может быть из/в любого регистра АЛУ, множителя и сдвигателя, кроме регистров обратной связи  $AF$ ,  $MF$  и регистра  $SB$ .

### 4. Команды вычислений

По этим командам производятся вычисления в АЛУ, множителе/ аккумуляторе и сдвигателе. Например, команда сложения с переносом в АЛУ  $AR = AXO + AYO + C$  – в регистр  $AR$  записывается сумма  $AXO$  и  $AYO$  плюс значение флага переноса  $C$ . Команда  $MAC$  множителя запишется как  $MR = MR + MXO * MYO (SS)$ . В регистр результата записывается сумма предыдущего значения с произведением операндов входных значений  $X$  и  $Y$ . Модификатор в скобках означает, что операнды знаковые. Возможны 5 вариантов модификаторов, например,  $(US)$  умножается незнаковый операнд на знаковый. Если в команде имеется модификатор  $(RND)$  то после умножения-суммирования производится округление результата, то есть результат берется из регистра  $MR1$ , а разряды  $MRO$  не учитываются. По условной команде  $IF MV SAT MR$  – если произошло переполнение (появился флаг  $MV$ ), то проводится операция насыщения (ограничения) результата. При этом в регистр  $MR$  записывается максимальное положительное или отрицательное число в зависимости от знака результата.

Приемником для всех операций сдвигателя является регистр результата SR, кроме операции определения экспоненты (порядка), где приемник это регистры SE или SB. SI – входной регистр содержит операнд, который сдвигается или нормализуется.

#### Контрольные вопросы

1. Какие основные блоки входят в ЦСП ADSP2181?
2. Перечислите шины процессора и их назначение.
3. Каковы области применения процессора ADSP2181?
4. Назначение и основные операции АЛУ.
5. Зачем нужен регистр AF?
6. Поясните назначение флагов АЛУ. Куда они записываются?
7. Как реализуется операция MAC в умножителе-аккумуляторе?
8. Как и зачем производится операция ограничения (насыщения) и округления?
9. Как производится умножение знаковых и незнаковых операндов?
10. Каковы основные операции кольцевого сдвигателя?
11. Как задается величина сдвига и как делается сдвиг?
12. Поясните смысл команд (H/L/O).
13. Как происходит преобразование числа с фиксированной точкой в число с плавающей точкой?
14. Назначение и состав формирователей адресов.
15. Как вычисляется следующий адрес?
16. В чем разница линейной и циклической адресации?
17. Как и с чего производится начальная загрузка формирователя адресов?
18. Для чего и когда делается операция перестановки бит?
19. Перечислите основные функции устройства управления программой. Что поступает на шину PMA?
20. Основные узлы устройства управления программой и их назначение.
21. Каково назначение регистров ASAT, MSTAT, SSTAT?
22. Что входит в периферию ADSP2181?
23. Назначение и структура последовательных портов. Для чего нужна двойная буферизация?
24. Поясните роль прямого доступа к памяти (DMA).
25. Назначение и структура таймера.
26. Что входит в систему команд процессора?
27. Перечислите команды управления программой.
28. Перечислите команды пересылок и прерывания.
29. Примеры команд вычислений.

## Глава 4. Реализация на процессоре ADSP 2181 типовых устройств обработки сигналов

### 4.1. Нерекурсивный цифровой фильтр

Достоинством нерекурсивных фильтров являются: абсолютная устойчивость, простота и возможность получения линейной фазовой характеристики [2, 3]. При реализации фильтра во временной области вычисляется дискретная свертка

$$Y(kT) = \sum h(nT) * x[(k-n)T],$$

где  $h(nT)$  – коэффициенты импульсной характеристики фильтра, определяющие АЧХ и ФЧХ фильтра;  $n$  меняется от 0 до  $N-1$ , где  $N$  – порядок фильтра.

Например, для  $N=3$  имеем разностное уравнение

$$y(n) = h(0)x(n) + h(1)x(n-1) + h(2)x(n-2),$$

где  $x(n)$  представляет последнюю входную выборку,  $x(n-1)$  – предпоследнюю входную выборку, а  $x(n-2)$  – предпредпоследнюю входную выборку.

Схема фильтра показана на рис 1.8. Данные на вход фильтра поступают с АЦП через последовательный порт ЦСП. Для хранения данных и коэффициентов фильтра выделяются  $N$  ячеек памяти данных и  $N$  ячеек памяти программ. В процессорах, имеющих две памяти данных, коэффициенты записываются во вторую память данных. Во время инициализации значения ячеек памяти данных обнуляются. Для получения выходного отсчета необходимо  $N$  раз выполнить типовую операцию МАС.

На сигнальных процессорах фильтры реализуются в программном виде [5, 6, 8]. Программа состоит из трех частей: главного модуля, подпрограммы инициализации `flt.init` и подпрограммы фильтрации (см. рис. 4.1). Главный модуль обеспечивает синхронизацию ввода/вывода данных через последовательный порт процессора. При этом задаются частоты тактовой и кадровой синхронизации, получаемые делением внутренней тактовой частоты ГТИ. Возможно использование внешней синхронизации. Ввод данных



выполняется по сигналам прерываний. В главном модуле программы `flt_routine` объявляются адреса подпрограмм `proc_start` и `init_start`, производится установка регистров последовательного порта и загрузка векторов прерываний и вызывается подпрограмма инициализации. В подпрограмме инициализации объявляется метка входа в подпрограмму `init_start`, объявляются буферы данных `data_buffer` и коэффициентов `coef_buffer`. Входные выборки хранятся в памяти данных, а коэффициенты фильтра в памяти программ. Также задаются возвращаемые значения – состояние регистров начала работы подпрограммы, изменяемые параметры – состояние регистров, используемых в подпрограмме. Для обработки изменений в буфере данных используется циклическая адресация.

Далее вызывается подпрограмма фильтрации `flt_proc`, выполняющая базовую операцию свертки для очередного отсчета входных данных с приемного регистра порта RXO. Выходной отсчет

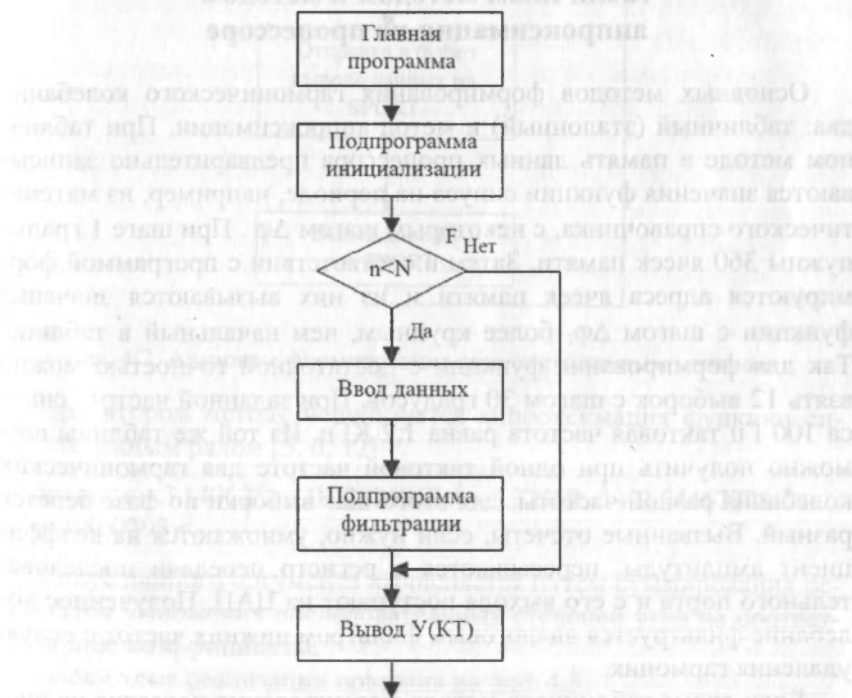


Рис. 4.1. Блок-схема реализации фильтра

данных записывается в передающий регистр последовательного порта ТХО. Все коэффициенты и значения данных при фильтрации представляются в формате 1.15. В программе содержатся параметры вызова: начальный адрес данных, порядок фильтра (длина цикла), начальный адрес буфера коэффициентов и значения регистров-модификаторов; возвращаемые значения и изменяемые регистры SI, MXO, MYO, MR, CNTR. Время вычислений для фильтра порядка  $N$  составляет  $(N + 6) T_{ци}$ .

Максимальный порядок фильтра определяется, во-первых, длительностью инструкции, во-вторых, числом входных выборок и требуемым числом операций.

Программа реализации нерекурсивного фильтра приведена в Приложении 2.

### 4.2. Формирователь гармонического колебания табличным методом и методом аппроксимации на процессоре

Основных методов формирования гармонического колебания два: табличный (эталонный) и метод аппроксимации. При табличном методе в память данных процессора предварительно записываются значения функции синуса на периоде, например, из математического справочника, с некоторым шагом  $\Delta\varphi$ . При шаге 1 градус нужны 360 ячеек памяти. Затем в соответствии с программой формируются адреса ячеек памяти и из них вызываются значения функции с шагом  $\Delta\varphi_1$  более крупным, чем начальный в таблице. Так для формирования функции с достаточной точностью можно взять 12 выборок с шагом 30 градусов. При заданной частоте синуса 100 Гц тактовая частота равна 1,2 КГц. Из той же таблицы возможно получить при одной тактовой частоте два гармонических колебания разной частоты: для этого шаг выборки по фазе берется разный. Вызванные отсчеты, если нужно, умножаются на коэффициент амплитуды, пересылаются в регистр передачи последовательного порта и с его выхода поступают на ЦАП. Полученное колебание фильтруется аналоговым фильтром нижних частот с целью удаления гармоник.

Блок-схема табличного метода формирования показана на рис. 4.2.



Рис. 4.2. Алгоритм формирования синуса табличным методом

При втором методе используется аппроксимация функции синус степенным рядом [5, 6, 12]

$$y = \sin x = 3,140625x + 0,02026367x^2 - 5,32519x^3 + 0,54467778x^4 + 1,800293x^5.$$

Синус данного аргумента вычисляется путем суммирования результатов умножения последовательных степеней угла на соответствующие коэффициенты.

Блок-схема реализации показана на рис. 4.3.

Вычисления по формуле выполняются для первого квадранта, поскольку при этом обеспечивается максимальная точность соот-

ветствующих значениям аргумента от 0 до 0.5 или от 0. до 90°. Для других квадрантов используются соотношения:  $\sin(-\varphi) = -\sin \varphi$ ,  $\sin(180 - \varphi) = \sin \varphi$ ,  $\sin(\varphi - 180) = -\sin \varphi$ .

В первом квадранте обеспечивается максимальная точность, в других квадрантах при прямом вычислении точность хуже, поэтому для расчетов  $\sin x$  в 2, 3 и 4 квадрантах используется указанное соотношение.

В программе имеются условные переходы определения квадранта и вычисление соответствующих значений. Значения  $x$  представляются в дробном формате 1.15. Углу 180° в этом формате соответствует максимальное положительное значение  $x$  0x7FFF (примерно единица), углу -180° максимальное отрицательное число  $x$  0x8000 (минус 1). Коэффициенты полинома инициализируются в буфере `sin_coef` памяти данных в формате 4.12, содержащем разряд знака, 3 разряда целой части и 12 разрядов дробной – мантиссы. Процессор выполняет вычисления значений функции с погрешностью до двух младших разрядов.

Подпрограмма вычисления синусоидальной функции методом аппроксимации приведена в Приложении 3. Для реализации функции синус используется 25 циклов. Частота дискретизации выбирается из условия  $F_d = 60 f_c$ . Точность реализации функции синус определяется, во-первых, числом членов ряда, во-вторых, разрядностью коэффициентов ряда, и точностью вычислений в процессоре, зависящей от разрядности.

### 4.3. Вычислитель быстрого преобразования Фурье на процессоре

При спектральном анализе и цифровой фильтрации широко используется алгоритм быстрого преобразования Фурье (БПФ) [2, 3]. Входная выборка из  $N$  временных отсчетов с выхода АЦП может быть действительной и комплексной. Комплексная выборка представляет собой, например, отсчеты косинусной и синусной составляющих с выхода квадратурного приемника. В данной главе рассматривается реализация БПФ по комплексной выборке.

Следует заметить, что результаты вычислений по алгоритму БПФ такие же, как и при вычислениях ДПФ, однако, БПФ более эффективен по числу выполняемых типовых операций. Наиболее

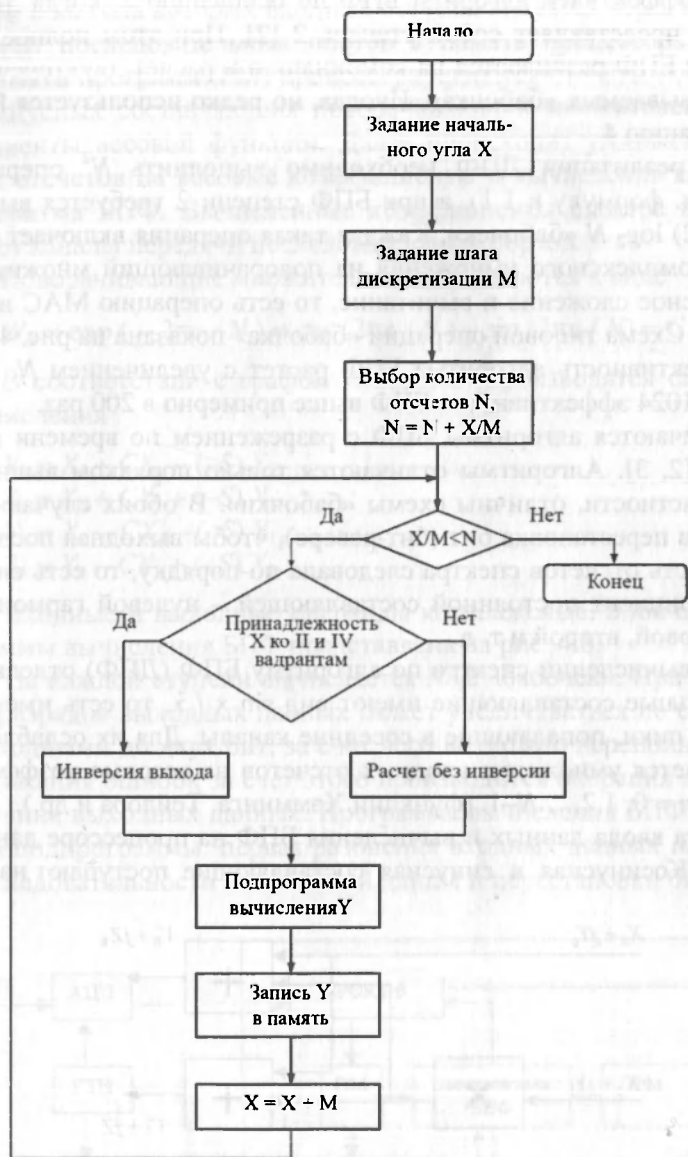


Рис. 4.3. Блок-схема реализации синусоидального колебания методом аппроксимации

прост и эффективен алгоритм БПФ по основанию 2, когда длина выборки представляет собой степень 2 [2]. При этом полное вычисление БПФ разбивается на комбинацию и расчет двухточечных ДПФ, называемых «бабочка». Иногда, но редко используется БПФ по основанию 4.

При реализации ДПФ необходимо выполнить  $N^2$  операций МАС (см. формулу в 1.1), а при БПФ степени 2 требуется вычислить  $(N/2) \log_2 N$  «бабочек». Каждая такая операция включает операцию комплексного умножения на поворачивающий множитель, комплексное сложение и вычитание, то есть операцию МАС и вычитание. Схема типовой операции «бабочка» показана на рис. 4.4.

Эффективность алгоритма БПФ растет с увеличением  $N$ , Так, при  $N = 1024$  эффективность БПФ выше примерно в 200 раз.

Различаются алгоритмы БПФ с разрежением по времени и по частоте [2, 3]. Алгоритмы отличаются только порядком вычислений, в частности, отличны схемы «бабочки». В обоих случаях необходима перестановка бит (бит-реверс), чтобы выходная последовательность отсчетов спектра следовала по-порядку, то есть сначала коэффициент постоянной составляющей – нулевой гармоники, затем первой, второй и т. д.

При вычислении спектра по алгоритму БПФ (ДПФ) отдельные спектральные составляющие имеют вид  $\sin x / x$ , то есть имеются боковые пики, попадающие в соседние каналы. Для их ослабления используется умножение входных отсчетов на весовые коэффициенты  $b_n, n = 0, 1, 2, \dots, N-1$ . (функции Хэмминга, Тейлора и др.).

Схема ввода данных и вычисления БПФ на процессоре дана на рис. 4.5. Косинусная и синусная составляющие поступают на два

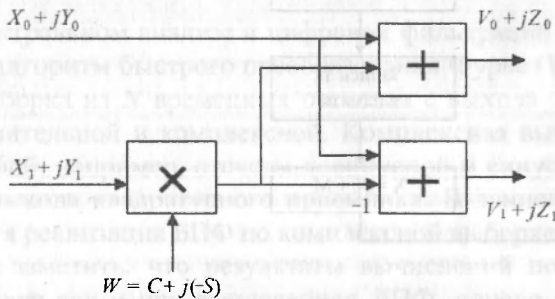


Рис. 4.4. Операция «бабочка»

#### Глава 4. Реализация на процессоре ADSP 2181 типовых устройств

АЦП, с выхода которых входные отсчеты вводятся через приемные каналы последовательных портов в память процессора ( DM ). В памяти программ (PM) процессора хранятся отсчеты косинусных и синусных составляющих поворачивающих множителей и коэффициенты весовой функции. Далее происходит умножение входных отсчетов на весовые коэффициенты и вычисления программе алгоритма БПФ. Вычисленные коэффициенты спектра выводятся через каналы передачи последовательных портов.

Поворачивающие множители представляются в виде

$$W_N = \exp(-2\pi n / N) = \cos(2\pi n / N) - j \sin(2\pi n / N) = C + jS.$$

В соответствии с графом «бабочки» производятся следующие вычисления

$$V_0 = X_0 + CX_1 - (-S) Y_1,$$

$$Z_0 = Y_0 + CY_1 + (-S) X_1,$$

$$V_1 = X_0 - CX_1 - (-S) Y_1,$$

$$Z_1 = Y_0 - CY_1 + (-S) X_1.$$

Входные и выходные операнды комплексные. Блок-схема программы вычисления БПФ представлена на рис. 4.6.

На каждой ступени вычисляется  $N/2$  «бабочек». При этом число разрядов выходных данных может увеличиваться по сравнению с входными до двух бит, за счет чего возможно переполнение. Для избежания ошибок за счет этого производится операция масштабирования выходных данных. Программа вычисления БПФ содержит три подпрограммы: первая разбиения входных данных на две подпоследовательности половинной длины и перестановки бит, вторая

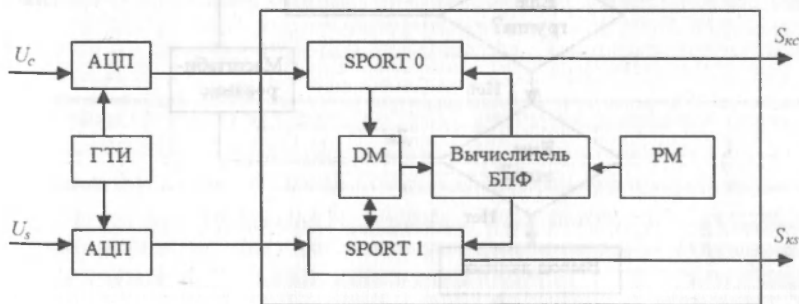


Рис. 4.5. Реализация БПФ на процессоре

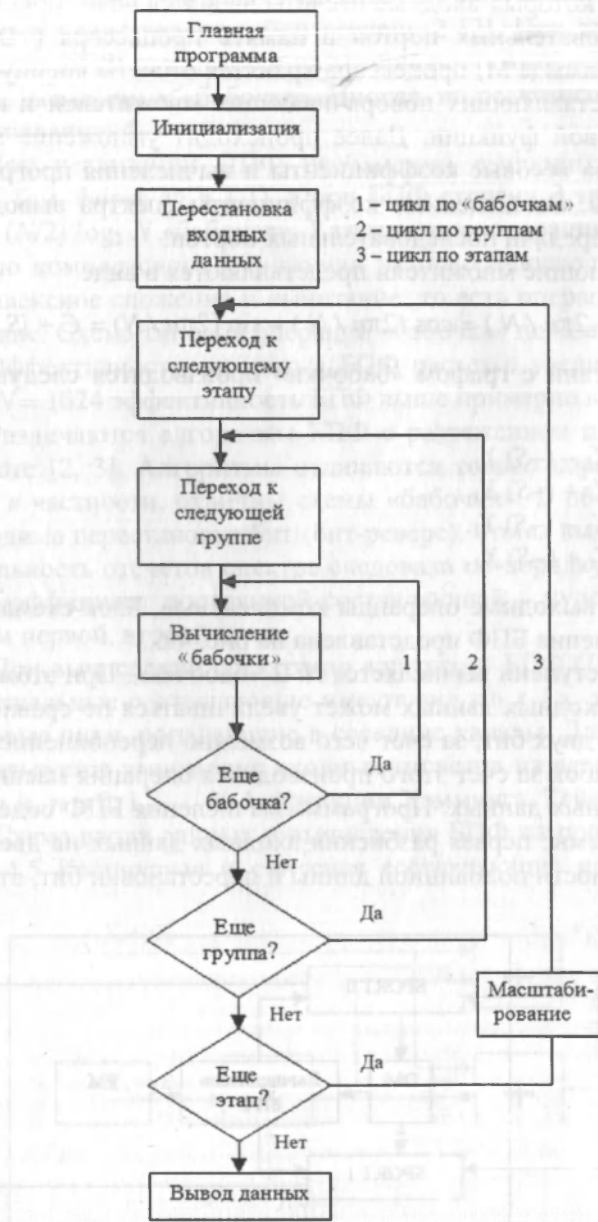


Рис. 4.6. Вычислитель БПФ



собственно вычисления БПФ и третья – масштабирования выходных данных. Изменяемыми величинами являются: число групп, число «бабочек» в группе, поворачивающие множители. Подпрограмма реализации алгоритма БПФ на ADSP 2181 приведена в Приложении 4.

### **Контрольные вопросы**

1. Какова структура программ цифровой фильтрации?
2. Подпрограмма вычисления свертки на ЦСП ADSP2181.
3. Каковы основные команды программы реализации нерекурсивного фильтра?
4. Где хранятся данные и коэффициенты, как формируются их адреса?
5. Как задаются данные и коэффициенты в программе?
6. Какие устройства процессора используются при реализации фильтра на ЦСП?
7. Поясните принцип формирования гармонического колебания табличным методом.
8. Чем определяется точность формирования гармонического колебания табличным методом?
9. Поясните принцип формирования гармонического колебания методом аппроксимации.
10. Поясните вычисления функции  $\sin x$  на процессоре.
11. Чем определяется точность формирования гармонического колебания методом аппроксимации?
12. Каковы основные команды вычисления функции  $\sin x$  на процессоре?
13. Поясните смысл типовой операции «бабочка» при вычислении БПФ.
14. Как реализуется вычисление БПФ в процессоре?
15. Каковы основные команды реализации БПФ на процессоре?
16. Чем определяется время вычисления БПФ при заданной длине выборки?

## Глава 5. Процессоры ADSP Blackfin и ADSP SHARC

### 5.1. Семейство сигнальных процессоров ADSP Blackfin

Фирма Analog Devices выпустила семейство 16-разрядных процессоров с фиксированной точкой на новой технологии 0,18 мкм и с развитой периферией BF531/533/535 Blackfin [A1, A2]. Архитектура ЦСП – модифицированная гарвардская плюс иерархическая структура памяти. Процессор базируется на микросигнальной структуре (MSA), соответствующей комбинации двойного MAC ЦСП и RISC микропроцессора на основе принципа «простые инструкции и много данных» (SIMD). Структурная схема процессора представлена на рис 5.1.

В состав процессора входят:

Ядро процессора, включающее в себя высокоэффективный параллельный вычислитель, адресный блок и устройство управления программой.

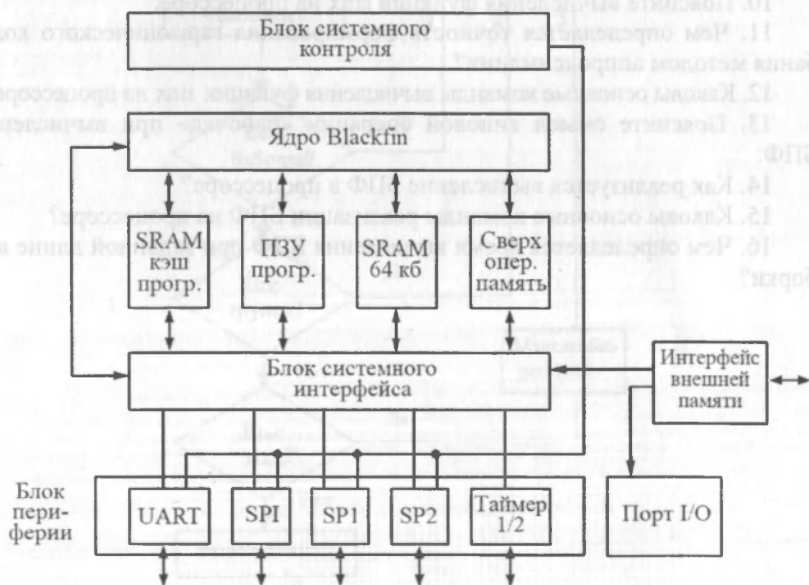


Рис. 5.1. Схема процессоров Blackfin

## Глава 5. Процессоры ADSP Blackfin и ADSP SHARC

Блок памяти (L), состоящий из ПЗУ программ, кэш памяти, SRAM данных 64К×16, сверхоперативной памяти 4К×16); памяти инструкций и данных конфигурируются как оперативная память (RAM), или как КЭШ память.

Периферия, включает блок системного интерфейса, параллельный порт вход/выход, интерфейс внешней памяти, порт ПДП и два таймера, один из которых – сторожевой таймер, следящий за периодом и длительностью тактовых импульсов. Блок системного интерфейса управляет – последовательными портами SPI, SP2 и SPI – последовательным периферийным интерфейсом и таймером, а также портом UART – (universal asynchronous receiver/transmitter) – универсальный асинхронный приемо-передатчик, производящий управление последовательными портами, преобразуя поток байтов в асинхронный поток бит и наоборот). В процессор также входят блоки системного контроля. Ядро процессора показано на рис. 5.2.

Вычислительный блок включает два 16 битных умножителя-аккумулятора (MAC), два 40 битных АЛУ, 4 видео АЛУ для обра-

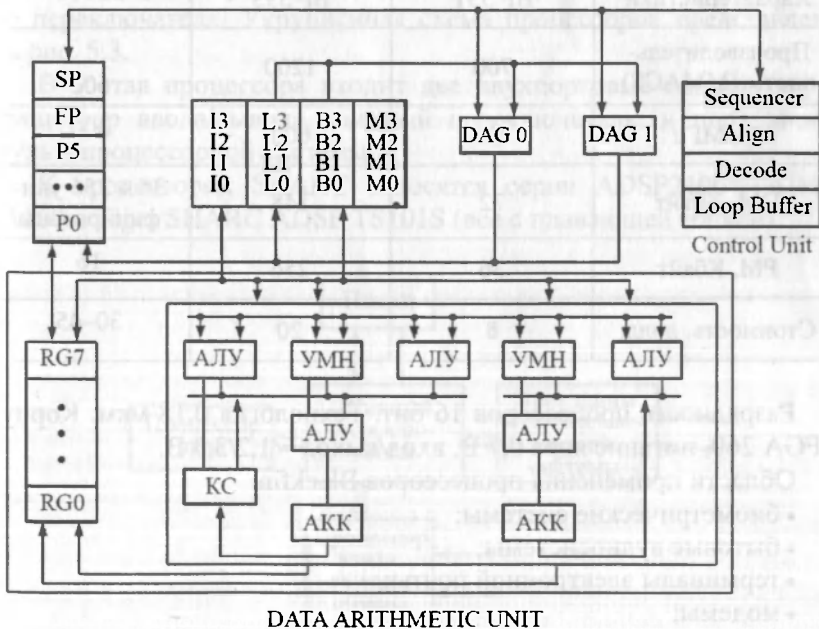


Рис. 5.2. Ядро процессоров Blackfin

ботки видеосигналов и 40 битный кольцевой сдвигатель (КС). Каждый МАС производит умножение  $16 \times 16$  за время одной инструкции. Два 40 разрядных АЛУ накапливают два 40 разрядных числа или четыре 16 битных.

Данные представляются 8-, 16- или 32-битными операндами. Для 8-разрядных данных реализуется алгоритм обработки RGB-пиксел. Также имеются специальные команды для обработки видеосигналов. Например, дискретное косинусное преобразование, суммирование абсолютных значений разностей, алгоритмы сжатия видеосигналов MPEG2, MPEG4, JPEG.

С вычислителем связан файл регистров R0-R7. Устройство управления программой состоит из устройства контроля следования команд (Sequencer), выравнивателя задержек (Align), декодера и кольцевого буфера (Loop Buffer), регистра указателя стека SP, устройства защиты файлов (FP) и указателей адресов P0-P5.

*Характеристики процессоров*

Характеристики	BF 531	BF 533	BF535
Производительность. (ММАС)	700	1200	600
$F_T$ , МГц	350	600	300
DM, Кбайт	72	148	36 + 256 унифицированы
PM, Кбайт	36	256	16
Стоимость. долл.	8	20	30-45

Разрядность процессоров 16 бит. Технология 0,18 мкм. Корпус RFGA 260, питание ядра 0,7 В, вход/выход – 1,2/3,3 В.

Области применения процессоров Blackfin:

- биометрические системы;
- бытовые аудиосистемы;
- терминалы электронной почты;
- модемы;
- Интернет-применение;
- игровое и обучающее оборудование;

- система глобального местоопределения;
- информационные системы;
- широкополосная передача данных.

