Новое ядро микроконтроллеров Cortex-M23. Часть 1

Илья АФАНАСЬЕВ

В настоящее время доступно восемь семейств процессоров с архитектурой ARM Cortex-M [1]. Микроконтроллерные процессоры Cortex (Microcontroller Processors, Cortex-M) разработаны с учетом получения наименьшей площади кристалла и высокой энергоэффективности. Представители этого семейства имеют короткий конвейер и относительно низкие тактовые частоты и весьма популярны в микроконтроллерных встраиваемых системах. В статье рассмотрены особенности нового ядра микроконтроллеров Cortex-M23, анонсированного компанией ARM в октябре 2016 г.

Обзор процессоров Cortex-M23. Сравнение с M0+

Процессорные ядра Cortex-M0/M0+/M1 имеют архитектуру ARMv6-M, использующую набор инструкций из 56 команд, большинство из которых 16-бит, но оперируют 32-бит данными. Такого набора команд достаточно для большинства задач управления и обработки информации. Процессорное ядро с малым набором команд может быть реализовано с применением малого количества гейтов (начиная примерно от 12 тыс. для Cortex-M0 и Cortex-M0+). Однако многие из инструкций не используют старшие регистры (R8–R12) и имеют ограничения на объем данных в инструкциях с прямой адресацией — это следствие компромисса между требованием обеспечения энергоэффективности и производительностью.

Процессорное ядро Cortex-M23 создано на архитектуре ARMv8-M в варианте Baseline, представляющем собой расширение архитектуры ARMv6-M. Дополнительные инструкции включают:

- команды деления;
- сравнение с переходом, 32-бит переходы;
- команды для поддержки расширения TrustZone;
- команды эксклюзивного доступа (exclusive access instructions), обычно используются для операций с семафорами;
- команды прямой адресации с 16-бит данными;
- атомарные операции загрузки (load) и сохранения (store) для поддержки языка C стандарта C11 и управления атомарными переменными;
- новая программная модель MPU, использует архитектуру PMSA v8 и более гибко настраиваемые блоки памяти.

