

Архитектура ЭВМ ARM-7.

Список архитектур ARM

В настоящее время значимыми являются несколько семейств процессоров ARM:

1. **ARM7** (с тактовой частотой до 60-72 МГц), предназначенные, например, для недорогих мобильных телефонов и встраиваемых решений средней производительности. В настоящее время активно вытесняется новым семейством **Cortex**.
2. **ARM9**, **ARM11** (с частотами до 1 ГГц) для продвинутых телефонов, карманных компьютеров и встраиваемых решений высокой производительности.
3. **Cortex A** — новое семейство процессоров на смену ARM9 и ARM11.
4. **Cortex M** — новое семейство процессоров на смену ARM7, также призванное занять новую для ARM нишу встраиваемых решений низкой производительности. В семействе присутствуют три значимых ядра: Cortex M0, Cortex M3 и Cortex M4.

В **2010 году** производитель анонсировал процессоры **Cortex-A15** под кодовым названием Eagle.

Ядро **Cortex A15** на 40 процентов производительнее на той же частоте, чем ядро **Cortex-A9** при одинаковом числе ядер на чипе.

Изделие, изготовленное по 28-нанометровому техпроцессу, имеет 4 ядра, может функционировать на частоте до 2,5 ГГц и будет поддерживаться многими современными операционными системами.

Популярное семейство микропроцессоров **xScale** фирмы **Marvell** (до 27 июня 2007 года — **Intel** (до 27 июня 2007 года — **Intel**), в действительности является расширением архитектуры ARM9, дополненной набором инструкций **Wireless MMX**, специально разработанных фирмой Intel для поддержки мультимедийных приложений.

Микроконтроллер **ARM7100** можно назвать микроконтроллером широкого применения, т.к. он ориентирован на использование в таких устройствах как:

1. персональные информационные устройства (PDA),
2. органайзеры,
3. интеллектуальные мобильные телефоны,
4. многофункциональные пейджеры,
5. карманные измерительные устройства,
6. системы сбора данных.

Микроконтроллер организован по модульному принципу с использованием внутренней шины AMBA, организующей взаимодействие ядра со стандартными библиотечными ячейками периферии.

Два других микроконтроллера **ARM7500** и **ARM7500FE** являются однокристальными микрокомпьютерами, ориентированными на:

1. реализацию мультимедиа устройств,
2. портативных и настольных компьютеров,
3. карманных вычислительных и измерительных устройств,
4. интерактивных приставок цифрового TV,
5. игровых консолей.

Их отличие – в приборе ARM7500FE есть ускоритель операций с плавающей точкой (FPA)



более **высокая производительность**.

Они также реализованы **по модульному принципу и объединяют ядро ARM7 с:**

1. самодостаточными макроячейками видео,
2. звука,
3. FPA (ARM7500FE),
4. стандартных библиотечных ячеек периферии.

Общим для всех трех микропроцессоров является:

1. использование ядра ARM7,
2. встроенного единого кэш команд и данных емкостью 8 Кбайт (ARM7100) и 4 Кбайт (ARM7500 и ARM7500FE),
3. MMU,
4. буфера записи,
5. наличие режимов энергосбережения.

Однокристалльная система

С технической точки зрения называть чипы архитектуры ARM процессорами не совсем верно,

т.к. помимо одного или нескольких вычислительных ядер они включают целый ряд сопутствующих компонентов.



термины **однокристалльная система** и **система-на-чипе**.

Например:

Новейшие однокристалльные системы для смартфонов и планшетных компьютеров включают:

1. контроллер оперативной памяти,
2. графический ускоритель,
3. видеodeкодер,
4. аудиокодек,
5. опционально модули беспроводной связи.

Отдельные компоненты однокристалльной системы могут быть разработаны как непосредственно ARM Limited, так и сторонними компаниями.