Возможна реализация алгоритмов MPEG-4, Windows Media. Подключение линеек видеокодеров/декодеров формата ITU-R656. Высокоскоростные АЦП/ЦАП до 65 MSPS (миллионов отсчетов в секунду).

Архитектура включает три вида операций: пользователя, супервизора и эмуляции. Архитектура оптимизирована под компилятор Си.

Analog Devices разработала новую версию отладки программного обеспечения Visual DSP ++3.1 для BF.

### 5.2. Процессоры с плавающей точкой ADSP SHARC [1]

Аббревиатура SHARC означает супергарвардская архитектура. Название «супергарвардская» связано с наличием мощного шинного переключателя. Укрупненная схема процессоров представлена на рис. 5.3.

В состав процессора входят две двухпортовые памяти, ядро, процессор ввода/вывода, шинный переключатель, и порт шины мультипроцессорной системы.

К процессорам SHARC относятся серии ADSP2106\*, ADSP 21160 и Tiger SHARC ADSP TS101S (все с плавающей точкой).



Рис. 5.3. Процессоры семейства SHARC

На рис. 5.4 представлена схема процессора ADSP2106\*. В состав процессора входят две двухпортовых памяти, процессорное ядро, процессор ввода/вывода и обмена данными и порт шины мультипроцессорной системы для подключения стандартных интерфейсов и шинный переключатель. К основным характеристикам процессора относятся:

$t_n = 25$  нс ,  $F_T = 40$  МГц.

Число разрядов  $m = 32$ .

Память  $2*4$  Мбит (65).

Для разных модификаций:

$2*2$  Мбит (62);

$2*0,5$  Мбит (60).

Технология КМОП, 0,3 мкм.

Корпус 240 отводов.

Питание 5 В, 3,3 В.

Стоимость до 10 долл. (минимальная).

Применяются в радиомодемах, для сотовой связи, для мультимедиа, в базовых станциях и для радиолокации и радионавигации и др.

В отличие от процессора ADSP2181 нет шины результата, но имеется общий файл регистров. В программе указывается какие регистры используются, например, при умножении и т. д. Все вычислители связаны между собой, поэтому необходимости в шине результата нет. Имеется КЭШ память, ускоряющая обмен командами. Шинный переключатель коммутирует шины команд и данных. Управляющий порт осуществляет обмен данными и командами и адресами через стандартные интерфейсы.

Процессор вход/выход I/O управляет работой порта ПДП, последовательных портов и линков. Имеется шесть портов – линков по 40 Мбит/с, всего 240 Мбит/с для организации многопроцессорных систем.

Одновременно может производиться одно умножение, одно сложение, чтение из памяти, обновление 2-х адресов ячеек, вычитание.

Активизируются последовательные порты, затем порты-линки, в блоке ПДП обновляются 2 адреса ячеек памяти.

Адресация: регистровая адресация, прямая адресация, косвенная адресация, непосредственная адресация. Могут быть 3 формата данных: целочисленные, беззнаковые, знаковые и с плавающей точкой.

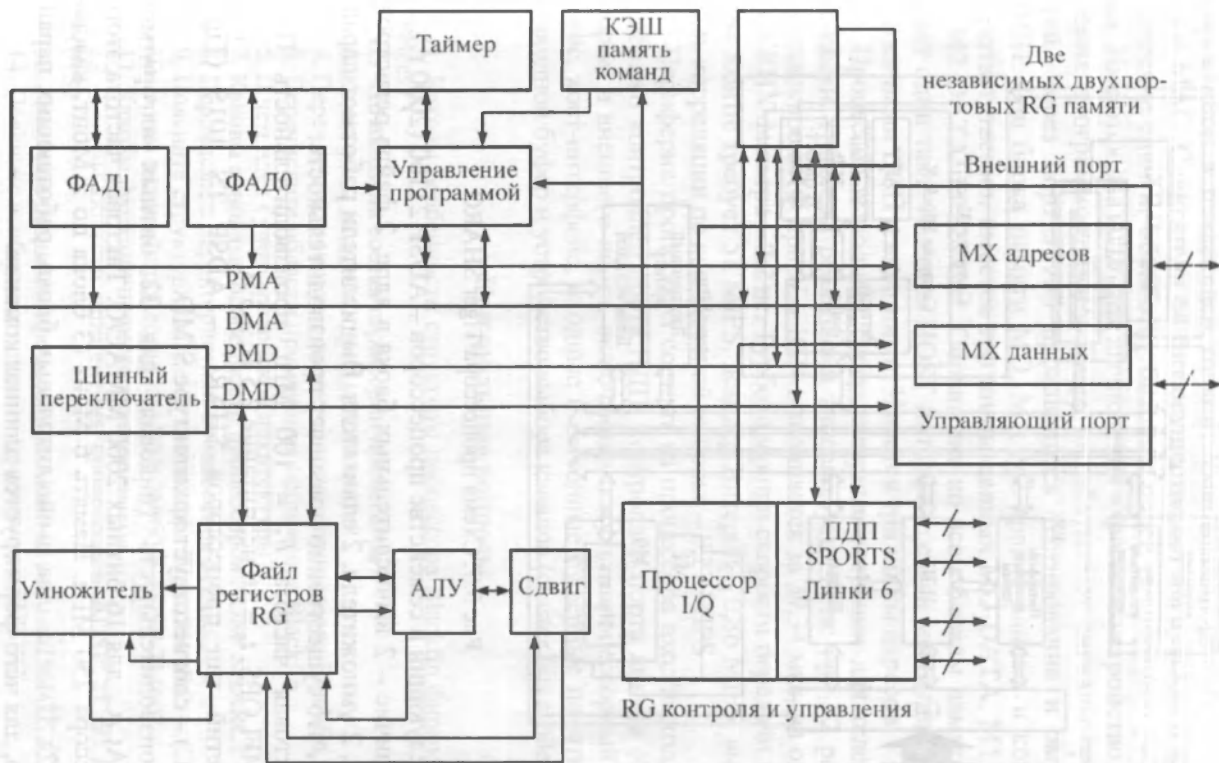


Рис. 5.4. Схема процессора ADSP 2106\*

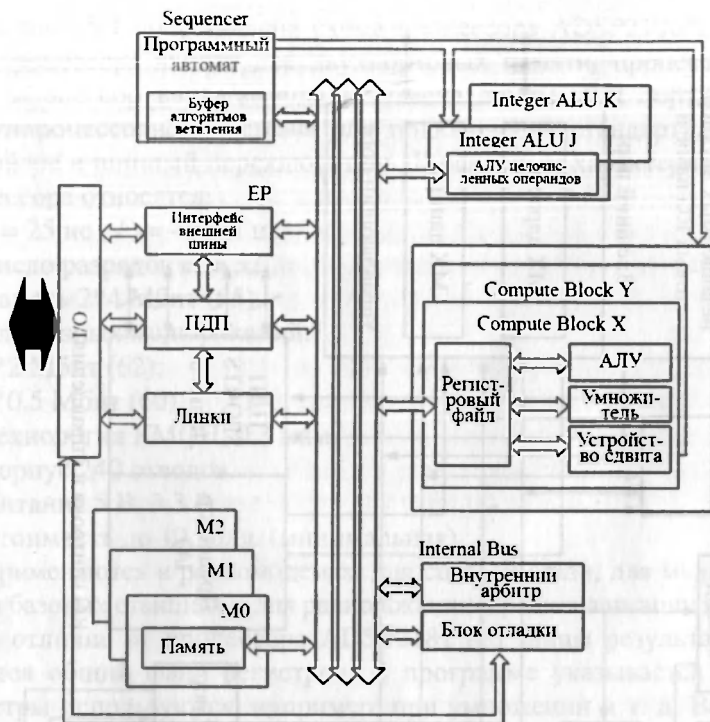


Рис 5.5. Схема процессора Tiger SHARC

Следующий в семействе процессоров – ADSP 21160 (2000 г.).

Отличие – 2 вычислительных блока в ядре, 2 файла регистров, 2 АЛУ, 2 умножителя, 2 сдвигателя. Вычислители работают параллельно, что существенно повышает производительность.

Тактовая частота  $F_T = 100$  МГц, производительность  $\Pi = 600$  MFLOPS,  $t_n = 10$  нс, питание 2,5 В; 3,3 В.

Третий тип процессоров SHARC – ADSP TS 101S (Tiger SHARC) – соответствует архитектуре SIMD.

Производительность (пиковая) для 32 битных алгоритмов 500 MAC/C, для 16 битных 2000 MMAC/C. Тактовая частота этого процессора 250 МГц. Память 6 Мбит (3 блока по 2 Мбит, каждая 64К×32). Имеется два вычислительных блока, работающих параллельно, для чего формируется длинная команда.

Структурная схема процессора показана на рис. 5.5.



АЛУ производят стандартные арифметические операции с фиксированной и плавающей точкой. Умножители реализуют операцию МАС. Сдвигатели 64 бит осуществляют логические и арифметические сдвиги, обработку битовых полей и операцию распаковки. Имеются два АЛУ для вычисления адресов и устройство управления программой. Три таймера вырабатывают сигналы прерываний. Через порт JTAG производится тестирование и эмуляция ЦСП. Три блока памяти М0, М1, М2 хранят данные и команды. Соответственно, имеется три шины данных М0 DATA, М1 DATA и М2 DATA по 128 бит. Обращение ко всем блокам памяти занимает один такт. Имеется HOST интерфейс связи между внутренними шинами ЦСП и внешними. Имеется три шины адресов.

Процессор выполняет суперскалярные операции для телекоммуникаций. Примеры реализации типовых устройств: при 32 разрядах комплексное БПФ при  $N = 1024$  выполняется за 39,34 мкс, на один отвод КИХ фильтра – 2,2 нс. Турбокодер при скорости передачи данных 384 Кбит/с требует 51 MIPS, декодер Витерби – 0,86 MIPS, вычислитель корреляции по комплексной выборке – 0,27 MIPS.

Периферия процессора состоит из процессора вход/выход, включающего контроллер порта ЦДП: контроллер порта линков (4 линк-порта) и внешнего порта, включающего мультипроцессорный интерфейс, хост-интерфейс, входные и выходные буферные памяти FIFO, выходной буфер и устройство выбора каналов (cluster bus arbiter).

### Контрольные вопросы

1. Каковы особенности структуры процессора Blackfin фирмы Analog Devices?
2. Что входит в состав периферии процессора?
3. Какие узлы входят в ядро ЦСП Blackfin?
4. Перечислите основные характеристики процессора Blackfin.
5. Перечислите области применения процессора Blackfin.
6. В чем идея супергарвардской архитектуры?
7. Каковы основные особенности процессоров ADSP SHARC?
8. Перечислите характеристики процессоров SHARC.
9. Поясните структурную схему процессора Tiger SHARC.
10. В чем разница процессоров ADSP 21060 и ADSP 21160?
11. Каковы функции процессора вход/выход I/O?
12. Каковы области применения процессоров ADSP 21160 и ADSP TS101S?
13. Поясните различия процессоров ADSP 21060 и ADSP TS101S.
14. Поясните особенности памяти процессоров SHARC.
15. Для чего нужны порты-линки?

## Глава 6. Цифровые процессоры и микроконтроллеры фирмы Motorola

### 6.1. Классификация и особенности процессоров Motorola [A3]

Процессоры классифицируются по числу разрядов на 16 разрядные и 24 разрядные с фиксированной точкой и 32 разрядные с плавающей точкой, как показано на рис. 6.1. На рисунке также приведены обозначения ЦСП.

К особенностям процессоров можно отнести:

1. Наиболее распространенные процессоры имеют 24 разряда, за счет чего увеличены точность вычислений и динамический диапазон, что существенно при применениях.

2. Имеются две памяти данных и память программ, что позволяет вызывать сразу два операнда и не занимать память программ данными.

3. В состав вычислителя входят два аккумулятора, что существенно при обработке сигналов квадратурных составляющих в системах подвижной связи.

4. Разрядность адресов 16 бит, а данных 24.

5. За один командный цикл производятся выборка двух операндов, две параллельных пересылки и выполнение вычисления, в последних модификациях одновременно делаются два вычисления.

6. Циклы DO реализуются аппаратно.

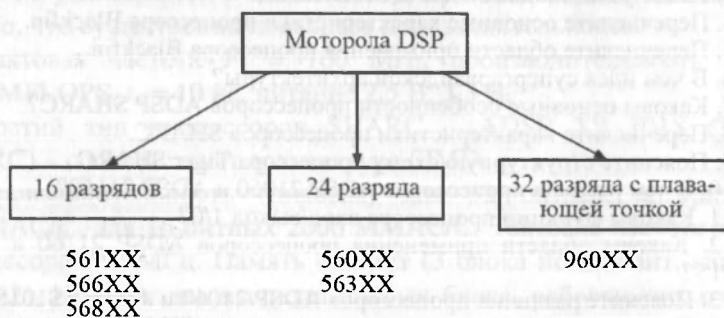


Рис. 6.1. Классификация сигнальных процессоров фирмы Motorola

7. Модульная, бит-реверсная и косвенная адресация.
8. В ряде модификаций в памяти процессора записана таблица синусов.
9. Глобальная шина данных GDB для связи с периферией и загрузки формирователей адресов.
10. В составе ЦСП имеется управляющий HOST (ХОСТ) порт.

### **6.2. Процессор DSP 56300**

DSP 56300 относится к семейству 24 разрядных процессоров с фиксированной точкой. Структурная схема DSP 56300 представлена на рис. 6.2.

В состав ЦСП входят ядро, две памяти данных X и Y, память программ и периферия, соединенные рядом шин: адресов данных XAB, YAB и программ PAB, шин данных XDB, YDB, GDB и команд PDB, объединенные во внутреннюю шину данных. Имеются две внешние шины адресов и данных, управляемых шинными переключателями, позволяющими подключить внешние шины к любой из внутренних.

Периферия включает:

- HOST порт для работы с управляющим микроконтроллером.
- Два последовательных порта – синхронный и асинхронный.
- Два таймера.
- Устройство связи (логика шины) с контроллером MC6800, переключатели внешней шины адреса и данных и устройство управления шинами.

Кроме того, имеются генератор тактовых импульсов, контроллер прерываний, устройство контроля выполнения программы. Для связи с управляющим микроконтроллером в процессоре имеется HOST порт.

Схема HOST порта процессора приведена на рис. 6.3.

Возможны три типа пересылок данных:

1. Из процессора через интерфейс.
2. Извне в процессор.
3. Из шины МК – в коммутатор управления процессором.

Особенность интерфейса – двойная буферизация (скорость обмена в два раза выше, так как имеются 2 регистра, соединенные последовательно).

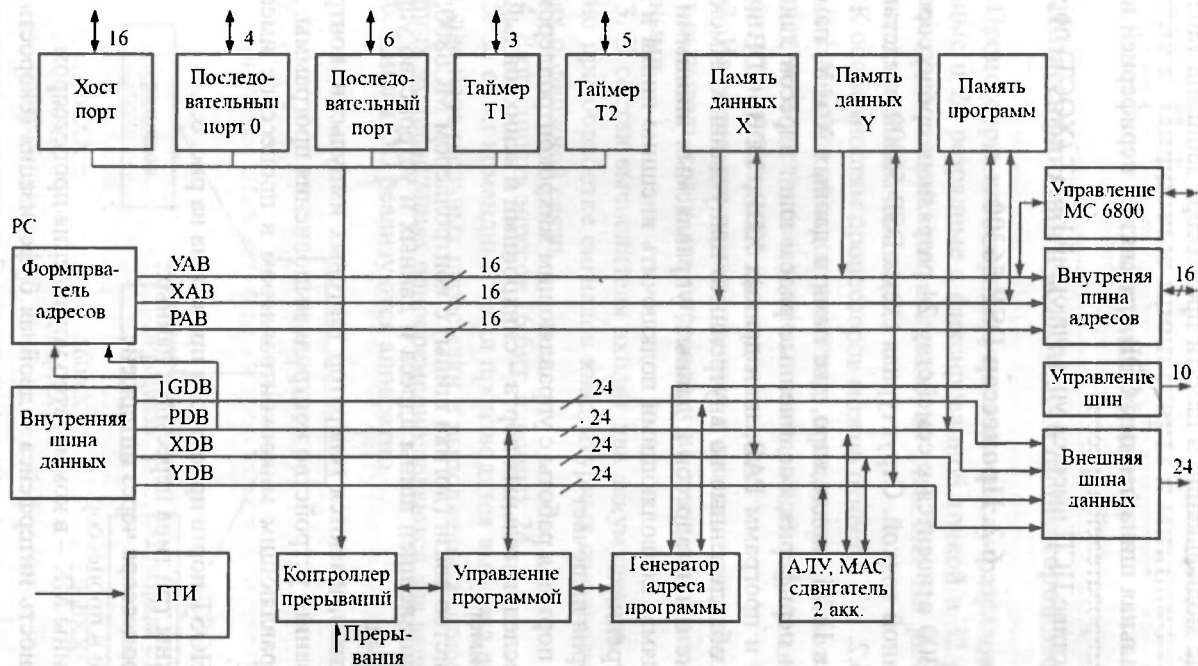


Рис. 6.2. Структурная схема процессора

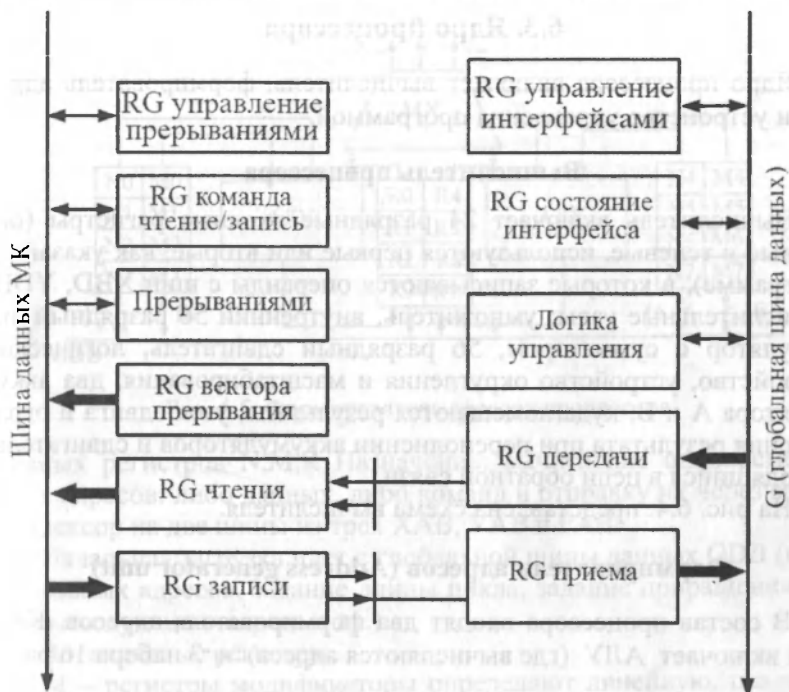


Рис. 6.3. HOST порт процессора

### Характеристики процессора

1. Производительность 40 MIPS.
2. Длительность цикла 25 нс,  $F_t = 40$  МГц.
3. Памяти данных X, Y состоят из RAM 2,5 К×24 р и ROM 8К×24 р.4. Память программ состоит из RAM 0,5×24 р и 64 К×24 р и ROM 37 К×24.
4. Технология 0,65 мкм (в настоящее время 0,25 мкм).
5. Питание 5 В и 2,5 В. Есть режим малого потребления.
6. Корпус содержит 208 выводов.

Области применения процессора: подвижная связь, системы радиодоступа, обработка речи, цифровое вещание, телеметрия, видеоконференции, проводные модемы и факс-модемы, в том числе и стандарта V34, разработанного фирмой.

### 6.3. Ядро процессора

Ядро процессора включает вычислитель, формирователь адресов и устройство управления программой.

#### Вычислитель процессора

Вычислитель включает 24 разрядные входные регистры (основные и теньевые, используются первые или вторые, как указано в программе), в которые записываются операнды с шин XDB, YDB, вычислительные узлы: умножитель, внутренний 56 разрядный аккумулятор с сумматором, 56 разрядный сдвигатель, логическое устройство, устройство округления и масштабирования, два аккумулятора А и В, куда помещаются результаты, узел сдвига и ограничения результата при переполнении аккумуляторов и сдвигатель, находящийся в цепи обратной связи.

На рис. 6.4. представлена схема вычислителя.

#### Формирователи адресов (Address generator unit)

В состав процессора входят два формирователя адресов. Каждый включает АЛУ (где вычисляются адреса) и 3 набора 16 раз-

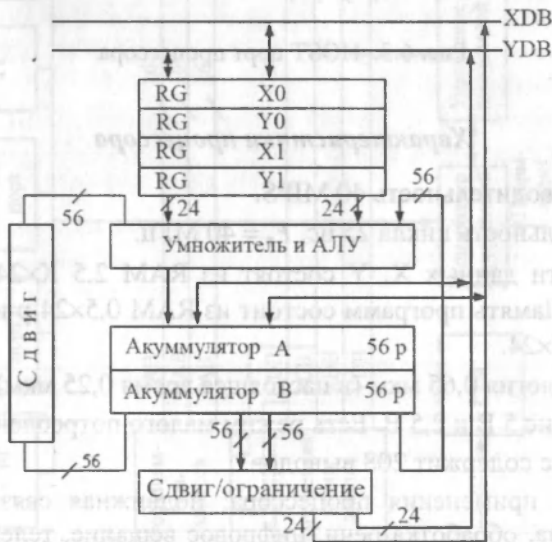


Рис. 6.4. Вычислитель процессора

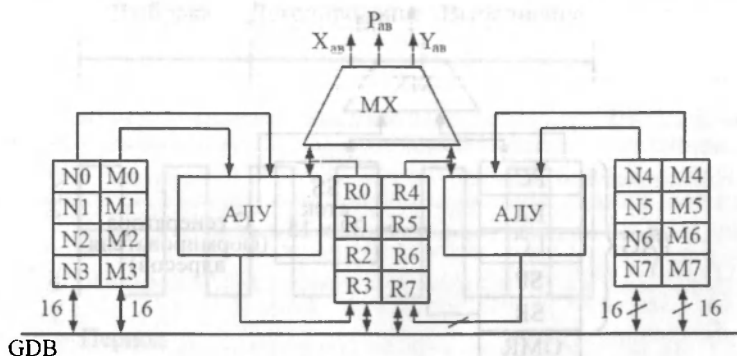


Рис. 6.5. Формирователь адресов процессора

рядных регистров N,M,R Назначение: вычисление одновременно двух адресов: либо данных, либо команд и отправку их через мультиплексор на две шины из трех XAB, YAB и PAB.

Начальная загрузка идет с глобальной шины данных GDB (ввод начальных адресов, задание длины цикла, задание приращения номеру следующего адреса и др.).

R – адресные регистры.

M – регистры модификаторы определяют линейную, циклическую или бит-реверсную адресацию.

N – регистры смещения.

$2^{16} = 65536$  – адресное пространство.

Адресация: линейная, циклическая и бит-реверсная.

АЛУ состоит из 3-х сумматоров.

Смещения – к содержимому регистра  $R_n$  добавляет или вычитает 1, то есть производит линейную адресацию.

Сумматора, который производит сравнение по модулю – при циклической адресации.

Сумматора с обратным переносом обеспечивает перестановку бит адресов в обратном порядке для реализации БПФ.

### Устройство программного управления (PCU)

PCU (Program Control Unit) производит конвейерную обработку команд.

Организация цикла.

Обращение к подпрограммам.

Обслуживание прерываний.

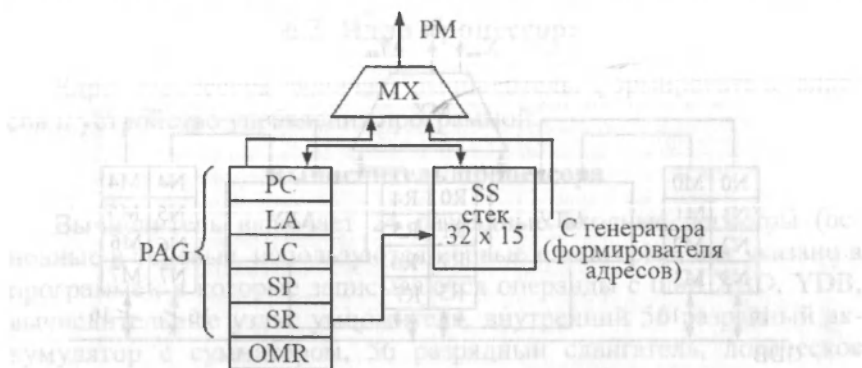


Рис. 6.6. Генератор адреса команды

В состав устройства управления входят:

Генератор адреса команд (PAG).

Контроллер декодирования (PDC).

Контроллер прерываний (PIC).

На рис. 6.б. показана схема генератора адреса команды.

Счетчик циклов (LC), регистр начального адреса (LA).

Программный счетчик (PC).

Регистр состояния. (SR).

Регистр режима обработки (OMR).

Системный стек поддерживает до семи циклов (SS).

Регистр состояния стека (SP).

PCU – блок контроля программ, который управляет пятью состояниями процессора:

– нормальным;

– прерываниями;

– сбросом;

– ожиданием;

– остановкой.

Обработка команд включает три фазы:

– выборка;

– декодирование;

– выполнение.

При выборке вызывается команда по заданному адресу и отправка ее по назначению. Далее производится декодирование команды и третья фаза – определение места операндов и выполнение.





Рис. 6.7. Циклы работы процессора

На рис. 6.7. показаны циклы и время выполнения.

Работа трехуровневого конвейера соответствует нормальной работе процессора.

Контроллер декодирования команд (PDC) генерирует сигналы управления конвейером.

Он включает логическую матрицу декодирования команд, генератор адреса команд в РМ, узел аппаратного цикла DO, узел обслуживания команды REP, Регистр кода условия для условных переходов, узел обслуживания прерываний и регистр, куда записывается 24 разрядная команда.

Контроллер программного прерывания PIC вырабатывает адреса команд прерываний и подтверждения прерывания, далее выполняется программа прерывания. Контроллер имеет четыре вывода: IRQA, IRQB, на которые поступают запросы от внешних аппаратных прерываний, NMI тоже при немаскируемых внешних прерываниях и вывод RESET, на который подается сигнал сброса. Имеется еще ряд команд:

Состояние ожидания – WAIT,

Состояние остановок – STOP (самое низкое потребление), ГТИ – выключен.

## 6.4. Система команд процессоров DSP 56\*\*\*

Основные виды адресации команд – прямая, косвенная, индексная и др [9].

Основные – 6 групп команд.

### 1. Команды пересылок:

MOVE – (пересылка содержимого управления регистра);

MOVC – (пересылка содержимого управления регистра);

MOVEM – пересылка;

LUA – загрузка регистров Rn и Nn в AGU.

### 2. Арифметические:

ADD – сложение ADDL со сдвигом влево, SUB вычитание;

ADS, SBS – сложение, вычитание с переносом;

ADDR – сложение операнда со сдвигом вправо;

SUBR – вычитание операнда со сдвигом вправо;

MPY – умножение;

MAC – умножение с накоплением;

MPYR – умножение с округлением;

ASR – арифметический сдвиг;

ASL – арифметический сдвиг со знаком;

CMP – установка бит (флажков);

TST – проверка аккумулятора на 0;

NEG – отрицательный результат;

ABS – взятие модуля;

NORM – нормализация;

RND – округление;

CLR – очистка.

### 3. Логические команды:

AND, OR, EOR (с отрицанием);

LSL – логический сдвиг влево;

LSR – Логический сдвиг вправо;

### 4. Команды обработки бит:

BCLR – проверка и обнуление бита;

BCHG – проверка и инверсия бит ;

BSET – установка бит;

BTST – проверка бит.

### 5. Команды цикла DO, END, DO.

### 6. Управление.

REP – повторить следующую команду;

JMP – безусловный переход;

JSP – обращение к подпрограмме;

RTS – возврат из подпрограмм;

RTI – возврат из прерывания;

NOP – нет операции;

SWI – программное прерывание;

WAIT – ожидание прерывания;

STOP – останов;

RESET – сброс.

### **6.5. Микроконтроллеры, особенности и применение**

В системах телекоммуникаций микроконтроллеры выполняют функции коммутации, в частности, обеспечения протоколов связи, управления, например, мощностью передатчика базовой станции [10]. Существенна роль МК в устройствах бытовой электроники.

#### **Классификация микроконтроллеров**

По числу разрядов (рис. 6.8).

По назначению:

а) общего назначения (универсальные);

б) для коммутации;

в) для бытовой электроники;

г) для автомобильной электроники;

д) для промышленной электроники;

е) для компьютерной техники.

#### **Характеристики и особенности микроконтроллеров**

МК обычно строятся по гарвардской архитектуре. Основные команды: пересылка данных, передача управления (JMP, RTI), арифметические, логические, битовые и специальные команды, например, WAIT, STOP. Число команд 30 – 65.

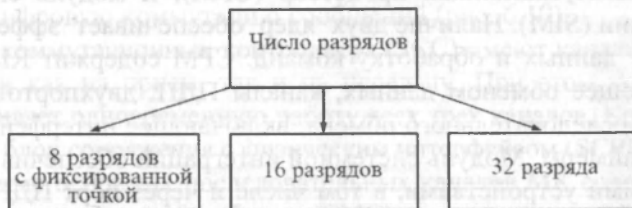


Рис. 6.8. Разрядность МК

### Виды адресации

1. Регистровая адресация. Операнд записывается в регистр адреса или данных.
2. Косвенно – регистровая. Операнд находится в ячейке памяти.
3. Косвенная адресация с инкрементом.

$RGM = 1$ .

4.  $RGM = -1$  (косвенная адресация с декрементом).
5. Прямая адресация. Операнд в ячейке памяти, и адрес задается в команде.
6. Относительная адресация. Операнд находится в ячейке памяти, а адрес – равен числу находящихся в счетчике команд + команда смещения.