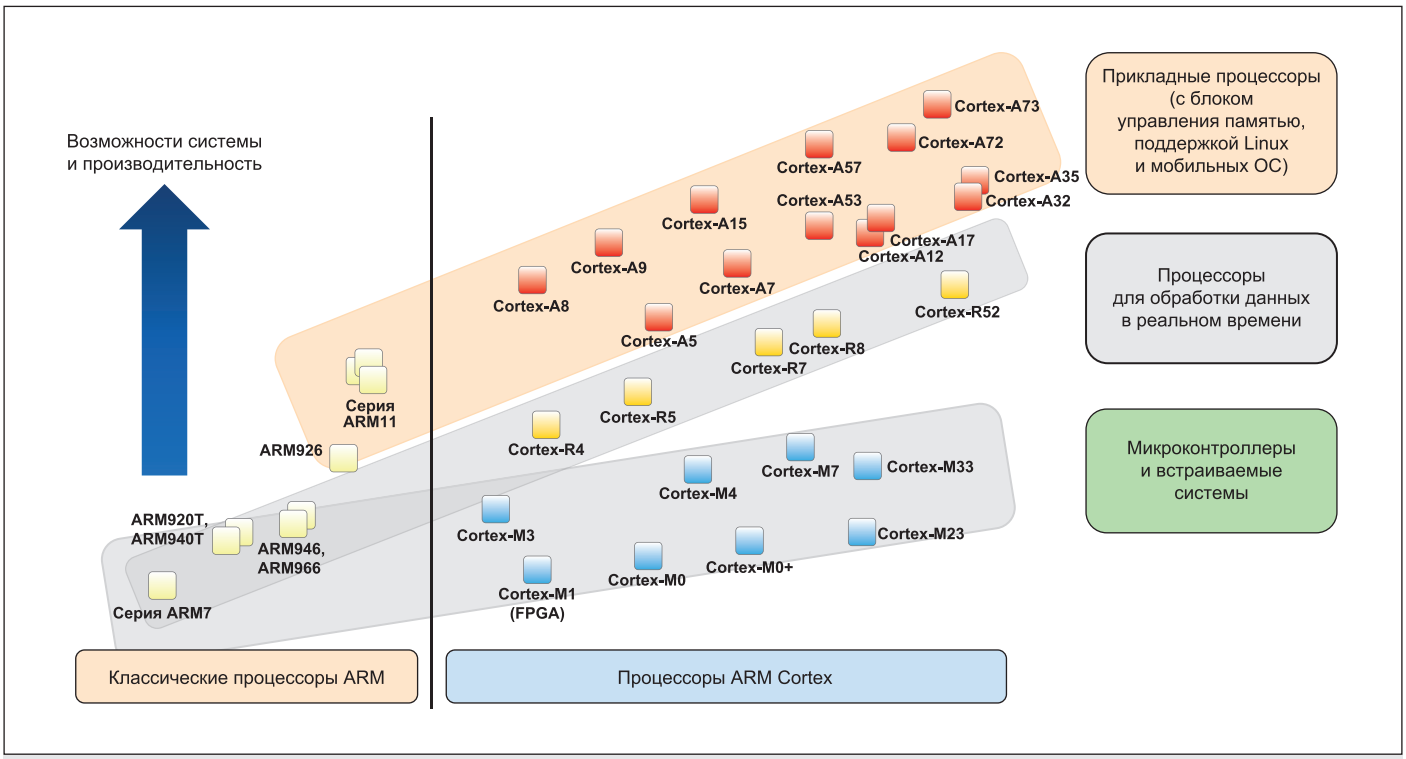

Рис. 1. Семейство ARM-процессоров

Таблица. Сравнение производительности ядер M0, M0+, M23

	Dhrystone DMIPS/МГц (v2.1)	Coremark/МГц (v1.0)
Cortex-M0	0,84	2,33
Cortex-M0+	0,94	2,42
Cortex-M23	0,98	2,5

Дополнительный набор команд позволяет увеличить производительность и может быть полезен для SoC-контроллеров, содержащих несколько процессорных ядер. За счет расширения системы команд ядро Cortex-M23 отличается от M0, M0+ чуть большей производительностью (таблица, рис. 1) [1].

Все процессоры Cortex-M имеют контроллер вложенных прерываний (Nested Vectored Interrupt Controller, NVIC) и схожую модель обработки исключений. Ядра Cortex-M0 и M0+ поддерживают до 32 прерываний и только исключения типа Hard Fault. Ядро M23, как и M3, M4, M7, поддерживает до 240 периферийных прерываний и может иметь две таблицы прерываний — для поддержки защищенных и незащищенных прерываний и исключений.

Cortex-M23: расширение безопасности TrustZone

В последние несколько лет «Интернет вещей» (Internet of Things, IoT) стал очень популярной темой для разработчиков встраиваемых приложений. По мере усложнения IoT-систем повышаются требования к обеспечению безопасности, однако растущий рынок предполагает сокращение сроков разработки. Традиционно вопросы обеспечения безопасности решаются путем разделения ПО на привилегированные и непривилегированные части. Привилегированное ПО может использовать модуль защиты памяти (Memory Protection Unit, MPU) для предотвращения доступа непривилегированных приложений к критическим системным ресурсам, включая информацию, для которой необходимы повышенные меры безопасности данных. Подобная схема отлично подходит для ряда систем IoT, но в отдельных случаях простое наличие двух разделов оказывается недостаточным. В частности, для систем, содержащих множество сложных привилегированных программных компонентов, одна уязвимость в одном из привилегированных приложений может допустить захват управления системой.

Наиболее значимым улучшением в архитектуре ARMv8-M является расширение безопасности TrustZone — это технология, предусматривающая новый уровень управления безопасностью и позволяющая иметь несколько доменов в однопроцессорном контроллере. Данное расширение добавляет аппаратное ортогональное разбиение на безопасную (Secure, Trusted) и незащищенную (Non-secure, Non-trusted) области:

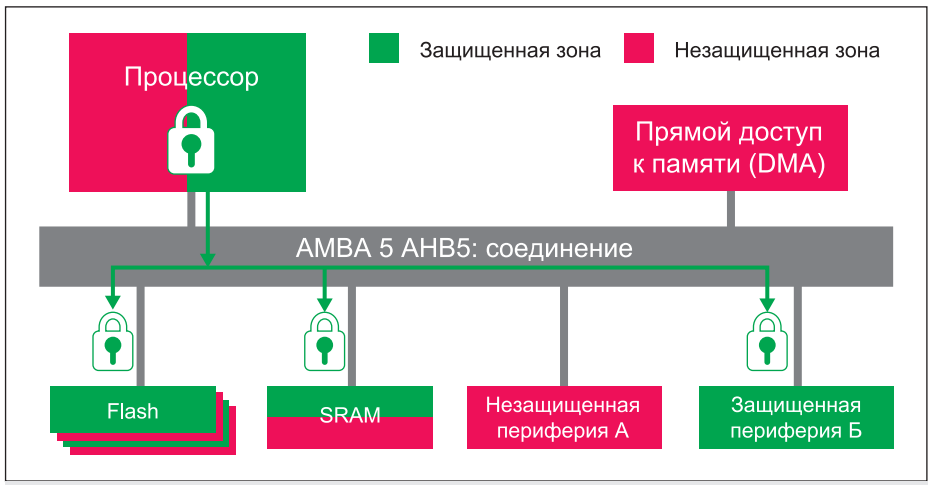

Рис. 2. Возможности шины AMBA 5 AHB5

- незащищенная область выделяется для приложений, не требующих особых мер безопасности;
- защищенная область предназначена для программных компонентов и ресурсов, требующих защищенного, разграниченного доступа (хранение чувствительной информации, работа с криптофункциями и т. д.).

Незащищенное программное обеспечение имеет доступ только к незащищенной памяти и периферии, в то время как код из защищенной части обладает доступом к обоим сегментам.

С такой системой разграничения разработчики ПО могут, как и раньше, создавать приложения в небезопасном сегменте, а также использовать, например, защищенные библиотеки связи, предоставляемые поставщиками чипов для обеспечения безопасных соединений IoT. В этом случае, даже если есть уязвимость в незащищенном программном обеспечении, механизм защиты TrustZone способен помешать захватить контроль над всем устройством, ограничивая влияние атаки и потенциально позволяя удаленно восстанавливать контроль над системой. Кроме того, в архитектуре ARMv8-M реализована проверка ограничений стека и расширенная архитектура MPU, что упрощает развертывание дополнительных мер безопасности.

Процессорное ядро в системе безопасности TrustZone является только одной из частей системы. Необходимы дополнительные аппаратные усовершенствования для поддержания безопасности на системном уровне. ARM выпустил спецификацию AMBA (Advanced Microcontroller Bus Architecture) 5 AHB5 (Advanced High-performance Bus) — усовершенствование спецификации AHB Lite из AMBA 3.0. Обновленный вариант шины обеспечивает взаимодействие с защищенными и незащищенными ресурсами (рис. 2).

Концептуально TrustZone в архитектуре ARMv8-M аналогична широко используемой TrustZone ARM в процессорах семейства Cortex-A. Однако основные операции TrustZone для ARMv8-M имеют свои особенности, поскольку они оптимизированы для встраиваемых систем, которые не только требуют реакции в реальном времени, но и обеспечивают высокую энергоэффективность и малую площадь на кристалле.

В отличие от процессоров Cortex-A деление на безопасную и небезопасную части основано на карте памяти, и обращения к соответствующим ресурсам происходят автоматически без помощи обработчика исключений (Secure Monitor), что увеличивает быстродействие.

Ядро Cortex-M23 предоставляет два возможных варианта для управления различными конфигурациями безопасности процессора (рис. 3).

Первый предполагает использование SAU (Security Attribution Unit) — аппаратного блока, подобного MPU (Memory Protection

