

# 1 ВВЕДЕНИЕ

Процессоры ADSP-BF531, ADSP-BF532 и ADSP-BF533 являются представителями семейства процессоров Blackfin с расширенными возможностями, которые обладают значительно большей производительностью и меньшей потребляемой мощностью по сравнению с предыдущими процессорами семейства Blackfin при сохранении простоты использования и совместимости кода. Три новых процессора полностью совместимы по выводам и различаются только производительностью и размером внутренней памяти, что позволяет избежать многих трудностей, возникающих при разработке новых продуктов.

Архитектура ядра процессора Blackfin является архитектурой с единым набором команд, включающей ядро обработки сигналов со сдвоенным блоком умножения-накопления, имеющей ортогональный набор команд, характерный для RISC-микропроцессоров, обладающей гибкостью команд типа SIMD и мультимедийными возможностями.

Особенностью продуктов семейства Blackfin является динамическое управление питанием. Возможность изменения как напряжения питания, так и рабочей частоты позволяет оптимизировать потребление мощности в соответствии с конкретной задачей.

## Периферийные устройства

Периферийные устройства системы процессора включают:

- Параллельный периферийный интерфейс (PPI)
- Последовательные порты (SPORT)
- Последовательный периферийный интерфейс (SPI)
- Таймеры общего назначения
- Универсальный асинхронный приёмник-передатчик (UART)
- Часы реального времени (RTC)
- Сторожевой таймер
- Порт ввода/вывода общего назначения (программируемые флаги)

Эти периферийные устройства соединены с ядром несколькими шинами с высокой пропускной способностью, как показано на рис. 1-1.

# Введение

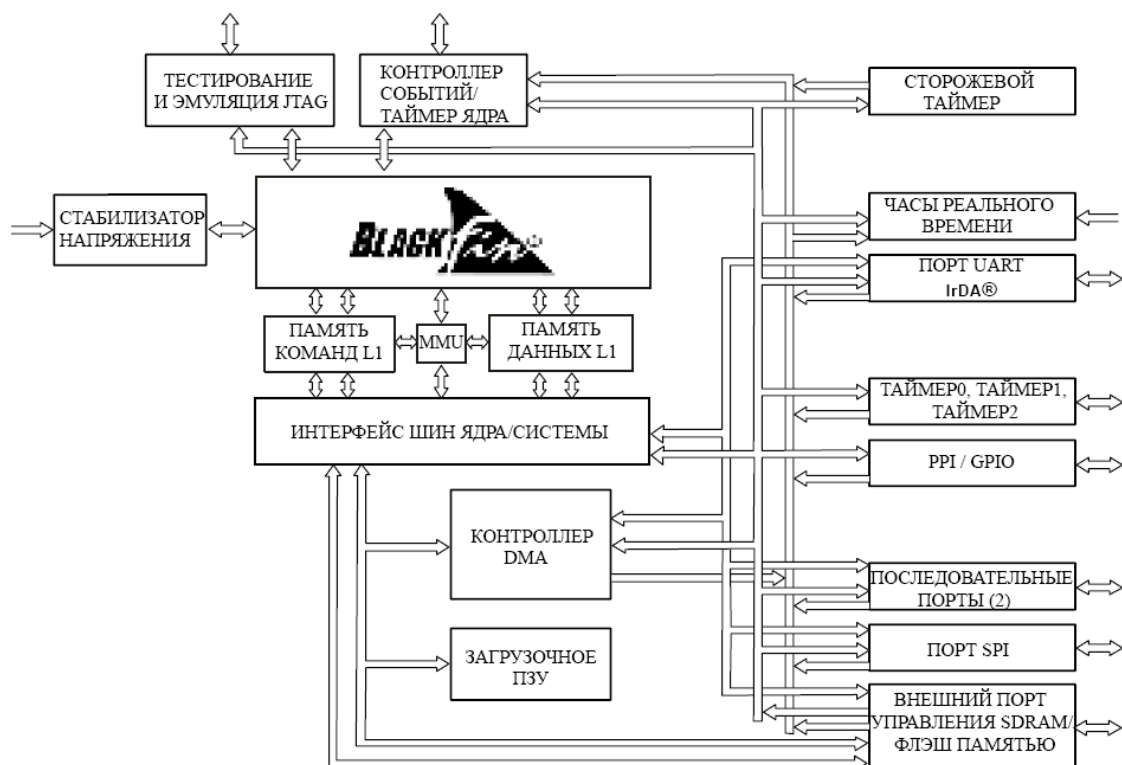


Рис. 1-1. Блок-схема процессора

Все периферийные устройства, за исключением порта ввода/вывода общего назначения, часов реального времени и таймеров, поддерживаются гибкой структурой прямого доступа к памяти (DMA). Также в процессоре имеются два отдельных канала DMA типа “память-память”, предназначенные для осуществления передач между пространствами памяти процессора, включая внешнюю SDRAM и асинхронную память. Набор внутренних шин обеспечивает пропускную способность, достаточную для поддержания работы ядра процессора, даже если задействованы все внутренние и внешние периферийные устройства.

## Архитектура ядра

Ядро процессора содержит два 16-разрядных умножителя, два 40-разрядных аккумулятора, два 40-разрядных арифметико-логических устройства (АЛУ), четыре 8-разрядных видео АЛУ и 40-разрядное устройство сдвига, которые показаны на рис. 1-2. Вычислительные устройства обрабатывают 8-, 16-и или 32-разрядные данные, поступающие из регистрационного файла.