### Виды команд

1. Операции пересылки.
2. Арифметические.
3. Логические.
4. Операции сдвига.
5. Сравнение и тестирование.
6. Битовые операции.
7. Операции управления.

Тактовые частоты 16 разрядных МК составляют 0–21 МГц, 32-разрядных от 15 до 40 МГц и выше.

### Телекоммуникационный микроконтроллер

Схема микроконтроллера (МК 68300) показана на рис. 6.9. Данный МК содержит три устройства: центральный процессор (CPU), коммутационный процессор (CPM), и модуль системной интеграции (SIM). Наличие двух ядер обеспечивает эффективную передачу данных и обработку команд. CPM содержит RISC ядро, управляющее обменом данных, каналы ПДП, двухпортовое ОЗУ, каналы последовательного обмена, включающие интерфейсы (SPI, SCI) и таймеры. Модуль системной интеграции обеспечивает связь с внешними устройствами, в том числе и через порт ПДП общего назначения и поддерживает ряд протоколов связи. В состав модуля SIM входят контроллер памяти, управляющий связью с внешней

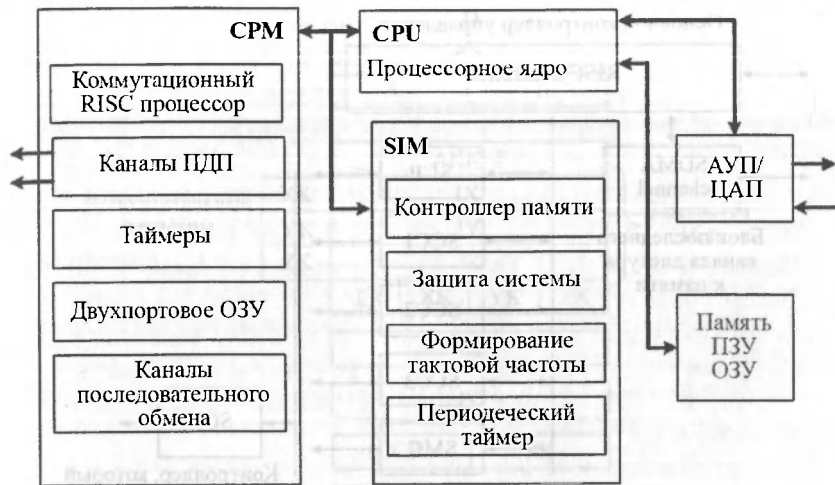


Рис. 6.9. Телекоммуникационный микроконтроллер

памятью, блок защиты от системных ошибок, таймер, осуществляющий прием и выдачу сигналов управления, и формирователь тактовых импульсов.

Коммутационное устройство (СРМ) микроконтроллера показано ниже. Его основное назначение – поддержка различных протоколов передачи данных. Реализация конкретного протокола осуществляется программно RISC ядром. В ПЗУ содержатся программы, поддерживающие протоколы HDLC (управления каналом передачи данных), BISDN (управления широкополосной сетью с комплексными услугами), UART (Universal Asynchronous Receiver/ Transmitter – управления последовательными портами, преобразует поток байтов в асинхронный последовательный поток бит и обратно ) и др. Программы нестандартных протоколов могут записываться в ОЗУ. Протоколы также производят синхронизацию по кадрам, пакетам, цифровую коммутацию сообщений (рис. 6.10).

Три коммутационных контроллера (SCC) имеют каналы доступа к памяти как на прием, так и на передачу. При этом контроллер обеспечивает одновременную работу всех трех каналов. Кроме того, имеется блок сопряжения с физическим интерфейсом (SCPI).

Помимо основных последовательных каналов МК содержит модуль последовательного обмена (SMC), поддерживающий межкристальный цифровой канал (IDL), а также синхронный последователь-



Рис. 6.10. Контроллер управления

ный интерфейс SCP передачи-приема данных. Последовательные каналы ПДП с временным разделением (STDMA) производят передачу данных, если необходимо между SCC и внешней памятью.

Таким образом, RISC контроллер пересылает данные между последовательными каналами и внутренней или внешней памятью, выполняет инструкции от процессорного ядра МК и формирует запросы на прерывания.

Применение коммутационных МК: цифровые АТС, базовые станции сотовой связи, модемы и терминалы.

Программная модель процессорного ядра CPU

Особенностью ядра является аккумуляторная архитектура, отсутствует быстрый умножитель, как показано на рис. 6.11. Все арифметически-логические операции выполняются над операндами, содержащимися в аккумуляторах (А, В, Е, М) или в памяти, откуда вызываются с использованием индексных регистров IX, IY, IZ и дополнительных регистров ХК, YK, ZK.

SP, PC – внутренняя память и счетчик команд.

SK – указатель стека, PK – расширитель программного счетчика.

CCR – регистр признаков (состояния) (C – признак переноса, Z – нулевой результат, N – знак)

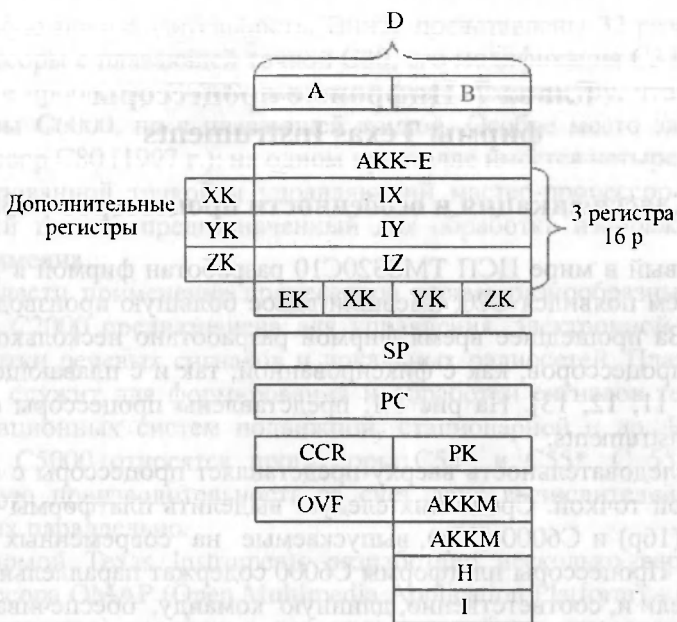


Рис. 6.11. Процессорное ядро МК

I – признак прерывания.

H – перенос.

### Контрольные вопросы

1. Перечислите основные типы процессоров и телекоммуникационных микроконтроллеров фирмы Motorola.
2. Каковы основные особенности процессоров фирмы Motorola?
3. Что входит в состав периферии ЦСП DSP 56300?
4. Какова структура ядра процессора DSP 56300?
5. Поясните принцип действия формирователя адресов.
6. Поясните принцип действия устройства управления программой.
7. Поясните назначение хост порта DSP 56300.
8. Что входит в систему команд DSP 56300?
9. Какова классификация микропроцессоров?
10. Что входит в ядро микроконтроллера M68HC16?
11. Что входит в состав коммутационного контроллера?
12. Перечислите основные операции вычислителя процессора DSP5630.
13. Поясните назначение аккумуляторов.
14. Поясните смысл операции ограничения.

## Глава 7. Цифровые процессоры фирмы Texas Instruments

### 7.1. Классификация и особенности процессоров [A5, A6]

Первый в мире ЦСП TMS320C10 разработан фирмой в 1982 г. [5]. Затем появился C20, имевший вдвое большую производительность. За прошедшее время фирмой разработано несколько поколений процессоров, как с фиксированной, так и с плавающей точкой [6, 11, 12, 13]. На рис 7.1. представлены процессоры фирмы Texas Instruments.

Последовательность сверху представляет процессоры с фиксированной точкой. Среди них следует выделить платформы C2000, C5000 (16p) и C6000 (32p), выпускаемые на современных технологиях. Процессоры платформы C6000 содержат параллельные вычислители и, соответственно, длинную команду, обеспечивающие

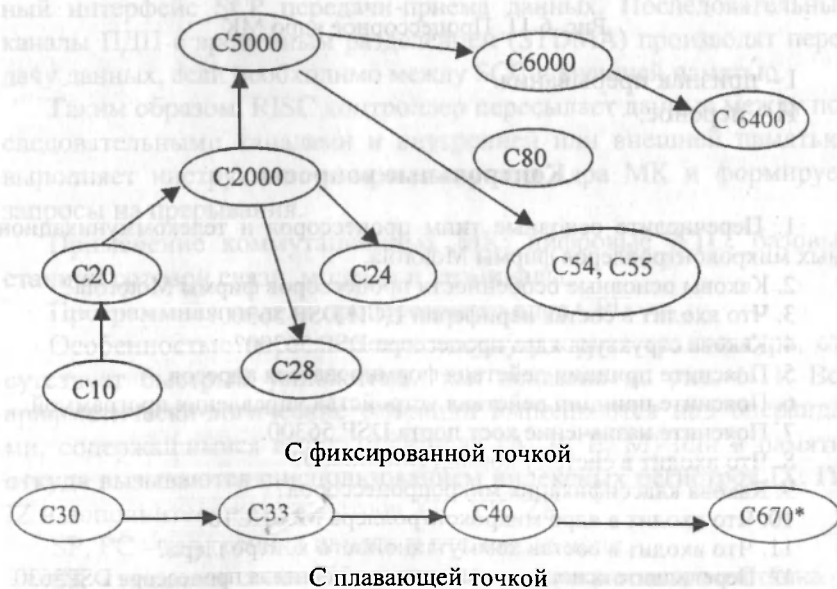


Рис. 7.1. Процессоры фирмы Texas Instruments



огромную производительность. Внизу представлены 32 разрядные процессоры с плавающей точкой С30, его модификация С33 и С40, а также процессор С6700, имеющий ту же архитектуру, что и процессоры С6000, но с плавающей точкой. Особое место занимает процессор С80 (1997 г.): на одном кристалле имеется четыре ЦСП с фиксированной точкой и управляющий мастер-процессор с плавающей точкой, предназначенный для обработки изображений и мультимедиа.

Области применения процессоров весьма разнообразны: платформа С2000 предназначена для управления, электронной почты, обработки речевых сигналов и локальных радиосетей. Платформа С5000 служит для формирования и обработки сигналов телекоммуникационных систем подвижной, стационарной и др. К платформе С5000 относятся процессоры С54\* и С55\*. С 55 имеет большую производительность за счет двух вычислителей, работающих параллельно.

Фирмой Texas Instruments разработано несколько вариантов процессора OMAP (Open Multimedia Application Platform) – составная (открытая) платформа для мультимедийных применений. На одном кристалле находится процессор С54 или С55 и сопроцессор ARM (Advance RISC Machines), с RISC архитектурой, предназначенный для обработки видео изображений (MPEG 4, JPEG, WINDOWS, Media Video) или для обработки аудио сигналов (MPEG 1, AAC, WMA, GSM речевой кодек) или для обработки речи (текст-речь, восстановление речи) и др. В 32 разрядных процессорах платформы С6000 используется архитектура параллельных вычислений, а именно в ядре содержится 8 вычислителей, работающих одновременно. Это потребовало формирования длинного командного слова (Very long instruction) – 8 команд в одном слове. За счет этого получена наибольшая производительность ЦСП. Высокопроизводительные процессоры С6000 позволяют реализовать достаточно сложные алгоритмы ЦОС, например, используются на базовых станциях сотовой связи, для обработки видео изображений, в системах радиолокации и радионавигации. Выпускаются процессоры С6000 как с фиксированной точкой – С6400, так и с плавающей – С6700 по той же архитектуре, но с арифметикой с плавающей точкой. К процессорам с плавающей точкой также относятся 32 разрядные процессоры С33 и С40. С33 изготовлен по современной технологии и предназначен для широкой области применений:

фильтрации, спектрального анализа, систем радиодоступа, обработки речи и др. Транспьютероподобный процессор C40 предназначен для работы в распределенных вычислительных устройствах. Для этого в процессоре имеется 6 коммуникационных портов для связи с другими процессорами.

## 7.2. Процессоры платформы C2000

Процессоры C2000 это интегрированные ЦСП для систем управления, обработки речи и электронной почты. К ним относятся цифровые сигнальные контроллеры TMS320C24\* и TMS32028\*. Первый имеет производительность 40MIPS, второй – 150 MIPS. В них используется 4-ступенчатый конвейер, предусмотрен режим малого потребления. Схема процессора C24\* показана на рис. 7.2. В состав процессора, кроме ядра, входят три памяти: данных, программ, а также Flash память.

Контроллер C24\* 16 разрядный, а C28\* 32-разрядный. Ядро C24\* включает АЛУ, кольцевой сдвигатель, аппаратный стек, аккумулятор, блок регистров и эмулятор. Ядро C28\* дополнено умножителем 32×32 бит, тремя таймерами и портом контроля и загрузки JTAG. Оба контроллера в составе периферии содержат аналого-цифровой преобразователь, что связано с применением в системах управления. Остальные устройства периферии стандартные.

Объемы памяти приведены в табл. 7.1.

Таблица 7.1

Тип памяти	Flash/ ROM	RAM	ROM
C24*	64 Кбит	5 Кбит	512 бит
C28*	256 Кбит	40 Кбит	8 Кбит

В блоке управления C28\* имеются 16 каналов с широтно-импульсной модуляцией, АЦП 12 разрядный с временем преобразования 200 нс.

## 7.3. Процессоры платформы C5000 (C54, C55)

Процессоры платформы C5000 предназначены в основном для применения в системах телекоммуникаций:

Цифровая сотовая связь (стандарты GSM, IS-54, CDMA IS-95).

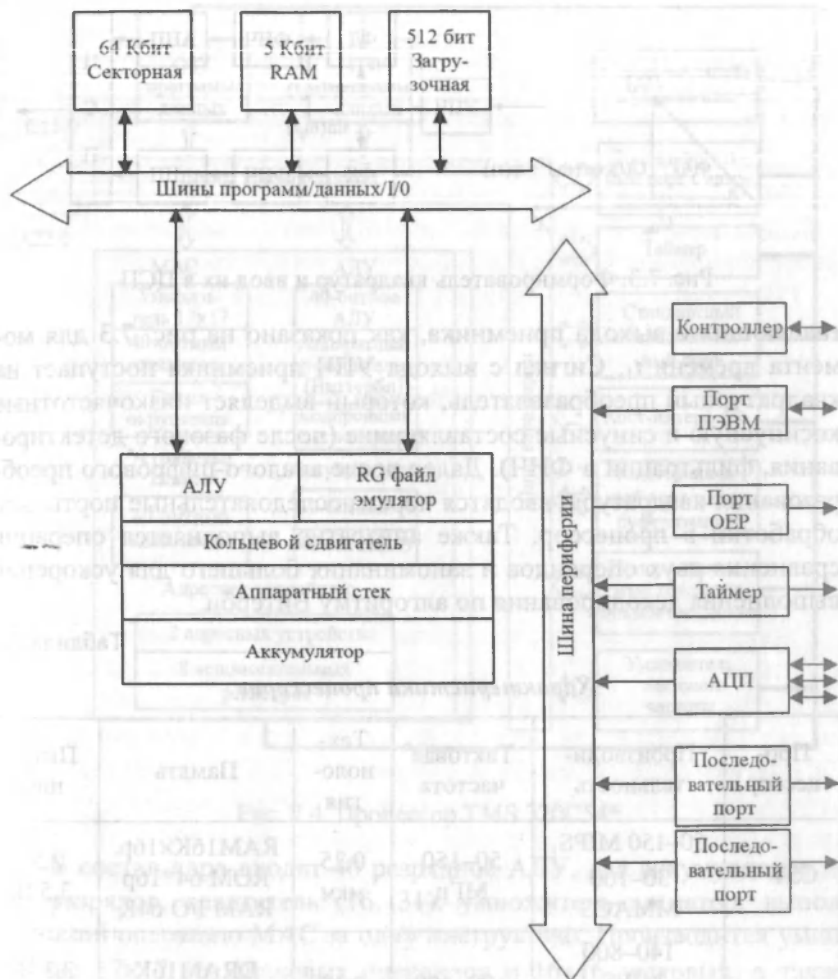


Рис. 7.2. Процессор TMS320C24\*

Радиотелефоны.

Системы радиодоступа (DECT, CT2 и др.).

Беспроводные локальные сети.

Пейджинговая связь.

Модемы радио и проводные (стандарты V32, ISDN).

С учетом применения в связи в процессоре имеются два аккумулятора для одновременной обработки двух квадратурных сос-

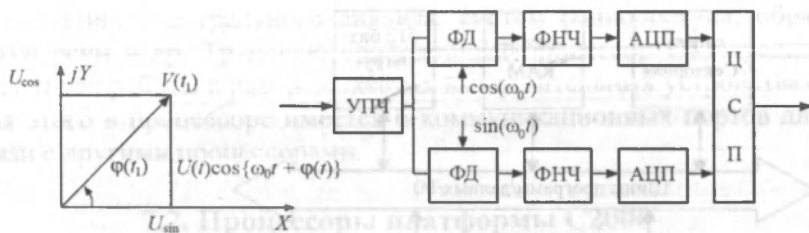


Рис. 7.3. Формирователь квадратур и ввод их в ЦСП

ставляющих с выхода приемника, как показано на рис. 7.3 для момента времени  $t_1$ . Сигнал с выхода УПЧ приемника поступает на квадратурный преобразователь, который выделяет низкочастотные косинусную и синусные составляющие (после фазового детектирования, фильтрации в ФНЧ). Далее после аналого-цифрового преобразования квадратуры вводятся через последовательные порты для обработки в процессор. Также аппаратно выполняется операция сравнения двух операндов и запоминания большего для ускорения выполнения декодирования по алгоритму Витерби.

Таблица 7.2

**Характеристики процессоров**

Про- цессор	Производи- тельность	Тактовая частота	Тех- ноло- гия	Память	Пита- ние
C54**	50–150 MIPS, 30–160 ММАС	50–150 МГц	0,25 мкм	RAM16K×16p. ROM 64*16p RAM I/O 64K	2,5; 3,5 В
C55**	140–800 MIPS, 140–800 ММАС	100–400 МГц	0,18 мкм	DRAM16K SRAM160K ROM256*16	3,3 В, ядро 0,9 В

Для реализации обработки сигналов по стандарту GSM C54 необходимы 12,5 MIPS, по стандарту IS-54 соответственно 12,8 MIPS, речевой кодек занимает 2,3 MIPS. Кроме этих устройств на C54 реализуются эхо-подавитель, многоканальные устройства подвижной связи и вокодеры.

На рис. 7.4 представлена структурная схема процессора C54.

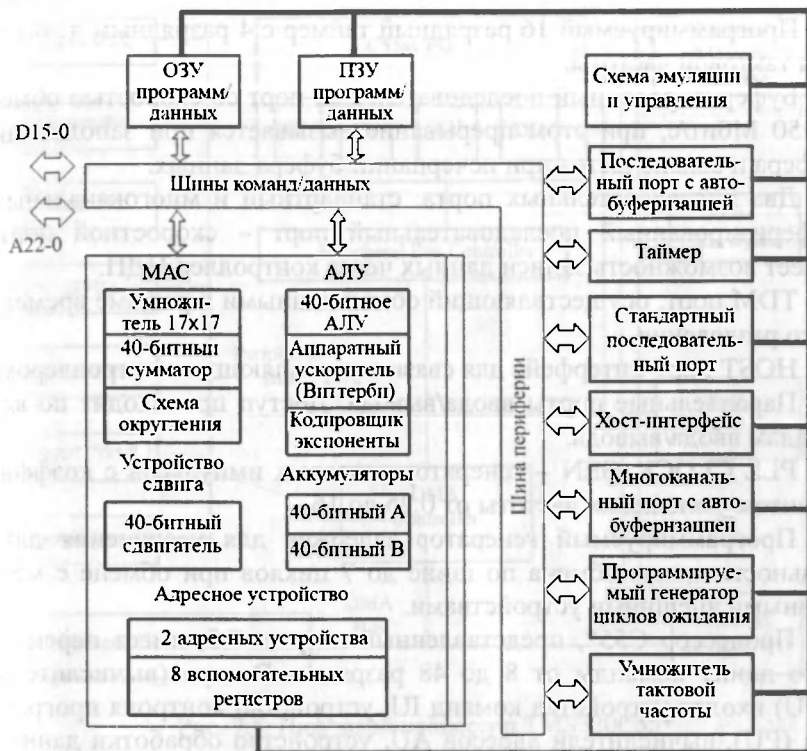


Рис. 7.4. Процессор TMS 320C54\*

В состав ядра входят 40 разрядное АЛУ, два аккумулятора по 40 разрядов, сдвигатель (16, 31), умножитель-сумматор, выполняющий операцию МАС за одну инструкцию. Производится умножение  $17 \times 17$  без знаковых операндов и  $16 \times 16$  знаковых, а также округление и насыщение результатов. Имеется 4 основные шины (3 шины данных и 1 шина команд). Устройство управления программой включает 8 вспомогательных регистров и 2 генератора адресов, счетчик команд и стек счетчика. В одном такте производятся 4 операции с памятью с использованием 4 адресных шин: РАВ, САВ, DАВ, ЕАВ и 4: для передачи команд РВ, чтения данных СВ, DB и записи данных ЕВ. Кроме того, имеется двунаправленная шина для связи с периферией.

Периферия процессора содержит:

Программируемый 16 разрядный таймер с 4 разрядным делителем тактовой частоты.

Буферизированный последовательный порт со скоростью обмена 50 Мбит/с, при этом прерывание вызывается при заполнении буфера и выключается при исчерпании буфера данных.

Два последовательных порта: стандартный и многоканальный буферизированный последовательный порт – скоростной порт. Имеет возможность записи данных через контроллер ПДП.

TDM порт, осуществляющий обмен данными в режиме временного разделения.

HOST порт-интерфейс для связи с управляющим контроллером.

Параллельные порты ввода/вывода. Доступ происходит по командам ввода/вывода.

PLL CLOCK GEN – генератор тактовых импульсов с коэффициентом умножения частоты от 0,25 до 16.

Программируемый генератор задержек для увеличения длительности цикла обмена по шине до 7 циклов при обмене с медленными внешними устройствами.

Процессор C55\*, представленный на рис. 7.5, имеет переменную длину команды от 8 до 48 разрядов. В ядро (вычислитель-CPU) входят устройства команд IU, устройство контроля программы (PU), вычислители адресов AU, устройство обработки данных (DU), содержащее два АЛУ и два умножителя, работающие параллельно, что обеспечивает высокую производительность. Однако сдвигатель один.

В C55\* также имеются статическая и динамические памяти RAM, память ROM и КЭШ-память команд, обеспечивающая максимальную скорость записи команд из внешней памяти, третий последовательный порт, дополнительное АЛУ для выполнения операций управления кодами. К периферии относятся также глобальный порт вход/выход GP I/O, контроллер DMA, связанный с интерфейсами управляющего HOST порта и внешней памяти. Система шин C55\* включает: три шины чтения данных, две шины записи данных по 16 разрядов, шину выборки команд – 32 разряда, шесть 24-разрядных шин адресов, шина периферии.

Развитием платформы C5000 являются процессоры OMAP – Open Multimedia Application Platform (составная платформа для мультимедийных применений). Основой платформы являются про-

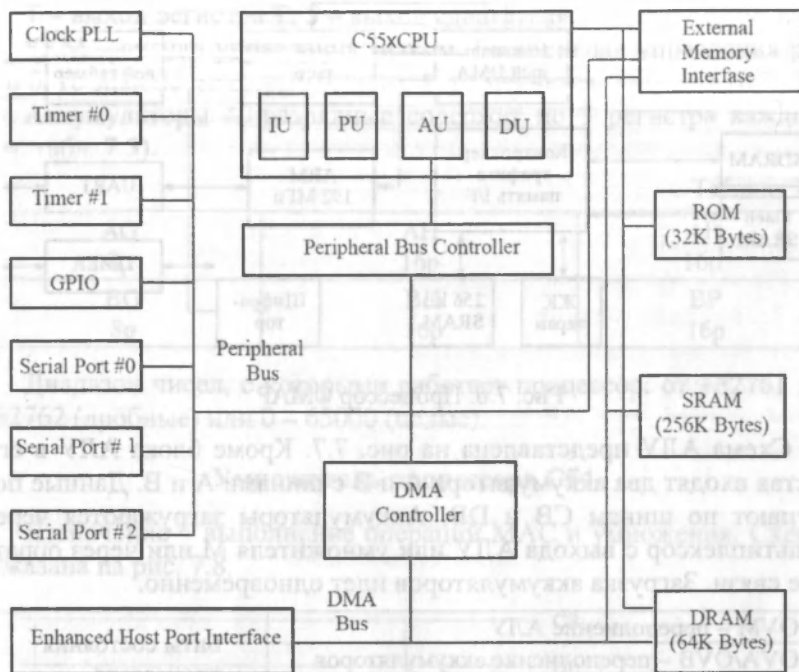


Рис. 7.5. Цифровой процессор TMS320C55\*

процессор C55 (C54), процессор ARM (Advance RISC Machine). Процессор предназначен для использования в системах радиодоступа стандартов 802.11, Bluetooth, обработки речи (текст-речь, восстановление речи и др.), обработка и контроль видео и изображений, обработка данных. Укрупненная схема процессора OMAP 5910 показана на рис. 7.6.

Кроме процессоров на кристалле расположены контроллер трафика, память SRAM, 16 канальный порт DMA, шифратор, универсальный порт UART, таймеры, один из которых «сторожевой». Более подробные сведения о платформе находятся на сайте фирмы TI [A5, A6].

#### 7.4. АЛУ и умножитель процессора C54

Назначение АЛУ заключается в выполнении арифметических и логических операций.

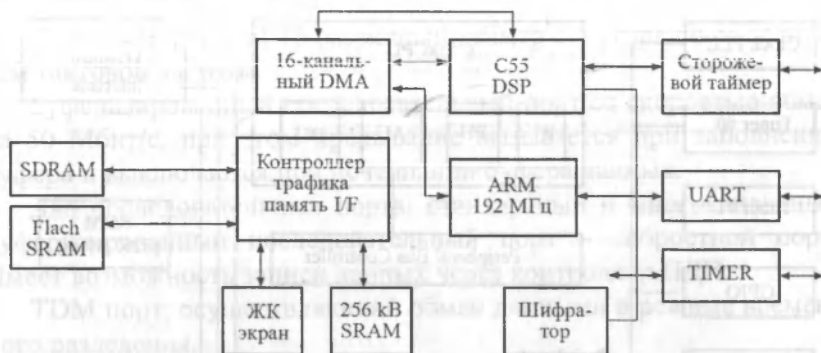


Рис. 7.6. Процессор OMAP

Схема АЛУ представлена на рис. 7.7. Кроме блока АЛУ в его состав входят два аккумулятора А и В с шинами А и В. Данные поступают по шинам СВ и DB. Аккумуляторы загружаются через мультиплексор с выхода АЛУ или умножителя М или через обратные связи. Загрузка аккумуляторов идет одновременно.

OVM – переполнение АЛУ	Биты состояния процессора
OVA/OVB – переполнение аккумуляторов	
ZA/ZB – нулевой результат	
C16 – уменьшенная точность вычисления (округление)	

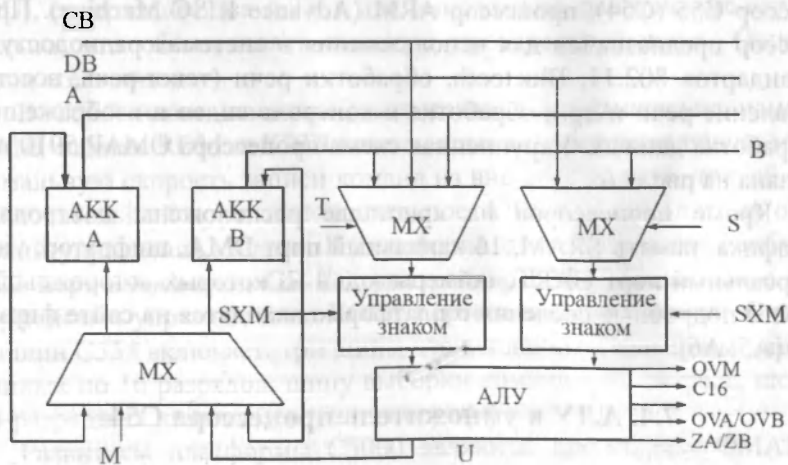


Рис. 7.7. АЛУ процессора C54



T – выход регистра T, S – выход сдвигателя.

SXM – сигнал управления знаком. Если сигнал управления равен 0, то знак не меняется, если же 1, то меняется.

Аккумуляторы 40-разрядные содержат по 3 регистра каждый (см. табл. 7.3).

Таблица 7.3

AG 8p	AH 16p	AP 16p
BG 8p	BH 16p	BP 16p

Диапазон чисел, с которыми работает процессор: от +32761 до -32762 (дробные) или 0 – 65000 (целые).

### Умножитель процессора C54

Назначение – выполнение операций MAC и умножения. Схема показана на рис. 7.8.

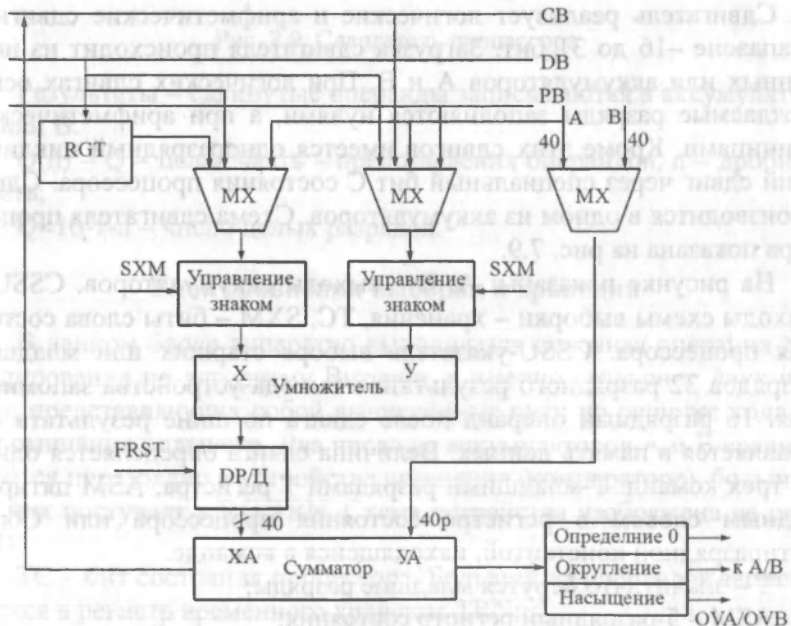


Рис. 7.8. Умножитель процессора C54

Умножение выполняется за 1 инструкцию.

