

процессоров на базе архитектуры Cortex-M3 даже в самых простых приложениях.

В основе процессора на базе архитектуры Cortex-M3 лежит ядро, выполненное по Гарвардской технологии с 3-ступенчатым конвейером, что обеспечивает такие привлекательные возможности, как предсказание переходов, однотактное умножение и аппаратно реализованное деление, что обеспечивает производительность порядка 1.25 MIPS/МГц (в тесте Dhrystone).

Несмотря на то, что ядро Cortex-M3 разрабатывалось как недорогое ядро, оно остается 32-битным микропроцессорным ядром и, в связи с этим, поддерживает два режима работы: потоковый режим (Thread) и режим обработчика (Handler). Благодаря этому появляется возможность разработки более интеллектуального программного обеспечения и поддержки операционных систем реального времени (OSPV).

В ядро Cortex также входит 24-битный автоматически перезагружаемый таймер, предназначенный для генерации

периодических прерываний и используемый ядром OCPB. Если у микропроцессоров ARM7 и ARM9 имеется два набора инструкций (32-битный ARM и 16-битный Thumb), то у семейства Cortex предусмотрена поддержка набора инструкций ARM Thumb-2. Этот набор представляет собой смесь 16- и 32-битных инструкций, которые позволяют добиться производительности 32-битного набора инструкций ARM и плотности кода, свойственной 16-битному набору инструкций Thumb.

Thumb-2 - обширный набор инструкций, ориентированный на компиляторы языков C/C++. Это означает, что программа для Cortex-микроконтроллера может быть полностью написана на языке программирования C/C++.

Семейство STM32 состоит из двух групп. Группа Performance Line работает на тактовых частотах до 72 МГц и оснащена полным набором устройств ввода-вывода (УВВ, англ. I/O), а группа Access Line работает на частотах до 36 МГц и интегрирует ограниченный набор устройств ввода-вывода.

Иногда фирма ARM ссылается на свои процессоры по наименованию версии архитектуры. В этих обозначениях версия архитектуры процессора Cortex-M3 определяется как ARMV7M. Иными словами, процессор Cortex-M3 выполнен по архитектуре ARMV7M и поддерживает исполнение инструкций Thumb-2.

Микропроцессорное ядро Cortex

Основой процессора Cortex является 32-битное микропроцессорное ядро, имеющее RISC-систему команд. Данное ядро использует упрощенную модель программирования ARM7/9, но при этом более обширный набор инструкций с хорошей поддержкой целочисленной арифметики, улучшенными битовыми операциями и более строгими реально-временными характеристиками.

Конвейер

Микропроцессорное ядро Cortex способно выполнять большинство инструкций за один цикл, что достигается с помощью

трехступенчатого конвейера: выборка-дешифрация-выполнение. Cortex поддерживает предсказание переходов для минимизации количества перезагрузок конвейера. Во время выполнения одной инструкции следующая инструкция дешифрируется, а третья инструкция считывается из памяти. Этот механизм отлично работает с линейным кодом, но если требуется выполнить переход, то, прежде чем продолжить выполнение кода программы, потребуется очистка и перезагрузка конвейера.

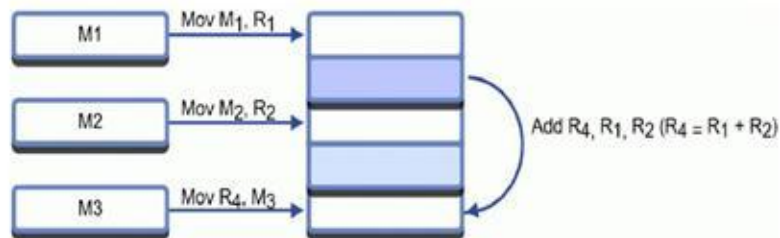
Трехступенчатый конвейер микропроцессора Cortex оснащен логикой предсказания переходов. Это означает, что при достижении инструкции условного перехода выполняется упреждающая выборка. В результате оба назначения инструкции условного перехода будут доступны для исполнения и не произойдет снижения производительности. Хуже обстоят дела с инструкциями косвенного перехода, т. к. в этом случае упреждающую выборку выполнить нельзя и перезагрузка конвейера может оказаться неизбежной.



Трехступенчатый конвейер

Модель программирования

ЦПУ Cortex является RISC-процессором, который выполнен по архитектуре чтения/ записи. Для выполнения операций обработки данных вначале необходимо поместить операнды из памяти в центральный регистровый файл, затем выполнить требуемую операцию над данными в регистрах и, наконец, перезаписать результат обратно в память.