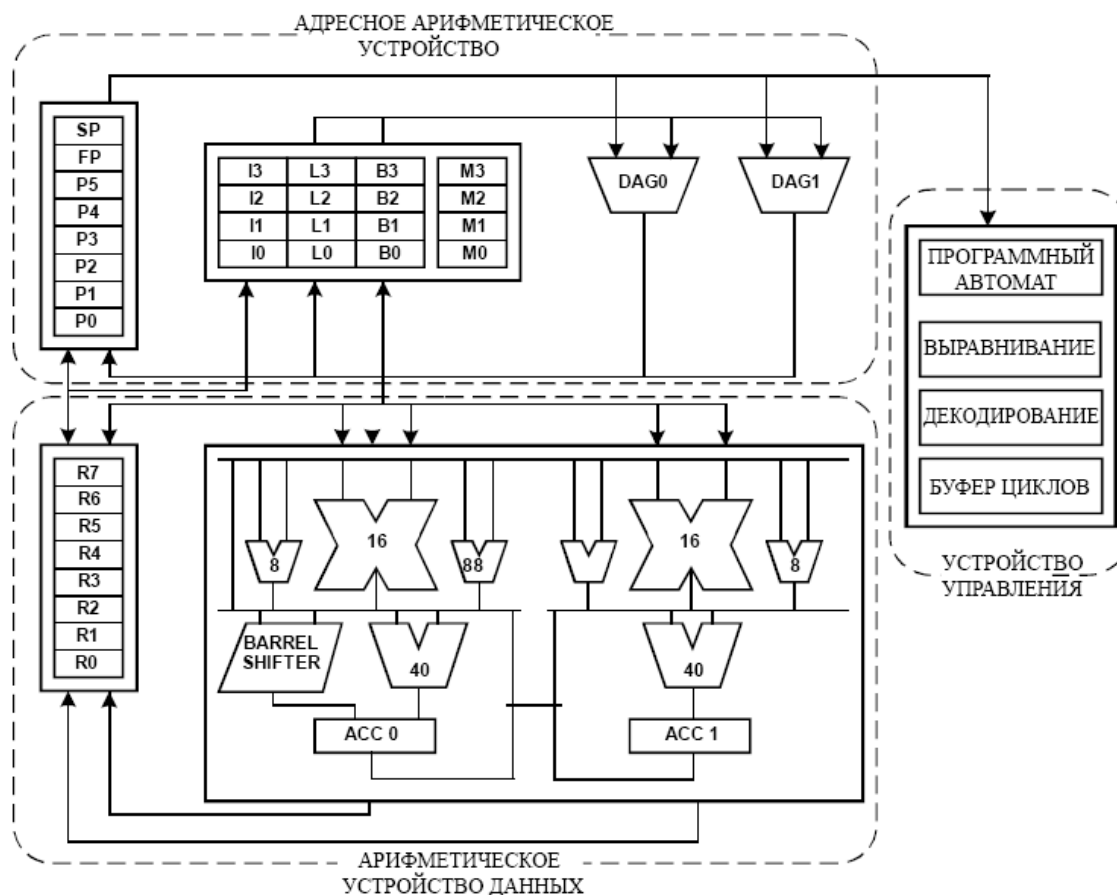


Рис. 1-2. Архитектура ядра процессора

Вычислительный регистровый файл содержит восемь 32-разрядных регистров. При выполнении вычислительных операций над 16-разрядными операндами регистровый файл функционирует как 16 независимых 16-разрядных регистров. Все операнды вычислительных операций поступают из многопортового регистрового файла или задаются константами в полях команды.

Каждый умножитель-накопитель (МАС) за один такт выполняет умножение двух 16-разрядных чисел и накопление, формируя 40-разрядный результат. Поддерживаются знаковый и беззнаковый форматы чисел, округление и насыщение.

АЛУ выполняет традиционный набор арифметических и логических операций над 16- или 32-разрядными данными. В него включены многие специальные команды, ускоряющие выполнение различных задач обработки сигналов. К ним относятся битовые операции, такие как извлечение поля, подсчет числа единиц, умножение по модулю  $2^{32}$ , примитивы деления, насыщение и округление, и определение знака/порядка. Набор видео-команд включает операции выравнивания и упаковки байтов, сложение 16-разрядных чисел с 8-разрядными с усечением результата, операции 8-разрядного усреднения и операции 8-разрядного вычитания/ взятия абсолютного значения/ накопления (SAA, subtract/ absolute value, accumulate). Также поддерживаются команды сравнения/выбора и векторного поиска. При использовании некоторых команд возможно одновременное выполнение двух 16-разрядных операций АЛУ над парами регистров (младшей и старшей 16-

# Введение

разрядными половинами вычислительного регистра). При использовании второго АЛУ возможно одновременное выполнение четырёх 16-разрядных операций.

40-разрядное устройство сдвига может осуществлять внесение данных и выполнять операции сдвига, циклического сдвига, нормализации и извлечения битового поля.

Программный автомат управляет процессом выполнения команд, включая выполнение операций выравнивания и декодирования команд. При управлении выполнением программы программный автомат поддерживает относительные (относительно счётчика команд) и косвенные условные переходы (со статическим предсказанием переходов) и вызовы подпрограмм. В процессоре реализована аппаратная поддержка циклов с нулевыми непроизводительными затратами. Архитектура программного автомата полностью замкнута, что гарантирует отсутствие видимых эффектов работы конвейера при выполнении команд с взаимосвязанными данными.

Адресное арифметическое устройство формирует два адреса для выполнения одновременных двойных выборок из памяти. Оно содержит многопортовый регистровый файл, состоящий из четырёх наборов 32-разрядных регистров индекса, модификации, длины и базового адреса (для организации циклических буферов) и восьми дополнительных 32-разрядных регистров указателей (для индексированной манипуляции стеком в стиле языка C).

Процессоры Blackfin поддерживают модифицированную Гарвардскую архитектуру с иерархической структурой памяти. Память уровня 1 (L1) обычно работает с полной скоростью процессора с небольшой задержкой или без задержки. Память команд на уровне L1 содержит только команды. Две памяти данных содержат данные, а выделенная сверхоперативная (блокнотная) память хранит информацию стека и локальные переменные.

Процессор имеет несколько блоков памяти L1, которые могут быть сконфигурированы как смешанный набор SRAM и кэша. Устройство управления памятью (MMU, Memory Management Unit) обеспечивает защиту памяти при выполнении ядром индивидуальных задач и может осуществлять защиту системных регистров от непреднамеренного доступа.

Архитектура процессора обеспечивает три режима работы: Пользовательский режим, режим Супервизора и режим Эмуляции. В Пользовательском режиме доступ к поднабору ресурсов системы ограничен, что обеспечивает организацию защищённой программной среды. В режимах Супервизора и Эмуляции доступ к ресурсам ядра и системы неограничен.

Набор команд процессора ADSP-BF53x Blackfin оптимизирован таким образом, что наиболее часто используемые команды представлены 16-разрядными кодами. Комплексные команды цифровой обработки сигналов (ЦОС) кодируются 32-разрядными кодами как многофункциональные команды. В продуктах семейства Blackfin реализована ограниченная поддержка многозадачности, которая заключается в возможности параллельного вызова 32-разрядной команды и двух

16-разрядных команд. Это позволяет программисту использовать многие ресурсы ядра в одном командном цикле.

В языке ассемблера процессора ADSP-BF53x Blackfin используется алгебраический синтаксис. Архитектура оптимизирована для использования совместно с компилятором языка C.

## Архитектура памяти

В архитектуре процессора Blackfin память структурировано в единое унифицированное 4-гигабайтное адресное пространство, использующее 32-разрядные адреса. Все ресурсы, включая внутреннюю память, внешнюю память и регистры управления ввода/вывода, занимают отдельные секции в общем адресном пространстве. Области памяти адресного пространства упорядочены в иерархическую структуру, которая обеспечивает баланс стоимости и производительности при использовании очень быстрой, обладающей малой задержкой внутренней памяти, такой как кэш или SRAM, и обладающими более низкой стоимостью и производительностью внешними системами памяти больших объёмов. В таблице 1-1 приведено сравнение памяти процессоров ADSP-BF531, ADSP-BF532 и ADSP-BF533.