Если  $FRST=1$ , то умножение дробных чисел.

$FRST=0$  – режим умножения целых беззнаковых чисел.

Если  $OVM=1$ , то имеется переполнение и результат заменяется либо на максимальное положительное или на максимальное отрицательное.

$OVM=0$ , то операция не выполняется, и процесс вычислений останавливается.

PB – внутренняя шина команд;

T – входной регистр;

A, B – выходы аккумуляторов;

Команда умножения-округления MACR – результат 16 бит берется из регистров АН или ВН аккумуляторов А или В, младшие разряды аккумуляторов отбрасываются.

## 7.5. Сдвигатель и устройство сравнения и хранения процессора C54

Сдвигатель реализует логические и арифметические сдвиги в диапазоне –16 до 31 бит. Загрузка сдвигателя происходит из шин данных или аккумуляторов А и В. При логических сдвигах освобождаемые разряды заполняются нулями, а при арифметических единицами. Кроме этих сдвигов имеется одноразрядный циклический сдвиг через специальный бит С состояния процессора. Сдвиг производится в одном из аккумуляторов. Схема сдвигателя процессора показана на рис. 7.9.

На рисунке показаны: А, В – выходы аккумуляторов, CSSU – выходы схемы выборки – хранения, ТС, SXM – биты слова состояния процессора. CSSU-указатель выбора старших или младших разрядов 32 разрядного результата с выхода устройства запоминания. 16 разрядный операнд после сдвига по шине результата направляется в память данных. Величина сдвига определяется одной из трех команд: Т-младшими разрядами Т регистра, ASM пятиразрядным словом в регистре состояния процессора или Const пятиразрядной константой, находящейся в команде.

T – значит, что берутся младшие разряды;

ASM – 5-разрядный регистр состояния;

const – пятиразрядная постоянная содержится в команде.

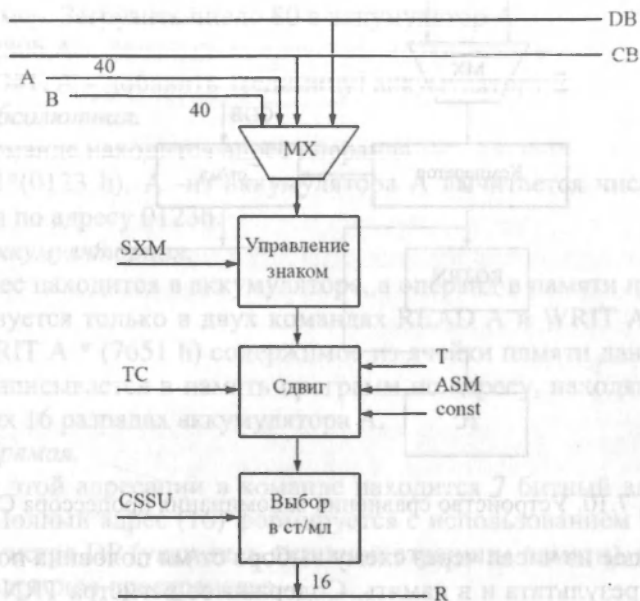


Рис. 7.9. Сдвигатель процессора

Результаты – сдвинутые операнды записываются в аккумулятор А или В.

$Q(n) - Q$  – целая часть – представления операндов,  $n$  – дробная часть;

$Q=16-1-n$  – число целых разрядов.

### Блок сравнения выборок и хранения

В данном блоке аппаратно выполняется основная операция декодирования по алгоритму Витерби, а именно сравнение двух чисел, представляющих собой вычисленные пути по решетке кода, и запоминание большего. Два числа из аккумуляторов А и В сравниваются поразрядно в устройстве сравнения (компараторе), большие из них поступает в RG TRN. Схема устройства изображена на рис. 7.10.

ТС – бит состояния процессора. Большой из операндов записывается в регистр временного хранения TRN;

ТС – бит слова состояния процессора; CDB – выход сдвигателя;

R – шина результата.

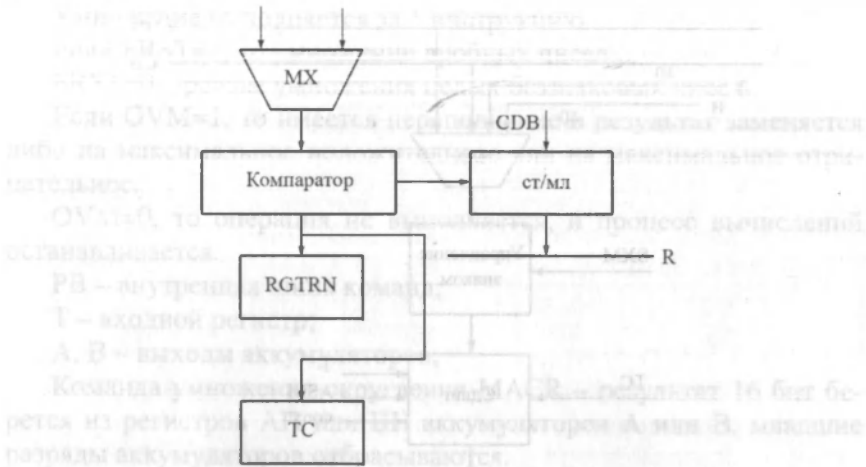


Рис. 7.10. Устройство сравнения-запоминания процессора C54

Большее из чисел через схему выбора ст/мл половина поступает на шину результата и в память. Содержимое регистра TRN сдвигается на 1 разряд влево и в младший освободившийся разряд записывается 0 или 1 в зависимости от выбора ст/мл половины. Этот же бит записывается в ТС.

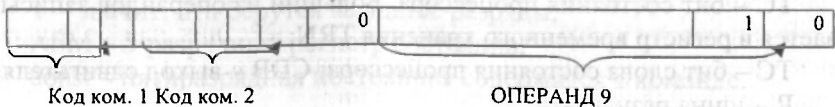
## 7.6. Адресация процессора C54

В процессоре C54 применяются следующие виды адресации:

1. Непосредственная.
2. Абсолютная.
3. Аккумуляторная.
4. Прямая.
5. Косвенная с помощью адресных регистров.
6. Адресация регистров, включенных в адресное пространство.
7. Стековая.

1. *Непосредственная.*

Операнд непосредственно находится в команде.



Пример. Загрузить число 80 в аккумулятор А

LD#80h,А

ADD#1, А – добавить 1(единицу) аккумулятору 2

### 2. Абсолютная.

В команде находится адрес операнда.

'SUB\*(0123 h), А -из аккумулятора А вычитается число, находящееся по адресу 0123h.

### 3. Аккумуляторная.

Адрес находится в аккумуляторе, а операнд в памяти программ. Используется только в двух командах READ А и WRIT А. Например, WRIT А \* (7651 h) содержимое из ячейки памяти данных операнда записывается в память программ по адресу, находящемуся в младших 16 разрядах аккумулятора А.

### 4. Прямая.

При этой адресации в команде находится 7 битный адрес операнда. Полный адрес (16) формируется с использованием 9 разрядного регистра DP (указатель активной страницы памяти), включенного в адресное пространство.

DP	Адрес операнда
----	----------------

LD # 10h постоянная 10 загружается в память по адресу, указанному в DP.

### 5. Косвенная.

При косвенной адресации используется блок дополнительных регистров, состоящий из 8 регистров ARO-AR7, дополнительного АЛУ и регистра циклического буфера. Все регистры включены в адресное пространство и находятся на нулевой странице памяти. Полный 16 разрядный адрес находится в одном из дополнительных регистров.

Адрес (номер) регистра, адресующего операнд, указывается непосредственно в команде.

Пример. STN А, \* AR6 – содержимое старшей половины аккумулятора А записать в ячейку, адрес которой находится в дополнительном регистре AR6. Признак косвенной адресации в символе \* в записи команды.

В дополнительном АЛУ выполняются действия над содержимым данного дополнительного регистра: декремент, инкремент, циклическая адресация, перестановка бит.

*6. Адресация регистров, включенных в адресное пространство.*

Все регистры находятся на нулевой странице памяти. Обращение к ним происходит вне зависимости от содержимого регистра DP.

Пример. LPM# 40h, SP – записать 40h в указатель стека SP.

*7. Стековая.*

Стековая адресация позволяет сохранить содержимое регистров при переходе к подпрограмме обработки прерываний и восстановления состояния регистров при возврате из прерываний. Имеется 4 команды стековой адресации:

PUSH – в стек;

POP – из стека;

PSHM, PSHD – запись регистра, включенного в адресное пространство, с предварительным уменьшением SP на 1;

POPM, POPD – считывание из стека с последующим инкрементом.

В устройство контроля программы входят:

1. Программный счетчик (PC).
2. Регистры повторения команд (RC).
3. Регистры повторения блока команд (BRC).
4. Регистры начального адреса, повторяемого блока команд (RSA).
5. Регистры конечного адреса повторяемого блока команд (REA).
6. Регистры расширения программного счетчика (XPC).

Программный счетчик хранит адрес текущей исполняемой команды. После выполнения очередной команды содержимое счетчика PC увеличивается на 1. Имеется ряд команд, изменяющих последовательность выполнения команд. К ним относятся, в частности, команды вызова подпрограмм, возврата из подпрограммы, команды переходов, а также повтора команды RPT.

### **7.7. Мультимедиа-видеопроцессор C80**

Данный процессор содержит 4 ЦСП, работающих параллельно, и управляющий мастер-процессор, что обеспечивает высокую производительность и позволяет обрабатывать изображения и мультимедиа.

Схема мультимедиа-видеопроцессора C80 показана на рис.7.11.

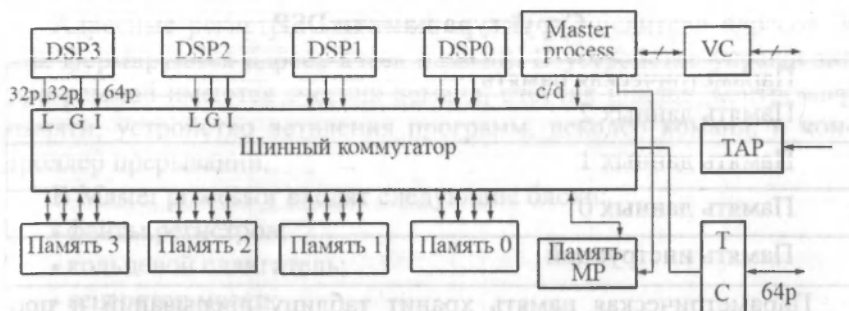


Рис. 7.11. Процессор C80

В состав процессора входят:

32 битный RISC мастер-процессор с плавающей точкой.

Четыре 32 битных параллельных DSP.

Память данных SRAM 50 кбит.

Видеоконтроллер VP для связи процессора с видеокамерой.

Каждый DSP имеет два 32 разрядных порта данных – глобальный (G) и локальный (L), а также 64 разрядный порт инструкций (I).

Мастер-процессор имеет 32 разрядный порт инструкций.

TAP-порт для проверки работоспособности процессора и отладки программ.

T/C-контроллер передачи данных между внутренней и внешней памятью.

### Характеристики процессора

Производительность П – 2000 MIPS.

M = 32 разряда.

Ft = 50 МГц ; tu = 0,02 мкс.

Память 4\*50К слов \* 32 разряда + 1\* 50К слов \* 32 разряда.

Четыре памяти данных и одна команд.

Технология КМОП 0,5 мкм.

Адресное пространство равно  $4 \times 10^9$  байт.

Шины обмена обеспечивают обмен с памятью 64p, 32p, 16p или 8p. За один цикл выполняются одновременно 5 инструкций и запись 10 данных. На корпусе находится 4 млн. транзисторов и имеется 305 выводов.

Питание 3,3 В.

### Структура памяти DSP

Параметрическая память
Память данных 2
Память данных 1
Память данных 0
Память инструкций

Параметрическая память хранит таблицу прерываний и программу обмена данными с портом ТС.

Применение процессора: видео-конференцная связь, формирование видеоизображений, обработка видеоизображения, мультимедиа, 2-мерная и 3-мерная графика, сжатие в реальном времени, система засекречивания информации, кабельное телевидение (сжатие картинки), передача документальных изображений.

Вычислитель DSP показан ниже.



- LAP – local address port (локальный порт адресов);
- GAP – global address port (глобальный порт адресов);
- IAP – порт адресов инструкций.
- В устройство обработки данных входят:
  - регистры данных;
  - умножитель;
  - кольцевой сдвигатель;
  - 3-х входное АЛУ;
  - генератор маскирования;
  - статусные регистры.



Адресные регистры включают в себя вычислители адресов (в них формируются адреса ячеек памяти). В устройстве управления программой имеются счетчик команд, счетчик циклов, контроллер памяти, устройство ветвления программ, декодер команд, и контроллер прерываний.

В Master processor входят следующие блоки:

- файлы регистров;
- кольцевой сдвигатель;
- генератор маски;
- компаратор;
- арифметическое и логическое устройства;
- таймер;
- устройство управления программой;
- контролер памяти инструкций;
- 4 аккумулятора с 2-ой точностью и с плавающей точкой;
- умножитель с плавающей точкой;
- вычислитель адресов;
- контроллер КЭШ памяти;
- интерфейс обмена данных, шинами.

### 7.8. Процессоры платформы C6000 [A6]

Платформа C6000 обеспечивает максимальную производительность для реализации сложных алгоритмов ЦОС с предельными скоростями обработки, как с фиксированной, так и с плавающей точкой. Это достигается за счет параллельных вычислений: ядро процессора содержит 8 вычислителей, работающих одновременно, и оптимальной архитектуры.

Для этого формируется так называемая «длинная команда» 8×32 разряда – Very Long Instruction Word.

#### Характеристики процессора

Производительность  $\Pi = 1600\text{--}2400$  MIPS, (C64x – до 8800 MIPS).

$F_t = 250$  МГц – 1 ГГц;  $t_u = 1\text{...}4$  нс.

Число разрядов 32.

8 команд × 32 разряда (команда 256р).

Технология КМОП 0,25 мкм.

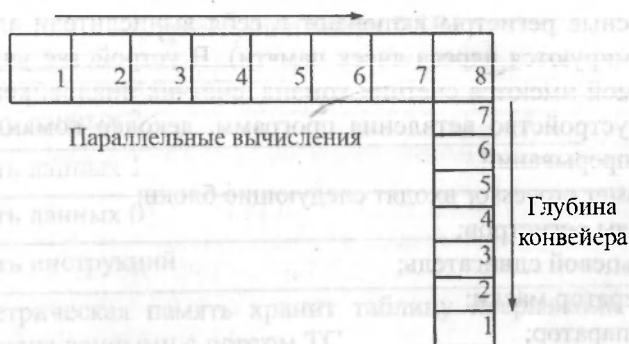


Рис. 7.12. Параллельно-конвейерная обработка

*Память.* Две памяти данных и программ по 512 К слов  $\times$  32р или память данных 1М $\times$ 32р. и память программ 2 М $\times$ 32р. Команда занимает 256р., данные представляются в виде 32р, 16р, 8р.

Наряду с параллельными вычислениями производится конвейерная обработка. Глубина конвейера до 8 операций, как показано на рис. 7.12.

### Области применения

Базовые станции сотовой связи, IP сети, Internet, DSL модели, мультимедийные системы и др., требующие высокой производительности. Схема процессора TMS320C6000 представлена на рис 7.13. Архитектура процессора – модифицированная гарвардская. Отличие от других ЦСП в структуре ядра.

Рассмотрим архитектуру ядра процессора. В ядро входят устройства обработки команд, формирователи потоков данных, два файла регистров данных и 8 вычислителей. Обработка команды, включает следующие фазы: обращение к командному слову (вызов команды), распаковка длинного командного слова и декодирование отдельных команд. Из памяти данных вызываются данные, образующие два потока, записываемые в два файла регистров А и В. Каждый включает 16 регистров по 32 разряда. Кроме того, имеются устройства управления и обработки прерываний. Вычислительные устройства выполняют следующие операции:

L – 32/40 разрядное сложение / вычитание и сравнение, логические операции;

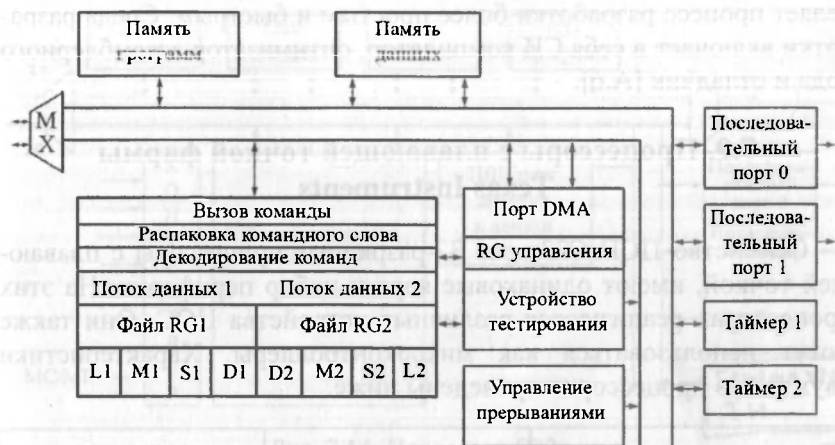


Рис. 7.13. Процессор платформы C6000

S – 32/40 разрядные сдвиги, сложение / вычитание, загрузку констант, логические операции, работа с битовыми полями;

M – целочисленное умножение 16×16;

D – адресный модуль, производит 32 разрядное сложение/вычитание, 32 разрядное сложение/вычитание адресов, обеспечивает передачу данных между регистровыми файлами и памятью, поддерживают циклическую и линейную адресации.

Таких наборов устройств имеется два, причем каждый работает со своим файлом регистров.

Каждый из регистровых файлов соединен 32 разрядными шинами с диспетчером памяти, который организует одновременную выборку из памяти по 4 шинам до 64 разрядов по 2 адресам. Память дробится на множество файлов, причем каждый байт адресуется отдельно как многопортовая память. Возможно подключение внешнего расширителя ОЗУ через 32 разрядный интерфейс внешней памяти. Два канала прямого доступа к памяти (DMA) могут выполнять функцию аппаратной загрузки программ. Связь с управляющим контроллером осуществляется через 16 разрядный интерфейс (Хост – порт доступа); имеется также два буферизированных последовательных порта. В состав процессора входят два таймера и ГТИ.

Данная платформа реализует переход от аппаратно-ориентированной среды разработки к программным моделям, что

делает процесс разработки более простым и быстрым. Среда разработки включает в себя СИ компилятор, оптимизатор ассемблерного кода и отладчик [А.6].

### 7.9. Процессоры с плавающей точкой фирмы Texas Instruments

Семейство ЦСП С33х это 32-разрядные процессоры с плавающей точкой, имеют одинаковые ядра и набор периферии. На этих процессорах реализуются различные устройства ЦОС. Они также могут использоваться как микроконтроллеры. Характеристики двух типов процессоров приведены ниже.

Процессор	VC33-120	VC33-150
Производительность	120 MFLOPS	150 MFLOPS
Время цикла 9 (нс)	17	13
Число разрядов	M = 32p	
ОЗУ (слов)	34К	34К
Память данных/ память программ	16М	16М
КЭШ-память	64	64
Технология КМОП	0,18 мкм	
Потребление	Меньше 200 мВ	
Питание	3,3 В, ядра 1,8 В.	

Применение процессоров С33х: – цифровая фильтрация, в том числе и адаптивная, корреляция, преобразование Гильберта, спектральный анализ, формирователи и демодуляторы сигналов, кодирующие и декодирующие устройства, ФАПЧ, обработка речи и др.

Упрощенная схема ЦСП С33х показана на рис. 7.14

Контроллер реализует команды сброса, прерываний и блокировки ветвлений. Ядро процессора 32-разрядное с плавающей точкой. Действия проводятся как с фиксированной точкой 16/32 разряда, так и с плавающей 32/40 разрядов.

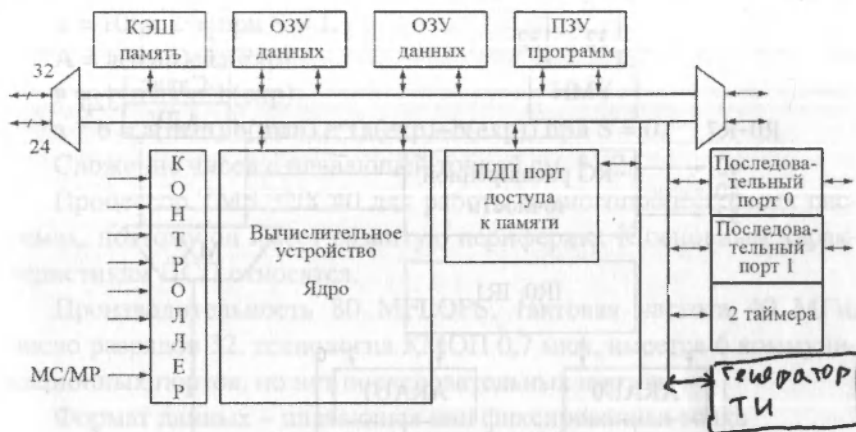


Рис. 7.14. Процессор C33x:

МС/МР – микро ЭВМ/цифровой сигнальный процессор

В состав процессора, кроме ядра, входят управляющий контроллер, две памяти данных, память программ и КЭШ память для хранения часто повторяющихся частей программы, уменьшая обращение к внешней памяти, и периферия, включающая порты ПДП, 2 таймера, 2 последовательных порта, 2 таймера, параллельные порты (16М×32), загрузчик программ и генератор тактовых импульсов с возможностью умножения частоты в 5 раз.

Работа ЦСП организована по 4 ступенчатому конвейеру. Основные операции конвейера: выборка команды, декодирование команды, чтение команды и выполнение.

Ядро ЦСП (см. рис. 7.15) включает вычислители, формирователи адресов и устройство управления программой. Имеются умножитель операндов с плавающей точкой (32×32) разряда, АЛУ и сдвигатель, выполняющие операции параллельно. Результаты хранятся в восьми 40 разрядных регистрах повышенной точности, причем 32 разряда содержат мантиссу, а 8 – порядок. Эти регистры являются как приемниками, так и источниками для любой арифметической операции. Наличие регистров повышенной точности позволяет производить вычисления без сохранения промежуточного результата в ОЗУ. В состав ядра входят два арифметических блока (АРАУ). Они вычисляют два 24 разрядных адреса, которые доступны через 8 дополнительных регистров (AR0-AR7). Дополнительные арифметические блоки выполняют следующие операции:

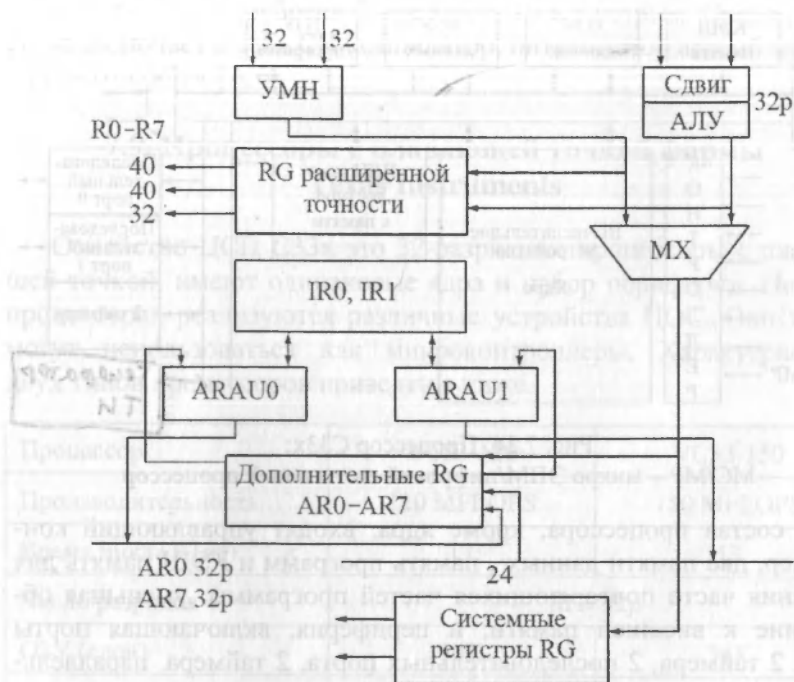


Рис. 7.15. Ядро процессора С30х

1. Декремент и инкремент адресов, в том числе и с шагом, отличным от 1 с использованием индексных регистров IR0, IR1.
2. Циклическую адресацию при использовании кольцевых буферов.
3. Бит-реверсную адресацию.

АРАУ – вычислители адресов, инкремент-декремент адреса. Изменение адреса с шагом больше единицы, кольцевая адресация бит-реверс – быстрое преобразование Фурье. Системные регистры: осуществляют управление программой. За инструкцию проводится пересылка двух операндов и одной команды.

### Представление чисел с плавающей точкой

$e$  – поле экспоненты (порядок) – 8 разрядов,  $s$  – знак 1 разряд,  $f$  – мантисса 23 разряда.

$$x = 0,1f \times 2^e \text{ при } S = 0.$$

$x = 10f \times 2^e$  при  $S = 1$ .

$A = a(\text{man})e^{a(\text{exp})}$ .

$v = b(\text{max})e^{b(\text{exp})}$ .

$a * b = a(\text{man}) b(\text{man}) e^{(a(\text{exp})+b(\text{exp}))}$  при  $S = 0$ .

Сложение чисел с плавающей точкой см. §1.2.

Процессор TMS 320C40 для работы в многопроцессорных системах, поэтому он имеет развитую периферию. К основным характеристикам ЦСП относятся.

Производительность 80 MFLOPS, тактовая частота 40 МГц, число разрядов 32, технология КМОП 0,7 мкм, имеется 6 коммуникационных портов, но нет последовательных портов.

Формат данных – плавающая или фиксированная точка

Поскольку процессор предназначен для использования в многопроцессорных системах он содержит 6 высокоскоростных телекоммуникационных портов со скоростью обмена 30 Мбит/с.

Ядро ЦСП такое же как у С30, также имеется 4 памяти.

КЭШ – память 512×32р. КЭШ n = 80 MFLDPS. Ft = 40 МГц, tu = 25 нс. ОЗУ 4К × 32р. ОЗУ2 4К × 32р. ПЗУ 4К × 32р.

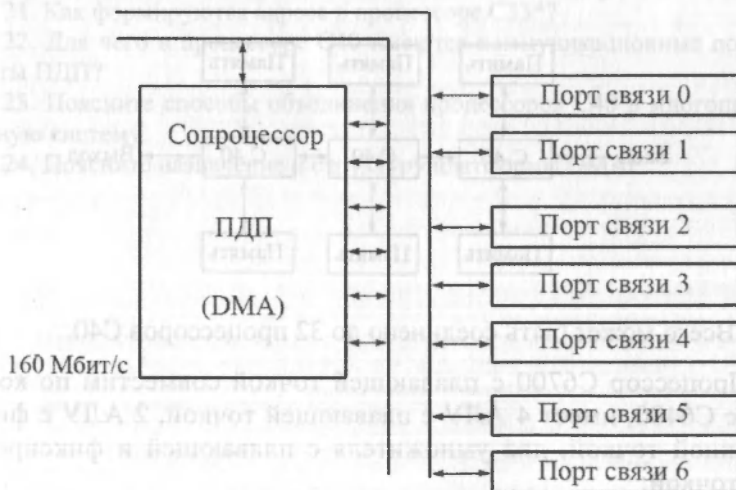
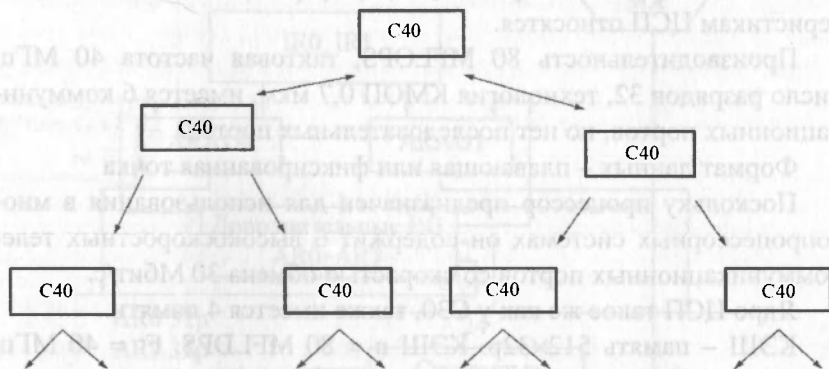


Рис. 7.16. Порты ввода/вывода

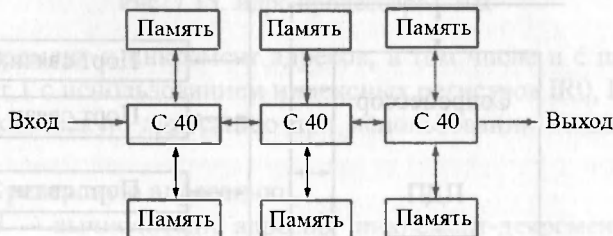
АЦП/ЦАП могут подключаться через коммуникационный порты. Порт DMA имеет 6 каналов для связи с внешними устройствами (рис. 7.16).

### Способы объединения в многопроцессорную систему

#### 1. Иерархическая.



#### 2. Конвейерная.



Всего может быть соединено до 32 процессоров C40.

