

32-разрядные микроконтроллеры AVR32 uC3: с чего начать?

Алексей КУРИЛИН
ak@efo.ru

В данной статье рассматриваются программные и аппаратные средства поддержки разработок для 32-разрядных микроконтроллеров AVR32 uC3.

Микроконтроллеры AVR32 uC3 вызывают интерес разработчиков своими привлекательными потребительскими характеристиками, такими как быстродействующее ядро, широкий набор периферийных узлов и очень низкое энергопотребление. Но любой привлекательный микроконтроллер очень сложно применить в проекте без доступных программных и аппаратных средств поддержки разработок, поэтому именно такие средства для AVR32 uC3 будут кратко описаны в этой статье.

Аппаратные средства поддержки разработок для AVR32 uC3

Микроконтроллеры с архитектурой AVR32 uC3 представлены двумя линейками микросхем — AT32UC3Axxxx и AT32UC3Bxxxx, для которых выпускаются оценочные наборы EVK1100 и EVK1101 соответственно. Платы обоих наборов построены классическим образом — на плате распаян старший из линейки микроконтроллеров и есть набор внешних компонентов, требуемых для запуска микроконтроллера, обеспечения отладки (с использованием внешнего эмулятора) и тестирования запускаемых на плате приложений.

На плате EVK1100 (рис. 1) находятся:

- микросхема памяти DataFlash объемом 8 Мбайт для хранения пользовательских данных, подключенная по интерфейсу SPI микроконтроллера;
- микросхема памяти SDRAM объемом 32 Мбайт, подключенная на шину EBI микроконтроллера;
- разъем mini-USB A-B, подключенный к интерфейсу USB OTG микроконтроллера;
- разъем RJ-45 с микросхемой, реализующей физический уровень Ethernet, подключенные к блоку MAC/Ethernet микроконтроллера;
- кварц на 12 МГц на OSC1, часовой кварц на OSC2; предусмотрена возможность установки кварца на OSC2;
- два разъема DB9, подключенные через преобразователи уровней к USART0 и USART1 микроконтроллера;
- символьный 4-строчный ЖК-дисплей по 20 символов в строке с голубой ШИМ-регулируемой подсветкой;
- два трехцветных и четыре одноцветных светодиода;
- джойстик с функцией нажатия и три отдельные кнопки;
- элементы с аналоговым выходным сигналом: потенциометр, датчик освещенности и температурный датчик;
- разъем для карт памяти SD/MMC, подключенный на SPI1;
- разъем для подключения внутрисхемного эмулятора.

На плате EVK1100 есть макетное поле (с шагом 2,54 мм). Все выводы микроконтроллера разведены на контактные площадки. Сигналы интерфейсов SPI0, TWI и скоростного отладочного интерфейса Nexus выведены отдельно.

EVK1100 может быть запитана по интерфейсу USB либо от внешнего источника питания напряжением 8–20 В. На плате есть 2 DC/DC-конвертора: для преобразования напряжения питания внешнего источника в напряжение 5 В и преобразования напряжения 5 В в напряжение 3,3 В, а также линейный стабилизатор для получения уровня напряжения 1,8 В. В системе питания предусмотрена схема последовательного включения всех трех преобразователей (power sequencer).

В плату EVK1101 (рис. 2), оценочный набор для микросхем серии AT32UC3B, входят:

- микроконтроллер AT32UC3B0256;
- микросхему памяти DataFlash объемом 8 Мбайт для хранения пользовательских данных, подключенную по интерфейсу SPI0 микроконтроллера;
- разъем mini-USB A-B, подключенный к интерфейсу USB OTG микроконтроллера;
- кварц на 12 МГц на OSC1, часовой кварц на OSC2; предусмотрена возможность установки кварца на OSC2;