Таблица 1-1. Сравнение памяти.

Тип памяти	ADSP-BF531	ADSP-BF532	ADSP-BF533
SRAM/кэш команд	16 Кбайт	16 Кбайт	16 Кбайт
SRAM команд	16 Кбайт	32 Кбайт	64 Кбайт
ПЗУ команд	32 Кбайт	32 Кбайт	-
SRAM/кэш данных	16 Кбайт	32 Кбайт	32 Кбайт
SRAM данных	-	-	32 Кбайт
Сверхоперативная память	4 Кбайт	4 Кбайт	4 Кбайт
<b>Общий объём памяти</b>	<b>84 Кбайт</b>	<b>116 Кбайт</b>	<b>148 Кбайт</b>

Система памяти L1 представляет собой первичную, обладающую наивысшей производительностью, память, доступную ядру. Система внешней памяти, доступ к которой осуществляется при помощи устройства интерфейса внешней шины (EBIU, External Interface Bus Unit), обеспечивает расширение памяти системы флэш-памятью, SDRAM и SRAM, позволяя адресовать до 132 Мбайт физической памяти.

Контроллер DMA типа “память-память” обеспечивает возможность организации пересылок данных с высокой скоростью. Он может выполнять передачи блоков кода или данных между пространствами внутренней и внешней памяти.

## Внутренняя память

Процессор содержит три блока внутренней памяти, обеспечивающих доступ к ядру с высокой пропускной способностью:

- Память команд L1, состоящая из SRAM и 4-входового наборно-ассоциативного кэша. В процессорах, содержащих ПЗУ, она также

## Введение

включает область ПЗУ, определяемую пользователем. Доступ к этой памяти осуществляется с полной скоростью процессора.

- Память данных L1, состоящая из SRAM и/или 2-входового наборно-ассоциативного кэша. Доступ к этому блоку памяти осуществляется с полной скоростью процессора.
- Сверхоперативное ЗУ L1, которое работает с той же скоростью, что память данных и память команд L1, но доступно только как SRAM данных и не может быть сконфигурировано как кэш-память.

## Внешняя память

Доступ к внешней (расположенной вне кристалла) памяти осуществляется при помощи устройства интерфейса внешней шины. 16-разрядный интерфейс обеспечивает не требующее использования дополнительных микросхем подключение банка синхронного DRAM (SDRAM) и до четырёх банков устройств асинхронной памяти, включая флэш-память, ПЗУ, EPROM, SRAM и устройства ввода/вывода, отображаемые в карте памяти.

PC-133 совместимый контроллер SDRAM может быть запрограммирован на интерфейс с SDRAM объёмом до 128 Мбайт.

Контроллер асинхронной памяти может быть запрограммирован на управление до четырёх банков устройств. Каждый банк занимает сегмент размером 1 Мбайт независимо от размера используемых устройств. Таким образом, банки являются неразрывными, только если каждый из них полностью заполнен памятью объёмом 1 Мбайт.

## Пространство памяти ввода/вывода

В процессорах Blackfin не определено отдельное пространство ввода/вывода. Все ресурсы отображены в 32-разрядном адресном пространстве. Регистры управления внутренних устройств ввода/вывода отображены в карте памяти по адресам, расположенным в верхней части 4-гигабайтного адресного пространства. Они разделены на два блока меньшего размера: один из блоков содержит регистры управления всеми функциями ядра, а другой содержит регистры, необходимые для настройки и управления, внутренними периферийными устройствами, располагающимися вне ядра. Регистры, отображённые в карте памяти доступны только в режиме Супервизора. Они представляют собой выделенное пространство внутренних периферийных устройств.

## Обработка событий

Контроллер событий процессора управляет всеми асинхронными и синхронными событиями, воздействующими на процессор. При обработке событий процессором поддерживается вложение и назначение приоритетов. Вложение допускает одновременную активность нескольких программ обслуживания событий.

Назначение приоритетов гарантирует, что прерывание с высоким приоритетом будет обслужено раньше прерывания с более низким приоритетом. Контроллер обеспечивает поддержку пяти различных событий:

- Эмуляция – Вызывает вход процессора в режим Эмуляции, позволяющий задавать команды и управлять процессором по интерфейсу JTAG.
- Сброс – Вызывает сброс процессора.
- Немаскируемое прерывание (NMI, Nonmaskable Interrupt) – Данное событие генерируется программным сторожевым таймером или входным сигналом NMI процессора. Немаскируемое прерывание часто используется в качестве индикатора выключения питания для инициации корректного завершения работы системы.
- Исключения – синхронные с процессом выполнения программы. Таким образом, исключение воспринимается до того, как будет разрешено завершение выполнения команды. Исключения вызываются такими условиями, как нарушения выравнивания данных и неопределённые команды.
- Прерывания – асинхронные с процессом выполнения программы. Они вызываются входными выводами, таймерами и другими периферийными устройствами.

Каждый тип события имеет соответствующий регистр, содержащий адрес возврата, и соответствующую команду возврата из события. При регистрации события состояние процессора сохраняется в стек супервизора.

Контроллер событий процессора состоит из двух уровней: контроллера событий ядра (CEC, Core Event Controller) и контроллера прерываний системы (SIC, System Interrupt Controller). CEC и SIC осуществляют назначение приоритетов и управление всеми событиями системы. В целом, прерывания от периферийных устройств поступают в SIC и преобразуются в прерывания общего назначения CEC.

## **Контроллер событий ядра (CEC)**

В дополнение к выделенным прерываниям и исключениям контроллер событий ядра поддерживает девять прерываний общего назначения (IVG15–7). Два прерывания с низшим приоритетом (IVG15–14) рекомендуется зарезервировать за обработчиками программных прерываний, оставляя для поддержки периферийных устройств семь входов прерываний с упорядоченными приоритетами.

## **Контроллер прерываний системы (SIC)**

Контроллер прерываний системы обеспечивает отображение событий многих периферийных источников прерываний во входы прерываний CEC общего назначения с упорядоченными приоритетами. Хотя в процессоре реализовано отображение по умолчанию, пользователь может изменять соответствия между прерываниями ядра и системы и приоритеты прерываний записью

# Введение

соответствующих значений в регистры назначения прерываний (IAR, Interrupt Assignment Registers).