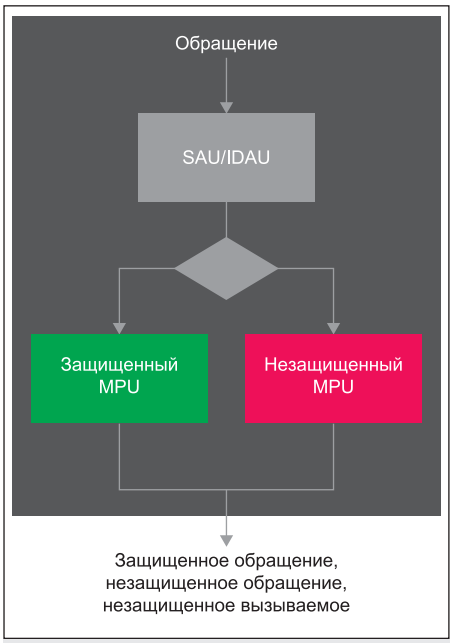

Рис. 3. Управление различными конфигурациями безопасности процессора

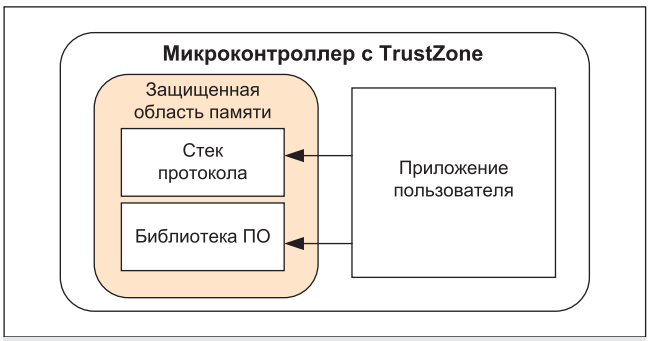

Рис. 4. Пример использования TrustZone при работе с защищенными библиотеками и стеками протоколов

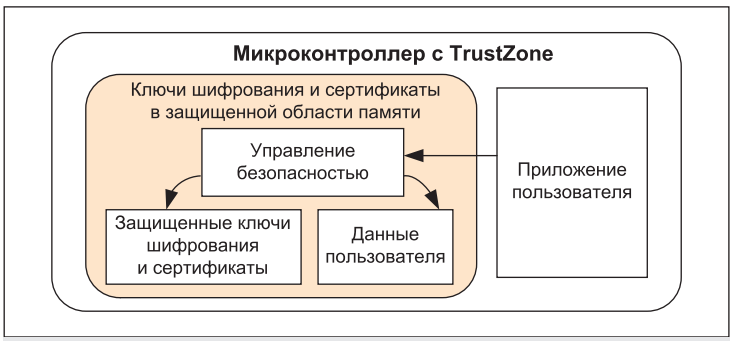

Рис. 6. Управление безопасностью с использованием TrustZone

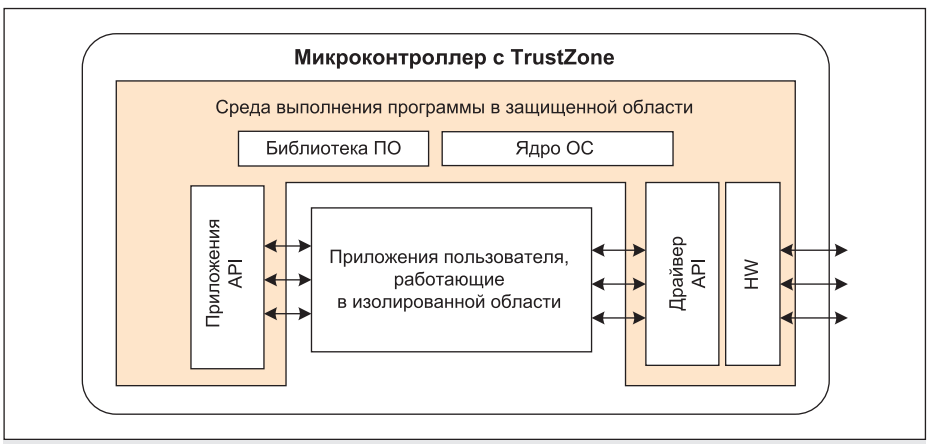

Рис. 5. Пример использования TrustZone при работе с защищенными библиотеками и драйверами периферийных устройств

Unit), который управляет всеми обращениями ядра к защищенному (безопасному) или незащищенному (небезопасному) регионам. Применение SAU подразумевает, что конфигурация должна располагаться в другом месте в архитектуре MCU для обеспечения безопасности.

Второй подход заключается в использовании блока определения атрибутов (Implementation Defined Attribution Unit, IDAU) — внешнего аппаратного блока по отношению к ядру.