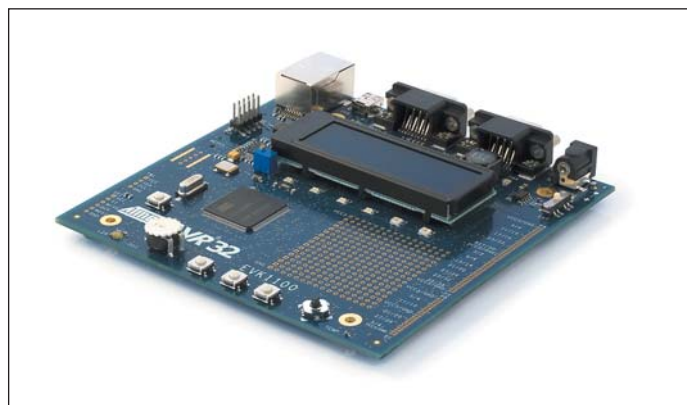


Рис. 1. Оценочный набор EVK1100

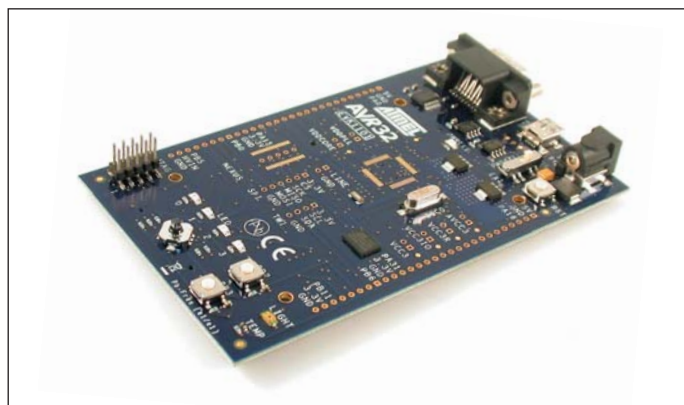


Рис. 2. Оценочный набор EVK1101



Рис. 3. Стартовый набор STK600

- разъем DB9, подключенный через преобразователь уровней к USART1 микроконтроллера;
- четыре одноцветных светодиода;
- джойстик с функцией нажатия и две отдельные кнопки;
- элементы с аналоговым выходным сигналом: датчик освещенности, температурный датчик и интегральный акселерометр на 3 оси;
- разъем для карт памяти SD/MMC, подключенный на SPI0;
- разъем для подключения внутрисхемного эмулятора.

Все выводы микроконтроллера разведены на контактные площадки, а интерфейсы Nexus, SPI0 и TWI микроконтроллера выведены отдельно. В отличие от EVK1100, на плате EVK1101 не предусмотрено макетного поля.

Так же как и EVK1100, плата EVK1101 может быть запитана по интерфейсу USB либо от внешнего источника питания напряжением 8–20 В. Цепи питания организованы похожим образом.

Плата набора EVK1101 лишена дорогостоящей периферии (например, дисплея) с целью минимизации стоимости платы. Пользователь может самостоятельно дополнить устройство такими внешними компонентами. Пример удачной реализации мезонина EVK1101 приведен в одном из выпусков AVR TV [1], где показано, как может выглядеть плата расширения, устанавливаемая в EVK1101. На этой плате есть графический дисплей, где продемонстрированы возможности AT32UC3B0256 по формированию динамической картинки размером 132×132 точек с частотой 25 кадров в секунду.

Помимо перечисленных выше двух оценочных плат недавно был анонсирован стартовый набор STK600 (рис. 3), который работает со всеми микроконтроллерами AVR32 uC3.