## Поддержка DMA

Процессор имеет несколько независимых контроллеров DMA, поддерживающих автоматизированные передачи данных с минимальным вмешательством ядра. Передачи DMA могут осуществляться между блоками внутренней памяти и любым из периферийных устройств процессора, поддерживающих DMA. Кроме того, передачи DMA могут осуществляться между любым периферийным устройством, поддерживающим DMA, и внешними устройствами, подсоединёнными к интерфейсам внешней памяти, включая контроллер SDRAM и контроллер асинхронной памяти. К периферийным устройствам, поддерживающим DMA, относятся последовательные порты, порт SPI, UART и PPI. Каждое отдельное периферийное устройство, поддерживающее DMA, имеет, по меньшей мере, один выделенный канал DMA.

Контроллер DMA поддерживает и одномерные (1D) и двумерные (2D) передачи DMA. Инициализация передач DMA может осуществляться либо при помощи регистров, либо при помощи набора параметров, называемых блоками дескрипторов.

Функция двумерного DMA поддерживает произвольные размеры строк и столбцов до 64К элементов на 64К элементов, и произвольные размеры шагов по строкам и столбцам до +/- 32К элементов. Более того, размер шага по столбцу может быть меньше размера шага по строке, что позволяет реализовать перемежение потоков данных. Это свойство особенно полезно в видеоприложениях, в которых может осуществляться разделение данных на лету.

Примеры поддерживаемых типов DMA:

- операции с одиночным линейным буфером, обращения к которому прекращаются по завершению передачи;
- передачи с использованием циклического, автоматически обновляемого буфера, по заполнению или частичному заполнению которого вызывается прерывание;
- одномерный и двумерный DMA с использованием связанного списка дескрипторов;
- двумерный DMA с использованием массива дескрипторов, определяющих только базовый адрес DMA в пределах общей страницы.

В дополнение к выделенным каналам DMA периферийных устройств, также имеется отдельный канал DMA типа “память-память”, предназначенный для передач между различными устройствами памяти системы. Его использование позволяет передавать блоки данных между любыми устройствами памяти – включая внешние SDRAM, SRAM, ПЗУ и флэш-память – с минимальным вмешательством процессора. Управление передачами DMA типа “память-память” может осуществляться при помощи очень гибкой методики, основанной на



использовании дескрипторов, или стандартного механизма автобуферизации, основанного на использовании регистров.

### **Устройство интерфейса внешней шины**

Устройство интерфейса внешней шины (EBIU) процессора реализует интерфейс с разнообразными устройствами памяти индустриального стандарта. Контроллер состоит из контроллера SDRAM и контроллера асинхронной памяти.

#### **Контроллер SDRAM PC133**

Контроллер SDRAM обеспечивает интерфейс с одним банком устройств SDRAM индустриального стандарта или DIMM. Банк, полностью совместимый со стандартом SDRAM PC133, может конфигурироваться на объём от 16 до 128 Мбайт памяти.

Для поддержки банком SDRAM более медленных устройств памяти имеется набор программируемых временных параметров. Банк памяти имеет разрядность 16 разрядов, что позволяет минимизировать количество устройств и снижает стоимость системы.

#### **Асинхронный контроллер**

Контроллер асинхронной памяти обеспечивает конфигурируемый интерфейс до четырёх отдельных банков памяти или устройств ввода/вывода. Временные параметры каждого банка могут независимо программно устанавливаться. Это свойство позволяет осуществлять интерфейс с разнообразными устройствами памяти, включая SRAM, ПЗУ и флэш-EPROM, а также устройствами ввода/вывода, имеющими интерфейс со стандартными линиями управления памятью. Каждый банк занимает окно размером 1 Мбайт в адресном пространстве процессора. Если окна заняты не полностью, контроллер памяти не делает их непрерывными. Банки имеют разрядность 16 разрядов, что позволяет организовать интерфейс с рядом устройств памяти и устройств ввода/вывода.

### **Параллельный периферийный интерфейс**

В процессоре имеется параллельный периферийный интерфейс (PPI, Parallel Peripheral Interface), который позволяет напрямую соединиться с параллельными ЦАП и АЦП, видеокодерами и декодерами стандарта ITU-R 601/656, и периферийными устройствами ввода/вывода общего назначения. PPI включает выделенный вывод входного тактового сигнала, до трёх выводов кадровой синхронизации и до 16 выводов данных. Входной тактовый сигнал допускает параллельную передачу данных со скоростями, достигающими половины тактовой частоты системы.

## Введение

При работе в режимах ITU-R 656 PPI принимает и разбирает поток 8-разрядных или 10-разрядных элементов данных. Поддерживается внутреннее декодирование встроенной управляющей преамбулы и информации синхронизации.

Поддерживается три различных режима ITU-R 656:

- Приём активной видеоинформации – PPI не выполняет чтение данных, поступающих между символами преамбул Конец Активного Видео (EAV, End of Active Video) и Начала Активного Видео (SAV, Start of Active Video), или данных, присутствующих в течение интервалов обратного хода луча. В данном режиме последовательности управляющих байтов не сохраняются в память; они фильтруются в PPI.
- Приём информации в интервалах обратного хода кадровой развёртки – PPI используется для передач данных в течение интервала обратного хода кадровой развёртки (VBI, Vertical Blanking Interval), а также информации в течение интервала обратного хода строчной развёртки и последовательности управляющих байтов по строкам VBI.
- Приём полного поля – PPI производит чтение всего поступающего битового потока. Он включает активное видео, последовательности управляющих преамбул и служебные данные, которые могут передаваться во интервалах обратного хода кадровой и строчной развёртки.

Возможности формата выходного сигнала ITU-R 656, явно не поддерживаемые, могут достигаться настройкой структуры всего кадра (включая активное видео, информацию, передаваемую в течение обратного хода развёртки, и информацию управления) в памяти и последовательной передачей данных из PPI в режиме без кадровой синхронизации. Свойства двумерного DMA процессора упрощают осуществление передач подобного типа, позволяя один раз разместить в памяти буфер статического кадра (коды управления и гашения обратного хода) и затем просто обновлять активную видеоинформацию для каждого кадра.

Режимы PPI общего назначения предназначены для удовлетворения потребностей различных приложений передачи и захвата данных. Эти режимы разделены на четыре основные категории, каждая из которых допускает передачу до 16 бит данных за такт PPI\_CLK:

- Приём данных с внутренней генерацией кадровой синхронизации
- Приём данных с внешней генерацией кадровой синхронизации
- Передача данных с внутренней генерацией кадровой синхронизации
- Передача данных с внешней генерацией кадровой синхронизации

Эти режимы поддерживают соединения с АЦП/ЦАП, а также аппаратную передачу видеоизображения. Многие из режимов поддерживают более одного уровня кадровой синхронизации. По желанию, между установлением сигнала кадровой синхронизации и приёмом/передачей данных может быть добавлена программируемая задержка.