Благодаря блоку IDAU всю встроенную память — например, память программ Flash, энергонезависимую память данных Data Flash и ОЗУ, — можно разделить на субрегионы, зарезервированные как для защищенного, так и незащищенного приложения.

С применением TrustZone архитектура безопасности расширяется до уровня системы, где не только части кода и данных, но и каждый из периферийных модулей (прерываний и т. д.) могут быть отнесены к безопасным или небезопасным сегментам. Последовательность обработки прерываний и исключений автоматически сохраняет и восстанавливает защищенные данные в регистрах, чтобы предотвратить утечку защищенной информации. В результате технология безопасности TrustZone позволяет системам сохранять работу в режиме реального времени и обеспечивает надежную основу безопасности для приложений IoT, без усложнений в разработке программного обеспечения.

Примеры применения технологии TrustZone

Защита от удаленных программных атак

Если через открытые каналы получения данных удаленный узел подвергается программной атаке, то инфицированный код будет обнаружен и удален. Микроконтроллеры с TrustZone могут иметь разделенные ресурсы. Защищенная область может содержать копию приложения незащищенной области. При детектировании изменений в незащищенной области защищенная часть кода переписывает приложение и восстанавливает работоспособность системы.

Защита интеллектуальной собственности

Защищенная область может содержать интеллектуальную собственность (алгоритмы, функции) и предоставлять доступ (API) из незащищенной области только к вызову определенных функций (рис. 4). Обращение к неразрешенным областям памяти для чтения или записи кода будет приводить к сбросу системы. Таким образом, существует возможность одной компании предоставлять другой компании запрограммированные микроконтроллеры с доступом к вызову своего кода, но без раскрытия своей интеллектуальной собственности и другой критичной информации.

Выполнение кода в устройствах с сертифицированным ПО

Многие контроллеры со встроенными беспроводными интерфейсами, такие как Bluetooth-чипсеты, содержат предзагруженное ПО и область памяти с доступом для разработчиков, чтобы добавлять свои программные компоненты. Благодаря технологии TrustZone предзагруженное ПО можно размещать в защищенной области, и его поведение нельзя изменить из приложения, исполняемого в незащищенной области (рис. 5). Это дает уверенность в том, что сертифицированное ПО сохраняется в неизменном виде и защищено от анализа и копирования.

Защита от копирования

Ключи и права доступа — важная часть конечного продукта в подключаемых устройствах и должны быть защищены от клонирования. Удаленный подключаемый узел и хост должны взаимно удостовериться (аутентифицироваться), чтобы предотвратить попытки клонирования. TrustZone может защитить ключи от попыток их считывания инфицированным ПО.

Управление безопасностью в устройствах IoT

Большинство IoT-устройств требует сохранения в секрете информации, такой как пользовательские данные и ключи шифрования. Технология TrustZone позволяет сохранять подобные сведения и связанное с ними ПО (которое имеет прямой доступ к защищаемым данным) в защищенной области памяти. Технология TrustZone позволяет пользовательскому ПО из незащищенной области иметь доступ к защищенной информации через API (рис. 6). Доступ может быть опосредованный — например, получать результат операций с защищенными ключами шифрования или получать доступ к ключам, пройдя процедуру аутентификации.

Защищенный загрузчик

Защищенный загрузчик (Secure Boot) — важная часть поддержания безопасности. Он обеспечивает проверку целостности и достоверности запускаемого приложения. При старте микроконтроллера криптомодуль производит аутентификацию и проверку целостности кода, предотвращая возможность запуска подмененного неавторизованного кода.

Защищенное обновление прошивки

В варианте обновления прошивки по воздуху (OTA) важно удостовериться в целостности и аутентичности обновляемой версии прошивки. Защищенный загрузчик и криптофункции контроллера могут дешифровать образ и удостовериться в его аутентичности до обновления прошивки.

Реализация работы с корневым доверительным центром

Подключаемые устройства с аутентификацией требуют поддержки корневого доверительного центра (Root of Trust). Особенно это актуально для устройств с возможностью обновления прошивки по воздуху (OTA). В системе с TrustZone код, предоставляющий поддержку обновления прошивки и аутентификацию, может быть расположен в защищенной области (рис. 7). Даже если устройство скомпрометировано на уровне приложения, его нельзя стереть и заменить прошивку.