STK600 построен аналогично популярной плате STK500 [2] для 8-разрядных AVR — на базовой плате размещаются все управляющие компоненты, а поддержка различных микроконтроллеров обеспечивается мезонинными модулями. Не останавливаясь на общих деталях, опишем отличия, которые появились в STK600:

- работает с любыми 8- и 32-разрядными микроконтроллерами AVR, имеющими Flash-

память (поддержка различных корпусов микросхем осуществляется с помощью мезонинных модулей);

- поддерживает новые интерфейсы программирования (JTAG, PDI);
 - соединяется с компьютером по интерфейсу USB;
 - питается по интерфейсу USB, если подключенная к плате периферия потребляет не более 300 мА.
- Помимо этого на плате STK600:
- усовершенствована схема формирования напряжения питания и опорного напряжения (более высокая стабильность и точность подстройки);
 - усовершенствована система тактирования (диапазон частоты тактового сигнала от 1,1 кГц до 32 МГц);
 - добавлен генератор тактового сигнала частотой 32 кГц;
 - установлено большее количество разъемов расширения, подключенных к портам ввода/вывода микроконтроллера;
 - предусмотрены разъемы интерфейсов CAN, LIN, USB;
 - распаяна микросхема памяти DataFlash объемом 4 Мбит;
 - добавлены цепи защиты от короткого замыкания на линиях ISP, PDI и JTAG (для использования STK600 в качестве программатора микросхем на внешних устройствах).

Отдельно следует отметить, что для минимизации расходов потребителя на дорогостоящие ZIF-панели, мезонинные модули STK600 теперь выпускаются не на определенный перечень микросхем, а для определенного типа корпуса микросхемы, независимо от его архитектуры. Например, мезонинный модуль с ZIF-панелью под корпус TQFP64 будет поддерживать все AVR/AVR32/Xmega, выпускаемые в этом корпусе. Но есть одна деталь: выводы микросхем в одинаковом корпусе, но различных архитектур, расположены по-разному. Поэтому для правильного соединения «ног» микроконтроллера, установленного в ZIF-панель, с элементами на базовой плате STK600, между ними размещается промежуточная плата, которая выполняет правильное соединение конкретной микросхемы с STK600. Благодаря такому решению Atmel поддержка всего ассортимента 8- и 32-разрядных AVR, выпускаемых в одинаковых корпусах, будет осуществляться небольшим набо-

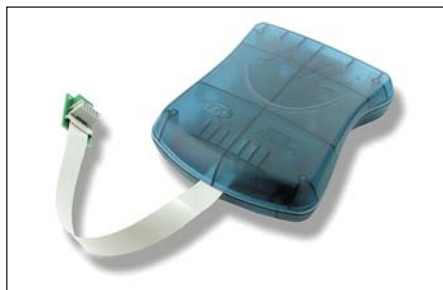


Рис. 4. Внутрисхемный эмулятор JTAGICE2

ром плат с ZIF-панелями и более широким набором переходных плат, стоимость которых сравнительно невелика.

На момент написания статьи STK600 серийно еще не выпущен, поэтому пока сложно сказать, каков базовый состав набора STK600 и какие наименования дополнительных мезонинных плат будут предложены производителем.

Внутрисхемный эмулятор JTAGICE2

Для обеспечения внутрисхемной отладки программы на пользовательской целевой плате или каком-либо из перечисленных выше стартовых и оценочных наборов предлагается изделие JTAGICE2. Это изделие уже знакомо многим разработчикам, работающим с 8-разрядными микроконтроллерами AVR. JTAGICE2 (рис. 4) — это универсальный внутрисхемный эмулятор, работающий с 8-разрядными микроконтроллерами AVR, 32-разрядными микроконтроллерами AVR32 и AVR32 uC3, а также новыми кристаллами AVR Xmega.

Для отладки микроконтроллеров AVR32 uC3 эмулятор JTAGICE2 работает с интегрированной средой разработки AVR32 Studio, о которой будет сказано позже. Эмулятор предоставляет возможность привычной пошаговой отладки с функцией просмотра состояния всех ресурсов процессора в любой момент, когда был выполнен останов выполнения программы в микроконтроллере. Поддерживается останов программы на каждом шаге и по точкам останова.