## Последовательные порты (SPORT)

Процессор имеет два двухканальных синхронных последовательных порта (SPORT0 и SPORT1), обеспечивающих последовательные и межпроцессорные соединения. Последовательные порты имеют следующие особенности:

- Двухнаправленная работа с поддержкой I<sup>2</sup>S.  
Каждый последовательный порт имеет два набора независимых выводов для передачи и приема, позволяющих организовать восемь каналов стереозвуча в формате I<sup>2</sup>S.
- Порты передачи и приема с буферизацией (глубиной 8 слов).  
Каждый порт имеет регистр данных для передачи слов данных в другие процессорные компоненты и приема из них, а также регистры сдвига, соединенные с регистрами данных;
- Тактирование  
Каждый порт приема и передачи может использовать внешний последовательный тактовый сигнал или генерировать собственный тактовый сигнал в широком диапазоне частот.
- Длина слова  
Каждый порт поддерживает последовательную передачу слов данных длиной от 3 до 32 разрядов. Данные могут передаваться, начиная со старшего разряда или начиная с младшего разряда.
- Кадровая синхронизация.  
Каждый порт передачи и приема может функционировать с сигналами кадровой синхронизации, сопровождающими каждое слово данных, или без них. Сигналы кадровой синхронизации могут генерироваться внутри порта или внешним источником, иметь активный низкий или активный высокий уровень. Возможна кадровая синхронизация с любой из двух длительностей импульса, а также выбор кадровой синхронизации с опережением или запаздыванием.
- Аппаратное компандирование.  
Каждый последовательный порт может выполнять А- или  $\mu$ -компандирование в соответствии с рекомендацией ITU G.711. Компандирование может выбираться для порта передачи и/или приёма, оно не вносит дополнительных задержек.
- Операция с DMA с непроизводительными затратами в один такт  
Каждый последовательный порт может автоматически принимать и передавать несколько буферов памяти данных. Процессор может осуществлять связанные передачи или цепочки последовательностей передач DMA между последовательным портом и памятью.
- Прерывания  
Каждый порт передачи и приема генерирует прерывание после завершения передачи слова данных или после передачи целого буфера с данными или нескольких буферов при помощи DMA.
- Многоканальность  
Каждый последовательный порт поддерживает работу 128 каналов из 1024-канального окна и совместим со стандартами H.100, H.110, MVIP-90 и NMVIP.

## Введение

# Порт последовательного периферийного интерфейса (SPI-порт)

Процессор имеет SPI-совместимый порт, который позволяет процессору соединяться с различными SPI-совместимыми устройствами.

SPI интерфейс имеет три вывода для передачи данных: два вывода данных и один вывод тактового сигнала. Входной сигнал выбора микросхемы SPI позволяет другим SPI-совместимым устройствам выбирать данный процессор, а семь выходных сигналов выбора микросхемы SPI позволяют процессору выбирать другие SPI-совместимые устройства. Выводы выбора SPI являются реконфигурируемыми выводами программируемых флагов. Используя эти выводы, SPI-порт обеспечивает полнодуплексный, синхронный последовательный интерфейс, который поддерживает режимы ведущего и ведомого, а также режим работы в системе с несколькими ведущими.

Скорость передачи данных и фаза/полярность сигналов тактовой синхронизации порта SPI могут программно изменяться. Порт имеет интегрированный контроллер DMA, который конфигурируется таким образом, чтобы поддерживать либо входные, либо выходные потоки данных. В отдельно взятый момент времени контроллер DMA SPI-порта может обслуживать только однонаправленные обращения.

Во время выполнения передач порт SPI одновременно передаёт и принимает данные, осуществляя их последовательный сдвиг по двум линиям данных. Выборка и сдвиг данных на этих линиях управляются по линии последовательного тактового сигнала.

## Таймеры

Процессор имеет четыре программируемых таймера общего назначения. Три таймера имеют внешний вывод, который может быть сконфигурирован для использования либо в качестве широтно-импульсного модулятора (PWM, Pulse Width Modulator) или выходного сигнала таймера, либо в качестве входа тактового сигнала таймера, либо в качестве механизма измерения ширины импульса внешнего сигнала. Эти таймеры могут синхронизироваться внешним тактовым сигналом, подаваемым на входной вывод PF1 или PPI\_CLK, или внутренним сигналом SCLK.

Таймеры могут использоваться совместно с UART для измерения ширины импульсов в потоке данных, обеспечивая функцию автоматического определения скорости передачи в последовательном канале.

Таймеры могут генерировать прерывания ядра процессора для обеспечения периодической синхронизации по тактовому сигналу процессора или по последовательности внешних сигналов.

В дополнение к трём таймерам общего назначения в процессоре также имеется дополнительный четвёртый таймер. Он тактируется внутренним тактовым сигналом процессора и обычно используется в качестве системных часов для генерации периодических прерываний операционной системы.

## Порт UART

Процессор имеет полудуплексный порт универсального асинхронного приёмника-передатчика (UART, Universal Asynchronous Receiver/Transmitter), полностью совместимый со стандартными портами UART персонального компьютера. Порт UART обеспечивает упрощённый UART-интерфейс с другими периферийными устройствами или хост-процессорами, реализующий полудуплексные асинхронные передачи данных в последовательном режиме с поддержкой DMA. Порт UART поддерживает использование от 5 до 8 битов данных, 1 или 2 стоповых битов, а также возможность передачи данных без проверки и с проверкой на чётность или нечётность. Порт UART поддерживает два режима работы:

- Запрограммированный ввод/вывод  
Процессор посылает или принимает данные путём записи в регистры UART, отображённые в карте памяти. При приёме и передаче данные подвергаются двойной буферизации.
- Прямой доступ к памяти (DMA)  
Передачи принимаемых и передаваемых данных осуществляются контроллером DMA. При этом уменьшается количество и частота прерываний, необходимых для передач данных в память и из памяти. UART имеет два выделенных канала DMA – для приёма и для передачи. Вследствие относительно низких скоростей обслуживания эти каналы DMA имеют приоритет, меньший, чем большинство других каналов DMA.

Скорость передачи в бодах, формат последовательно передаваемых данных, состояние и генерация кода ошибки, и прерывания UART могут настраиваться программно, что позволяет поддерживать:

- широкий диапазон скоростей передачи в битах;
- форматы данных от 7 до 12 битов на кадр;
- генерацию маскируемых прерываний процессора при операциях приёма и передачи.

При использовании порта UART совместно с таймерами общего назначения поддерживается автоматическое определение скорости передачи в канале.

Возможности порта UART, дополнены поддержкой протокола спецификации физического уровня инфракрасного канала последовательной передачи данных (SIR, Serial Infrared Physical Layer Link Specification), разработанного ассоциацией передачи данных по инфракрасному каналу (IrDA<sup>®</sup>, Infrared Data Association).

