

Программный пакет разработки для ARM-микроконтроллеров RealView Keil

Программный пакет разработки RealView Microcontroller Development Kit (MDK-ARM) компании Keil объединяет компилятор C/C++ ARM RealView и интегрированную среду разработки Keil μ Vision. Этот продукт предоставляет в распоряжение разработчика многофункциональную среду разработки, оптимизированную для работы с широкой линейкой микроконтроллеров на базе ядра ARM. MDK-ARM обеспечивает поддержку устройств, базирующихся на ARM7, ARM9 и Cortex-M3 таких производителей, как Analog Devices, Atmel, Freescale, Luminary, OKI, NXP, Samsung, Sharp, STMicroelectronics и TI.

Использование MDK-ARM позволяет уменьшить цикл проектирования и существенно сократить время выхода продукта на рынок. MDK-ARM — идеальное средство, стандартизированное для промышленных проектов с развитой системой отладки и поддержкой реального времени.

Сергей КОПЫТИН
tools@microcontroller.ru
Кирилл ДОРОФЕЕВ
info@microcontroller.ru

Интегрированная среда разработки μ Vision IDE

Интегрированная среда разработки (Integrated Development Environment) — μ Vision IDE — фирмы Keil сочетает в себе уникальные возможности: управление проектами, создание отдельных программ, редактирование текста программы, отладку программ, позволяет непосредственно вызывать симулятор или внутрисхемный эмулятор. Редактор и отладчик объединены в одно приложение, что упрощает процесс разработки проекта. μ Vision проста в использовании и содержит богатый набор опций:

- **Device Database** — интеллектуальная база данных с детальной информацией обо всех контроллерах, поддерживаемых инструментальными средствами Keil. База данных автоматически конфигурирует ассемблер, компилятор C/C++ и компоновщик для выбранного микроконтроллера, генерирует файлы описания регистров, конфигурирует симулятор CPU и периферии, корректирует код инициализации и программные алгоритмы. Device Database содержит подробные инструкции по конфигурированию, ссылки на другие источники информации и включает более чем 200 ARM-микроконтроллеров (полный список этих устройств можно найти на сайте www.keil.com/dd).
- **Project Manager** — менеджер проекта, дает методику создания проекта из исходных файлов, различных опций разработки и директорий. Программный проект состоит из большого числа файлов, которые обра-

батываются индивидуально. Например, часть файлов подлежит компиляции, а другие следует ассемблировать. При этом достигается простая интеграция различных исходных файлов в проект.

- **Building Projects** — менеджер проекта, позволяет создавать в одном проекте отдельные файлы для симуляции, отладки с помощью программы-эмулятора и программирования EPROM. Ассемблер и компилятор автоматически генерируют зависимости между файлами и добавляют их в проект. При глобальной оптимизации μ Vision неоднократно компилирует исходный файл для достижения оптимального использования регистров. Все параметры проекта сохраняются в специальном файле, то есть компиляция и линковка проекта происходят по нажатию одной клавиши.
- **μ Vision Editor** — интегрированный редактор облегчает подготовку исходного текста за счет многооконности, выделения синтаксиса цветом и исправления ошибок в режиме диалога. Редактор настраивается в соответствии со вкусами пользователя. Интерактивная система исправления ошибок позволяет отслеживать ошибки и предупреждения, которые появляются в отдельном окне во время отладки программы. Существует возможность исправления файлов проекта, пока μ Vision продолжает проверку в фоновом режиме. Номера строк, содержащих ошибку или предупреждение, автоматически обновляются при изменении исходного файла.

- **μ Vision Utilities** — мощные интегрированные утилиты, облегчающие создание проекта. Source Browser — база данных программных символов для быстрой навигации по исходному файлу, Find in Files — полный поиск во всех файлах, PC-Lint — анализ синтаксиса исходного кода, Flash tool — утилиты загрузки флэш-памяти и многие другие.
- **On-line help** — встроенная система помощи, содержит как краткую информацию об использовании программного обеспечения, так и полный перечень руководств пользователя On-line Manuals.

Отладчик-симулятор μ Vision Debugger

μ Vision Debugger — это надежный, полнофункциональный отладчик, который позволяет вести отладку программ, написанных на C и ассемблере или в смешанном формате, а также сделать выбор между симулятором, монитором, JTAG-отладчиком и внутрисхемным эмулятором.

Полная симуляция включает быструю симуляцию системы команд и встроенной периферии (АЦП, ЦАП, таймеров, UART, CAN, I²C, прерываний, внешних сигналов и I/O) плюс управление. Симуляция предоставляет дополнительные возможности, не достижимые при JTAG-отладке: точный временной расчет и детальный анализ исполнения программы при различных параметрах.