Эмулятор JTAGICE2 — сложно организованное изделие, которое содержит свой внутренний микрокод и подключается к компьютеру по интерфейсу USB. Это значит, что при подключении к компьютеру потребуются драйвер USB, а для поддержки всех программных новшеств и функций в JTAGICE2 необходимо следить за обновлениями микрокода эмулятора. Несмотря на наличие отдельной интегрированной среды разработки для AVR32 uC3 — AVR32 Studio, драйвер USB и обновления микрокода включаются только в состав AVR Studio, выпускаемой для 8-разрядных AVR. Поэтому перед первым подключением эмулятора к компьютеру необходимо установить последнюю версию AVR Studio для 8-разрядных кристаллов, при установке которого необходимо установить галочку “Install USB driver”. После подключения JTAGICE2 драйвер USB будет автоматически подставлен операционной системой. После этого необходимо обновить микрокод эмулятора в меню “Tools” > “JTAGICE2 Upgrade”. В дальнейшем нужно следить за очередными выпусками AVR Studio, в информации о ревизии к которым сказано, есть ли обновление для JTAGICE2.

Внутрисхемный эмулятор JTAGICE2 сложно назвать дешевым изделием — это плата за универсальность. Для серийного производ-

ва изделий с кристаллами AT32UC3 необходима только функция программирования Flash-памяти. Фирма Atmel пока не предложила фирменного внутрисхемного программатора, но предоставила возможность изготовить собственный — протокол работы JTAG-автомата микросхем AT32UC3A/B с внутренней Flash-памятью кристаллов представлен в руководстве по применению “AVR32708: AVR32 UC3A and UC3B Flash JTAG Programming Algorithms”. Вероятно, в скором времени появятся программаторы третьих фирм.

В микроконтроллерах AT32UC3 помимо интерфейса JTAG предусмотрен высокоскоростной отладочный интерфейс Nexus с расширенными возможностями внутрисхемной отладки: точки останова по условию в данных, запись трассы программы и данных, а также прямой доступ к данным во время выполнения программы [3]. Эмулятор JTAGICE2 не работает с этим интерфейсом. Фирма Atmel планирует выпуск эмулятора с предварительным маркетинговым наименованием “AVR ONE!”, который будет работать с интерфейсом Nexus. К сожалению, Nexus пока поддерживается только эмулятором Vita фирмы Ashling, стоимость которого около \$10 000.

В завершение повествования об аппаратных средствах поддержки разработок скажем несколько слов и том, как изготовить собственный макет изделия на базе AT32UC3. Дизайн оценочных плат EVK1100 и EVK1101 представляет собой пример того, как правильно применить микросхемы AT32UC3. Производитель свободно распространяет схемы этих отладочных плат, библиотеку для схемотехнического САПР Orcad с элементами AT32UC3A и AT32UC3B, а также руководства по применению “AVR32714: AVR32 UC3A Schematic Checklist” и “AVR32715: AVR32 UC3B Schematic Checklist”, в которых описано, на что нужно обратить особое внимание при проектировании собственного изделия.

Программные средства поддержки разработок для AVR32 uC3

Для поддержки разработок проектов на AVR32 uC3 существует 2 типа программных средств: платные и бесплатные. Фирма Atmel активно развивает направление бесплатных программных средств, оптимизируя имеющиеся open-source пакеты для их использования в качестве программных средств для AVR32 uC3. Именно они будут кратко описаны в данной статье.

Состав бесплатных программных средств поддержки разработок включает:

- набор пакетов программ для компиляции и генерации исполняемого кода GNU Toolchain;
- набор библиотек, драйверов и примеров программ uC3 Software Framework;

- интегрированную среду разработки AVR32 Studio.

Набор пакетов программ GNU Toolchain предназначен для компиляции и компоновки исходных текстов программ в объектный код, необходимый для выполнения отладки программы, и в исполняемый код, готовый к использованию в конечном изделии.

GNU Toolchain включает следующие основные пакеты: GnuCompilerCollection (GCC), GnuBinutils и GnuDebugger. На рис. 5 изображена структура взаимодействия пакетов GNU Toolchain в полной цепочке от исходных кодов программ до его отладки и окончательного программирования в конечном изделии.