Процессор C6700 с плавающей точкой совместим по командам с C6400, имеет 4 АЛУ с плавающей точкой, 2 АЛУ с фиксированной точкой, два умножителя с плавающей и фиксированной точкой.

К основным характеристикам C6700 относятся производительность до 1200 MFLOPS, 334 ММАС,  $t_n = 5$  нс, технология 0,18 и 0,13 мкм, памяти – статическая ОЗУ 1 Мбит и 512 Кбит.



### Контрольные вопросы

1. Перечислите основные типы процессоров фирмы Texas Instruments.
2. Поясните назначение и особенности платформ C2000, C5000, C6000.
3. Назначение и особенности процессоров C2000.
4. Перечислите основные устройства процессора C55\*.
5. Сравните характеристики ЦСП C54\* и C55\*.
6. Для чего в процессоре C54\* используются два аккумулятора?
7. Каковы функции и структура АЛУ процессора C54\*?
8. Каковы структура и операции умножителя C54\*?
9. Как производится операция сдвига в C54\*?
10. Поясните назначение устройства сравнения-запоминания C54\*.
11. Какие типы адресации используются в ЦСП C54?
12. Что входит в устройство управления программой C54?
13. Каковы особенности и назначение процессора C80?
14. Что входит в ядро процессора C80?
15. Поясните структуру памяти C80.
16. Поясните фазы обработки команд процессора C6000.
17. Что такое «длинная» команда и как она используется?
18. За счет чего обеспечивается высокая производительность C6000?
19. Поясните назначение вычислителей C6000.
20. Структура и характеристики процессоров C33\*.
21. Как формируются адреса в процессоре C33\*?
22. Для чего в процессоре C40 имеются коммуникационные порты и порты ПДП?
23. Поясните способы объединения процессоров C40 в многопроцессорную систему.
24. Поясните назначение и структуру платформы OMAP.

## Глава 8. Процессоры NM640x фирмы Модуль с векторным сопроцессором и микроконтроллер «Мультикор» фирмы Элвис

### 8.1. Структурная схема процессоров NM640x и его характеристики

К этим процессорам относятся ЦСП NM6403 и NM6404 фирмы «Модуль» [А.10].

Процессоры сочетают в себе архитектуры VLIW (Very Long Instruction Word) и SIMD (Single Instruction Multiple Data). Процессоры спроектированы для целочисленных данных.

Структурная схема процессора NM6403 показана на рис. 8.1.

#### Характеристики процессора

Производительность:

- Скалярные операции 40 MIPS; 120 MIPS для 32 p;
- Векторные операции от 40 до 11.5 ММАС;

Тактовая частота 40 МГц.

Технология КМОП 0,5 мкм.



Рис. 8.1. Структурная схема процессора NM 6403

Число разрядов:

- RISC ядро –
  - Команды 32 р и 64 р;
  - Данные 32 р (целые);
- VCP 1 – 64 р (длина векторных операндов переменная).

RISC процессор производит арифметические, логические и сдвиговые операции над 32-разрядными скалярными данными и формирует 32-разрядные адреса команд и данных при обращении к внешней памяти. Длина команды 32 и 64 разряда (две команды формируются одновременно).

Обмен данными между основными узлами процессора происходит по трем внутренним шинам, двум входным и одной выходной.

GM1 и LM1 – два блока программируемого интерфейса с локальной и глобальной внешними шинами, к каждой может подключаться внешняя память до  $2^{31} \times 32$  разрядных ячеек. Обмен происходит словами по 32 и 64 разряда (две соседние ячейки памяти).

CP1 и CP2 – коммуникационные порты, обеспечивающие обмен по двунаправленной 8-разрядной шине. Каждый коммуникационный порт имеет контроллер ПДП (DMA), позволяющий обмениваться 64-разрядными данными с памятью на внешних шинах CP1 и CP2, совместимыми с интерфейсами процессора TMS320C40.

RISC ядро использует 5-ступенчатый конвейер, две операции выполняются в одной команде. В ядро входят АЛУ, сдвигатель, адресный генератор с адресным пространством 16 GB, также 2 интерфейса с SRAM / DRAM. Имеются регистры: 8×32р общего назначения; 8×32р адресных; RG управления и состояния процессора.

Векторный сопроцессор VCP включает три внутренних памяти FIFO – 34×64р.

Имеет два типа насыщения на кристалле, поддержку матричных и векторно-матричных операций.

Корпус процессора типа BGA имеет 256 контактов. Напряжение питания 3,5 В. Потребляемая мощность 1,3 Вт.

Для увеличения производительности процессора используется параллельная работа в виде линейки процессоров, либо матрицы из процессоров. На рис. 8.2 представлена линейная многопроцессор-

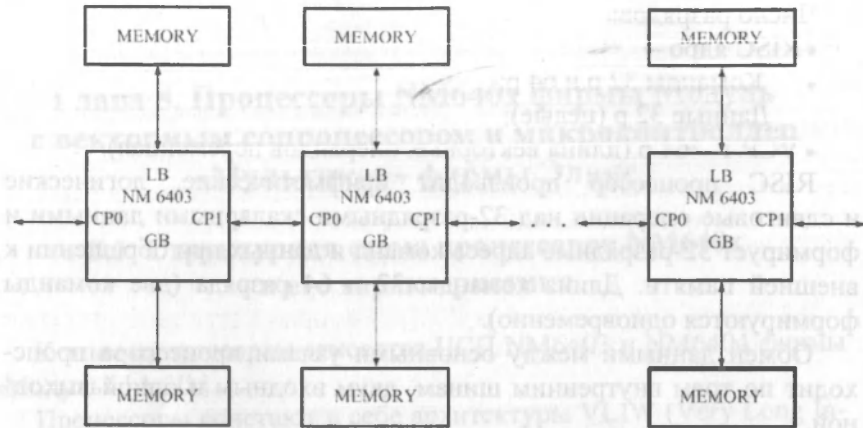


Рис. 8.2. Многопроцессорная система

ная структура на базе NM6403. Для соединения с внешними памятьями используются глобальные и локальные интерфейсы, а для связи процессоров между собой коммуникационные порты.

Области применения:

- нейронные сети;
- ЦОС (БПФ, фильтрация и др.);
- обработка видеоизображений;
- векторно-матричные вычисления;
- телекоммуникации.

Одним из существенных недостатков процессора NM6403 является отсутствие внутренней памяти, кроме трех буферных памяти FIFO в векторном сопроцессоре. В 2005 г. фирма «Модуль» выпустила модернизированный ЦСП NM6404, имеющий два внутренних ОЗУ каждое по 16 К×64 бит. К особенностям этого процессора можно отнести повышенную вдвое тактовую частоту 80 МГц, технологию 0,25 мкм.

Производительность при выполнении скалярных операций увеличена вдвое, векторные операции типа «умножение с накоплением» – за один такт при 32-разрядных данных выполняется 2 МАС, а при 16-разрядных – 4 МАС операций. За один такт процессор выполняет до трех скалярных операций (вычисление в АЛУ, модификация адреса, операция ввода-вывода – обращение к внешней памяти). Питание ядра 2,5 В, периферии – 3 В.

Кроме указанных выше применений, фирма рекомендует использовать процессор для обработки изображений, в гидро- и радиолокации, в базовых станциях сотовых сетей CDMA, TDMA и криптографии.

## **8.2. Векторный сопроцессор и реализация нейронной сети**

Наличие векторного сопроцессора (BCP) отличает NM6403 от других ЦСП. Сопроцессор выполняет векторно-матричные операции и реализует операции нейросети. На рис. 8.3 изображена функциональная схема сопроцессора. В нее входят следующие устройства:

OU – операционное устройство (рис. 8.2, матрица вычислений);

RAM – буфер внутренней памяти – очередь глубиной 32×64р слова, организованных по принципу FIFO;

VR – векторный регистр 64р, используемый в качестве операнда в операции умножения с накоплением (буфер из заданного количества одинаковых слов);

AFIFO – буфер накопления результата, 32×64 р по принципу FIFO;

F2CR – регистр хранения параметров функции насыщения;

RCS (Re loadable control storage) – перезагружаемая управляемая память;

SU – блоки сегментации.

В OU реализуются арифметические и логические операции в матричном виде. Имеется две матрицы 64×64р: рабочая и теневая. Результат любой операции процессора сохраняется в AFIFO. Он также может использоваться как один из аргументов в операциях умножения с накоплением и в операциях на векторном АЛУ. Чтобы получить доступ к результатам в AFIFO их надо перезагрузить в память.

Векторное арифметическо-логическое устройство (АЛУ) совершает операции над парами 64р слов, каждое из которых разделено на малоразрядные элементы. Буфер весовых коэффициентов (WFIFO) – очередь глубиной 32×64 слова, память организованная по принципу FIFO. В нее загружаются весовые коэффициенты и хранятся до записи в теневую матрицу. Загрузка может происходить по частям.

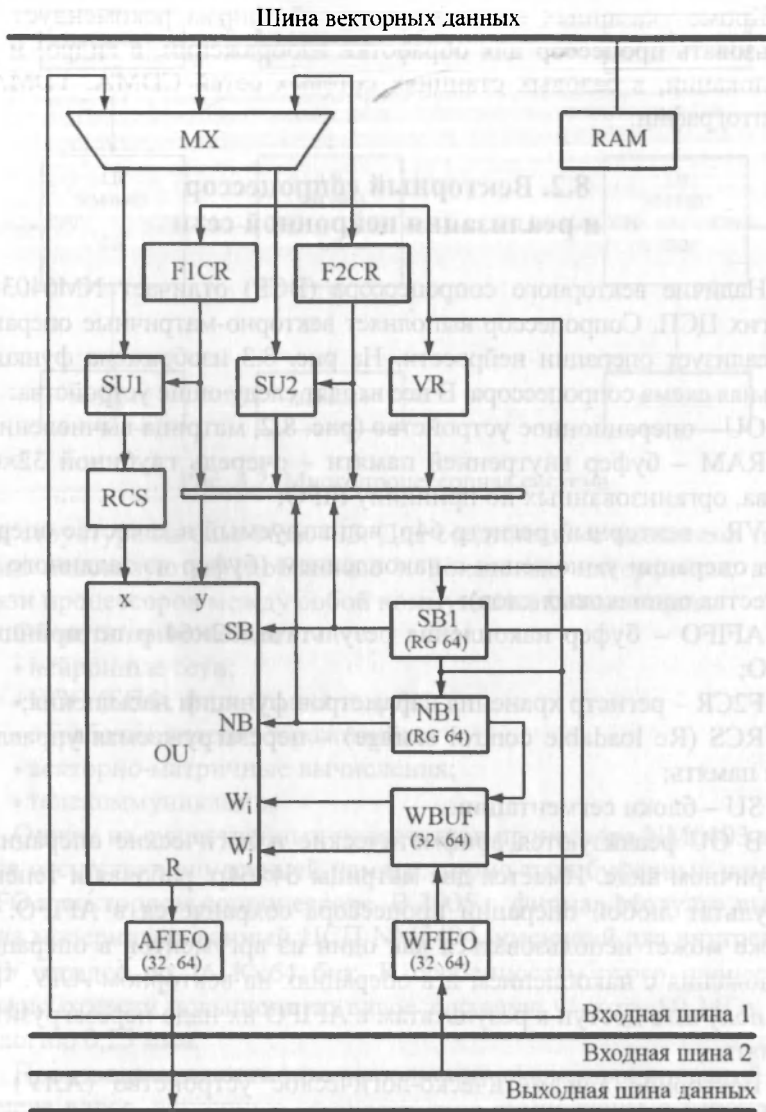


Рис. 8.3. Векторный процессор

Рабочая матрица (OU) производит умножение с накоплением. С рабочей матрицей связаны устройства разделения на строки SB1 и столбцы NB1 (такие же и для теневой матрицы).

Теневая матрица (64×64) – используется для ускорения записи весовых коэффициентов в рабочую матрицу. При вычислениях в рабочей матрице в теневую записываются новые коэффициенты.

### Нейронная сеть

Многослойная нейронная сеть (НС) типа персептрона содержит несколько слоев одноптипных нейронов [24].

В нейроне вычисляется значение функции

$$z_i = f[y_i + \sum_{j=1}^N W_{ij}x_j],$$

где  $f(*)$  – нелинейная (пороговая) функция активации;  $y_i$  – начальные условия;  $x_j$  – входные данные;  $W_{ij}$  – весовые коэффициенты.

Схема нейрона представлена на рис. 8.4.

Нейрон можно представить в виде скалярной функции от вектора численных входных значений.

В целях лучшей обучаемости градиентными методами пороговую функцию делают гладкой и нелинейной. Нейроны одного слоя, например персептрона, получают один и тот же вектор входных значений. Изменение выходных характеристик сети достигается за счет смены весовых коэффициентов, которые называются обучением.

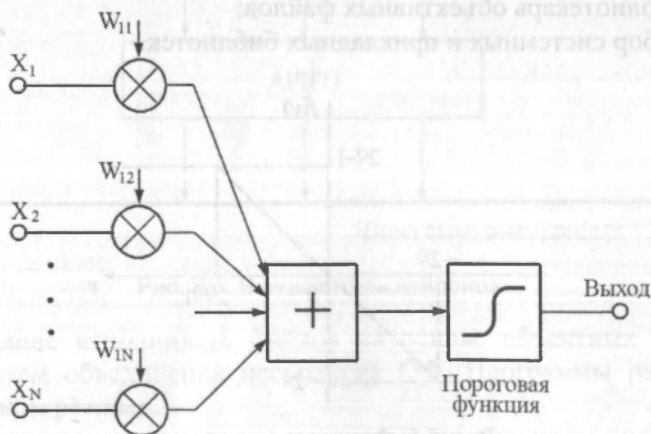


Рис. 8.4. Схема нейрона

Значения пороговой функции вычисляются по таблице в диапазоне от  $-32768$  до  $32767$ . если результат больше, то используется функция насыщения – наиболее положительное или отрицательное число  $f = -(x/2048)$  – числа в таблице целые.

Матрица произвольно делится на строки и столбцы. В макро-ячейки загружаются коэффициенты  $W_{ij}$ . На вход матрицы подается входной вектор  $x = (x_1, x_2, \dots, x_N)$ . Каждому его элементу соответствует строка матрицы. Ширина строки в бит соответствующей разрядности элемента входных данных.

Для каждого столбца вычисляется

$$y_i + \sum_j W_{ij} x_j.$$

Для снижения разрядности выходных данных и защиты от переполнения используется программируемая функция насыщения (SATUR), ограничение за счет уменьшения числа разрядов. На рис. 8.5. приведен вид функции.

Базовое программное обеспечение включает:

- компилятор C++;
- ассемблер;
- редактор связей;
- программный эмулятор;
- символьный отладчик;
- библиотекарь объективных файлов;
- набор системных и прикладных библиотек.

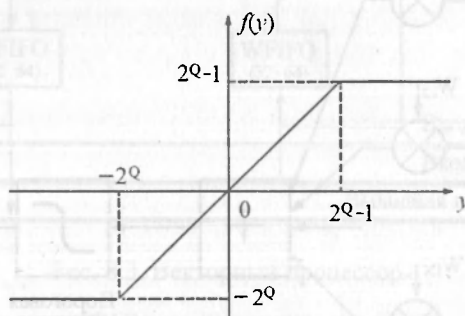


Рис. 8.5. Функция ограничения



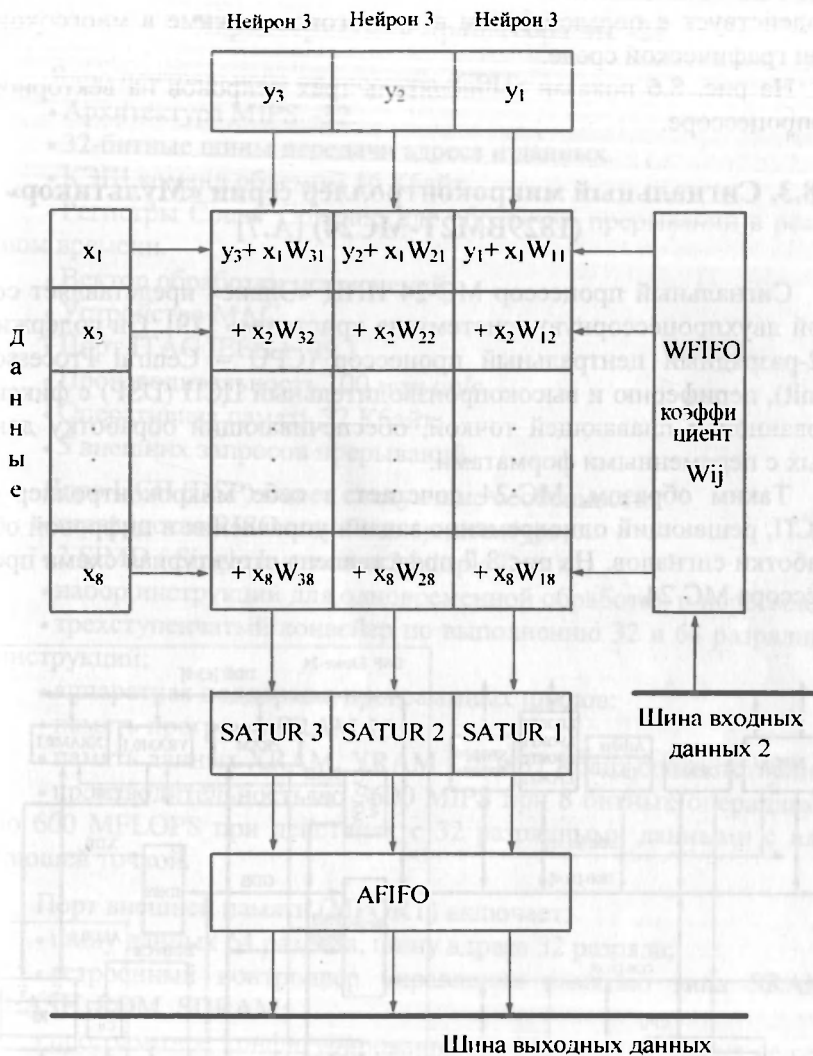


Рис. 8.6. Вычислитель нейронов

Создание исполнимых файлов на основе объектных файлов (ОФ) путем объединения нескольких ОФ. Программы работают в 32-битном режиме.

Все компоненты, за исключением символического отладчика, имеют интерфейс командной строки. Символический отладчик взаи-

модействует с пользователем в диалоговом режиме в многооконной графической срдс.

На рис. 8.6 показан вычислитель трех нейронов на векторном сопроцессоре.

### 8.3. Сигнальный микроконтроллер серии «Мультикор» (1829VM2T-МС24) [А.7]

Сигнальный процессор МС-24 ИПЦ «Элвис» представляет собой двухпроцессорную «систему на кристалле» [19]. Он содержит 32-разрядный центральный процессор (CPU – Central Processor Unit), периферию и высокопроизводительный ЦСП (DSP) с фиксированной и плавающей точкой, обеспечивающий обработку данных с переменными форматами.

Таким образом, МС-24 сочетает в себе микроконтроллер и ЦСП, решающий одновременно задачи управления и цифровой обработки сигналов. На рис 8.7 представлена структурная схема процессора МС-24.

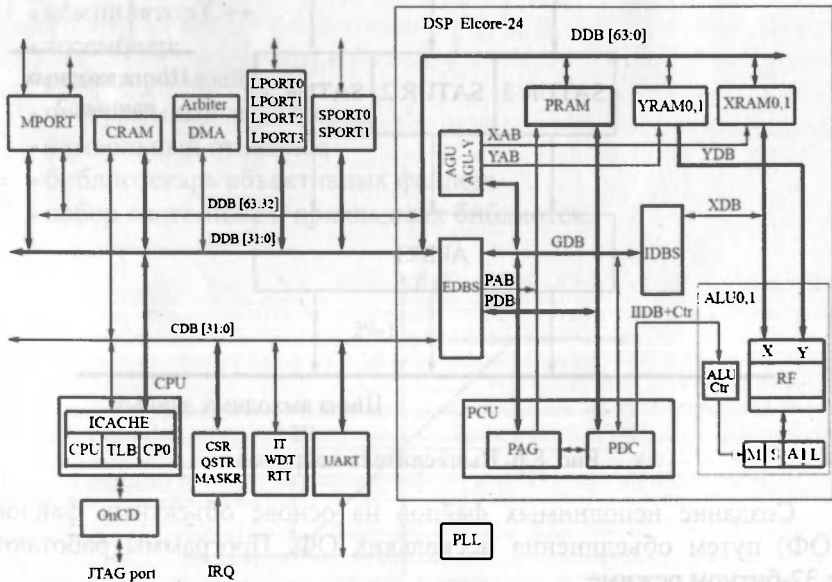


Рис 8.7. Микроконтроллер МС-24

### **Характеристики процессора MC-24**

Ядро центрального процессора (CPU):

- Архитектура MIPS –32.
- 32-битные шины передачи адреса и данных.
- КЭШ команд объемом 16 Кбайт.
- Регистры Count/ Compare для обработки прерываний в реальном времени.
- Вектор обработки исключений.
- Устройство MAC.
- Порт JTAG IEEE 1149.1.
- Производительность 100 млн оп/с.
- Оперативная память 32 Кбайт.
- 5 внешних запросов прерываний.

Ядро ЦСП (DSP) имеет следующие особенности:

- гарвардская RISC подобная архитектура;
- 2 SIMD ( Single Instruction Multiple Data);
- набор инструкций для одновременной обработки и пересылок;
- трехступенчатый конвейер по выполнению 32 и 64 разрядных инструкций;
- аппаратная поддержка программных циклов;
- память программ PRAM 16;
- память данных XRAM, YRAM 128 и 32 Кбайт соответственно;
- производительность до 3600 MIPS при 8 битных операндах и до 600 MFLOPS при действиях с 32 разрядными данными с плавающей точкой.

Порт внешней памяти (MPORT) включает:

- шину данных 64 разряда, шину адреса 32 разряда;
- встроенный контроллер управления памятью типа SRAM, FLASH, ROM, SDRAM;
- программное конфигурирование типа памяти и объема ее сегмента, а также циклов ожидания;
- обслуживание 4-х внешних прерываний.

Периферийные устройства состоят из следующих узлов:

- 12 канальный контроллер ПДП (DMA);
- 2 последовательных порта;
- 4 линк-порт;
- универсальный порт UART типа 16550;

- 32-разрядный интервальный таймер;
- 32-разрядный таймер реального времени;
- 32-разрядный сторожевой таймер.

Дополнительные блоки:

- генератор тактовой частоты (PLL) с множителем и делителем частоты входных сигналов;
- встроенные устройства отладки (OnCD);
- порт JTAG;
- поддержка режима энергосбережения.

Области применения сигнальных микроконтроллеров фирмы «Элвис»:

- акустика и гидролокация;
- связь и обработка сигналов (БПФ. Фильтрация и пр.);
- мультимедийная обработка изображений и цифровое телевидение;
- управление объектами с использованием высокоточных адаптивных алгоритмов;
- мультимедийная обработка звука (звуковые кодеки разных стандартов).

RISC ядро является ведущим в двухпроцессорной системе и выполняет основную программу. RISC ядро управляет работой DSP ядра путем передачи ему команд. Ядро DSP формирует ряд прерываний для RISC ядра. Внутренние интерфейсы микросхемы представляют собой стандартные шины архитектуры AMBA (Advanced Microcontroller Bus Architecture).

DSP ядро микроконтроллера MC-24 имеет две секции обработки данных с архитектурой SIMD, а ядро микроконтроллера MC-12 имеет одну секцию. Параллелизм обработки данных на уровне SIMD секции определяется возможностью выполнения по одной инструкции нескольких операций.

### Регистры ядра

Регистры ядра делятся на регистры данных и управления. Регистры данных состоят из файла данных RF RO-R31 – 32×16р. или 16×32р., двух аккумуляторов ACO и AC1, используемых в операции MAC как накопитель, регистра хранения результатов вычислений CCR и регистров формирователей адресов. Аккумуляторы могут объединяться в один 64-разрядный регистр. Ядро включает три

формирователя адресов: инструкций AGU, содержащий 8 наборов по 3 регистра – регистры адреса AT, смещения In и модификации Mn; два формирователя адресов данных AGU Y0, AGU Y1, в которые входят по 4 регистра – адреса AT, смещения IT, DT, модификатор MT. Регистр PDNR служит для управления, при операции денормализации, блочным порядком и режимом масштабирования (Scaling).

Устройство программного управления содержит регистры, контролирующие ход выполнения программы. Программный счетчик PC хранит 16-разрядный адрес инструкции в программе. Системный стек SS (15×16) хранит содержимое программного счетчика при входе в подпрограммы. Регистры CSL, CSH содержат длину цикла и адрес последней команды. Регистр управления и состояния SR содержит параметры управления и состояния DSP и заносит инкрементированное значение счетчика PC в системный стек при инициализации нового программного цикла DO, DOFOR и при входе в программу.

В регистре LA записывается адрес окончания цикла, регистр LC является счетчиком цикла, CNTR – счетчик циклов.

### **Форматы и типы данных**

При вычислениях используются следующие форматы данных с фиксированной точкой:

- 32р. целое со знаком в дополнительном коде;
- 32р. целое без знака;
- 32р. дробное в дополнительном коде (эти форматы используются при умножении операндов);
- длинный двойной формат, данные помещаются в двух 32-разрядных регистрах файла RF или в аккумуляторах AC0 и AC1, используются 64-разрядные данные целые со знаком, без знака и дробные.
- комплексный 16-разрядный формат дробных или целых чисел (Re X), (Im X), размещаемых в 32-разрядных регистрах RF, причем действительная часть помещается в старшем слове, а мнимая в младшем; имеется также комплексный 8 разрядный формат – две пары чисел по 8 разрядов.

Умножение целых и дробей отличается представлением результата: при умножении дробей значащие биты результата ( $2N - 1$ )

выравниваются по левой границе, а младший бит должен быть заполнен нулем; при умножении целочисленных операндов ( $2N - 1$ ) значащие биты результата выравниваются по правому краю и знаковый разряд дублируется.

При действиях с плавающей точкой используется формат 32 разряда. При этом используются регистр файла или одно слово памяти данных.

### Форматы и классы команд

Все инструкции процессора делятся на условно выполняемые и безусловные, по количеству одновременно выполняемых операций, по типу и количеству операндов (имеются трехадресные и двухадресные операции), по типу и числу одновременно выполняемых операций пересылок. Операции программного управления и пересылок обеспечиваются специальными форматами ( всего 8 ).

### Команды RISC ядра

Все команды являются однословными (32 бит). Типы команд: немедленного выполнения (I – immediate), перехода (J – jump), регистровые (R) представлены в табл. 8.1.

Таблица 8.1

I	Код операции	Номер регистра	Номер регистра	Операнд		
J	Код операции	Адрес перехода				
R	Код операции	Номер регистра	Номер регистра	Номер регистра	Величина сдвига	Поле функций 6 бит

К командам RISC относятся команды загрузки/ записи (память - RF), при этом следующий адрес =  $AT + In.$ ; вычислительные – арифметические; команды переходов/ветвлений, меняющие ход выполнения команд; сопроцессорные команды, выполняемые в сопроцессоре; спец. команды в формате R.

## Глава 8. Процессоры NM640x и микроконтроллер «Мультикор»

Основные параметры быстродействия микросхемы MC-24 (в тактах при рабочей тактовой частоте 100 МГц).

Количество MAC операций (умножения с накоплением) – од-  
нобитовый формат – 64.

Количество MAC операций 16×16/32 – 4.

MAC 32×32 + 32 (с плавающей точкой) – 2.

Нерекурсивная фильтрация комплексный формат плавающая  
точка.

Число тактов на отвод – 2.

Скалярная задержка – 4.

БПФ 1024 комплексное 16 битный формат данных и коэффици-  
ентов – 5800.

Деление, формат с плавающей точкой – 5.

Квадратный корень формат 32 бит с плавающей точкой – 8.

### **Контрольные вопросы**

1. На основе какой архитектуры реализованы процессоры NM6403 и NM6404?

2. Перечислите основные характеристики процессоров NM6403 и NM6404 и области применения.

3. Какие операции производит RISC процессор?

4. Как используются коммутационные порты и локальный и глобаль-  
ный интерфейсы процессоров?

5. Какие узлы входят в состав RISC ядра?

6. Какие узлы входят в векторный сопроцессор?

7. Как реализуется многопроцессорная система на процессорах NM640\*?

8. Перечислите области применения процессоров NM640\*.

9. Каково назначение памяти FIFO?

10. Какие операции производятся в операционном устройстве (рабо-  
чей матрице)?

11. Зачем нужна теневая матрица?

12. Поясните функцию нейрона.

13. Как реализуется в сопроцессоре операция ограничения?

14. Как реализуется функция нейронов в векторном сопроцессоре?

15. Что входит в состав сигнального микроконтроллера MC-24?

16. Каковы особенности и характеристики ядра центрального процес-  
сора микроконтроллера MC-24?

17. Каковы особенности и характеристики ядра ЦСП процессора?

18. Что входит в состав периферийных устройств микроконтроллера?

19. Каковы области применения микроконтроллера МС-24?
20. Какие порты содержит МС-24?
21. Перечислите основные шины микроконтроллера и их назначение.
22. Что входит в состав регистров ядра микроконтроллера?
23. Какие узлы входят в состав устройства программного управления МС-24?
24. Перечислите основные форматы данных с фиксированной и с плавающей точкой МС-24.
25. Каковы типы команд RISC ядра?
26. Перечислите основные параметры быстродействия МС-24.