# Введение

## Часы реального времени

Часы реального времени (RTC, Real-Time Clock) процессора предоставляют набор свойств цифровых часов, включающий функции будильника, секундомера и индикации текущего времени. RTC тактируется внешним кварцевым резонатором с частотой 32,768 кГц. Так как периферия RTC имеет выделенные выводы питания, она может находиться во включённом состоянии и тактироваться, даже когда остальная часть процессора находится в состоянии пониженного потребления мощности. RTC обеспечивает несколько возможных программируемых прерываний, включающих прерывания по секундам, минутам, часам или дням, прерывание программируемого секундомера, или прерывание будильника по наступлению программно заданного времени.

Сигнал частотой 1 Гц получается делением входного тактового сигнала частотой 32,768 кГц в делителе. Функция счёта таймера осуществляется четырьмя счётчиками: 60-секундным счётчиком, 60-минутным счётчиком, 24-часовым счётчиком и 32768-дневным счётчиком.

При разрешении функции будильника по совпадению значения таймера со значением, запрограммированным в регистре управления будильником, генерируется прерывание. Существует два режима будильника. В первом режиме будильник настраивается на время дня. Во втором режиме будильник настраивается на день и время в течение заданного дня.

Функция секундомера выполняет обратный отсчёт от запрограммированного значения с минутным разрешением. При разрешении функции будильника по достижению счётчиком нуля генерируется прерывание.

Как и остальные периферийные устройства, RTC может выводить процессор из режима Сна или режима Глубокого Сна при генерации любого события RTC. По событиям RTC может производиться также вывод внутреннего стабилизатора напряжения из выключенного состояния.

## Сторожевой таймер

Процессор содержит 32-разрядный таймер, который может использоваться для реализации функции программного сторожевого таймера. Использование программного сторожевого таймера может повысить устойчивость системы при помощи принудительного перевода процессора в известное состояние путём генерации аппаратного сброса, немаскируемого прерывания (NMI) или прерывания общего назначения, если таймер истекает до того, как он будет сброшен программно. Программист инициализирует счётчик таймера, разрешает соответствующее прерывание и затем разрешает работу таймера. После этого, счётчик должен быть перезагружен программно до достижения им нуля. Это защищает систему от попадания в неизвестное состояние в случае, когда выполнение программы, которая при нормальной работе сбросила бы таймер, останавливается из-за воздействия внешнего шума или программной ошибки.

Если сторожевой таймер настроен на генерирование аппаратного сброса, он выполняет сброс и ЦПУ и периферийных устройств. После выполнения сброса программа может определить, являлся ли источником аппаратного сброса сторожевой таймер, путём проверки бита состояния в регистре управления сторожевым таймером.

Таймер работает от тактового сигнала системы (SCLK) с максимальной частотой  $f_{SCLK}$ .

## Программируемые флаги

Процессор имеет 16 двунаправленных выводов программируемых флагов (PF) или ввода/вывода общего назначения, PF[15:0]. Каждый вывод может конфигурироваться индивидуально с использованием регистров прерываний, состояния и управления флагами.

- Регистр управления направлением флага – определяет направление каждого отдельного вывода PFx на вход или выход.
- Регистры состояния и управления флагами – в процессоре реализован механизм “записи-1-для-модификации” (“write-1-to-modify”), позволяющий производить модификацию любой комбинации отдельных флагов одной командой без воздействия на уровень других флагов. Имеются четыре регистра управления. В первый регистр производится запись для установки значений флагов, во второй регистр производится запись для сброса значений флагов, в третий регистр производится запись для изменения значений флагов на противоположные, и в четвёртый регистр производится запись для задания любого числа значений флагов. Программа осуществляет опрос значений флагов путём чтения регистра состояния флагов.
- Регистры маскирования прерываний флагов – Два регистра маскирования прерывания флагов позволяют реализовать функцию прерывания процессора для каждого отдельного вывода PFx. Аналогично двум регистрам управления флагами, используемыми для установки и сброса значений отдельных флагов, один регистр маскирования прерывания флагов устанавливает биты, разрешая функцию генерирования прерывания, а другой – сбрасывает биты, запрещая функцию генерирования прерывания. Входные выводы PFx могут быть сконфигурированы таким образом, чтобы генерировать аппаратные прерывания. Выходные выводы PFx могут вызывать программные прерывания.
- Регистры чувствительности прерываний флагов – Два регистра чувствительности прерываний флагов определяют чувствительность отдельных выводов PFx к фронту или уровню и, при чувствительности к фронту, определяют, происходит ли срабатывание по переднему фронту или по обоим фронтам. Один регистр определяет тип чувствительности, другой определяет, к какому фронту чувствителен вывод.

## Введение

### Сигналы тактовой синхронизации

Тактирование процессора может осуществляться сигналом внешнего кварцевого резонатора, входным синусоидальным сигналом или буферизированным тактовым сигналом, полученным из сигнала внешнего тактового генератора.

Внешний тактовый сигнал подаётся на вывод CLKIN процессора. В режиме нормальной работы процессора сигнал на входе CLKIN не может фиксироваться, изменяться или иметь частоту ниже определённого значения. Тактовый сигнал должен иметь TTL-совместимые уровни.

Тактовый сигнал ядра (CCLK) и тактовый сигнал периферии системы (SCLK) формируются из входного тактового сигнала (CLKIN). Внутренняя схема фазовой автоподстройки частоты (PLL, Phase Locked Loop) может производить умножение частоты сигнала CLKIN на определяемый пользователем (от 1x до 63x) множитель (ограниченный максимальной и минимальной частотами генератора, управляемого напряжением). По умолчанию используется множитель 10x, который может изменяться при выполнении определённой последовательности команд программы. Изменение частоты на лету может осуществляться записью в регистр PLL\_DIV.

Все внутренние периферийные устройства тактируются тактовым сигналом системы (SCLK). Тактовая частота системы программно устанавливается битами SSEL[3:0] регистра PLL\_DIV.

### Динамическое управление питанием

Процессор имеет четыре рабочих режима, каждому из которых соответствует определённый профиль производительности/потребляемой мощности. Кроме того, динамическое управление питанием обеспечивает динамическое изменение напряжения питания ядра процессора, что позволяет осуществить дальнейшее уменьшение рассеиваемой мощности. Управление тактированием каждого периферийного устройства также уменьшает потребление мощности.

#### **Режим работы с полной мощностью (Максимальная производительность)**

В режиме работы с полной мощностью разрешена работа и использование PLL, что обеспечивает максимальную рабочую частоту. Это состояние является нормальным состоянием выполнения программы, в котором может достигаться максимальная производительность.



## **Активный рабочий режим (Умеренное снижение потребляемой мощности)**

В активном режиме работа PLL разрешена, но она не используется. Вследствие этого, частота тактового сигнала ядра (CCLK) и тактового сигнала системы (SCLK) равны частоте входного тактового сигнала (CLKIN). При работе в данном режиме возможны изменения отношения частоты выходного сигнала VCO к частоте сигнала CLKIN, однако, до входа в режим работы с полной мощностью они не будут иметь эффекта. В данном режиме возможно осуществление доступов DMA к сконфигурированным соответствующим образом блокам памяти L1.