Например:

1. Графические ускорители от:
2. Qualcomm (графика Adreno),
3. NVIDIA (графика GeForce ULP),
4. Imagination Technologies (PowerVR) → гаджеты Apple и Amazon, планшетики Samsung Galaxy Tab 2, смартфоны на базе процессоров MTK.

Cortex™ -A9

ARM CoreSight™ Multicore Debug and Trace

ARMv7 32b CPU

NEON™
Data Engine

Floating Point
Unit

16-64k
I-Cache

16-64k
D-Cache

Core
1 2 3 4

ACP

SCU

Dual 64-bit AMBA3 AXI

Cortex™ -A5 MPCore

ARM CoreSight™ Multicore Debug and Trace

Generic Interrupt Control and Distribution

FPU/NEON
Data Engine

FPU/NEON
Data Engine

FPU/NEON
Data Engine

FPU/NEON
Data Engine

Integer CPU

Integer CPU

Integer CPU

Integer CPU

L1 Cache

L1 Cache

L1 Cache

L1 Cache

Snoop Control Unit (SCU)

Direct Cache
Transfers

Snoop
Filtering

Private
Peripherals

Accelerator
Coherence

Dual 64-bit AMBA3 AXI

Interrupt Control

Cortex-A15
MPCore

CPU

CPU

L2 Cache

Cortex-A7
MPCore

CPU

CPU

L2 Cache

CCI-400 Coherent Interconnect

Устаревшие и современные поколения чипов.

Морально устаревшими, но все еще широко распространенными процессорными архитектурами являются:

1. ARM9 (семейство ARMv5),
2. ARM11 (семейство ARMv6).

ARM9:

1. могут достигать тактовой частоты 400 МГц, установлены в:
 - мобильный телефон Sony Ericsson K750i,
 - мобильный телефон Nokia 6300.
2. набор инструкций Jazelle – позволяет комфортно работать с Java-приложениями (Opera Mini, Jimm, Foliant и др.).

ARM11:

обладают:

1. расширенным набором инструкций,
2. более высокой тактовой частотой.

До сих пор применяются в смартфонах начального уровня:

- Samsung Galaxy Pocket,
- Nokia 500.

Современные поколения чипов

Чипы семейства ARMv7 – восемь ядер, тактовая частота свыше 2 ГГц.

Разработанные непосредственно ARM Limited процессорные ядра принадлежат к **линейке Cortex** и большинство производителей однокристальных систем используют их без существенных изменений.

Лишь компании **Qualcomm** и **Apple** создали собственные модификации на основе ARMv7:

1. Scorpion и Krait,
2. Swift,

Соответственно.

ARM Cortex-A8:

Основа таких известных SoC своего времени как:

1. Apple A4 (iPhone 4 и iPad),
2. Samsung Hummingbird (Samsung Galaxy S и Galaxy Tab).

Обладает:

1. вдвое более высокой производительностью по сравнению с ARM11,
2. ядро Cortex-A8 получило сопроцессор NEON для обработки видео высокого разрешения и поддержку плагина Adobe Flash



Но энергопотребление значительно выше, чем у ARM11.

ARM Cortex-A9.

1. Производительность ядер выросла примерно втрое по сравнению с Cortex-A8,
2. Появилась возможность объединять их по два или даже четыре на одном чипе.
3. Сопроцессор NEON стал уже необязательным, т.к.



компания NVIDIA в однокристальной системе Tegra 2 его упразднила для освобождения места под графический ускоритель.

В это время:

появились первые реализации предложенной **ARM Limited концепции big.LITTLE –**

однокристальные системы должны иметь одновременно мощные и слабые, но энергоэффективные процессорные ядра.



Первой реализацией концепции big.LITTLE стала система-на-чипе NVIDIA **Tegra 3** с четырьмя ядрами Cortex-A9 (до 1,7 ГГц) и пятым энергоэффективным ядром-компаньоном (500 МГц) для выполнения простеньких фоновых задач.