На рис. 5 видно, что исходный open-source набор GNU Toolchain дополнен утилитами GNU debugger и AVR32 program для взаимодействия с аппаратными средствами поддержки разработок.

Дистрибутив GNU Toolchain распространяется свободно на сайте Atmel [4]. Там предлагаются несколько дистрибутивов, скомпилированных на разные операционные системы: Windows, Linux Ubuntu, Linux SUSE и Linux Fedora. GNU Toolchain также предоставляется в исходных кодах — для желающих модифицировать код дистрибутива, либо скомпилировать его для любой другой ОС.

В среде ОС Windows GNU Toolchain по умолчанию устанавливается в папку C:\Cygwin. Так как проекты GCC и GNU Binutils изначально разработаны для UNIX-систем, в папке C:\Cygwin также размещается система Cygwin, которая эмулирует операционную систему Linux. Здесь есть практически все ее стандартные утилиты. Для открытия терминала Linux необходимо запустить файл C:\cygwin\cygwin.bat — это и есть среда разработки, в которой теперь можно компилировать проекты. Как это делается, можно посмотреть в руководствах по применению “AVR32006: Getting Started with GCC for AVR32” и “AVR32119: Getting Started with AVR32 UC3A Microcontrollers”.

Альтернативной средой разработки, использующей GNU Toolchain, является инте-

грированная среда разработки AVR32 Studio, о которой будет сказано позже.

Особого внимания заслуживает регулярное обновление GNU Toolchain, так как инженеры фирмы Atmel постоянно совершенствуют его, исправляют ошибки и добавляют новые возможности в новые версии пакета. Автору данной статьи не всегда удавалось корректно обновить пакет (установить новую версию поверх старой) — последнюю версию получалось корректно установить только после полного удаления папки C:\Cygwin. Возможно, в будущих дистрибутивах эта задача будет решаться проще.

Следующим в списке бесплатных программных средств, перечисленных выше, идет набор библиотек, драйверов и примеров программ uC3 Software Framework, который включает:

- драйверы для каждого внутреннего периферийного блока микроконтроллеров AT32UC3, а также внешних узлов, размещенных на отладочных платах EVK1100 и EVK1101;
- библиотеку цифровой обработки данных, оптимизированную для архитектуры AVR32 uC3;
- библиотеку для организации и работы с файловой системой FAT16;
- примеры различных приложений;
- код операционной системы реального времени FreeRTOS, портированный на AT32UC3;
- код стека протоколов TCP/IP lwIP, портированный на AT32UC3;
- реализацию стандартных классов USB: CDC, DFU, HID и MSD;
- файлы проектов для GNU GCC и IAR Embedded Workbench;
- подробную документацию, руководства по быстрому старту и учебные пособия на весь программный код.

Весь набор Software Framework предназначен для быстрой разработки проектов на AT32UC3, их запуска на аппаратных средствах поддержки разработок и на готовых изделиях. Все модули в Software Framework представлены в исходных кодах.