В активном режиме возможно запрещение работы PLL при помощи регистра управления PLL (PLL\_CTL). Если работа PLL запрещена, при переходе в режим работы с полной мощностью и режим сна её необходимо разрешить.

## **Режим сна (Значительное снижение потребляемой мощности)**

В режиме сна потребление мощности уменьшается за счёт запрещения тактирования ядра процессора (CCLK). Однако в данном режиме продолжается работа PLL и подаётся тактовый сигнал системы (SCLK). Обычно, процессор выводится из режима сна внешним событием или в результате работы часов реального времени. Когда процессор находится в режиме сна, по возникновению любого прерывания он считывает значение бита обхода PLL (BYPASS) в регистре управления PLL (PLL\_CTL). Если PLL используется, то процессор переходит в режим работы с полной мощностью. Если PLL не используется, процессор переходит в активный режим.

Организация доступов системы к памяти L1 при помощи DMA в режиме сна не поддерживается.

## **Режим глубокого сна (Максимальное снижение потребляемой мощности)**

В режиме глубокого сна потребление мощности максимально снижается за счёт запрещения тактовых сигналов ядра процессора и синхронной части системы (CCLK и SCLK). Асинхронные системы, такие как часы реального времени, могут продолжать работу, но не могут осуществлять доступ к внутренним ресурсам или внешней памяти. Вывод процессора из этого режима может быть вызван только прерыванием сброса или асинхронным прерыванием, генерируемым часами реального времени. Если процессор находится в режиме глубокого сна, по асинхронному прерыванию часов реального времени он переходит в активный режим, по установлению активного уровня сигнала  $\overline{\text{RESET}}$  – в режим работы с полной мощностью.

# Введение

## Стабилизация напряжения

Процессор имеет внутренний стабилизатор напряжения, способный формировать уровни внутреннего напряжения (от 0,8 В до 1,2 В) из внешнего напряжения питания (от 2,25 В до 3,6 В). На рис. 1-3 показаны типичные внешние компоненты, необходимые для завершения системы управления питанием.

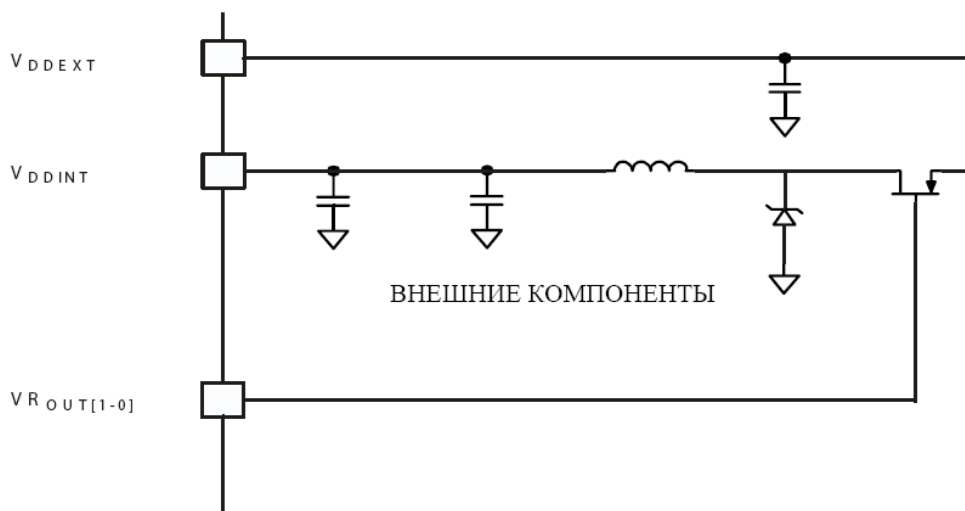


Рис. 1-3 Схема стабилизатора напряжения

Стабилизатор управляет уровнями напряжения внутренней логики, которые могут программно изменяться в регистре управления стабилизатором напряжения ( $VR\_CTL$ ) с инкрементом 50 мВ. Внутренний стабилизатор напряжения может быть запрограммирован таким образом, чтобы отключать питание ядра процессора, сохраняя подачу питания устройств ввода/вывода, в целях снижения потребляемой в режиме ожидания мощности. При нахождении в таком состоянии можно сохранять подачу  $V_{DDEXT}$ , что устраняет необходимость использования внешних буферов. Также, по желанию пользователя, можно отключать стабилизатор и обходить его (запрещать его использование).

## Режимы загрузки

Процессор имеет два механизма автоматической загрузки внутренней памяти команд L1 после сброса. Третий режим предназначен для выполнения команд из внешней памяти, не используя последовательность загрузки:

- Выполнение команд из 16-разрядной внешней памяти – выполнение программы начинается с адреса 0x2000 0000, команды упаковываются по 16 бит. В данном режиме загрузочное ПЗУ не используется. Устанавливаются конфигурационные настройки для самого медленного устройства (время удержания – 3 такта, времена доступа чтения/записи – 15 тактов; время предустановки – 4 такта).
- Загрузка из 8- или 16-разрядной внешней флэш-памяти – выполняется программа, расположенная в пространстве памяти загрузочного ПЗУ, которая реализует загрузку из флэш-памяти с использованием банка асинхронной

памяти 0. Устанавливаются конфигурационные настройки для самого медленного устройства (время удержания – 3 такта, времена доступа чтения/записи – 15 тактов; время предустановки – 4 такта).

- Загрузка из последовательной EEPROM по SPI (с 8-, 16- или 24-разрядной адресацией) – порт SPI использует выходной вывод PF2 для выбора SPI-совместимого устройства памяти EEPROM, последовательно выдаёт команды чтения по адресам 0x00, 0x0000 и 0x000000 до обнаружения устройства памяти EEPROM с 8-, 16- или 24-разрядной адресацией, и начинает синхронную с тактовым сигналом передачу данных в начало памяти команд L1.

В каждом из режимов загрузки сначала выполняется чтение 10-байтного заголовка из внешней памяти. Заголовок определяет число передаваемых байтов и адрес в памяти, в который будут помещены команды. В любой из процедур загрузки может выполняться загрузка нескольких блоков. После загрузки всех блоков начинается выполнение программы из начала SRAM команд L1. При использовании процессора, содержащего область ПЗУ, определяемую пользователем, загрузочное ПЗУ не используется, и выполнение программы начинается из начала ПЗУ команд L1 (0xFFA0 0000).

Кроме того, программа приложения может устанавливать бит 4 регистра конфигурации при сбросе для обхода (отказа от выполнения) нормальной последовательности загрузки при программном сбросе. В данном случае процессор выполняет переход к началу памяти команд L1.