## **Глава 9. Программируемые логические интегральные схемы и их применение [А8, А9]**

### **9.1. Классификация ПЛИС**

Программируемые логические интегральные схемы (ПЛИС) благодаря своим достоинствам получили широкое применение в различных областях. Эта элементная база высокой степени интеграции позволяет разработчику за короткое время создать на основе стандартных компонент устройства обработки сигналов с высоким быстродействием и низким потреблением. На базе ПЛИС могут быть изготовлены логические блоки и устройства, например, преобразователи кодов, контроллеры, умножители, сигнальные процессоры с не очень сложными алгоритмами (вычислитель БПФ, различные фильтры и др.). ПЛИС относятся к полужаказным интегральным схемам, поскольку внутрисхемная топология частично формируется при производстве самих ПЛИС, а частично программируется в соответствии с требованиями потребителя. Программирование пользователем ПЛИС производится путем изменения внутренней структуры, то есть соединения имеющихся на кристалле элементов в соответствии с требуемым алгоритмом. Настройка ПЛИС на заданный алгоритм и заданные спецификации производится программатором или с использованием САПР, что существенно уменьшает время проектирования.

ПЛИС делятся на стандартные (Standart Programmable Logic Devices-SPLD) или классические, сложные (Complex PLD), решетки программируемых элементов (Field Programmable Gate Array – FPGA) и программируемые матрицы (PLM). Классификация ПЛИС представлена на рис. 9.1. Стандартные PLD – это ПЛМ, ППЗУ (PROM) и программируемые матрицы логики. Сложные (complex PLD) это несколько стандартных, объединенных коммутационной матрицей. Основным программируемым компонентом ПЛИС являются логические матрицы [14, 15].

Структура PLD это совокупность двух матриц: программируемой матрицы «И», фиксированной матрицы «ИЛИ» и макроячейки. Входные сигналы поступают на  $n$  входов матрицы «И», обеспечивая

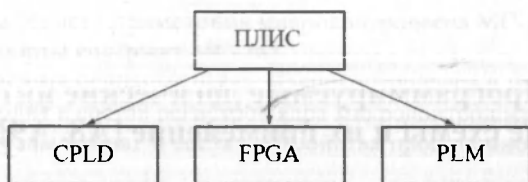


Рис. 9.1. Классификация ПЛИС

любые конъюнкции входных переменных. Выходы матрицы «И» соединены с матрицей «ИЛИ», которая реализует дизъюнкции поступающих сигналов, как показано на рис. 9.2.

PAL (Programmable Array Logic) – это соединение программируемой матрицы «И» и фиксированной «ИЛИ». Макроячейка является выходным буфером, содержащим мультиплексоры и коммутируемую логику. Также имеются цепи обратных связей с макроячейки на матрицу «И».

Возможность коммутации обеспечивается наличием программируемого элемента – переключки, представляющей собой ячейку памяти. Количество переключек конфигурации может быть до пяти.

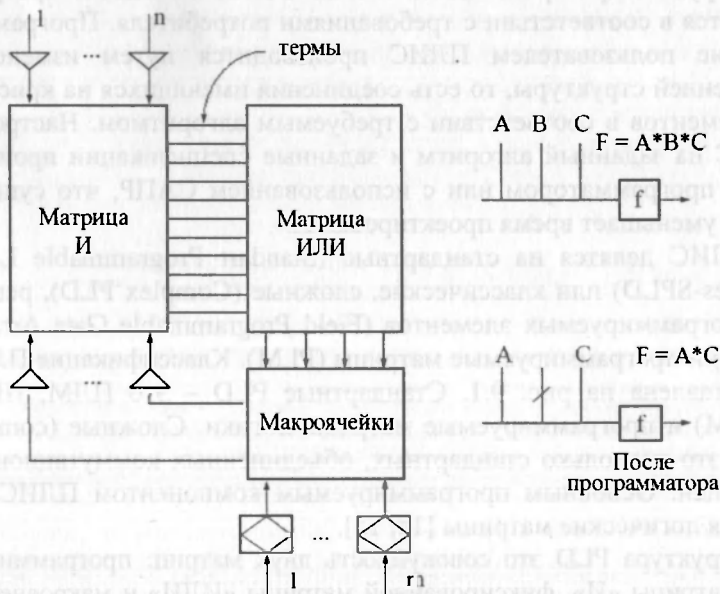


Рис. 9.2. Структура PAL

На рисунке показан один из конъюнкторов матрицы «И». В ПЛИС такой конъюнктор называется термом.

Сигналы А, В, С являются входом конъюнктора, образуя логическую функцию «И». Разрывая одну или несколько перемычек можно получить любую конъюнкцию от этих сигналов (см. рисунок). В ПЛИС каждый терм содержит не только прямые, но и обратные связи.

Программируемые пользователем матрицы логических элементов (FPGA) имеют архитектуру, показанную на рис. 9.3. В центре находится матрица настраиваемых (конфигурируемых) логических блоков (КЛБ), пространство между которыми заполнено программируемыми межсоединениями, состоящими из горизонтальных и вертикальных каналов. В точках пересечения межсоединений находятся переключающие точки. По краям кристалла для согласования уровней внутренних каналов с внешними выводами расположены блоки ввода/ вывода (БВВ). Они выполняют функции буферов: входного, выходного, с тремя состояниями. Программируемые межсоединения позволяют получить сложные логические функции за счет объединения входов и выходов любых БВВ КЛБ. Кроме указанных элементов современные FPGA содержат блок памяти, который может конфигурироваться как двухпортовое ОЗУ, встроенный умножитель (18×18) и цифровые блоки управления синхронизации. Настройка FPGA на заданный алгоритм выполняется каждый раз перед началом ее работы. Необходимая для этого программа настройки предварительно записывается в ПЗУ. После включения питания производится загрузка информации из ПЗУ и инициализация FPGA.

КЛБ конфигурируемые логические блоки. G, F, C – это входы общего назначения. X, Y, XQ, YQ – это выходы. K – тактовый вход. Cin, Cout – вход и выход переноса.

### 9.2. Свойства ПЛИС и порядок проектирования устройств

#### Общие свойства ПЛИС

1. Высокое быстродействие (тактовые частоты от 100 МГц до гигагерц).
2. Высокая степень интеграции, технология КМОП – десятки доли микрон, на кристалле находятся от сотен тысяч до нескольких миллионов вентиляей.

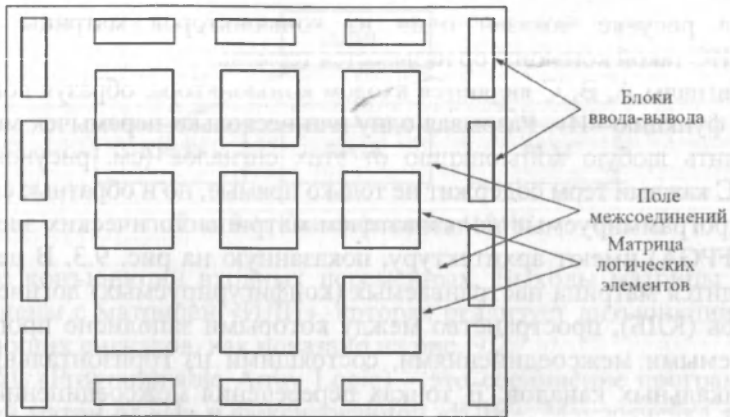


Рис. 9.3 Обобщенная структура FPGA

На рис. 9.4 представлен конфигурируемый логический блок и коммутирующие матрицы.

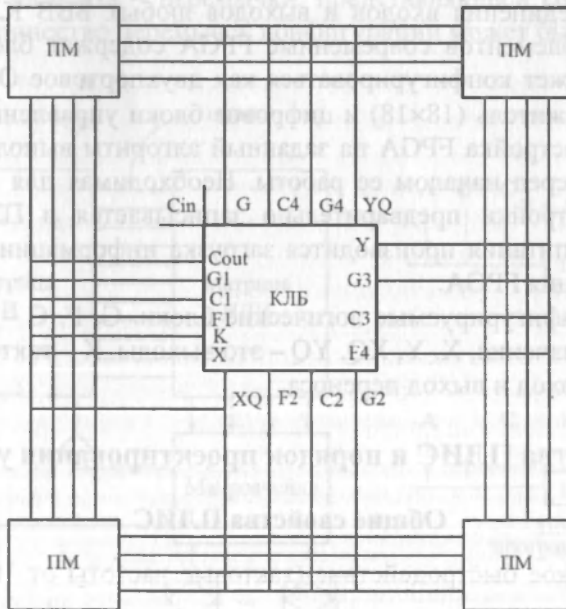


Рис. 11.4 Блок КЛБ

Рис. 9.4. Блок КЛБ

## ***Глава 9. Программируемые логические интегральные схемы***

3. Функциональная полнота, позволяющая реализовать законченное устройство на одной ПЛИС.
4. Универсальность: на одном кристалле реализуются разные устройства; многократность перепрограммирования.
5. Высокий процент выхода годных изделий и высокая надежность.
6. Защита от считывания.
7. Сброс регистров в ноль при включении питания;
8. Совместимость с TTL логикой.

### **Системные свойства ПЛИС**

1. Наличие бита Turbo при программировании значения бита. ON – это максимальное быстродействие при номинальном потреблении, OFF – это минимальное потребление при номинальном быстродействии.
2. Пониженное энергопотребление.
3. Питание 5 В и 3,3 В.
4. Защита информации – несколько бит.
5. Наличие нескольких входов для сигналов синхронизации и разрешение выходов.
6. Буферизация входных данных.

### **Проектирование устройств на ПЛИС**

#### ***Свойства проектирования***

1. Относительно малое время проектирования.
2. Наличие средств программирования и аппаратных средств проектирования (САПР).
3. Простота разработки устройств.
4. Простота конструкторского проектирования.
5. Высокая надежность.

Проектирование ПЛИС производится с использованием САПР и включает следующие этапы:

1-й этап: описание схемы, с помощью программного обеспечения, одним из стандартных способов (буквы, уравнения, таблицы истинности, принципиальная схема);

2-й этап: составление таблицы прошивки ПЛИС;

3-й этап (заключительный): технологическое программирование с использованием программатора, который разрывает ненужные связи (перемычки) в соответствии с таблицей прошивки.

Обобщенная схема САПР ПЛИС представлена на рис. 9.5.

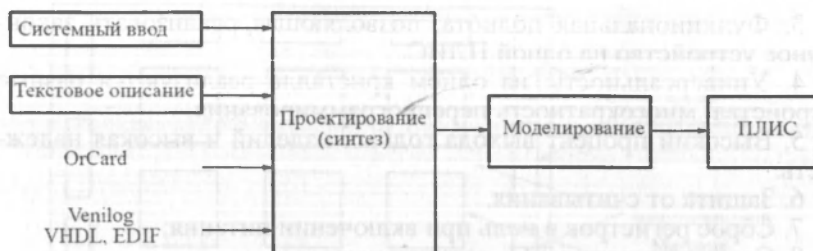


Рис. 9.5. Обобщенная схема САПР ПЛИС

### *I. Описание задания:*

1. Схемный ввод (библиотеки, готовые модули – регистры, счетчики и т.д.).
2. Язык функционального описания: булевы уравнения, таблица истинности, описание «черного ящика».
3. С использованием специализированных библиотек, так называемых мегафункций.
4. В виде временных диаграмм (графическое представление таблиц истинности).
5. Интерфейсы PDS, OrCAD, Verilog, VHDL, XNF.

### *II. Компиляция проекта – синтез.*

1. Выбор ПЛИС.
2. Разбивка на несколько корпусов.
3. Возможность назначения сигналов на конкретные выводы
4. Оптимизация результатов синтеза.

*III. Моделирование – функциональное и временное (диаграммы с отображением реальных задержек).*

## 9.3. Основные характеристики ПЛИС фирм Altera и XILINX [A.9]

Семейство ПЛИС фирмы Altera представлено на рис. 9.6. Номенклатура микросхем, выпускаемых фирмой, достаточно разнообразна, имеются мощные средства проектирования и набор мегафункций, то есть файлы типовых устройств ЦОС [A.9].

### Характеристики ПЛИС Apex

1. Число выводов (250 – 780).
2. Минимальное число эквивалентных вентилей ( $2,63 \cdot 10^5 \dots 2,67 \cdot 10^6$ )

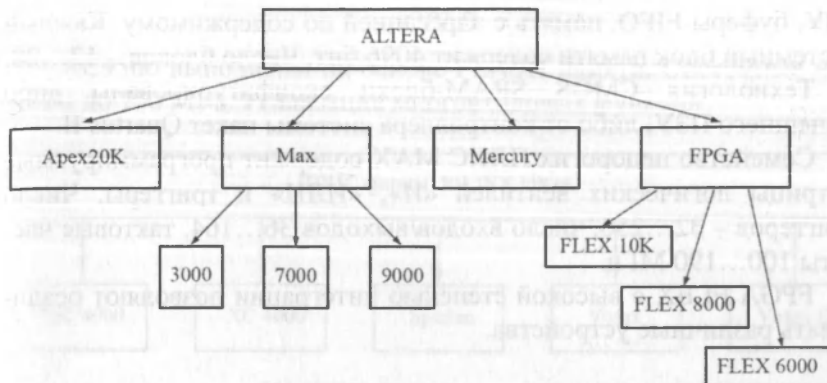


Рис. 9.6. Классификация ПЛИС фирмы Altera

3. Число логических элементов ( $4 \cdot 10^5 - 4 \cdot 10^7$ ).
4. Встроенные блоки памяти (26 ... 264).
5. Максимальный объем памяти ( $5,3 \cdot 10^4 \dots 5,4 \cdot 10^7$ ); число макроячеек (400...4000).

### **Особенности ПЛИС Mercury**

1. Интегрированный приемо-передатчик, поддерживающий синхронную передачу данных (для ЦОС и связи). Скорость передачи до 1,25 ГГб/с.
2. Архитектура логического элемента основана на традиционной таблице перекодировок (Look-up-table), оптимизация под высокие скорости обработки данных.
3. Новая архитектура межсоединений внутри кристалла для уменьшения задержек.
4. Элементы ввода-вывода поддерживают множество интерфейсов обмена данными.
5. ПЛИС содержат до 14 тысяч элементов [(ЛЭ), LE)].
6. Тактирование ФАПЧ с программируемым коэффициентом умножения частоты до 12 выходов ФАПЧ.
7. Специальная схема аппаратного умножения знаковых и без знаковых операндов  $16 \times 16$ р.
8. Встроенные системные блоки памяти ( Embedded system block): 4-портовые встроенные ОЗУ, двуканальные 2-портовые

ОЗУ, буферы FIFO, память с адресацией по содержимому. Каждый системный блок памяти содержит 4096 бит. Число блоков – 12...28.

Технология CMOS SRAM-блоки сконфигурированы либо с внешнего ПЗУ, либо от контроллера системы пакет Quartus II.

Семейство недорогих ПЛИС MAX содержит программируемые матрицы логических вентилей «И», «ИЛИ» и триггеры. Число триггеров – 32...256, число входов/выходов 36...164, тактовые частоты 100...190 МГц.

FPGA FLEX с высокой степенью интеграции позволяют реализовывать различные устройства.

### Мегафункции для ПЛИС Altera

1. FFT / IFFT – быстрое преобразование Фурье.
2. FIR Compiler – компилятор КИХ фильтров.
3. IIR Filter Compiler – компилятор БИХ фильтров.
4. Numerically Controlled Oscillator Compiler генератор с цифровым управлением.
5. Turbo Encoder / Decoder – кодер/ декодер турбокодов.
6. Reed-Solomon Compiler Encoder / Decoder. – кодер/декодер Рида-Соломона.

### Семейство ПЛИС фирмы XILINX [A.8, 16]

Семейство ПЛИС фирмы XILINX представлено на рис. 9.7.

Характеристики некоторых ПЛИС фирмы приведены ниже.

#### XC9500 (CPLD)

1. Высокая производительность. Время задержки вх/вых 5 нс. Тактовая частота 125 МГц 16 разрядного счетчика от 38 до 288 макроячеек. Не требуется программатор.

2. Возможно перепрограммирование до 10 тысяч раз. Основа макроячейки включена в функциональные блоки (ФБ), блоки ввода/вывода (БВВ), соединенные переключающейся матрицей (ПМ).

Структурные схемы CPLD и функционального блока показаны на рис 9.8 и 9.9.

I / O – логические выводы. GSK, GSR, GTS – управляющие выходы.



CPLD

XC9500 выполнены на основе FLASH памяти. Системные частоты до 150 МГц. Реализация многовходовых функций.

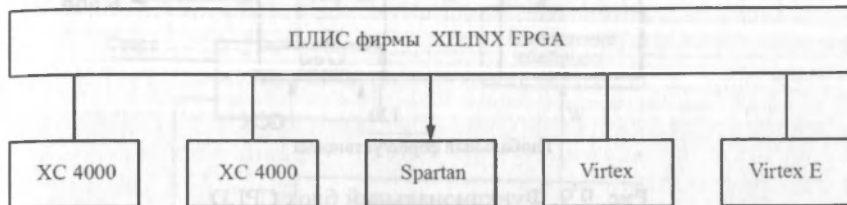


Рис. 9.7. ПЛИС фирмы XILINX

Высокоскоростные дешифраторы. Сложные синхронные счетчики. Программирование через порт JTAG, низкое энергопотребление < 75 мВт.

Virtex (FPGA) с технологией КМОП 0.22 мкм, содержит 1 млн. эквивалентных вентиляей. Используется в телекоммуникационных устройствах, сетевых решениях, и пр.

1. Тактовая частота 200 МГц, скорость обмена (chip on chip) до 200 МГц.

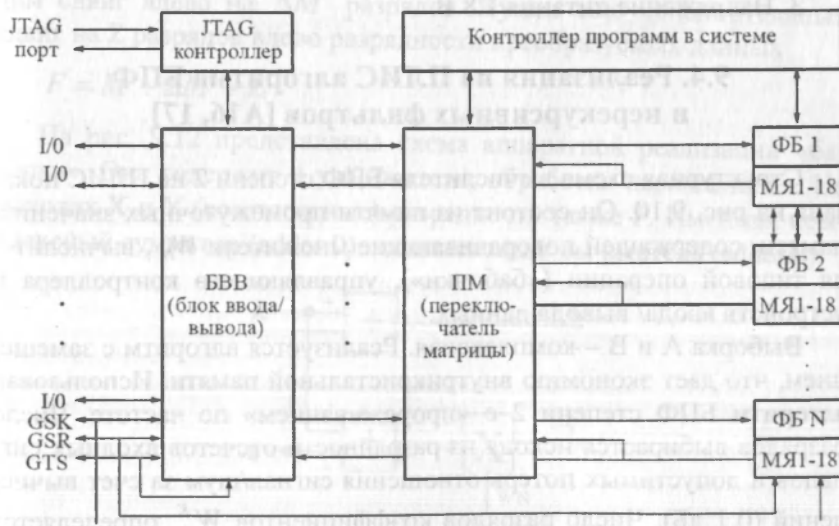


Рис. 9.8. Схема CPLD



Рис. 9.9. Функциональный блок CPLD

2. Умножение  $16 \times 16$  за 6 нс.
3. Технология 0,22 мкм, напряжение питания 2,5 В.
4. Несколько типов ОЗУ

Процесс проектирования близок к проектированию СБИС в системе HDL (Hardware Description Language) пакет Foundation Express.

Virtex E – 0.18 мкм, до 3.3 млн. вентиляей. Может использоваться для реализации системы на кристалле (System on Chip) (см. далее).

1. Тактовая частота 250 МГц
2. ОЗУ – 811 Кбит
3. Напряжение питания 1.8 В.

### 9.4. Реализация на ПЛИС алгоритма БПФ и нерекурсивных фильтров [A16, 17]

Структурная схема вычислителя БПФ степени 2 на ПЛИС показана на рис. 9.10. Он состоит из памяти промежуточных значений, памяти, содержащей поворачивающие множители  $W_N$ , вычислителя типовой операции («бабочки»), управляющего контроллера и устройств ввода/ вывода данных.

Выборка А и В – комплексная. Реализуется алгоритм с замещением, что дает экономию внутрикристалльной памяти. Использован алгоритм БПФ степени 2 с «прореживанием» по частоте. Число разрядов выбирается исходя из разрядности отсчетов входных сигналов и допустимых потерь отношения сигнал/шум за счет вычислений (0,1 дБ). Число разрядов коэффициентов  $W_N^K$  определяется по формуле



Рис. 9.10. Блок-схема вычислителя БПФ

Структура вычислителя типовой операции БПФ дана на рис. 9.11.

$$R = M - 1,7 \lg\left(\frac{N}{2} + 1\right) + 8,$$

где  $M$  бит – разрядность входного сигнала.

Разрядность промежуточных данных  $M + \Delta M$ . Далее необходим сдвиг влево на  $\Delta M$  разрядов. Нужен еще дополнительный сдвиг на  $Z$  разрядов влево разрядности преобразуемых данных

$$F = M + \Delta M + Z.$$

На рис. 9.12 представлена схема аппаратной реализации «бабочки». Она содержит 4 умножителя; обработка параллельная. На выходах  $X$  и  $Y$  (комплексные) разрядность также  $F$ . Имеются комплексный сумматор (adder 0), комплексный вычитатель (adder 0),

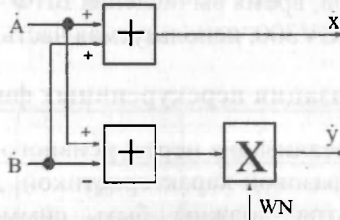


Рис. 9.11. «Бабочка» для БПФ с прореживанием по частоте

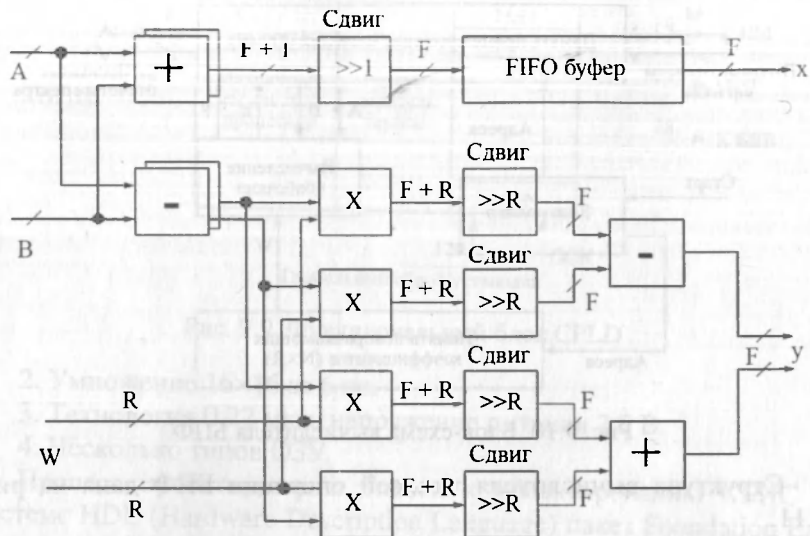


Рис. 9.12. Блок-схема аппаратной реализации «бабочки»

умножители ( $\text{mul } 0, \dots, \text{mul } 3$ ), сумматор ( $\text{adder } 1$ ). Для компенсации сдвигов в разных каналах используется FIFO-буфер (delay line). Для хранения промежуточных данных используется внутренняя 2-портовая память ПЛИС. Одновременно считываются числа  $A$  и  $B$  и записываются числа  $X$  и  $Y$ .

Устройства сдвига производят нормировку. В ПЛИС сдвиг осуществляется путем подключения ячеек со сдвигом по номерам битов входных и выходных шин элементов и не требует аппаратных затрат.

Вычислитель БПФ на ПЛИС имеет параметры: размер выборки  $N = 512$ , тактовая частота  $F_T = 80$  МГц, разрядность выходного сигнала  $M = 14$  разрядов, время вычисления БПФ – 30 мкс, тип кристалла: XILINX FPGAХCV300, используемая часть кристалла <25%.

### Реализация нерекурсивных фильтров

Ниже описана реализация нерекурсивного фильтра с КИХ 8 порядка с линейной фазовой характеристикой. Для ее получения коэффициенты фильтра должны быть симметричными, то есть  $h_m = h_{N-m}$ . Это позволяет сократить число перемножителей вдвое.

Схема фильтра показана на рис. 9.13.

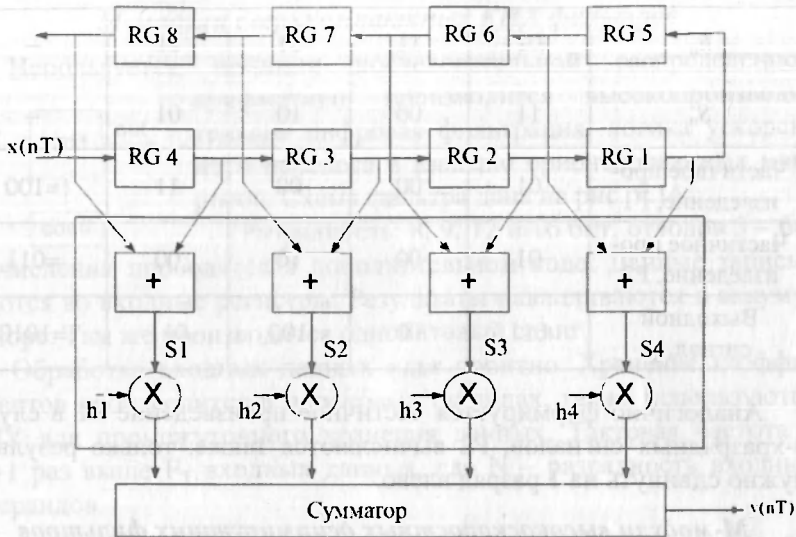


Рис. 9.13. Нерекурсивный фильтр (N = 8)P

В качестве параллельного векторного множителя (ПВП) удобно использовать таблицу перекодировок, входящую в состав логического элемента (ЛЭ). Работа ПВП описывается уравнением

$$y = S_1 h_1 + S_2 h_2 + S_3 h_3 + S_4 h_4$$

Рассмотрим реализацию 2-разрядного параллельного векторного умножителя.

Вектор коэффициентов имеет вид.

$h_1$	$h_2$	$h_3$	$h_4$
01	11	10	11

Вектор сигналов

$S_1$	$S_2$	$S_3$	$S_4$
11	00	10	01

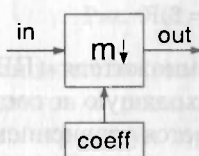
Значения коэффициентов, частичных сумм произведений и результатов представлены ниже в таблице.

## Сигнальные микропроцессоры

$i_n$	01	11	01	11	—
$S_n$	11	00	10	01	—
Частичное произведение, P1	01	00	00	11	=100
Частичное произведение, P2	01	00	10	00	=011
Выходной сигнал, у	011	00	100	011	=1010

Аналогично формируется частичное произведение P1 в случае 4-хразрядных сигналов, P2 вычисляются также, только результат нужно сдвинуть на 1 разряд влево.

### *М-модули высокоскоростных децимирующих фильтров*



Фильтры на основе параллельной арифметики производят фильтрацию в реальном времени с понижением частоты дискретизации в  $m$  раз. Используется принцип параллельной арифметики с временным мультиплексированием.

Для хранения коэффициентов используется распределенное ОЗУ. Архитектура оптимизирована под ПЛИС.

### *Характеристики М модулей по занимаемому объему на ПЛИС*

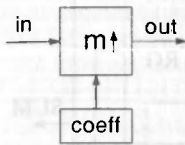
Децимация	16 отводов	256 отводов
Частоты	$F_{вх} = 0...48$ МГц	$F_{вх} = 0...3$ МГц
2	1800 КЛБ	1780 КЛБ
32	240 КЛБ	230 КЛБ

### *М-модули высокоскоростных интерполирующих фильтров*

Разрядность 12 и 16 бит; число отводов 16, 32, 64, 128, 256. Работа с данными производится в дополнительном коде. Тактовая частота увеличивается в  $m$  раз.

***M-модули сверхкомпактных КИХ фильтров***

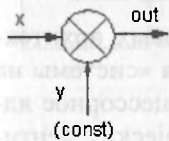
Используется принцип последовательной распределенной арифметики, производится высокопроизводительная цифровая фильтрация, логика ускоренного переноса и взаимно ориентированных макросов. Схема фильтра дана на рис. 9.14



Разрядность: 8, 9, 12 и 16 бит, отводов 5 – 64, вычисления проводятся в дополнительном коде. Данные записываются во входные регистры. Результаты накапливаются в аккумуляторе. Там же производится однобитовый сдвиг.

Обработка входных данных идет побитно. Хранение коэффициентов производится в поисковых таблицах, также используются ОЗУ для промежуточного хранения данных. Тактовая частота в N+1 раз выше  $F_T$  входных данных, где N – разрядность входных операндов.

***M-модули быстродействующих параллельных умножителей***



Используется логика ускоренного переноса и взаимного относительного ориентирования макросов (RLOC).

8×8, 10×10, 12×12, 16×16 быстродействующие параллельные умножители.

Асинхронные или синхронные вход (выход) конвейерные вариант используют принцип Бута с последующей сверткой сумматорами разрядов одинакового веса.

Асинхронные не содержат элементов дополнительной памяти, а синхронные содержат регистры на вход и выходе, результат выдает с задержкой на такт.

Конвейерные умножители обеспечивают наибольшую производительность и выдают результат за 4 такта.