μ Vision Debugger предоставляет разработчику следующие возможности:

- **Breakpoints** — задание точек останова осуществляется через результат выражения или обращение к ячейке памяти/ переменной. Для редактирования и просмотра параметров контрольных точек служит окно Breakpoint. Точки останова могут остановить исполнение программы, запустить команду или сценарий отладчика.
- **Memory & Register** — просмотр областей памяти и состояний регистров в специальных окнах. Окно Serial I/O делает наглядной симуляцию последовательного ввода/вывода. μ Vision предлагает несколько путей для просмотра и изменения переменных и памяти.
- **Performance Analyzer** — анализатор производительности, фиксирует время исполнения программных модулей. Задавая список модулей для анализа, пользователь получает диаграмму затрат времени на каждую часть программы.
- **Code Coverage** — анализатор эффективности кода локализует части программы, к которым редко происходит обращение, что позволяет удалить ненужный код. Анализ эффективности кода осуществляется на уровне C и ассемблера. Подробная статистика: время исполнения, число обращений.
- **Logic Analyzer** — логический анализатор, осуществляет графический мониторинг аналоговых и цифровых сигналов, внутренних переменных, импульсов на входах/выходах, сигналов JTAG. Анализ временных диаграмм ведется одновременно с трассировкой кода, используется цветной курсор, виден временной отсчет. Анализ кода — отображение в окне анализатора. При несанкционированном доступе в память происходит останов симуляции.
- **Execution Profiler** — профилировщик. Это новая опция, которая разрешает или запрещает отображать информацию о времени исполнения команд в окне отладки. Она помогает оптимизировать текст программы для достижения максимальной производительности.
- **Target Monitor** — монитор, обеспечивает прямой интерфейс при отладке программ на плате и легко настраивается на любой микроконтроллер. Отладка ничем не отличается от режима симуляции. Требования к ресурсам микроконтроллера со стороны монитора минимальны.
- **JTAG Interface** — поддержка разнообразных опций отладки через интерфейс JTAG для связи с различными устройствами либо с помощью адаптера USB-JTAG ULINK2 (поддержка ARM, Cortex-M3, XC16x и μ PSD), а также отладчиков третьих фирм с ARM RDI (Remote Debugger Interface).
- **Real-Time Agent** — это небольшой программный модуль на C (занимает в приложении пользователя около 1500 байт), который позволяет вести отладку «на лету» и не требует остановки системы. Коммуникация осуществляется через адаптер

USB-JTAG ULINK2 или ULINK-ME. Отладка «на лету» дает возможность во время исполнения программы осуществить чтение и запись памяти, доступ к переменным, установку точек останова, Serial I/O (printf).

- **Serial Wire Debug** — новый 2-pin интерфейс для процессоров на базе Cortex-M3, который заменяет стандартный интерфейс JTAG, предлагая дополнительно к его возможностям доступ к памяти в реальном времени без останова процессора и какого-либо резидентного кода. Serial Wire Viewer использует еще один дополнительный контакт и позволяет просматривать значения переменных и сообщения об отладке при работающем на полной скорости процессоре.
- **ETM Trace** — технология трассировки Embedded Trace Macrocell для ARM9, которая позволяет загружать инструкции и данные трассировки из процессора. У ETM есть существенное ограничение: поскольку выводы ETM часто совмещены с другими I/O (например, ADC или GPIO), требуются дополнительные аппаратные затраты. ETM Trace поддерживается устройством трассировки Signum JTAGjet-Trace.

Средства компиляции RealView Compilation Tools

Средства компиляции RealView Compilation Tools состоят из компилятора C/C++, библиотеки MicroLib, ассемблера и компоновщика. RealView Compilation Tools для ARM транслирует исходные файлы на C в объектные файлы, которые содержат полную символическую информацию для отладки с помощью μ Vision Debugger или внутрисхемного эмулятора. Кроме объектных файлов компилятор генерирует файл листинга, который опционально может включать таблицу символов и перекрестные ссылки.

RealView C Compiler (armcc)

RealView C Compiler (armcc) — компилятор, который оптимизирует код и позволяет уменьшить объем памяти и сократить потребление за счет повышения производительности при той же тактовой частоте. Имеет следующие характеристики:

- Одновременное использование команд ARM и Thumb в исходном файле. Режим ARM необходим для обработки прерываний и быстрых алгоритмов обработки сигналов, тогда как режим Thumb обеспечивает минимальный размер кода.
- Атрибуты функций, которые обеспечивают доступ к аппаратным ресурсам ARM: `o_irq` и `_fiq` — для создания стандартных и быстрых процедур; `_swi(id)` — процедура обработки прерываний.
- Фрагменты ассемблера можно вставлять в функции на C с доступом со стороны ассемблера к переменным C, что необходимо для задач обработки сигналов. При ас-

семблерных вставках поддерживается полная оптимизация программы.