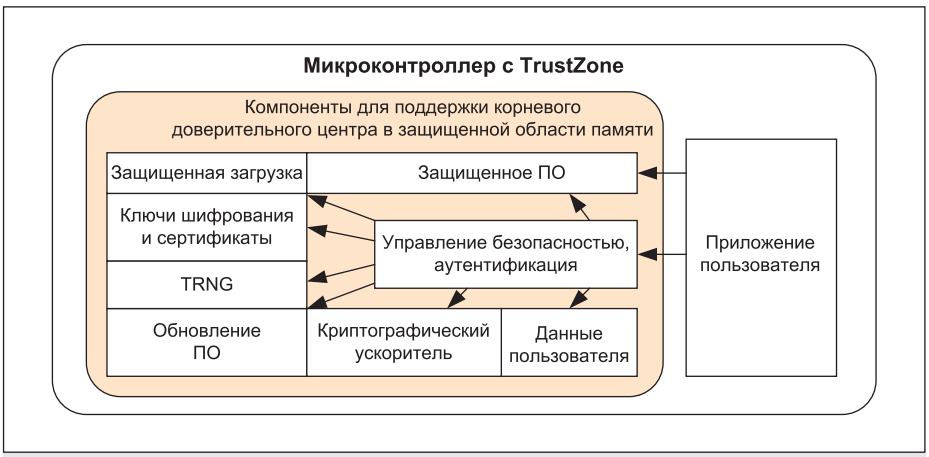

Рис. 7. Реализация корневого доверительного центра (Root of Trust) в системе с TrustZone

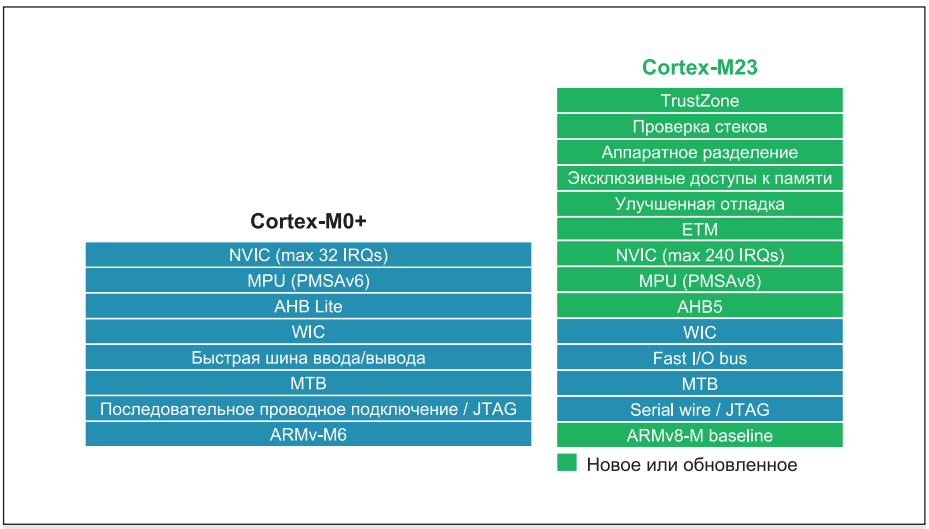

Рис. 8. Отличия ядер Cortex M0+ и Cortex-M23

Выводы

Ядро Cortex-M23 является развитием популярных Cortex-M0, M0+ и получило ряд усовершенствований, нацеленных на повышение производительности и защищенности (рис. 8).

Благодаря компактному ядру контроллеры Cortex-M0 и M0+ потребляют меньше энергии, чем Cortex-M3, M4, M7. Процессорное ядро Cortex-M23 не такое маленькое, как Cortex M0+, но в целом, при аналогичной конфигурации, имеет сравнимое потребление и обладает большей производительностью. Возможность ядра Cortex-M23 поддерживать TrustZone позволяет упростить разработку устройств с повышенной степенью безопасности, что востребовано в растущем рынке устройств «Интернета вещей». ■

Продолжение следует.

Литература

1. Yiu J. ARM Cortex-M for Beginners. An overview of the ARM Cortex-M processor family and comparison. ARM, March 2017.
2. Yiu J. ARMv8-M Architecture Technical Overview. White paper. ARM, 2015.