Разряды	Асинхронные		Синхронные		Конвейерные	
	время, нс	КЛБ	время, нс	КЛБ	время, нс	КЛБ
8×8	29,7	57	23,7	63	9,8	63
16×16	52,9	218	49	226	20,6	223

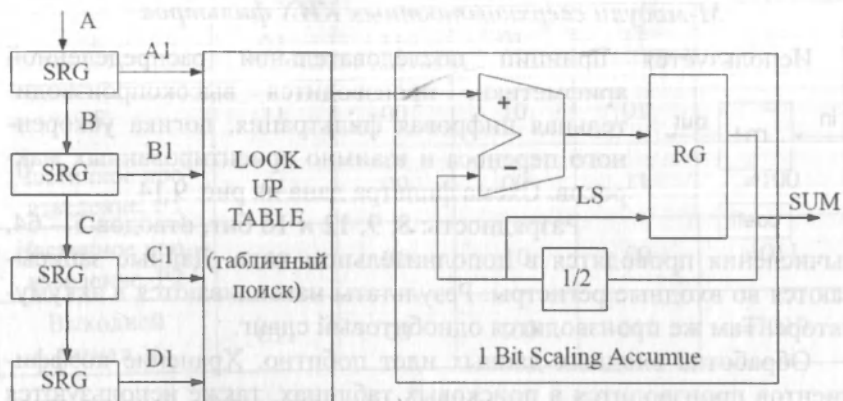


Рис. 9.14. Модуль сверхкомпактного КИХ фильтра

### 9.5. Система на кристалле как развитие ПЛИС и программируемых процессоров [19]

Усложнение методов и алгоритмов ЦОС требует новых архитектурных решений системного уровня. К ним относится «системы на кристалле», в которых на кристалле объединены процессорное ядро, внутреннее ОЗУ, массив программируемых логических вентилей, выполняющих отдельные операции цифровой обработки, и периферийные устройства ввода/вывода. В частности, к системам на кристалле относится архитектура ARM, включающая высокопроизводительное 32-разрядное RISC процессорное ядро, поддерживающее команды ЦСП, блок памяти из скоростного ОЗУ и кэш-память для увеличения производительности ядра, вычислители высокого уровня с жесткой логикой (после предварительного программирования), выполняющие стандартные операции обработки и гибкий интерфейс к внешним устройствам. Встроенные периферийные устройства, включающие контроллер прерываний, таймеры-счетчики, асинхронные приемо-передатчики, многоканальный порт ПДП, обеспечивают необходимую поддержку вычислений. Структура системы на кристалле показана на рис. 9.15.

Система на кристалле обычно содержит одно или несколько процессорных ядер, которые могут быть микроконтроллерами или ЦСП, массив программируемой логики и различные цифровые или аналоговые устройства. Различаются аппаратные и программные



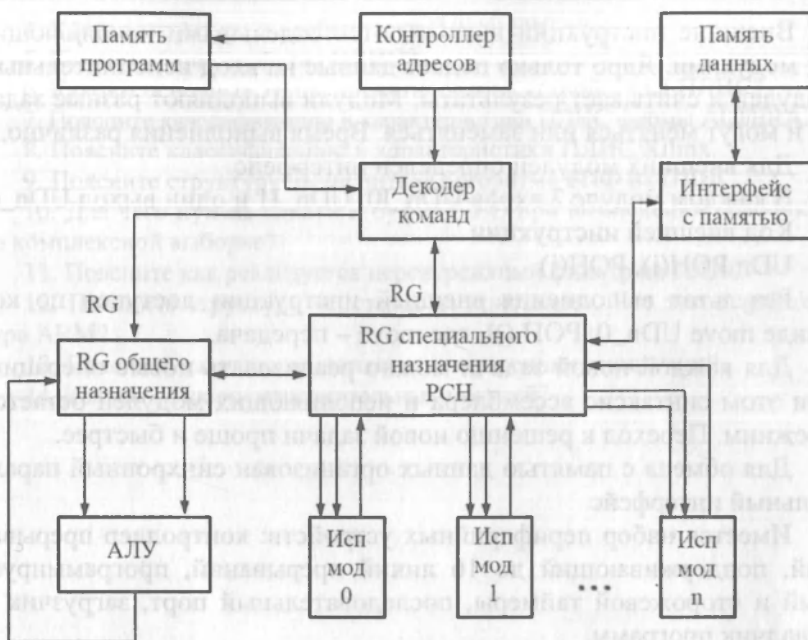


Рис. 9.15 Система на кристалле

ядра. Во втором случае ядро процессора содержит обычный набор команд и программируемую логику, система реализуется на кристалле FPGA. В первом случае ядро процессора и массив программируемой логики представляют отдельные устройства, связанные через интерфейсы.

Рассмотрим программное процессорное ядро, имеющее гарвардскую архитектуру.

Набор инструкций делится на три группы:

- 1) адресные инструкции,
- 2) условного и безусловного переходов,
- 3) вызова и выход из подпрограммы. Они выполняются контроллером указателя адреса памяти программ.

Внутренние инструкции (логические операции, команды переноса, сдвиг, запись констант) выполняются АЛУ. Внутренние инструкции изменяют состояние регистров общего назначения (РОН), с которыми работают.

Внешние инструкции выполняются отдельными исполняющими модулями. Ядро только подает данные на вход исполнительных модулей и считывает результаты. Модули выполняют разные задачи и могут меняться или заменяться. Время выполнения различно.

Для внешних модулей определен интерфейс.

В каждом модуле 2 входа  $UDn\_I0$ ,  $UDn\_I1$  и один выход  $UDn\_O$

Код внешней инструкции

$UDn\ POH(i), POH(j)$

Результат выполнения внешней инструкции доступен по команде  $move\ UDn\_O, POH(i)$ , где  $move$  – передача.

Для каждой новой задачи можно реализовать новые операции, при этом синтаксис ассемблера и исполняющих модулей остается прежним. Переход к решению новой задачи проще и быстрее.

Для обмена с памятью данных организован синхронный параллельный интерфейс

Имеется набор периферийных устройств: контроллер прерываний, поддерживающий до 16 линий прерываний, программируемый и сторожевой таймеры, последовательный порт, загрузчик и отладчик программ.

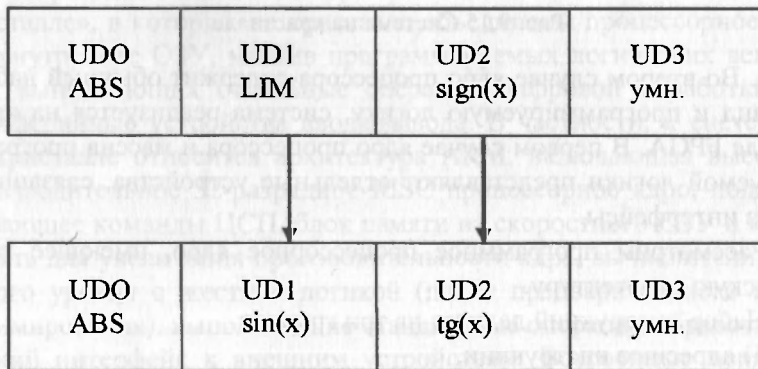


Рис. 9.16. Переопределение внешних команд

### Контрольные вопросы

1. Перечислите основные типы ПЛИС.
2. Какие устройства реализуются на ПЛИС?
3. Какова структура стандартной ПЛИС PLD (PAL)?

## Глава 9. Программируемые логические интегральные схемы

4. Какова структура и особенности ПЛИС FPGA?
5. Каковы общие свойства ПЛИС?
6. Каковы основные этапы проектирования устройств на ПЛИС?
7. Поясните классификацию и характеристики ПЛИС фирмы ALTERA.
8. Поясните классификацию и характеристики ПЛИС Xilinx.
9. Поясните структуру вычислителя алгоритма БПФ на ПЛИС.
10. Для чего нужны сдвиги и буфер F1EO при вычислении «бабочки» по комплексной выборке?
11. Поясните как реализуется нерекурсивный фильтр на ПЛИС.
12. Поясните структуру «системы на кристалле». Что такое архитектура ARM?
13. Поясните назначение внешних и внутренних инструкций.
14. Каковы функции исполнительных модулей?

**Список сокращений английских терминов**

- DSP (digital signal processing) – цифровой сигнальный процессор  
 MC (microcontroller) – микроконтроллер  
 PLD (programmable logic Devices) – программируемые логические ИС (ПЛИС)  
 RISC (reduced instruction on set computing) – вычислитель с ограниченным числом команд.  
 CISC (continual instruction on set computing) – вычислитель с бесконечным числом команд.  
 MIPS – миллионов инструкций в секунду  
 MFLOPS – миллионов операций с плавающей точкой в сек  
 MMACS – миллион операций MAC в сек  
 Floating point – плавающая точка  
 MS (multiprocessor Systems) – многопроцессорная система  
 RAM – ОЗУ с произвольной выборкой,  
 SRAM (read only memory) – статическая ОЗУ  
 ROM – постоянное запоминающее устройство  
 Flash memory – «быстрая память»  
 DM (data memory) – память данных  
 PM (program memory) – память программ  
 DMA (direct memory access) – прямой доступ к памяти (ПДП)  
 (порт прямого доступа к памяти)  
 RG – регистр  
 MUX (MX) – мультиплексор  
 CORE – ядро процессора  
 MAC – умножитель – накопитель (аккумулятор)  
 RND – округление результата  
 SAT (saturation) – операция насыщения, ограничения  
 ALU – арифметически логическое устройство  
 SHIFTER – кольцевой сдвигатель  
 LSHIFT – логический сдвиг  
 ASHIFT – арифметический сдвиг

- DAG (data address generator) – генератор адресов данных
- Programm sequencer – устройство управления программой
- Timer – таймер
- SPORT (serial port) – последовательный порт
- Host Port – управляющий порт
- Clock PLL – генератор тактовых импульсов
- SCLK – генератор тактовых импульсов
- FIR – фильтр с конечной импульсной характеристикой (нерекурсивный фильтр)
- IIR (infinite impulse response) – фильтр с бесконечным импульсным откликом (рекурсивный фильтр)
- FFT (fast fourier transformation) – быстрое преобразование Фурье
- VC (video control) – видеоконтроллер
- FIFO (first-in, first-out) – память типа «первый пришел, первый вышел»

### Программа реализации нерекурсивного фильтра на процессоре ADSP 2181

0100	IDLE; {цикл ожидания прерывания SPORT0}
0101	CNTR = LENGTH-1; {задание длины циклического буфера коэффициентов}
0102	MR = 0; MX0 = DM(I0,M0), MY0 = PM(I4,M4); {обнуление накопителя, считывание значений $x(n-N+1)$ и $h(N-1)$ из памяти}
0103	DO Co UNTIL CE; {цикл умножения $h(k)*x(n-k)$ с накоплением}
0104	MR = MR + MX0*MY0(SS), MX0 = DM(I0,M0), MY0 = PM(I4,M4)
0105	MR = MR + MX0*MY0(RND); {умножение $h(0)*x(n-k)$ с накоплением}
0106	IF MV SAT MR; {и округление с насыщением}
0107	TX0 = MR1; {запись $y(n)$ из MR1 в регистр передачи SPORT0}
0108	JUMP W1; {переход к началу цикла (возврат к режиму ожидания следующего прерывания)}
000C	SR1 = RX0; {чтение отсчета данных $x(n)$ с порта SPORT0}
000D	DM(I0,M0) = SR1; {пересылка $x(n)$ в начало очереди данных}
000E	RTI; {возвращение в главную программу}

I0 – начало очереди данных.

I4 – начало буфера коэффициентов.

M0, M4 = 1-возвращаемые значения

TX0 – выходной отсчет (с округлением).

I0 – указывает начало очереди данных.

I4 – указывает начало таблицы коэффициентов.

Изменяемые регистры: SI, MX0, MY0, MR, CNTR.

Программа реализации цифрового фильтра содержит главный модуль, подпрограмму инициализации и модуль подпрограммы обработки данных. Подпрограмма обработки данных вызывается из главного модуля по прерыванию приема очередного отсчета данных.

Главный модуль обеспечивает управление синхронизированным вводом/выводом данных и их алгоритмической обработкой.

Ввод/вывод данных осуществляется через последовательный порт сигнального процессора SPORT0. Порт работает в режиме внутренней тактовой и кадровой синхронизации. Источниками – приемниками данных могут быть любые внешние последовательные устройства, в том числе АЦП и ЦАП, непосредственно сопрягаемые с последовательными портами процессора. При этом импульсы внутренней кадровой синхронизации порта, выводимые наружу, используются также в качестве импульсов частоты дискретизации преобразуемого в код сигнала. Возможен также вариант с внешней кадровой синхронизацией порта с частотой выборок сигнала, то есть с внешней частотой дискретизации. Ввод данных в процессор и их обработка выполняются по прерываниям приема данных, генерируемых портом, также как и вывод обработанных данных.

В главном модуле программы осуществляется настройка прерываний и системных регистров процессора, выполняется программирование последовательного порта SPORT0, организуется цикл ожидания прерывания приема данных. По генерируемому портом прерыванию вызывается подпрограмма обработки данных цифрового фильтра, при возврате из которой процессор снова переводится в цикл ожидания очередного отсчета данных из последовательного порта.

Значения  $h(N - 1)$ ,  $h(N - 2)$ , ...  $h(1)$ ,  $h(0)$  в указанной последовательности размещаются по возрастающим адресам памяти программ, образуя массив или циклический буфер коэффициентов. Они умножаются на соответствующие отсчеты входного сигнала  $x(n-N+1)$ ,  $x(n-N+2)$ , ...  $x(n-1)$ ,  $x(n)$ , которые образуют сигнальную память фильтра или циклический буфер отсчетов сигнала в памяти процессора. Ему соответствует структура данных типа очередь постоянной длины. Очередной отсчет сигнала  $x(n)$  после его обработки замещает самое старое значение сигнала в очереди  $x(n-N)$ . Адрес этого отсчета определяет начало очереди, которое циклически перемещается в буфере в процессе обработки сигнала. Такому способу обработки соответствует алгоритм программной реализации нерекурсивного фильтра без сдвига сигнальной памяти. Исходный адрес начала очереди соответствует обычно началу буфера в памяти данных.

В модуле подпрограммы, реализующем базовую операцию свертки осуществляется алгоритмическая обработка очередного отсчета сигнала, который считывается с приемного регистра порта RX0. Выходной отсчет данных записывается в передающий регистр последовательного порта TX0. Все коэффициенты и значения данных для нерекурсивного фильтра представляются в формате 1.15.

**Программа формирования гармонического  
колебания методом аппроксимации  
на процессоре ADSP 2181**

Программа генерации синуса (sin.dsp)

```
.MODULE/RAM sin;
.VAR/DM sin_coeff[5];
.INIT sin_coeff: H#3240, H#0053, H#AACC, H#08B7, H#1CCE;
(инициализация коэффициентов)
.ENTRY sin;
sin: I3 = ^sin_coeff; (загрузить указатель на коэффициенты)
    AY0 = 0x4000;
    AR = AX0, AF = AX0 AND AY0; (проверка квадранта синуса)
    IF NE AR = -AX0; (инверсия)
    AY0 = 0x7FFF; (удаление знакового разряда)
    AR = AR AND AY0;
    MY1 = AR; (вычисление синуса)
    MF = AR * MY1(RND), MX1 = DM(I3, M3);
    MR = MX1 * MY1(SS), MX1 = DM(I3, M3);
    CNTR = 3;
    DO approx UNTIL CE;
    MR = MR + MX1 * MF(SS);
approx:
    MF = AR * MF(RND), MX1 = DM(I3, M3);
    MR = MR + MX1 * MF(SS);
    SR = ASHIFT MR1 BY 3 (HI); (преобразование к формату
1.15)
    SR = SR OR LSHIFT MR0 BY 3 (LO);
    AR = PASS SR1; (округление и инвертирование)
    IF LT AR = PASS AY0;
    AF = PASS AX0;
    IF LT AR = -AR;
    RTS;
.ENDMOD;
```



**Программа реализации алгоритма БПФ  
на процессоре ADSP 2181**

{ Параметры вызова:

inplacereal = действительные входные данные в скремблированном порядке

inplaceimag = все нули (предполагается, что входные данные действительные числа)

twid\_real = значения косинуса

twid\_imag = значения синуса

groups = N/2

bflys\_per\_group=1

node\_space=1

Возвращаемые значения

inplacereal = действительные результаты БПФ по порядку

inplaceimag = мнимые результаты БПФ по порядку

Изменяемые регистры

I0, I1, I2, I3, I4, I5, L0, L1, L2, L3, L4, L5

M0, M1, M2, M3, M4, M5

AX0, AX1, AY0, AY1, AR, AF

MX0, MX1, MY0, MY1, MR, SB, SE, SR, SI

Изменяемые области памяти

inplacereal, inplaceimag, groups, node\_space, bflys\_per\_group, blk\_exponent }

.MODULE fft;

.CONST log2N=10, N=1024, nover2 = 512, nover4 = 256;

.EXTERNAL twid\_real, twid\_imag;

.EXTERNAL inplacereal, inplaceimag;

.EXTERNAL groups, bflys\_per\_group, node\_space;

.EXTERNAL bfp\_adj;

.ENTRY fft\_strt;

fft\_strt: CNTR=log2N-2; {Инициализация счетчика стадий}

M0=0;

M1=1;

L1=0;

```
L1=0;
L3=0;
L4=%twid_real;
L5=%twid_imag;
L6=0;
SB=-2;
```

```
{----- Стадия 1 -----}
```

```
I0=^inplacereal;
I1=^inplacereal+1;
I2=^inplaceimag;
I3=^inplaceimag+1;
M2=2;
CNTR=nover2;
AX0=DM(I0,M0);
AY0=DM(I1,M0);
AY1=DM(I3,M0);
DO group_lp UNTIL CE;
group_lp: AR = AX0 + AY0, AX1=DM(I2,M0);
SB=EXPADJ AR, DM(I0,M2) = AR;
AR = AX0-AY0;
SB=EXPADJ AR;
DM(I1,M2) = AR, AR = AX1+AY1;
SB=EXPADJ AR, DM(I2,M2) = AR;
AR = AX1-AY1, AX0 = DM(I0,M0);
SB=EXPADJ AR, DM(I3,M2) = AR;
AY0 = DM(I1,M0);
AY1=DM(I3,M0);
CALL bfp_adj;
```

```
{----- Стадии с 2 до N-1 -----}
```

```
DO stage_loop UNTIL CE; {вычисление всех стадий БПФ}
I0=^inplacereal; {I0->x0 в 1-ой группе стадии}
I2=^inplaceimag; {I2->y0 в 1-ой группе стадии}
SI=DM(groups);
SR=ASHIFT SI BY -1(LO); {groups/2}
```

## Приложения

```
DM(groups)=SR0; { groups=groups/2 }
CNTR = SR0; { CNTR=счетчик групп }
M4=SR0; { M4=модификатор коэффициента с тильдой }
M2=DM(node_space); { M2=модификатор пересекающейся об-
ласти }
I1=I0;
MODIFY(I1,M2); { I1->y0 в 1-ой группе стадии }
I3=I2;
MODIFY(I3,M2); { I3->y1 в 1-ой группе стадии }
DO group_loop UNTIL CE;
I4=^twid_real; { I4->C от W0 }
I5=^twid_imag; { I5->(-S) от W0 }
CNTR= DM(bflys_per_group); CNTR=счетчик bfly
MY0=PM(I4,M4), MX0=DM(I1,M0);
MY1=PM(I5,M4), MX1=DM(I3,M0)
DO bfly_loop UNTIL CE;
MR=MX0*MY1(SS),AX0=DM(I0,M0);
MR=MR+MX1*MY0(RND),AX1=DM(I2,M0);
AY1=MR1,MR=MX0*MY0(SS);
MR=MR-MX1*MY1(RND);
AY0=MR1,AR=AX1-AY1;
SB=EXPADJ AR,DM(I3,M1)=AR; { проверка увеличения бит }
AR=AX0-AY0,MX1=DM(I3,M0), MY1=PM(I5,M4);
SB=EXPADJ AR, DM(I1,M1)=AR; { проверка увеличения бит }
AR=AX0+AY0, MX0=DM(I1,M0), MY0=PM(I4,M4);
SB=EXPADJ AR,DM(I0,M1)=AR; { проверка увеличения бит }
AR=AX1+AY1;
SB=EXPADJ AR,DM(I2,M1) = AR; { проверка увеличения бит }
bflyloop: MODIFY(I0,M2); { I0->первое x0 в следующей группе }
MODIFY(I1,M2); { I1->первое x1 в следующей группе }
MODIFY(I2,M2); { I2->первое y0 в следующей группе }
MODIFY(I3,M2); { I3->первое y1 в следующей группе }
group_loop: CALL bfp_adj; { компенсация увеличения бит }
SI=DM(bflys_per_group);
SR=ASHIFT SI BY 1(LO);
stage_loop: DM(node_space)=SR0;
DM(bflys_per_group)=SR0;
{----- Последняя стадия -----}
```

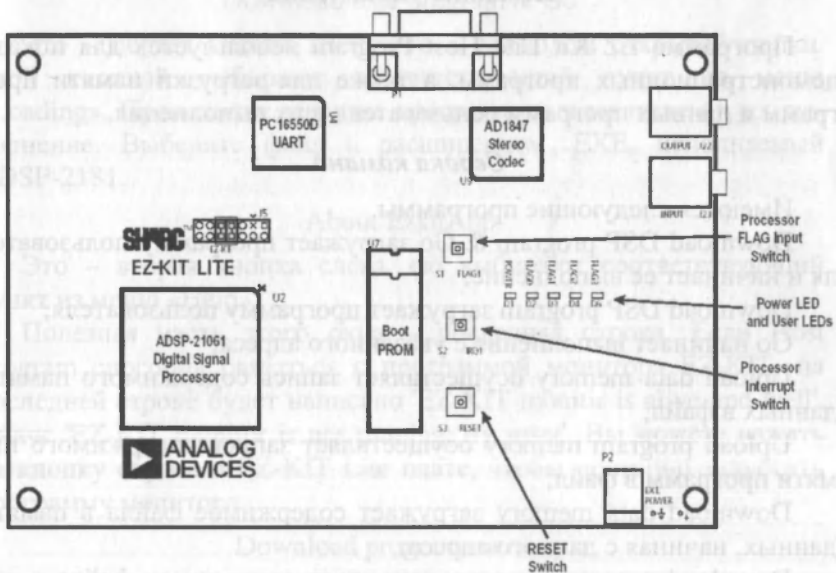
```

I0=^inplacereal;
I1=Ainplacereal+nover2;
I2=^inplaceimag;
I3=^inplaceimag+nover2;
CNTR=nover2;
M2=DM(node_space);
M4=1;
I4=^twid_real;
I5=^twid_imag;
MY0=PM(I4,M4), MX0=DM(I1,M0);
MY1=PM(I5,M4), MX1=DM(I3,M0);
DO bflyjp UNTIL CE;
MR=MX0*MY1(SS), AX0=DM(I0,M0);
MR=MR+MX1*MY0(RND), AX1=DM(I2,M0);
AY1=MR1, MR=MX0*MY0(SS);
MR=MR-MX1*MY1(RND);
AY0=MR1, AR=AX1-AY1;
SB=EXPADJ AR,DM(I3,M1)=AR;
AR=AX0-AY0, MX1=DM(I3,M0), MY1=PM(I5,M4);
SB=EXPADJ AR,DM(11,M1)=AR;
AR=AX0+AY0, MX0=DM(11,M0), MY0=PM(I4,M4);
SB=EXPADJ AR,DM(I0,M1)=AR;
AR=AX1+AY1;
SB=EXPADJ AR, DM(I2,M1)=AR;
bfly_lp: CALLbfp_adj;
RTS;
.ENDMOD;

```

### Платформа EZ-Kit Lite

Платформа EZ-Kit Lite входит в состав комплекта пользователя, содержащего также программное обеспечение, кабель RS232 и источник питания. Комплект пользователя предназначен для изучения семейства процессоров ADSP21xx и создания оригинальных программ реализации устройств ЦОС. Плата разработана для демонстрации возможностей сигнального процессора ADSP-2181 и может быть использована как платформа для разработки новых приложений ADSP-2181. На плате собраны (см. рисунок): процессор ADSP-2181, память EPROM, аналого-цифровой интерфейс AD1847 CODEC, служебные схемы и разъемы.



Платформа EZ-Kit Lite

Доступ к памяти EPROM осуществляется через порт прямого побайтового доступа (порт BDMA). Аналого-цифровой интерфейс AD1847 CODEC подключен к последовательному порту SPORT0 процессора, но при необходимости он может быть использован и

для других целей. Отключение аналого-цифрового интерфейса производится путем установки низкого уровня сигнала CODECDIS, сигнал этот доступен на разъеме P3. Последовательный порт SPORT 1 обеспечивает связь процессора с персональным компьютером (PC) через интерфейс RS-232 (COM-порт). Порт работает в альтернативном режиме как приемник сигналов FI и IRQ1 и источник сигнала FO. Сигналы последовательных портов связи согласуются с помощью установленной на плате микросхемы ADM232AAR. Вход данных, передаваемых в компьютер, этой микросхемы подключен к выводу FO SPORT 1, а выход данных с компьютера соединен с выводами FI и IRQ1 SPORT 1. Вход IRQ1 используется в качестве детектора стартового бита принимаемого последовательного кода. Протокол обмена RS-232 эмулируется программно процессором ADSP-2181.

### Команды управления

Программа EZ Kit Lite Host Program используется для показа демонстрационных программ, а также для загрузки памяти программ и данных программ пользователя и их выполнения.

#### Сводка команд

Имеются следующие программы.

Download DSP program & Go загружает программу пользователя и начинает ее выполнение;

Download DSP program загружает программу пользователя;

Go начинает выполнение с указанного адреса;

Upload data memory осуществляет запись содержимого памяти данных в файл;

Upload program memory осуществляет запись содержимого памяти программ в файл;

Download data memory загружает содержимое файла в память данных, начиная с данного адреса;

Download program memory загружает содержимое файла в память программы, начиная с данного адреса;

About указывает, установлена ли последовательная связь с платой.

Все команды доступны через главное Меню и его элементы. Обратите внимание, что дополнением к стандартным элементам Главного Меню типа «Файла» и «Справки» служат несколько оп-

ций, связанных с EZ-KIT Lite. Все опции Меню описаны подробно в следующих разделах.

Справка (еще один элемент главного меню).

### *Меню File*

Exit – выход из EZ-KIT Host Program.

Меню View. Меню View имеет дело с инструментальной панелью и строкой состояния.

### *Инструментальная панель*

Выбором этой опции осуществляют вставку и удаление панели инструментов. Панель инструментов – серая полоса, находящаяся ниже полосы меню, с несколькими маленькими квадратными кнопками слева. Эти кнопки обеспечивают быстрый доступ к командам главного меню.

### *Download user program & Go*

Это – первая кнопка из левого с символом открытой папки. Этой кнопкой выбирают соответствующий пункт из меню «Loading». Происходит загрузка программ пользователем и их выполнение. Выберите файл с расширением .EXE, исполняемый ADSP-2181.

### *About EzkitApp*

Это – вторая кнопка слева, ею выбирают соответствующий пункт из меню «Help».

Полезная часть этого окна – последняя строка. Если Host Program способна связаться с программой монитора EZ-KIT, на последней строке будет написано 'EZ-KIT monitor is alive and well'. Иначе 'EZ-KIT monitor is not running; try reset'. Вы можете нажать на кнопку сброса в EZ-KIT Lite плате, чтобы повторно запустить программу монитора.

### *Download program memory*

Это – третья кнопка слева с символом «P» и вниз указывающей стрелкой. Этой кнопкой выбирают соответствующий пункт из меню 'Loading'. Память программы из файла PCI загружается в память DSP, начиная с определенного адреса. Для выполнения программы нужно выбрать команду платы «GO» (см. соответствующий пункт в меню «Loading»).

### Download data memory

Это – четвертая кнопка слева с символом «D» и вниз указывающей стрелкой. Этой кнопкой выбирают соответствующий пункт из меню «Loading». Происходит загрузка памяти данных из файла в память DSP, начиная с определенного адреса. Для выполнения программы нужно выбрать команду «GO» (см. соответствующий пункт в меню Loading).

### Upload program memory

Это – пятая кнопка с символом «P» и стрелкой, указывающей вверх. Этой кнопкой выбирают соответствующий пункт из меню «Loading». Этим осуществляется запись памяти программ DSP в файл. Вы указываете начальный адрес и число ячеек памяти данных, которые будут записаны (см. соответствующий пункт меню «Loading»).

### Upload data memory

Это – шестая кнопка с символом «d» и стрелкой, указывающей вверх. Этой кнопкой выбирают соответствующий пункт из меню «Loading». Этим осуществляется запись данных DSP в файл. Вы указываете начальный адрес и число данных, которые будут записаны (см. соответствующий пункт меню «Loading»).

### Строка состояния

Выбором этой кнопки осуществляют вставку и удаление строки состояния основного окна.

Когда строка состояния присутствует, в ней дано более детализированное описание выбранного пункта меню.

### Меню Demo

Из опции меню выбирают одну из доступных демонстрационных программ для выполнения на EZ-KIT Lite.

DTMF – формирователь двухтонового многочастотного сигнала.

Эта демонстрационная программа генерирует двухтоновый сигнал (DTMF), который используется в телефонной сети для передачи сигнала. Сигнал DTMF составлен из двух одиночных тонов частоты. Полная реализация DTMF стандартного генератора тонов может генерировать 16 различных сигналов (только 12, обычно используются на телефонных трубках США).



В процессе получения DTMF тона, EZ-KIT генерирует два синусоидальных колебания, масштабирует и суммирует их. далее выводит результаты на ЦАП для преобразования в аналоговый сигнал. При запуске этой программы, EZ-KIT генерирует тон набора кода и передает его в выходной громкоговоритель. Возможны следующие параметры.

### Generate dial tone

Этот параметр может быть выбран нажатием соответствующей кнопки или печатью символа 't'. Генерируется стандартный сигнал набора, являющийся суммой непрерывных синусоидальных колебаний.

### Generate DTMF digits

Этот параметр может быть выбран нажатием соответствующей кнопки или печатью символа 'd'. Номер телефона программируется в демонстрационной программе. Выбирайте этот параметр, чтобы генерировать номер телефона.

### Close

Эта операция завершает программу и восстанавливает программу EZ-KIT монитора перед возвратом пользователя в меню. Она позволяет выбрать новую программу.

Filtering – программа реализации рекурсивного символа.

Выберите параметры программы, демонстрирующей фильтр. Эта программа показывает эффективность 4 полосовых фильтров по сравнению с сигналом на входе микрофона или сгенерированного шумового источника.

Выберите вид программы, демонстрирующий фильтр. Эта программа показывает эффект влияния 4 полосовых фильтров по сравнению с сигналом на входе микрофона или сгенерированного шумового источника.