- Для преобразования аргументов функций автоматически используются регистры CPU.
- Большинство библиотечных функций и все операции с плавающей точкой можно вызывать из главной программы и из прерываний без специальных схем защиты вызова библиотек.
- Операции одинарной и двойной точности с плавающей точкой в соответствии с IEEE-754.

MicroLib

Высоко оптимизированная библиотека для приложений на C, использование которой сокращает размер системного кода от 50 до 90% по сравнению со стандартной библиотекой. Обе библиотеки MicroLib и ARM Standard Library включены в состав RealView MDK ARM. Основные характеристики MicroLib следующие:

- Оптимизирована для встраиваемых приложений с жесткими требованиями по объему памяти.
- Разработана для работы без операционной системы, однако может работать совместно с любой из ОС, например RTX.
- Минимальные ресурсы для неиспользуемых функций ОС.
- Неиспользуемые функции удаляются из памяти.
- Большинство функций инициализируется в момент вызова.
- Не содержит файлов I/O или wide character support.
- Ряд инструкций выполняется более медленно, чем в Standard Library.

RealView Assembler (armasm)

Ассемблер транслирует инструкции ARM и Thumb в объектные файлы, обрабатываемые Linker/Locator или Library Manager, и отличается высокой скоростью исполнения, малым объемом кода и четким управлением аппаратной частью. Имеет следующие преимущества:

- Гибкие директивы классов памяти позволяют создавать переменные и фрагменты программ для всех методов доступа и типов памяти.
- Поддержка макросов для повторов последовательностей команд.
- Условное ассемблирование для создания нескольких приложений из одного исходного файла.
- Файл листинга включает перекрестные ссылки, дающие детальную символическую информацию об исходном файле.

RealView Linker (armLink)

Компоновщик осуществляет генерацию и оптимизацию кода, объединяет объектные модули ARM, создает исполняемые программы, распознает ссылки и назначает абсолют-

ные или фиксированные адреса для сегментов программы. На выходе линкера — абсолютные объектные модули для загрузки в μ Vision Debugger или Intel HEX файл для программирования устройств. Дает следующие преимущества:

- Увеличение скорости исполнения часто используемых функций без свойственных вызову функции задержек на изменение параметров и возврат.
- Глобальная регистровая оптимизация сокращает требуемый объем стека при увеличении скорости исполнения программы.
- Полный листинг, который содержит конфигурацию памяти, входные модули, карту памяти, таблицу символов и перекрестные ссылки.
- Глобальный листинг генерируемого кода, содержащий информацию о символьном дизассемблировании.
- Вычисление требуемого размера стека избавляет от проверки стека во время исполнения программы.

При помощи RealView Compilation Tools удается достичь меньшей стоимости системы в целом за счет высокой плотности кода (уменьшает требуемый объем программной памяти) и высокой производительности (позволяет реализовать задачу на более медленном процессоре, что ведет к снижению потребления энергии).

Библиотека RL-ARM — RealView Real-Time Library

С помощью операционной системы реального времени разработчик получает следующие преимущества: разбиение программы на простые задачи, модульный принцип построения многозадачных приложений, сокращение затрат времени на разработку и тестирование приложения. В качестве примера операционной системы реального времени для ARM микроконтроллеров можно привести RealView Real-Time Library компании Keil (RL-ARM). Библиотека RL-ARM интегрирована с компилятором RealView, легко конфигурируется для любого ARM-микроконтроллера и содержит следующие стандартные компоненты для создания современных встраиваемых систем:

- RTX Real Time Kernel — полнофункциональная операционная система реального времени RTX Keil, которая осуществляет управление процессорным временем, семафорами (для распределения ресурсов), событиями (для синхронизации задач) и почтовыми ящиками (для связи между задачами). RTX легко настраивается при помощи файла конфигурации.
- TCP/IP Networking Suite — коммуникационный модуль на базе стека TCP/IP, специально адаптированный для встраиваемых приложений. Он включает в себя TCP- и UDP-сокеты, PPP- и SLIP-интерфейсы, DNS, Telnet, TFTP, SMTP, встроенный веб-