Cortex-M3 выполнен по архитектуре чтения/записи. Все данные, перед выполнением инструкции их обработки, необходимо поместить в центральный регистровый файл. Следовательно, вся активность программы фокусируется вокруг регистрового файла ЦПУ. Данный регистровый файл образуют шестнадцать 32-битных регистров.

Регистровый файл

Микропроцессорное ядро Cortex-M3 содержит регистровый файл, состоящий из шестнадцати 32-битных регистров (R0-R15). У регистров R13-R15 имеются особые функции.



Рис. 9. Регистр статуса программы: ICI - возобновляемая прерыванием инструкция;

IT - поле 'if then'; ISR - процедура обработки прерывания

R13 выступает в роли указателя стека (SP). Данный регистр является банковым, что делает возможной работу Cortex в двух режимах работы, в каждом из которых используется свое собственное пространство стека. Данная возможность обычно используется операционными системами реального времени, которые могут выполнять свой системный код в защищенном режиме. У двух стеков Cortex имеются собственные наименования: основной стек и стек процесса.

R14 - регистр связи (LR). Он используется для хранения адреса возврата из подпрограммы. Благодаря этому регистру Cortex быстро переходит к подпрограмме и выходит из нее. Если же в программе

используется несколько уровней вложений подпрограмм, то компилятор будет автоматически сохранять R14 в стек.

R15 - счетчик программы (PC). Он является частью центрального регистрового файла, его чтение и обработка могут выполняться аналогично любым другим регистрам.

Регистр статуса программы

Помимо регистрового файла, имеется отдельный регистр XPSR, который называется регистром статуса программы. Он не входит в основной регистровый файл, а доступ к нему возможен с помощью двух специальных инструкций. В XPSR хранятся значения полей, влияющих на исполнение инструкций Cortex. Регистр статуса программы содержит поля статуса, от которых зависит исполнение инструкций. Данный регистр разделен еще на три поля: статуса прикладной программы, исполнения программы и прерываний. Регистр статуса программы содержит поля, от которых зависит исполнение инструкций. Данный регистр разделен на три поля:

статуса прикладной программы, исполнения программы и прерываний (рис. 9). Биты регистра XPSR разделены на три группы, к каждой из которых возможен доступ по собственному наименованию.

Верхние пять бит (флаги кода условия) именуются полем статуса прикладной программы. Первые четыре флага кода условия N, Z, C, V (индикация отрицательного (N) или нулевого (Z) результата, переноса (C) и переполнения (V)) устанавливаются и сбрасываются по итогам выполнения инструкции обработки данных. Пятый бит Q используется при выполнении математических инструкций с насыщением алгоритмов цифровой обработки сигналов для индикации достижения переменной своего максимального или минимального значения. Так же как и 32-битные инструкции ARM, некоторые инструкции Thumb-2 выполняются только при условии совпадения кода условия инструкции и состояния флагов регистра статуса прикладной программы.

Если коды условия инструкции не совпадают, то инструкция проходит по конвейеру как NOP (нет операции). Этим гарантируется

равномерность прохождения инструкций по конвейеру и минимизируется число перезагрузок конвейера. У Cortex данный способ расширен полем статуса исполнения программы, который связан с битами регистра XPSR. Это поле состоит из трех полей: поле 'if then' (IT), поле возобновляемой прерыванием инструкции и поле инструкции Thumb. Набор инструкций Thumb-2 реализует эффективный метод выполнения компактных блоков инструкций типа 'if then'. Если проверяемое условие истинно, записью значения в поле IT можно сигнализировать микропроцессору о необходимости выполнения до четырех следующих инструкций. Если же проверяемое условие - ложное, то данные инструкции пройдут по конвейеру как NOP.

Несмотря на то, что большинство инструкций Thumb-2 выполняются за один цикл, некоторые инструкции (например, инструкции чтения/записи) требуют для выполнения несколько циклов. Чтобы точно знать время отклика микропроцессорного ядра Cortex на прерывания, данные инструкции должны быть прерываемыми. В

случае преждевременного прекращения исполнения инструкции в поле возобновляемых прерываниями инструкций запоминается номер следующего регистра, подлежащего обработке инструкцией многократного чтения или записи. Таким образом, сразу после завершения процедуры обработки прерывания выполнение инструкции многократного чтения/записи может быть восстановлено. Последнее поле Thumb предусмотрено для совместимости с предшествующими версиями микропроцессорного ядра ARM. Данное поле сигнализирует, что в настоящий момент микропроцессорное ядро выполняет инструкцию ARM или Thumb. У микропроцессорного ядра Cortex-M3 данный бит всегда равен единице. Наконец, в поле статуса прерывания хранится информация о любых приостановленных запросах прерывания.