АЦП AD1847 оцифровывает сигнал микрофона и передает данные на последовательный порт DSP. DSP считывает данные с последовательного порта и повторно передает их обратно на ЦАП. ЦАП преобразовывает данные в аналоговый сигнал, который подается в громкоговоритель. Над данными никакая цифровая обработка не выполняется. Когда Вы говорите в микрофон, то должны слышать свой голос в громкоговорителе.

Верхние кнопки строки используются для ввода голоса из микрофона.

Фильтры являются полосовыми и их АЧХ равномерно располагаются на логарифмической оси частоты. Все фильтры – 256-го порядка и имеют неравномерность АЧХ 0,1.

Фильтр Нижняя граница Полоса пропускания Верхняя граница

FIR1 0-269 Hz 328-448 Hz 547-4000 Hz

FIR2 0-426 Hz 521-710 Hz 866-4000 Hz

FIR3 0-675 Hz 825-1125 Hz 1375-4000 Hz

FIR4 0-1070 Hz 1308-1783 Hz 2179-4000 Hz

Close

Эта операция завершает программу EZ-KIT демонстрации реализации фильтра и позволяет выбрать новую программу.

Echo Cancellation – подавление эхо сигналов.

Эта программа моделирует эхо и демонстрирует реализацию алгоритма его подавления.

АЦП AD1847 оцифровывает сигнал аналогового источника и передает данные в последовательный порт DSP. DSP формирует эхо сигнал, суммируя вводимый сигнал с сигналом моделируемого эхо канала. Эхо канал состоит из фильтра, формирующего задержку, и адаптивного FIR фильтра с 16 отводами. Обработанный сигнал посылается в ЦАП для восстановления. Первоначально программа выводит сигнал без ЭХО.

Next

При нажатии этой кнопки, по программе вводится эхо-сигнал. Дальнейшее нажатие кнопки изменит характеристику подавляющего фильтра.

Close

Эта операция завершает программу и восстанавливает программу EZ-KIT перед возвратом пользователя к меню. Она позволяет выбрать новую программу.

ADPCM (адаптивный дифференциальный импульсно-кодированный модулятор)

ADPCM состоит из ряда алгоритмов сжатия речи действующих в реальном времени. Каждый период получения выборки, линейный прогнозирующий фильтр генерирует предсказанную выборку.

Разность между предсказанной выборкой и ее фактическим значением посылается в канал связи. Так как динамический диапазон дифференциальной ошибки значительно ниже чем у речевого сигнала, то этим достигается более низкая разрядность передаваемого сообщения.

### ADPCM coding

Эта опция допускает ADPCM (кодирование / декодирование). Каждая цифровая выборка сигнала микрофона из АЦП сначала кодируется, затем декодируется с использованием ADPCM. После того, как она декодируется, выборка следует обратно в ЦАП для восстановления. Красный LED светит, когда кодирование ADPCM включено.

### Straight through

При этой опции кодирования не происходит. Цифровые выборки сигнала микрофона следуют непосредственно в ЦАП для восстановления.

### Close

Эта операция завершает программу и восстанавливает программу EZ-KIT монитора перед возвратом пользователя к меню. Она позволяет выбрать новую программу.

### 7.8K LPC (линейный кодер речи с предсказанием)

Эта программа моделирует 7.8K LPC..

### Interrupt button on EZ-KIT Lite

Нажмите на кнопку Interrupt button on EZ-KIT Lite для переключения режима работы речи без кодирования и с 7.8K LPC. Красный LED светит, когда происходит кодирование LPC происходит в действительности.

### Close

Эта операция завершает программу и восстанавливает программу EZ-KIT монитора перед возвратом к меню. Она позволяет выбрать новую программу.

## **Симулятор процессора SIM 2181**

Симулятор процессора ADSP 21\*\* предназначен для моделирования процессора (в нереальном времени) на ПЭВМ. Симулятор представляет собой программное обеспечение, ориентированное на работу в диалоговом режиме с помощью окон дисплея ПЭВМ и моделирования выполнения инструкций ADSP 21\*\* и отладки программ пользователя. Конфигурация симулятора соответствует архитектуре системы потребителя.

Симулятор обеспечивает следующие функции:

- моделирование на уровне инструкций загрузки и указание команд;
- симуляцию портов с использованием файла управления данными;
- симуляцию внутренних и внешних прерываний;
- ассемблирование и дисассемблирование инструкций ADSP 21\*\*;
- выполнение условий прерываний;
- показ содержимого регистров процессора и взаимодействия их содержимого.

### **Описание работы симулятора SIM2181 для ЦСП ADSP2181 фирмы Analog Devices**

Все команды, которые будут рассматриваться в дальнейшем, должны быть напечатаны в командной строке с учетом указания адреса (C:\ >) если работа идет в основном меню компьютера. В большинстве случаев, когда запускается симулятор из рабочего каталога, в котором находится сам симулятор, этого делать нет необходимости.

#### ***Системные требования***

На первом шаге в разработке системы DSP нужно определить какие возможности системы необходимы. Эти возможности будут зависеть от типов реализуемых алгоритмов, типов используемых сигналов и типов устройств ввода-вывода, которые должны быть соединены с процессором DSP.

Оценка необходимого размера для памяти данных и памяти программы основана на количестве получаемых вводом – выводом данных и количеством выполняемой обработки. Оцененный размер

программы, созданной, учитывая используемые алгоритмы, также определяет память обязательной программы.

Например, можно строить систему обработки речевых сигналов, при этом выбирают соответствующий алгоритм, типа LPC, для сжатия речи. Этот алгоритм требует некоторого количества хранения данных которое определено алгоритмом и некоторое количество памяти (хранения) команды программы определенное количеством кода, необходимого для создания алгоритма.

Конечно, этого мало для реализации алгоритма, это лишь необходимые условия – только приближенные. Для этого необходимо 4 Кбайт памяти данных и 1 Кбайт память программы. Внутренняя память ADSP-2181 достаточно большая. Можно использовать аналого-цифровой ввод – вывод для аналогового сигнала. При этом регистр нужно подключить к аудио кодер-декодер с последовательным портом DSP. Чтобы процессор DSP был квалифицированным и не отставал от скорости выборки данных, приходящих в от ввода – вывода: 33 МГц для ADSP-2181 больше чем достаточная скорость.

### *Описание файла архитектуры*

Система должна быть описана в файле описания архитектуры. Система ADSP-2181 уже определена, этот шаг был уже сделан за Вас. Файл назвали ADSP2181.ACH (файл, произведенный системным составителем программы) включен в EZ-KIT Lite программное обеспечение. Текстовый файл содержит показанное ниже описание архитектуры типичной системы ADSP-2181 и используется как ввод(вход) для системного составителя программы.

```
.system demo;  
.adsp2181;  
.mmap0;  
.seg/pm/ram/abs=0/code/data int_pm_lo[8192];  
.seg/pm/ram/abs=8192/code/data int_pm_hi[8192];  
.seg/dm/ram/abs=0/data int_dm_lo[8192];  
.seg/dm/ram/abs=8192/data int_dm_hi[8160];  
.endsys;
```

Первые три строки файла определяют название системы как демонстрационная версия использование ADSP-2181 с его программируемым ПЗУ загружают допускаемый полнометражный фильм ( MMAP).

Этот текстовый файл (.SYS файл) используется как ввод/вывод системного составителя программы, чтобы создать файл АСН.

Противоречия флага между программным обеспечением и использованием аппаратных средств.

Файл архитектуры, названный ADSP2181 АСН, снабжен программным обеспечением системы EZ-KIT Lite. Эта система содержит составителя программы также как номер из дополнительных программных средств и библиотек.

### Запуск симулятора

Симулятор реализует код в виртуальной среде, что позволяет проверить программное обеспечение без подключения фактической системы аппаратных средств. Процесс моделирования используется для реализации программной работы прежде, чем это выполнено на аппаратных средствах. Симулятор полностью эмитирует работу ADSP2181 и не нуждается в подключении отладочной платы с ADSP2181.

Запуск симулятора производится под управлением операционной системы MS-DOS и при помощи операционной оболочки типа Norton Commander (запуск симулятора под оболочкой Windows 3.xx может привести к зависанию системы) с минимальной конфигурацией компьютера 386SX/DX с не менее чем 4 Мбайт ОЗУ. Перед запуском симулятора необходимо войти в директорию, где находится симулятор SIM2181 (название запускающего файла SIM2181.EXE) и убедиться в наличии в той же директории файлов EZKIT\_LT.ACH или ADSP2181.ACH (эти файлы описывают архитектурное строение процессора ADSP2181). Если отсутствует какой-то из перечисленных файлов, то необходимо скопировать данный файл из другого каталога. Для запуска программы необходимо в командной строке набрать:

```
SIM2181.EXE -A ADSP2181.ACH -E < имя запускаемой программы > ( например echo.exe )
```

К примеру: sim2181.exe -a adsp2181.ach -e echo.exe и далее нажать «ввод».

Если симулятор не запустился, то необходимо проверить правильность написания текста в командной строке и правильность названия файла находящегося в директории.

### Содержимое главного меню симулятора SIM2181

File Register Memory Stacks Command Execution Display Misc Exit

#### Главное меню симулятора SIM2181

После того как симулятор запустился, на время появится рамка с логотипами

Analog Devices ADSP2181 Simulator. Когда она пропадет появится командное меню, которое необходимо закрыть нажатием клавиши Esc.

Далее в окне появится главное меню симулятора SIM2181 (шапка заголовков оболочки симулятора.).

Рассмотрим более детально некоторые из них. На рисунке показан пример открытия окон (окна в симуляторе открываются нажатием клавиши Enter и закрываются нажатием клавиши Esc).

#### Меню File

Load Executable File – загрузить исполнимый файл (пользовательская программа). Этим меню необходимо воспользоваться в том случае, если запуск симулятора производился без подачи в командной строке имени исполнимой программы т.е. вот так: sim2181.exe -a ADSP2181.ACH

Load Window File – загрузить файл состояния открытых окон. Эта опция может пригодиться для того, чтобы при каждом последующем запуске симулятора не приходилось заново открывать необходимые окна. Эту опцию разумно применять лишь в том случае, когда совместно с ней используется следующая опция.

Save Window File – сохранить в файле положение окон на экране.

Display File – показать содержимое файла.

#### File

- Load Executable File
- load Window File
- Save Window File
- Display File
- Playback keystroke File
- Start Keystroke Recording
- Stop Keystroke Recording
- Operating System Shell
- Shell an OS Command
- SPORT Files

Playback keystroke File – воспроизвести последнее нажатие клавиши сохр. в файле.

Start Keystroke Recording – задусть последнее нажатие клавиши сохраненного файла.

Stop Keystroke Recording – остановить запуск последнего нажатия клавиши сохраненного файла.

Operating System Shell – временно выйти в DOS.

Shell an OS Command – выполнить команду DOS.

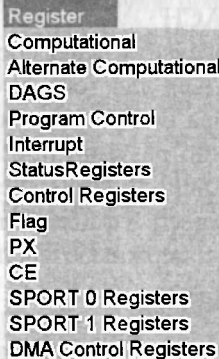
SPORT Files – задать соответствие между файлами на диске и входными/выходными потоками данных последовательного порта.

### *Меню Register*

Опция Computational позволяет показать содержимое основных активных регистров процессора [AX1, AX0, AR и т.д.] (показывает регистры вычислителя).

Alternate Computational показывает на экране содержимое дополнительных наборов регистров процессора (это резерв основных регистров).

Опция DAGS позволяет отобразить на экране содержимое генераторов адресов процессора (устройства формирователя адреса).



Register  
Computational  
Alternate Computational  
DAGS  
Program Control  
Interrupt  
StatusRegisters  
Control Registers  
Flag  
PX  
CE  
SPORT 0 Registers  
SPORT 1 Registers  
DMA Control Registers

Program Control показывает регистры управления программой, т.е. позволяет осуществить доступ к регистру PC (Program Counter), при помощи которого можно управлять очередностью исполнения команд, проще говоря, установить процессор для исполнения инструкции, находящейся по определенному адресу.

Опция Interrupt позволяет имитировать различные прерывания, приходящие на процессор (регистры управления прерываниями).



Status Registers позволяет наблюдать процесс изменения состояния флагов процессора, как результат той или иной операции, исполняемой в нем.

Control Registers показывает регистры управления.

Flag показывает регистр флагов.

PX регистр ввода значений.

SE показать регистры.

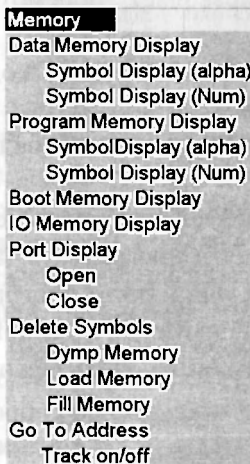
SPORT 0 Registers позволяет показать регистры первого последовательного порта.

SPORT 1 Registers позволяет показать регистры второго последовательного порта.

DMA Control Registers показывает регистры системы ПДП.

### Меню Memory

При помощи данной опции пользователь получает доступ к различным участкам памяти процессора.



Data Memory Display – показывает содержимое памяти данных процессора и позволяет редактировать их.

Symbol Display (alpha) – показывает символы памяти данных в алфавитном порядке.

Symbol Display (Num) – показывает значения памяти данных в порядке нумерации.

Program Memory Display – показывает память программ пользователя и позволяет редактировать программу.

Symbol Display (alpha) – показывает символы памяти программы в алфавитном порядке.

Symbol Display (Num) – показывает значения памяти программы в порядке нумерации.

Boot Memory Display – показывает содержимое памяти загрузки.

IO Memory Display – показывает содержимое памяти ввода-вывода.

Port Display – показывает соответствие между портом и файлом на диске.

Open – открывает меню установки соответствующих портов и файлов на диске.

Close – закрывает меню установки соответствующих портов и файлов на диске.

Delete Symbols – удаляет символы.

Dump Memory – позволяет просмотреть блок памяти в 16-ных кодах.

Load Memory – позволяет загружать содержимое памяти данных или памяти программ.

Fill Memory – позволяет заполнить блок памяти каким либо значением.

Go To Address – позволяет выполнить переход на заданный адрес памяти данных и памяти программ (в зависимости от того какое окно (памяти данных или памяти программ) активно в данный момент: Load Memory, Dump Memory или Fill Memory ).

Track on/off – включить/выключить режим слежения.

### *Меню Stacks*



Stacks  
Loop Address  
Loop Counter  
PC  
Status

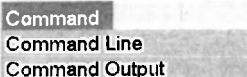
Loop Address – показать содержимое стека адресов циклов.

Loop Counter – показать содержимое стека значений переменной.

PC – показать содержимое счетчика команд.

Status – показать регистр состояния стека.

### Меню Command



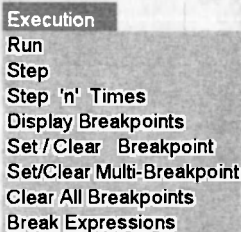
Command  
Command Line  
Command Output

Command Line – задать командную строку.

Command Output – создать файл для записи сообщения программы.

### Меню Execution

Эта опция симулятора SIM2181 позволяет получить контроль над исполнением программы пользователя.



Execution  
Run  
Step  
Step 'n' Times  
Display Breakpoints  
Set / Clear Breakpoint  
Set/Clear Multi-Breakpoint  
Clear All Breakpoints  
Break Expressions

Run – производит запуск программы пользователя с адреса, указанного в регистре PC ( смотрите в окне Program Control).

Step (клавиша F10) – служит для пошагового выполнения программы, исполняет одну инструкцию процессора, расположенную по адресу указанному в регистре PC (см. в окне Program Control).

Step 'n' Times – служит для пошагового исполнения программы, исполняет n инструкций процессора, расположенную по адресу указанному в регистре PC (см. в окне Program Control).

Display Breackpoints\* – показать точки останова процессора.

Set / Clear Breakpoint – установить/сбросить точку останова процессора (срабатывает при достижении определенного количества переходов через нее).

Set/Clear Multi-Breakpoint – установить/сбросить точку приостановки процессора.

Clear All Breackpoints – позволяет снять все точки останова в программе.

\* (Breackpoints – точки останова предназначены для отладки программ пользователя они прерывают исполнение программы для последующего ее анализа и отладки).

## Предметный указатель

- А**
- Адрес 17, 31
  - Адресация 18
    - косвенная 31
    - линейная 31, 69
    - непосредственная 42, 43, 90
    - прямая 42, 74, 91
    - циклическая 31
  - Аккумулятор 26, 68
  - Аналого-цифровое преобразование 5, 7
  - Алгоритм 49
  - Арифметико-логическое устройство 17, 24
  - Арифметика
    - с фиксированной точкой 12
    - с плавающей точкой 13, 100
  - Архитектура процессора
    - фон Неймана 17
    - гарвардская 17
  - Ассемблер 20
- Б**
- Байт 14, 75
  - Бит 12
  - Буферизация 36
- В**
- Ввод-вывод 17
  - Векторный сопроцессор 107
  - Время доступа к памяти 60
  - Вычитание 25
  - Вычислитель БПФ 50, 128, 143
- Г**
- Генератор тактовых импульсов 17
- Д**
- Дискретизация сигнала 5
- Длинная команда** 21, 79, 95
- Длина команды** 24, 71
- Доступ к памяти прямой (ПДП)**
- З**
- Запрос на прерывание 71
- И**
- Интерфейс параллельный 18
- К**
- Команды
    - вызова подпрограммы 43
    - пересылки данных 44
    - прямой адресации 42
    - управления программой 49
  - Конвейер 18, 71, 99
  - Кольцевой сдвигатель 17, 28
- Л**
- Линейная
    - операция 10
    - система 9
- М**
- Мантисса 14, 100
  - Модем 24, 58, 67
  - Микропроцессор 101
  - Машинный цикл 71
  - Микроконтроллер 73
- Н**
- Нерекурсивный цифровой фильтр 11, 46
  - Нейронная сеть 109
- О**
- Операнд 25, 26
  - Операции ввода/вывода 61
  - Отладка программы 148, 154

## Предметный указатель

- П**
- Память
- двухпортовая 60
  - динамическая 67
  - статическая 56
  - произвольного доступа 56,81
  - энергонезависимая 127
- Перенос 25
- Подпрограмма 69, 140
- Порядок 13, 14
- Порт ввода/вывода 17
- коммутационный 75
  - параллельный 18
  - последовательный 17, 34
  - хост-порт 67
- Представление чисел (операндов)
- в дополнительном коде 12
  - в обратном коде 12
- Прерывания
- аппаратные 134
  - немаскируемые 38, 71
  - программируемые 39
- Программа
- главная 46
  - инициализации 46
  - вычислений 46
- Периферия процессора 17
- Процессор цифровой 16
- мультимедийный 84
- Р**
- Разряд двоичный 5, 39
- Регистр
- модификатор 31
  - адреса длины цикла 31
  - индексный 31
  - состояния 25, 37, 70
  - буферный 17
- С**
- Система
- команд 42, 72
  - прерываний 43
  - «на кристалле» 134
- Способ адресации 42
- Стек 33, 70
- Страница памяти 91
- Счетчик команд 31, 70, 92
- Сигнальный процессор 1, 3
- Т**
- Таймер 17, 39
- Точность обычная 28
- У**
- Умножитель 27, 87
- Умножитель-аккумулятор 26
- Устройство ввода-вывода 60
- Ф**
- Файл регистров 17, 61
- Фильтрация 12, 46
- Формат чисел
- с фиксированной точкой 12
  - с плавающей точкой 13, 100
- Формирователь адресов 30, 68
- Формирователь гармонического колебания
- табличным методом 48
  - методом аппроксимации 49
- Ш**
- Шина адресов данных 22
- адресов команд 22
  - глобальная 69
  - данных 22
  - команд 22
  - периферии 83
  - результата 22
- Э**
- Эмулятор 110
- Я**
- Ядро процессора 17, 22, 57

## Список литературы

1. **Баскаков С.И.** Радиотехнические цепи и сигналы. – М: Высшая школа, 1988. –448 с.
2. **Цифровая обработка сигналов:** Справочник. Л. М. Гольденберг, Б. Д. Матюшин, Л.М. Поляк – М: Радио и связь, 1985. –312 с.
3. **Сергиенко А.Б.** Цифровая обработка сигналов. – СПб: Питер, 2002. – 608 с.
4. **Айфичер Э., Джервис Б.** Цифровая обработка сигналов. Перевод с английского. – М-СПБ-К: Вильямс, 2004. –992 с.
5. **Марков С.** Цифровые сигнальные процессоры. Кн.1. – М: Микроарт, 1996. – 144 с.
6. **Сперанский В.С.** Цифровые сигнальные процессоры и их применение для формирования и обработки сигналов – М.: МТУСИ, 1997. –92 с.
7. **Корнеев В.В., Киселев А.В.** Современные микропроцессоры – М: Нолидж, 2000. – 320 с.
8. **Глинченко А.С., Голенок А.И.** Принципы организации и программирования сигнальных процессоров ADSP 21\*\* – Красноярск: КГТУ, 2000. – 86 с.
9. **Солонина А., Улахович Д., Яковлев Л.** Цифровые процессоры обработки сигналов – СПб: БХВ, 2000. – 512 с.
10. **Куприянов М.С., Иванова В.Е., Матвиенко Н.И.** Микропроцессоры фирмы Motorola. Ч 2. – М: Motorola, 1998. – 51 с.
11. **Круг П.Г.** Процессоры цифровой обработки сигналов. – М: Изд. МЭИ, 2002. – 128 с.
12. **Кузин А.А.** Процессор TMS320C40\*. – Н. Новгород: 1999, 210 с.
13. **Процессоры** цифровой обработки сигналов фирмы Texas Instruments – М: ЗАО СКАН Инжиниринг, 1998. – 70 с.
14. **Мальцев П.П., Гарбузов Н.И., Шарапов А.П., Кнышов Д.А.** Программируемые логические ИМС на КМОП структурах и их применение. – М: Энергоатомиздат, 1998, – 98 с.
15. **Соловьев В.В., Васильев А.С.** Программируемые логические интегральные схемы и их применение. – Минск: Белорусская наука, 1998, – 270 с.
16. **Кузелин М.О., Кнышов Д.А., Зотов В.Ю.** Современные семейства ПЛИС фирмы XILINX. – М: Горячая линия-Телеком, 2004. – 440 с.

17. **Пяткин А.К., Никитин М.В.** Реализация на ПЛИС быстрого преобразования Фурье для алгоритмов ЦОС в многофункциональных РЛС // Цифровая обработка сигналов. – 2003. – № 3. С. 21 – 26.

18. **Шибяев С.В.** Проблемы и методы повышения производительности цифровых сигнальных процессоров // Радиотехника. – 2005. – № 9. С. 43 – 49.

19. **Фефилов И.И., Матюшин П.М.** Перестраиваемое процессорное ядро – основа устройств «Система на кристалле» // Цифровая обработка сигналов. – 2003. – № 3. С. 32 – 37.

20. **Морган Д.** Устройство обработки сигналов на поверхностных акустических волнах. – М.: Радио и связь, 1990. – 416 с.

21. **Применение методов Фурье-оптики.** Пер. с англ., под ред. Г. М. Старка. – М.: Радио и Связь, 1988. – 535 с.

22. **Кочемасов В. Н.** и др. Акустические Фурье-процессоры. – М.: Радио и связь, 1988. – 299 с.

23. **Шеннон К.** Работы по теории информации и кибернетике. – М.: Иностранная литература, 1963. – 830 с.

24. **Комашинский В. И., Смирнов Д.А.** Нейронные сети и их применение в системах управления и связи. – М.: Горячая линия-Телеком, 2003. – 94 с.

### Электронные адреса фирм

A.1. [www.analog.com](http://www.analog.com)

A.2. [www.autex.ru](http://www.autex.ru)

A.3. [www.mot.com](http://www.mot.com)

A.4. [www.design-net.com](http://www.design-net.com)

A.5. [www.texas.ru](http://www.texas.ru)

A.6. [www.ti.com/sc/c6000](http://www.ti.com/sc/c6000), [www.ti.com/omaptraining](http://www.ti.com/omaptraining)

A.7. [www.elvees.ru](http://www.elvees.ru)

A.8. [www.xilinx.ru](http://www.xilinx.ru)

A.9. [www.altera.com/literature](http://www.altera.com/literature)

A.10. [www.module.ru/](http://www.module.ru/)

## Оглавление

Предисловие.....	3
Глава 1. Основы цифровой обработки сигналов.....	5
1.1. Основные операции цифровой обработки сигналов.....	5
1.2. Форматы представления данных при обработке.....	12
Контрольные вопросы.....	14
Глава 2. Классификация, архитектура и характеристики сигнальных процессоров	
2.1. Классификация процессоров обработки сигналов.....	15
2.2. Архитектура цифровых сигнальных процессоров.....	16
2.3. Характеристики цифровых сигнальных процессоров.....	19
Контрольные вопросы.....	21
Глава 3. Цифровой сигнальный процессор ADSP 2181 фирмы Analog Devices.....	22
3.1. Структурная схема и характеристики процессора.....	22
3.2. Арифметически-логическое устройство.....	24
3.3. Умножитель-аккумулятор.....	26
3.4. Кольцевой сдвигатель.....	28
3.5. Формирователи адресов.....	30
3.6. Устройство управления программой.....	32
3.7. Последовательные порты.....	34
3.8. Порт прямого доступа к памяти.....	38
3.9. Таймер.....	39
3.10. Система команд процессора ADSP2181.....	42
Контрольные вопросы.....	45
Глава 4. Реализация на процессоре ADSP2181 типовых алгоритмов обработки сигналов.....	46
4.1. Нерекурсивный цифровой фильтр.....	46
4.2. Формирователи гармонического колебания табличным методом и методом аппроксимации.....	48
4.3. Реализация алгоритма быстрого преобразования Фурье.....	50



Контрольные вопросы .....	55
Глава 5. Процессоры ADSP Blackfin и ADSP SHARC .....	56
5.1. Семейство сигнальных процессоров ADSP Blackfin .....	56
5.2. Процессоры с плавающей точкой ADSP SHARC .....	59
Контрольные вопросы .....	63
Глава 6. Цифровые сигнальные процессоры и микроконтроллеры фирмы Motorola .....	64
6.1. Классификация и особенности процессоров Motorola .....	64
6.2. Процессор DSP 56300, характеристики и применение .....	65
6.3. Ядро процессора .....	68
6.4. Система команд процессора DSP 56*** .....	72
6.5. Микроконтроллеры, структура и применение в телекоммуникациях .....	73
Контрольные вопросы .....	77
Глава 7. Цифровые процессоры фирмы Texas Instruments .....	78
7.1. Классификация и особенности процессоров .....	78
7.2. Процессоры платформы C2000 .....	80
7.3. Процессоры платформы C5000 схемы (C54x, C55x) .....	80
7.4. Арифметически-логическое устройство и умножитель процессора C54x .....	85
7.5. Сдвигатель и устройство сравнения и хранения процессора C54x .....	88
7.6. Адресация процессора C54x .....	90
7.7. Мультимедиа-видеопроцессор C80 .....	92
7.8. Процессоры платформы C6000 .....	95
7.9. Процессоры с плавающей точкой фирмы Texas Instruments .....	98
Контрольные вопросы .....	103
Глава 8. Процессор NM6403 и микроконтроллер «Мультикор» фирмы Элвис .....	104
8.1. Структурная схема процессора NM6403 с векторным сопроцессором и его характеристики .....	104
8.2. Векторный сопроцессор и реализация нейронной сети .....	107
8.3. Сигнальный микроконтроллер серии «Мультикор» (1829VM2T-МС24) .....	113
Контрольные вопросы .....	117

Глава 9. Программируемые логические интегральные схемы и их применение.....	119
9.1. Классификация ПЛИС.....	119
9.2. Свойства ПЛИС и порядок проектирования устройств .....	121
9.3. Основные характеристики ПЛИС фирм Altera и XILINX.....	124
9.4. Реализация на ПЛИС алгоритма БПФ и нерекурсивных фильтров .....	128
9.5. Система на кристалле как развитие ПЛИС и программируемых процессоров .....	134
Контрольные вопросы .....	136
Приложение 1. Список сокращений английских терминов.....	138
Приложение 2. Программа реализации нерекурсивного фильтра на процессоре ADSP 2181 .....	140
Приложение 3. Программа формирования гармонического колебания методом аппроксимации на процессоре ADSP 2181 .....	142
Приложение 4. Программа реализации алгоритма БПФ на процессоре ADSP 2181 .....	143
Приложение 5. Платформа EZ-Kit Lite .....	147
Приложение 6. Симулятор процессора SIM 2181.....	154
Предметный указатель .....	162
Список литературы.....	164

**В. С. Сперанский**

# **Сигнальные микропроцессоры и их применение в системах телекоммуникаций и электроники**

Рассмотрены необходимые теоретические вопросы и практика использования цифровых микропроцессоров обработки сигналов. Кратко рассмотрены основные операции цифровой обработки сигналов, архитектура, классификация и характеристики современных сигнальных процессоров. Существенное внимание уделено процессорам фирмы Analog Devices, приведены алгоритмы реализации типовых устройств. Описаны процессоры фирм Motorola, Texas Instruments, векторный процессор ИТЦ Модуль и микроконтроллер Мультикор (ИПЦ Элвис). Отдельный раздел посвящен программируемому логическим интегральным схемам и системе на кристалле. В приложениях содержатся программы реализации нерекурсивного фильтра, формирователя гармонического колебания, быстрого преобразования Фурье, а также описания отладочной платы процессора ADSP2181 и его симулятора.

Для студентов, обучающихся по специальностям «Радиосвязь, радиовещание и телевидение», «Средства связи с подвижными объектами», «Многоканальные телекоммуникационные системы», будет полезна специалистам.

**Книги издательства «Горячая линия — Телеком»**  
можно заказать через почтовое агентство DESSY: 107113, г.Москва, а/я 10,  
а также интернет-магазин: [www.dessy.ru](http://www.dessy.ru)

Сайт издательства:

**[www.techbook.ru](http://www.techbook.ru)**

ISBN 978-5-9912-0035-6



9 785991 200356