ARM Cortex-A5 и Cortex-A7.

1. компромисс между минимальным энергопотреблением ARM11 и приемлемым быстродействием Cortex-A8,
2. возможность объединения ядер по два-четыре.



многоядерные чипы Cortex-A5 и Cortex-A7 –Qualcomm MSM8625 и MTK 6589.

ARM Cortex-A15.

1. логическое продолжение Cortex-A9,
2. примерно сравнялся по быстродействию с Intel Atom.



В продаже гаджеты на базе NVIDIA Tegra 4 с четырьмя ядрами ARM Cortex-A15 и пятым ядром-компаньоном Cortex-A7.



Вслед за NVIDIA концепцию big.LITTLE подхватила компания Samsung: «сердцем» смартфона Galaxy S4 стал чип Exynos 5 Octa с четырьмя ядрами Cortex-A15 и таким же количеством энергоэффективных ядер Cortex-A7.

Дальнейшие перспективы

1. Развитие семейства процессоров ARMv8, представители которого в обязательном порядке будут 64-разрядными.
2. Развитие RISC-процессоров ядра Cortex-A53 и Cortex-A57:
3. первое энергоэффективное,
4. второе высокопроизводительное,
5. оба способны работать с большими объемами оперативной памяти.
6. Планируется вывести чипы ARM на серверный рынок: AMD и Calxeda.

Процессоры ARM конструктивно имеют множество особенностей, существенно выделяющих их из ряда подобных разработок:

1. использование RISC-архитектуры,
2. оптимизированная система команд,
↓
3. процессорные ядра с малым числом транзисторов,
↓
4. низкая стоимость,
5. малое энергопотребление,
↓
6. предикатное исполнение команд
↓
7. избегание многих «близких» условных переходов внутри программы
8. оптимизированная система команд
↓
9. улучшение быстродействия.

Также ARM процессоры могут содержать и другие интересные решения:

в зависимости от модели процессора в нем могут применяться специализированные блоки и модули:

1. блок Thumb, позволяет процессору использовать не 32 а 16 разрядные инструкции



2. блок Jazelle позволяет исполнять байт-код языка Java.

Учитывая особенности реализации процессоров для разных устройств, разработчики ввели обозначения, позволяющие определить нацеленность процессора.

Существуют три базовых направления:

- A** – Ядра для классического применения
- M** – Ядра для микроконтроллеров
- R** – Ядра для встраиваемых систем, работающих в режиме реального времени.

Внутрифирменная маркировка разработчиков предусматривает отображение большого количества информации.

Выглядит эта маркировка следующим образом:

ARM[NN][R][Z][Ext].

Здесь:

NN – номер семейства

R – тип блока защиты или управления памятью

Z – кэш-память

Ext – расширения процессора. В настоящее время имеются следующие варианты: T – поддержка режима Thumb; D – JTAG порт; M – быстрый умножитель; I – встроенный блок эмуляции; E – расширенный набор инструкций (подразумевает обязательное наличие функций TDMI, поэтому в случае использования индекса E индексы TDMI из названия опускаются); J – поддержка Java инструкций (режим Jazelle); F – блок векторной арифметики над числами с плавающей точкой; S – синтезируемая версия (поставляется производителю кристалла в виде исходного текста, требующего компиляции (синтеза), в отличие от несинтезируемых версий, которые имеют заданную и неизменяемую топологию).

Например:

процессорное ядро

ARM7TDMI

- принадлежит семейству ARM7,
- не имеет кэш-памяти и блока защиты памяти,
- поддерживает набор 16-разрядных команд,
- оснащено JTAG-отладчиком,
- имеет встроенные средства аппаратного умножения и блок эмуляции.