Режимы работы микропроцессора

Процессор Cortex разрабатывался как быстродействующее, простое в использовании и с малым числом логических элементов

микроконтроллерное ядро, в нем была учтена поддержка для работы операционных систем реального времени (ОСРВ). Процессор Cortex поддерживает два режима работы: режим Thread (или потоковый режим) и режим Handler (или режим обработчика).

Микропроцессор запускается в режиме Thread при непрерываемом, фоновом выполнении инструкций и переключается в режим Handler при обработке исключительных ситуаций. Кроме того, микропроцессор Cortex может выполнять код программы в привилегированном или непривилегированном режиме. В привилегированном режиме ЦПУ имеет доступ ко всему набору инструкций, а в непривилегированном некоторые инструкции отключаются (например, инструкции MRS и MSR, осуществляющие доступ к регистру XPSR и его битовым группам).

Системный интерфейс

Процессор Cortex-M3 выполнен по гарвардской архитектуре, которая подразумевает использование отдельных шин данных (D-bus)

и инструкций (I-bus). Обе эти шины могут осуществлять доступ к инструкциям и данным в диапазоне адресов 0x00000000-0x1 FFFFFFFF. Также имеется дополнительная системная шина (System), которая предоставляет доступ к области системного управления по адресам 0x20000000-0xDFFFFFFF и 0xE0100000-0xFFFFFFFF.

У встроенной отладочной системы процессора Cortex имеется еще одна дополнительная шинная структура, которая называется локальной шиной UBB (Private Peripheral Bus, PPB).

Системная шина и шина данных подключаются к внешнему микроконтроллеру через набор высокоскоростных шин, называемых матрицей шин. Она образует несколько параллельных соединений между шинами Cortex и другими внешними шинными мастерами, как, например, каналы ПДП к встроенным ресурсам, статическое ОЗУ и UBB. Если два шинных мастера (например, микропроцессорное ядро Cortex и канал ПДП) предпринимают попытку доступа к одному и тому же UBB, то вступит в действие внутренний арбитр, который

разрешит конфликт, предоставив доступ к шине тому, кто имеет наивысший приоритет. Однако, благодаря тесной взаимосвязи блоков ПДП с микропроцессорным ядром Cortex, необходимость арбитража во многих случаях исключается.

Организация памяти Cortex-M3

Процессор Cortex-M3 является стандартизованным микроконтроллерным ядром, и поэтому его карта памяти четко расписана (рис. 10). Несмотря на использование нескольких внутренних шин, адресное пространство является линейным и имеет размер 4 Гб.

будет более медленным, чем из встроенной флэш-памяти, расположенной в области кода программы.

Следующие полгигабайта памяти - область встроенных устройств ввода-вывода (УВВ). В этой области находятся все предоставляемые пользователю производителем микроконтроллера УВВ. Первый мегабайт в каждой области памяти (статическое ОЗУ и УВВ) является битноадресуемым. Для этого используется метод bit banding. Таким образом, все данные, хранящиеся в этих областях, могут обрабатываться как пословно, так и побитно.

Следующие два гигабайта адресного пространства выделены для внешних статического ОЗУ и устройства ввода-вывода (УВВ).

Последние полгигабайта зарезервированы для системных ресурсов и будущих расширений процессора Cortex. Все регистры процессора Cortex расположены по фиксированным адресам во всех Cortex-микроконтроллерах. Благодаря этому облегчается портирование программ. Для процессора Cortex-M3 определена фиксированная карта памяти размером 4 Гб, в которой выделены конкретные области для

хранения кода программы, статического ОЗУ, устройств ввода-вывода, внешней памяти и устройств, а также системных регистров. Карта памяти одинакова для всех Cortex-микроконтроллеров.