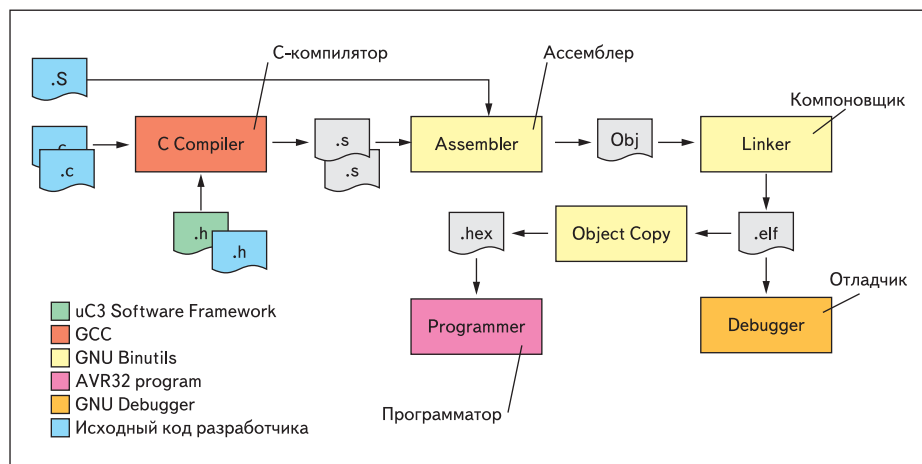


Рис. 5. Организация средств GNU Toolchain

Software Framework насыщен кодом настолько, что для помощи начинающим в его освоении было выпущено отдельное руководство по применению “AVR32008: How to Compile the UC3 Software Framework in AVR32 Studio”. Для заинтересованных в использовании библиотеки цифровой обработки данных есть руководство “AVR32718: AT32UC3 Series Software Framework DSPLib”.

Software Framework распространяется свободно на сайте Atmel [4] двумя архивами — для микроконтроллеров серий AT32UC3A и AT32UC3B. Каждый архив имеет номер версии. На момент написания статьи была выпущена версия 1.1.1 пакета. По словам команды разработчиков центра AVR, в ближайшее время должен быть выпущен очередной релиз, в котором будет усовершенствована интеграция Software Framework в AVR32 Studio.

AVR32 Studio — интегрированная среда разработки, которая предоставляет привычный интерфейс для создания, компиляции и отладки проекта. Для создания проектов в AVR32 Studio есть менеджер проектов, обеспечивающий наглядное изменение настроек проекта, изменения состава его файлов; включен редактор кода программ с развитой системой подсветки синтаксиса. Компиляция и отладка проекта осуществляется вызовами к соответствующим инструментам GNU Toolchain, а в среде AVR32 Studio предусмотрены элементы управления отладкой и средства просмотра ресурсов отлаживаемого кристалла. В файле справки AVR32 Studio размещена подробная документация по всем возможностям этой интегрированной среды разработки.

Создать свой первый проект в AVR32 Studio проще всего с помощью руководства по применению “AVR32015: AVR32 Studio getting started”, а более серьезный проект с использованием Software Framework — с помощью уже упомянутого руководства “AVR32008: How to Compile the UC3 Software Framework in AVR32 Studio”.

Дистрибутив AVR32 Studio распространяется свободно на сайте Atmel [4]. Есть дистрибутив для операционных систем Windows и Linux, но пока нет полноценной поддержки Windows Vista — совместимость будет обеспечена с AVR32 Studio версии 2.0.

Также на сайте Atmel распространяется пакет AVR32 Studio Development Tools, который включает дистрибутив AVR32 Studio и дистрибутив GNU Toolchain.

Прочее программное обеспечение

Отдельно следует отметить, что на стартовых наборах EVK1100 и EVK1101 препрограммирован USB-загрузчик из набора Software Framework, обеспечивающий возможность программирования Flash-памяти микроконтроллера на этих платах по интерфейсу USB без программатора. Основная информация по загрузчику содержится в отдельном документе “AVR32 UC3 USB DFU Bootloader”, ко-

торый дает полную картину организации загрузчика и взаимодействия с ним.

Единственное, о чем не написано в данном документе, — как в AVR32 Studio сконфигурировать проект, чтобы пользовательская программа размещалась в области, куда передается управление загрузчик в случае, если не выполнены аппаратные условия его активации. Следующей задачей является конфигурирование готового проекта таким образом, чтобы компоновщик разместил пользовательскую программу во Flash-памяти, начиная с адреса 80002000h:

1. В меню **File > Import** выбираем **General > File system**, нажимаем **Next**.
2. В строке **From Directory** добавляем путь *[папка, куда распакован Software Framework]* \SERVICES\USB\CLASS\DFU\EXAMPLES\ISP, переходим в левое окно обзора папок.
3. Выделяем файлы **conf_isp.h** и **BOOT\trampoline.S**, нажимаем кнопку **Finish**.
4. Так как платформа Eclipse чувствительна к регистру в именах файлов, необходимо переименовать **trampoline.S** в **trampoline.s**. Для этого в менеджере проекта находим файл **trampoline.S**, выделяем его, щелкаем правой кнопкой мыши и выбираем пункт **Rename**.
5. Открываем свойства проекта (меню **Project > Properties**), переходим в раздел **C/C++ Build**.
6. В zakładке **Tool settings** выбираем **AVR32/GNU Assembler**.
7. Заменяем в поле **command** значение **avr32-as** на **avr32-gcc**.
8. Переходим к пункту **General** внутри **AVR32/GNU Assembler**.
9. В поле **Assembler Flags** добавляем **-x assembler-with-cpp-c**.
10. В разделе **Include Paths** добавляем пути *[путь к Software Framework]*\UTILS и *[путь к Software Framework]*\UTILS\PREPROCESSOR.
11. Переходим в раздел **AVR32/GNU C Linker**, выбираем раздел **miscellaneous**.
12. В строке **Linker Flags** вводим **-Wl,-e,trampoline**, нажимаем **OK**.

Задача выполнена. Теперь можно компилировать проект.

Единственная проблема заключается в том, что реализация загрузчика была включена в Software Framework версии 1.1.0. До его выхода была выпущена партия оценочных наборов EVK1100, которые не препрограммированы этим загрузчиком, а значит, попытки активировать его не увенчаются успехом. В этом случае необходимо обратиться к поставщику набора за информацией по решению возникшей проблемы.

Добавим еще несколько слов о примерах применения, которые доступны на сайте Atmel и могут помочь в разработке. В качестве примера организации обмена данными по интерфейсу USB между микроконтроллером и ПК выпущен пакет “AVR32 UC3B Control Panel PC Demo Installer”, включающий готовый набор программ и библиотек как для нижнего, так

и для верхнего уровня. Для правильной разработки программного обеспечения для различных режимов энергосбережения кристаллов AT32UC3 выпущено руководство по применению “AVR32739: AVR32 UC3 Low power software design”, где приведены примеры кодов для перевода микроконтроллера в режимы энергосбережения таким образом, чтобы обеспечить наилучший результат.

Все перечисленные выше программные средства распространяются свободно и поддерживаются производителем. Если читатель не нашел нужного программного средства, то, возможно, оно производится другими фирмами, распространяющими его за деньги. Среди коммерческого ПО популярным и востребованным программным обеспечением являются:

- компилятор C фирмы IAR Systems;
- среда визуального программирования visualSTATE фирмы IAR Systems;
- операционная система реального времени uC/OS-II фирмы Micrium;
- операционная система реального времени TreadX фирмы Express Logic;
- надстройки на две предыдущие RTOS: стек протоколов TCP/IP, реализация файловой системы FAT, библиотеки для интерфейса USB фирм Micrium и Express Logic;
- графическая библиотека uC/GUI фирмы Micrium.

Более подробную информацию о перечисленных программных продуктах можно найти на сайтах производителей.

Заключение

Данная статья написана с целью упорядочить имеющуюся на сайте фирмы Atmel документацию и программное обеспечение. Так как статья ограничена в объеме, ее содержание указывает на документы, которые могут помочь в разработке проектов на AVR32 uC3. Дополнительную информацию по программно-аппаратному комплексу поддержки разработок для микроконтроллеров AVR32 uC3, помимо сайта www.atmel.com, можно найти на сайтах, приведенных в списке литературы к данной статье. ■

Литература

1. www.avrtv.com
2. Курилин А. И., Золотуха Р. Н. AVR-микроконтроллеры: семь ярких лет становления. Что дальше? Часть 3. Программные и аппаратные средства поддержки разработок для микроконтроллеров AVR // Компоненты и технологии. 2005. № 1.
3. Курилин А. И., Ламберт Е. В. Отладочные средства для микроконтроллеров AVR в 2007 году // Компоненты и технологии. 2007. № 3.
4. www.atmel.com
5. www.avrfreaks.net
6. www.atmel.no/beta_ware
7. Раздел FAQ на сайте support.atmel.no