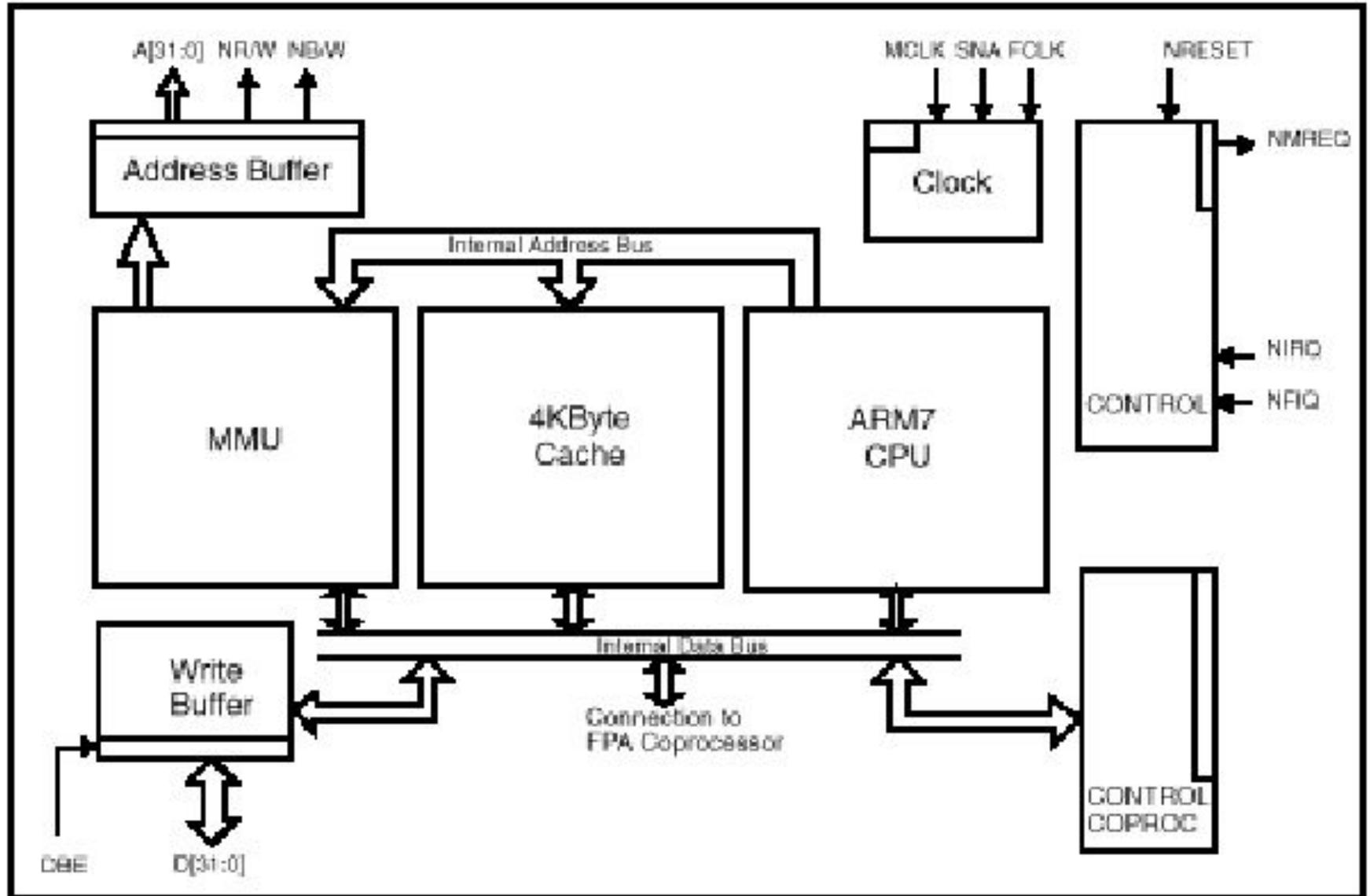
Структура процессора ARM7.