Мультиядерные ARM

Процессоры с ядрами ARM на данный момент являются достаточно популярными для мобильных устройств и встраиваемых систем различного применения. Самые массовые из них — это мобильные телефоны, смартфоны, коммуникаторы, мультимедийные устройства. Архитектура обладает такими привлекательными свойствами, как удобная и эффективная система команд, мощная поддержка при разработке аппаратной базы и программного обеспечения, высокая энергоэффективность. Относительно недавно было заявлено о разработке мультиядерной архитектуры на базе ARM, что открывает данным процессорам путь на рынок высокопроизводительных приложений. В частности, одно из возможных применений —

ноутбуки, сравнимые, а то и превосходящие их аналоги на архитектуре x86.

На данный момент компанией ARM представлены нескольких мультиядерных архитектур: ARM11 MPCore, Cortex-R5 MPCore, Cortex-R7 MPCore, Cortex-A7 MPCore, Cortex-A9 MPCore и Cortex-A15 MPCore, Cortex-A17 MPCore. Каждая из этих архитектур может масштабироваться четырех несколькими ядрами. Код, написанный для одиночных процессоров, также может исполняться и на мультиядерных.

Целевая область применения процессоров MPCore лежит в области мобильных приложений с высокими требованиями по производительности совместно с ограниченными энергетическими ресурсами. Благодаря масштабируемой пиковой производительности данный процессор может достаточно легко справляться с требованиями современных высокопроизводительных встраиваемых приложений при сохранении инвестиций в программное обеспечение в условиях развивающегося рынка.

Общая характеристика мультиядерных ARM-процессоров

Процессоры MPCore поддерживают полностью когерентный кэш данных, существенно упрощая как симметричный, так и асимметричный мультипроцессинг, собственно, как и любую другую мультипроцессорную технологию.

Производительность приложений увеличивается благодаря возможности разделения ядрами данных кэша, возможности распределения и балансирования вычислительной нагрузки между процессорами, портирования многозадачных приложений, а также масштабируемости приложений за счет эффективной загрузки процессора многопоточными приложениями, характерными для современного программного обеспечения. Возможность передачи данных между кэшами процессоров позволяет процессорам эффективно разделять данные без необходимости доступа в память. Оптимизированный кэш первого уровня существенно ускоряет операции с данными при сохранении достаточно низкого энергопотребления. Аппаратная реализация индексации и тэгирования

данных кэша снимают временные издержки при устранении наложения адресов или необходимости очистки кэша в ходе смены контекста в операционной системе.

Кэш данных используется как при операциях чтения, так и при записи данных совместно с адаптируемым буфером записи, который позволяет существенно снизить количество обращений к основной памяти и может формировать запросы на массивованную передачу данных из нескольких запросов к памяти. Уникальная система кэш-памяти ускоряет выделение пространства кэша, в результате чего оно выполняется всего за один цикл.

Процессоры MPCore позволяют производителю использовать одни и те же ядра с различными конфигурациями для продуктов с различными свойствами и требованиями.

К настоящему времени лицензия на выпуск процессоров ARM MPCore приобретена более чем 15 компаниями, включая Broadcom, NEC Electronics, NVIDIA, Renesas Technology, Toshiba and Sarnoff Corporation, и воплощена в большом количестве приложений и

устройств, представленных на современном рынке. Технология существенно расширяет спектр приложений, предлагая более эффективные модели операций.

Все мультитядерные решения от ARM базируются на шинной архитектуре AMBA 3 AXI, дающей возможность подключать к процессорам не только память и периферийные устройства, но и другие процессоры.

Шинный интерфейс процессоров MPCore и масштабируемость позволяют настраивать производительность системы, оптимизировать ее энергопотребление и снижать общую стоимость решения и риск морального старения при переходе к следующему поколению цифровых устройств.

Интеграция с существующими системными компонентами также снижает риски, связанные, например, с поддержкой операционных систем и продуктов на базе данных процессоров. Работает стандартная для ARM- архитектур модель программирования с поддержкой существующих операционных систем и приложений. Доступны

совместимые с Linux 2.6 SMP операционные системы и инструменты разработки.

Занимаемая процессорами площадь на кристалле, диапазон рабочих частот и потребляемая мощность зависят от использованного при реализации технологического процесса, библиотек компонентов и оптимизации.

Несмотря на различия в ядрах и некоторые различия в построении мультиядерных вариантов процессоров, есть ряд технологий, поддерживаемый ими всеми.

Технологии ускорения выполнения Java-приложений — **Jazelle DBX** и **Jazelle RCT** для оптимизации процесса адаптивной компиляции "на лету" (Just In Time (JIT) and Dynamic Adaptive Compilation (DAC)), а также для уменьшения расхода памяти — максимум в три раза.