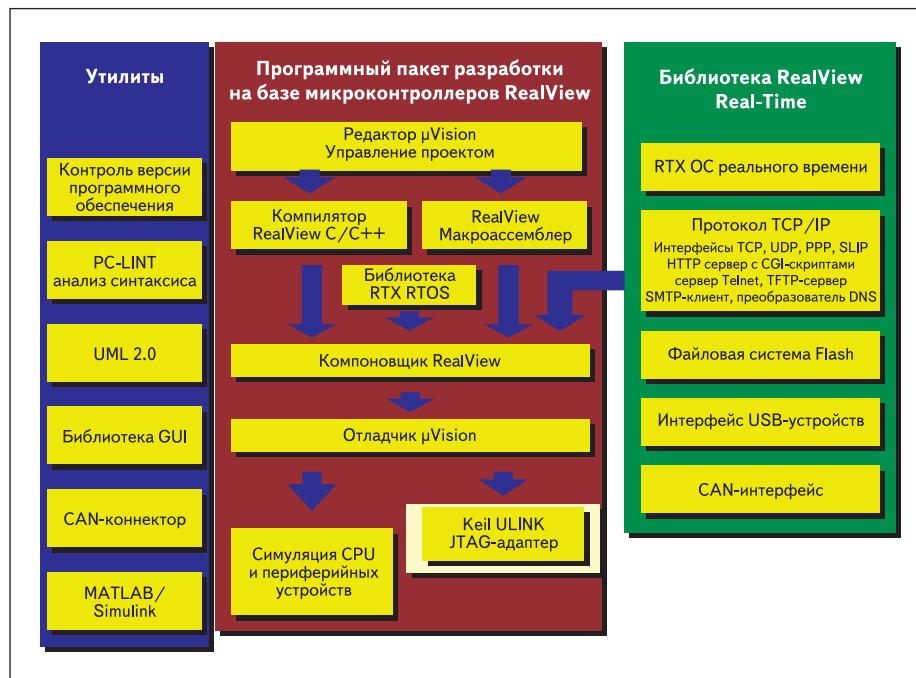


Рисунок. Обзор решений для разработки от Keil MDK для микроконтроллеров

сервер и CGI с защитой паролем. Стек работает как с Ethernet, так и с последовательным интерфейсом (модемом). Сконфигурованные примеры предназначены для работы на нескольких стандартных демонстрационных платах.

- Flash File System — система, которая позволяет сохранять файлы на Flash, ROM или RAM. Это полезно в решениях, где требуется большое количество памяти для хранения данных или для систем с интернет-интерфейсами HTTP или FTP.
- USB Device Interface — драйверы USB, которые позволяют подключать стандартные устройства, такие как HID или Mass Storage.
- CAN Interface — в RTX Keil входят библиотеки CAN, которые предоставляют мощный и простой интерфейс для различных контроллеров CAN.

Библиотека RL-ARM создана для решения типовых задач, которые встают перед разработчиком, и позволяет:

- управлять несколькими заданиями на одном CPU;
- контролировать время выполнения задачи;
- осуществлять связь и конфигурировать взаимодействие между различными программами в системе;
- осуществлять доступ к Internet через Ethernet или последовательное соединение (модем);
- реализовать встроенный веб-сервер, включающий в себя CGI-скрипты;
- реализовать E-mail через Simple Mail Transfer Protocol (SMTP).

Программный пакет RealView MDK-ARM включает в свой состав операционную систему RTX, но не содержит полностью библиотеку RL-ARM, которая является самостоятельным программным пакетом. RL-ARM до-

полнительно к компонентам, включенным в состав MDK-ARM, содержит исходный код RTX Source Code, Flash File System, TCP/IP Protocol Suite и драйверы USB/CAN. Однако как RTX, так и RL-ARM полностью интегрированы в MDK-ARM.

Все компоненты MDK-ARM и RL-ARM поставляются как royalty-free, то есть без дополнительных лицензионных отчислений на каждый создаваемый с их помощью проект. Каждая приобретаемая лицензия позволяет одному инженеру использовать эти программные продукты для неограниченного числа проектов.

Функции RTX описаны на стандартном языке C и компилируются при помощи компилятора RealView. Описание на языке C позволяет легко определять функции задачи без конфигурирования стека и переменных.

Интегрированная среда разработки μ Vision IDE обеспечивает полную поддержку RTX при отладке и предоставляет информацию о задаче RTX, ее приоритете и состоянии. Эта информация дает понимание работы приложений и показывает очередность и продолжительность исполняемых задач, их переключений и прерываний.

Keil RTX оптимизирована для 32-разрядной архитектуры ARM и имеет следующие характеристики: малое время переключения <5 мкс, 255 уровней прерываний, до 256 выполняемых одновременно задач, неограниченное число почтовых ящиков и таймеров пользователя. Несмотря на то, что RTX является многофункциональной и мощной системой, ее системные требования невелики: по крайней мере, один таймер, минимум 500 байт ROM и менее 5 кбайт RAM.

Заключение

Программный пакет Keil MDK-ARM предлагает реальные преимущества для разработчиков встраиваемых систем (рисунок). Обеспечение работоспособности системы, создание сложных сетевых решений и сов-

местимость различных компонентов системы теперь становятся интуитивно понятными. Использование MDK-ARM позволяет уменьшить цикл проектирования и существенно сократить время выхода продукта на рынок. MDK-ARM — идеальное средство, стандартизированное для промышленных

проектов с развитой системой отладки и поддержкой реального времени. ■

Литература

1. www.keil.com
2. www.arm.com