Основные характеристики ядра ARM7:

1. 32-разрядный RISC процессор (32-разрядные шины данных и адреса) с производительностью 17 MIPS при тактовой частоте 25 МГц (пиковая производительность 25 MIPS)
2. 32-разрядная адресация - линейное адресное пространство в 4 Гбайта - исключает потребность в сегментированной, разделенной на банки или оверлейной памяти
3. Тридцать один 32-разрядный регистр общего назначения и шесть регистров состояния
4. Регистры адресов, записи и конвейера
5. Циклическое сдвиговое устройство и перемножитель
6. Трехуровневый конвейер (выборка команды, ее декодирование и выполнение)
7. Рабочие режимы Big Endian и Little Endian
8. Напряжение питания 3,3 и 5 В
9. Малое потребление 0,6 мА/МГц, при изготовлении по CMOS технологии с топологическими нормами 0,8 мкм.
10. Полностью статическая работа, позволяющая дополнительно снижать потребление за счет уменьшения тактовой частоты, что идеально для критичных к потреблению применений
11. Быстрый отклик на прерывания применений реального масштаба времени
12. Поддержка систем виртуальной памяти
13. Простая но мощная система команд

Структура процессора ARM7.

Блок-схема ядра ARM7



32-разрядная система команд ядра ARM7 содержит одиннадцать базовых типов команд:

1. Два типа используют встроенное арифметико-логическое устройство, циклическое сдвиговое устройство и умножитель при операциях над данными в банке из 31 регистра, форматом по 32 разряда каждый;
2. Три класса команд управления перемещением данных между памятью и регистрами, один оптимизированный на обеспечение гибкости адресации, другой под быстрое контекстное переключение и третий под подкачку данных;
3. Три команды управляют потоком и уровнем привилегии выполнения;
4. Три типа предназначены для управления внешними сопроцессорами, что позволяет расширить функциональные возможности системы команд за пределами ядра.

Особенности системы команд:

- Система команд ARM хорошо обрабатывается компиляторами языков высокого уровня.
- Процессор ARM7, при возникновении необходимости в некотором уменьшении объема кодов, допускает программирование и на ассемблере.

Программы, подготовленные даже для довольно эффективной 32-разрядной ARM системы команд, требуют памяти значительного объема



рост общей стоимости системы



Решение проблемы –

технология Thumb –

технология, позволяющую существенно сократить объем кодов, необходимых для реализации той же программы, что выполняется на 32-разрядной ARM системе команд.

Подходы для решения проблемы большого размера кода.

Архитектура с расширенными возможностями:

1. малое потребление мощности,
2. малый размер кристалла,
3. высокая производительность

+

**решения проблемы размера кодов –
новая система команд Thumb.**

Технология Thumb.

Существует несколько подходов, решающих проблему размера кода:

1. **Написание кода вручную на ассемблере**
Для получения минимального размера кода программист может писать коды вручную - на ассемблере. Однако, этот процесс может занять непрактично большое время и в результате может быть получен код, который трудно поддерживать и всего лишь на 10-20% более компактный чем код, полученный при использовании хорошего компилятора. Корень проблемы, неэффективность кода, все еще не решается.
2. **Использование улучшенного компилятора**
Технология компилирования может улучшить код, но опять таки меньшим размер кода будет при ручном кодировании на ассемблере.

3. **Использование компрессированного кода**
Одним из вариантов может быть использование некоторой формы сжатого кода, который разворачивается во время выполнения. Однако, быстрая декомпрессия, которая не будет снижать производительность процессора при выполнении этого кода, достаточно сложна и требует использования дополнительных ресурсов системы.

Решение от фирмы ARM

Сочетание программных и аппаратных средств.



Реальные преимущества:

1. Ведущее в отрасли соотношение производительность/потребление (MIPS/Watt)
2. Превосходную плотность кода
3. Малый размер кристалла
4. Признание и активную поддержку большим количеством фирм-разработчиков специализированных ИС для различных применений