Технология **TrustZone** предназначена для обеспечения безопасности транзакций, управления цифровыми сертификатами, создания базы для проверки и защиты прав (Digital Rights Management (DRM)).

ARM11 MPCore

Синтезируемый процессор ARM11 MPCore поддерживает микроархитектуру ARM11 и может содержать от одного до четырех процессоров (рис. 11), достигая производительности до 2600 DMIPS; имеет расширенную полосу пропускания памяти порядка 1,3 Гб/с для одиночного процессора.

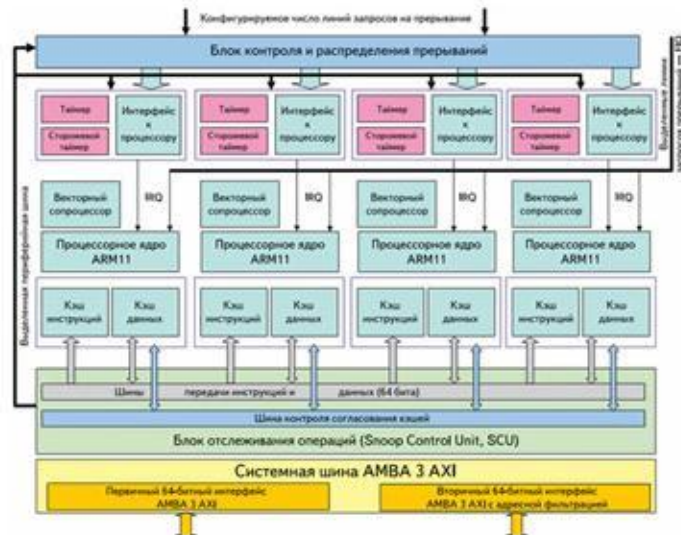


Рис. 11. Структура процессора ARM11 MPCore

Процессоры ARM11 MPCore поддерживают архитектуру ARMv6 с поддержкой Thumb, расширений цифровой обработки сигналов, SIMD мультимедийной обработки данных и ARM Jazelle Java.

У процессора высокопроизводительная подсистема памяти. Каждый процессор имеет свои независимые кэши данных и инструкций с поддержкой согласования данных. Размер кэшей инструкций и данных можно независимо изменять в пределах от 16 Кб до 64 Кб индивидуально для каждого ядра.

Поддерживается 64-битный интерфейс AMBA AXI с одиночной или двойной 64-битной шинной системой AMBA 3 AXI. Системная 64-битная шина AMBA 3 AXI упрощает обмен данными в системе при достаточно большой полосе пропускания и простой системе тактирования.

Векторный сопроцессор (Vector Floating Point coprocessors) работает с числами в формате с плавающей точкой.

Блок контроля и распределения прерываний является программируемым — возможно сконфигурировать до 255 независимых источников аппаратных прерываний

Добавлена система управления энергопотреблением: мультипроцессор имеет возможность отключать неиспользуемые ресурсы и процессоры (Adaptive Shutdown), что в итоге дает динамическое энергопотребление порядка 0,49 мВт/МГц — экономится до 85% энергии.

ARM11 MPCore позволяет разработчикам систем на кристалле рассматривать отдельный процессор как одиночный процессор, упрощая тем самым процесс разработки и уменьшая время выхода продукта на рынок.

Серия Cortex-A MPCore

Процессоры семейства Cortex-A MPCore — Cortex-A5 MPCore и Cortex-A9 MPCore — помимо небольшой занимаемой площади и

энергоэффективности обладают богатым арсеналом возможностей и функциональностью архитектуры ARMv7, что в итоге дает высокую производительность и низкое энергопотребление, как на специфических прикладных приложениях, так и для устройств общего плана. Поддерживается также технология Thumb-2, обеспечивающая высокую производительность при одновременном уменьшении размера кода на 30%.

Процессоры, входящие в состав мультипроцессоров Cortex-A-MPCore, имеют блок операций с плавающей точкой, способный выполнять операции с одинарной и двойной точностью. Он обладает примерно вдвое большей производительностью, чем предыдущие версии ARM FPU (Floating Point Unit).