Также имеется вторичный программный загрузчик, который расширяет возможный набор режимов, добавляя дополнительные механизмы загрузки. Вторичный загрузчик обеспечивает возможность загрузки из 16-разрядной флэш-памяти, быстрой флэш-памяти, памяти с переменной скоростью передачи и других источников.

## Описание набора команд

В наборе команд языка ассемблер семейства процессоров ADSP-BF53x применяется алгебраический синтаксис, разработанный для облегчения написания и чтения кода. Команды специально оптимизированы таким образом, чтобы обеспечить гибкий и сжатый набор команд, компилирующийся в код, занимающий очень малый объем памяти. В наборе команд также имеются полностью поддерживаемые многофункциональные команды, позволяющие программисту одновременно использовать многие ресурсы ядра процессора в одной команде. Использование этих свойств набора команд совместно со свойствами, чаще присущими микроконтроллерам, делает его очень эффективным при компиляции исходного кода, написанного на языках C и C++. Кроме того, архитектура процессора поддерживает как Пользовательский режим работы (код приложений/алгоритмов), так и режим Супервизора (ядро операционной системы, драйверы устройств, отладчики, программы обслуживания прерываний), что позволяет организовать несколько уровней доступа к ресурсам ядра.

Язык ассемблер даёт возможность воспользоваться следующими преимуществами уникальной архитектуры процессора:

## Введение

- Интеграция свойств ЦСП и ЦПУ, оптимизированных для выполнения операций над 8- и 16-разрядными данными
- Модифицированная Гарвардская архитектура с использованием многозадачности и принципа “загрузки регистров/сохранения в память” (load/store), поддерживающая выполнение за один такт двух операций умножения/накопления с 16 разрядными операндами или четырёх операций АЛУ с 8-разрядными операндами + две операции загрузки регистра/сохранения в память + два обновления указателей
- Все регистры, устройства ввода/вывода и память отображены в унифицированном 4-гигабайтном пространстве памяти, что упрощает модель программирования
- Свойства микроконтроллера, такие как манипуляция, внесение и извлечение произвольных битов и битовых полей; операции над 8-, 16- и 32-разрядными целочисленными данными; отдельные указатели пользовательского стека и стека супервизора.

Дополнительные возможности уплотнения кода включают смешивание 16- и 32-разрядных команд без переключения режима или изоляции отдельных частей кода. Часто используемые команды кодируются 16 битами.

## Средства разработки

Процессор поддерживается полным набором средств разработки программного и аппаратного обеспечения CrossCore™, включающим эмуляторы производства Analog Devices и среду разработки VisualDSP++™. Те же аппаратные эмуляторы, которые поддерживают другие продукты Analog Devices, полностью эмулируют процессоры семейства ADSP-BF53x.

Среда разработки проектов VisualDSP++ позволяет программистам разрабатывать и выполнять отладку приложений. Эта среда включает лёгкий в использовании ассемблер, основанный на алгебраическом синтаксисе, архиватор(средство создания библиотек), компоновщик, загрузчик, потактовый симулятор уровня команд, компилятор C/C++ и библиотеку исполняемых функций C/C++, включающую математические функции и функции ЦОС. Ключевой особенностью средств разработки программного обеспечения является эффективность кода, написанного на языках C/C++. Компилятор разработан таким образом, чтобы эффективно транслировать C/C++ код в ассемблер процессора Blackfin. Процессор Blackfin имеет архитектурные особенности, улучшающие эффективность компилированного C/C++ кода.

При отладке как C/C++ программ, так и программ на языке ассемблера в отладчике среды VisualDSP++ программист может:

- просматривать смешанный код на C/C++ и языке ассемблера (с перемежением исходной и объектной информации);
- устанавливать точки останова;
- устанавливать условные точки останова по содержимому регистров, памяти и стеков;
- производить трассировку выполнения команд;

- выполнять линейное или статистическое профилирование выполнения программы;
- заполнять, выгружать и графически отображать содержимое памяти;
- выполнять отладку на уровне исходной программы;
- создавать собственные окна отладчика.

Интегрированная среда разработки (IDE, Integrated Development Environment) VisualDSP++ позволяет программисту определять процесс разработки программного обеспечения и управлять им. Диалоговые окна и страницы свойств среды позволяют программисту конфигурировать и управлять всеми возможностями средств разработки, включая выделение цветом синтаксических конструкций в редакторе VisualDSP++. Эти возможности позволяют программисту:

- управлять процессом обработки входной информации и формирования выходной информации средствами разработки;
- поддерживать однозначное соответствие свойств, задаваемых в диалоговых окнах и страницах свойств, с ключами командной строки.

Ядро операционной системы VisualDSP++ (VDK, VisualDSP++ Kernel) включает возможности планирования и управления ресурсами, специфичные для адресации памяти и временных ограничений при программировании ЦСП. Эти свойства позволяют инженерам более эффективно разрабатывать код, устраняя необходимость начинать разработку нового приложения с нуля. Свойства VDK включают поддержку потоков, критических и незапланированных областей, семафоров, событий и флагов устройств. VDK также поддерживает основанное на приоритетах, упреждающее, кооперативное планирование и планирование процессов с квантованием времени. Кроме того, VDK разрабатывалось как масштабируемая операционная система. Если приложение не использует определённое свойство, то код, реализующий это свойство, исключается из разрабатываемой системы.

Так как VDK является библиотекой, решение о его применении оставляется на усмотрение разработчика. VDK интегрировано в среду разработки VisualDSP++, однако оно также может использоваться совместно со стандартными средствами командной строки. Среда разработки VDK помогает в управлении ресурсами системы, автоматизируя генерацию различных объектов, основанных на VDK, и визуализируя состояние системы в течение процесса отладки приложения.

Эмуляторы производства Analog Devices используют порт доступа тестирования JTAG IEEE 1149.1 процессора для наблюдения и управления процессором на отлаживаемой плате при эмуляции. Эмулятор обеспечивает полноскоростную эмуляцию, позволяющую отслеживание состояния и модификацию памяти, регистров и стеков процессора.

Внутрисхемная эмуляция без вмешательства в работу процессора обеспечивается использованием интерфейса JTAG – эмулятор не влияет на нагрузку или временные характеристики тестируемой системы.

В дополнение к средствам разработки программного и аппаратного обеспечения, предлагаемым фирмой Analog Devices, имеется широкий спектр средств,

## Введение

поддерживающих семейство процессоров Blackfin, предоставляемых третьими сторонами. Аппаратные средства включают автономные оценочные платы ADSP-BF533 EZ-KIT Lite™. Программное обеспечение, предлагаемое третьими сторонами, включает библиотеки ЦОС, операционные системы реального времени и средства разработки блок-схем.