Архитектура ARM Cortex-A5/A9 имеет мультимедийное 128-битное SIMD (одиночный поток команд, множественный поток данных) расширение архитектуры — NEON, предназначенное для поддержки мультимедийных операций и функций цифровой обработки сигналов (например, для ускорения работы таких алгоритмов, как H.264 или

MP3). Также данный модуль расширяет систему команд набором инструкций ARM **NEON** Advanced SIMD, впервые представленным с процессором Cortex-A8.

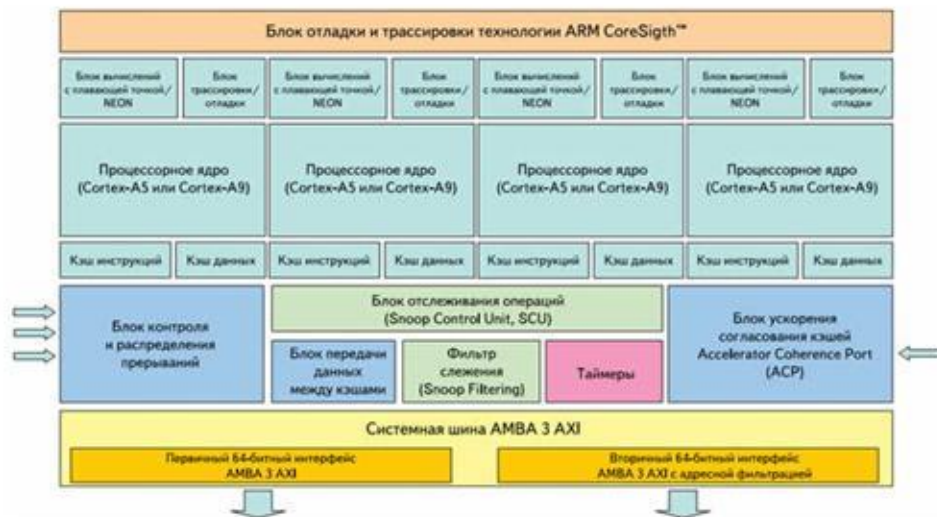


Рис. 12. Структура процессоров Cortex-A MPCore

ARM Cortex-A9 MPCore

Процессор Cortex-A9 MPCore имеет возможность оптимизации производительности приложений и по скорости выполнения, и по потребляемой мощности.

Основные возможности включают в себя следующее:

Энергоэффективный суперскалярный конвейер
производительностью более 2,0 DMIPS/МГц.

Оптимизированный по производительности и потребляемой мощности кэш первого уровня совмещает минимальное время задержки и минимальное энергопотребление. Добавлен контролер кэша второго уровня, позволяющий осуществлять доступ с малыми временами задержки и высокой пропускной способностью к кэш памяти размером до 2 Мб.

Мультипроцессор Cortex-A9 MPCore иллюстрирует практически линейную масштабируемость производительности на различных тестах.

Процессоры ARM Cortex-A9 — и одиночный вариант (рис. 13), и мультипроцессор — поддерживают ряд специфических расширений ARM-архитектуры, в том числе: DSP, SIMD, Jazelle, TrustZone, Intelligent Energy Manager (IEM TM). В дополнение к этому ARM разработала ряд поддерживающих технологий для сокращения времени разработки и сокращения времени выхода продукта на рынок.

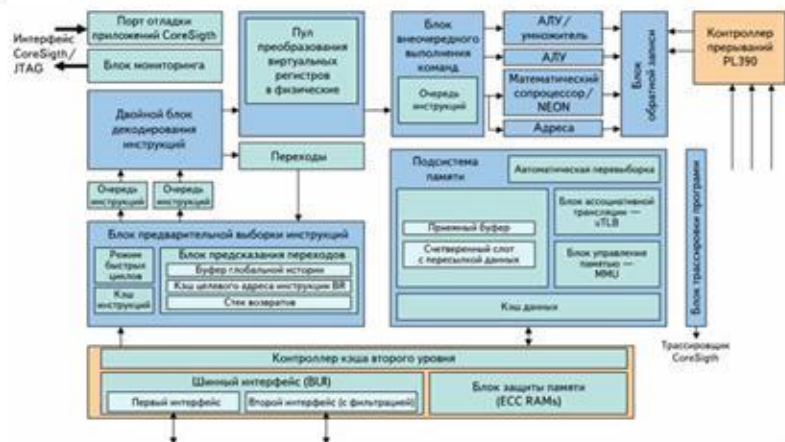


Рис. 13. Структура процессора ARM Cortex-A9