

Новые возможности в интегрированной среде разработки Keil μ Vision4

Сергей КОПЫТИН
info@microcontroller.ru
Кирилл ДОРОФЕЕВ
kirill@simecs.ru

Интегрированная среда разработки Keil μ Vision IDE объединяет функции управления проектами, создания и редактирования исходного кода, отладки приложений, полной симуляции устройств и Flash-программирования в одну мощную среду разработки. Платформа проста в изучении и использовании и помогает разработчикам встраиваемых систем создавать приложения для микроконтроллеров. Последняя версия μ Vision4 поддерживает более чем 1600 микроконтроллеров на базе процессоров ARM, 166 и 8051 и имеет некоторые новые функции для повышения эффективности программирования и производительности приложения.

Пакет μ Vision объединяет все средства разработчика в одну мощную среду разработки, что повышает производительность приложений для встраиваемых систем (рис. 1).

До 1995 года компиляторы, ассемблеры и компоновщики запускались под DOS из командной строки. Командные файлы должны были вызывать эти программы, а затем обрабатывать исходные файлы при помощи различных контрольных инструкций. В командных файлах использовались элементы, которые не были интуитивно понятны пользователю. Это усложняло процесс разработки приложения и требовало больших временных затрат.

Когда компания Keil Software выпустила интегрированную среду разработки μ Vision IDE для Windows-приложений, стало возможным редактировать исходные файлы, устанавливать контрольные инструкции, компилировать и отлаживать приложения — и все это с помощью нескольких кликов мышки.

В 1998 году Keil Software выпустила интегрированную среду разработки μ Vision2 с множеством новых функций. Во втором релизе μ Vision2 была впервые представлена база данных устройств (**Device Database**), содержащая список всех микроконтроллеров, которые поддерживаются μ Vision, что сделало простым процесс создания кода для конкретного микроконтроллера (рис. 2). Эта база также доступна на сайте компании Keil [4], где благодаря существующей форме поиска можно подобрать контроллер, наилучшим образом подходящий под требования приложения. Пользователю нужно всего лишь выбрать конечное устройство из базы данных, и μ Vision автоматически устанавливает необходимые инструменты и «мастер» кон-

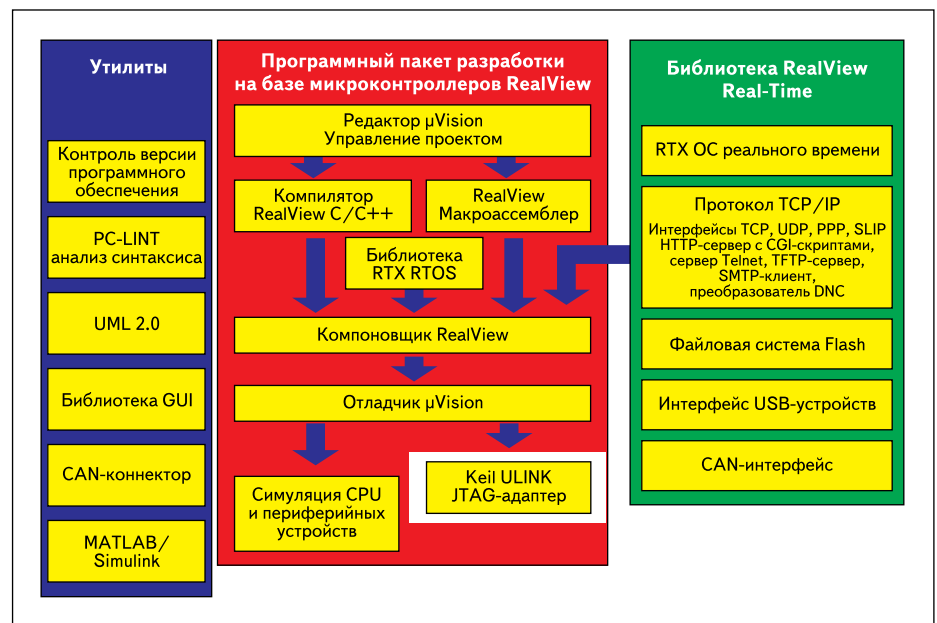


Рис. 1. Обзор решений для разработки от Keil MDK для микроконтроллеров

фигурации. μ Vision позволяет использовать только те инструменты и команды, которые возможно применять к выбранному микроконтроллеру, и предотвращает использование несовместимых друг с другом инструкций. Теперь разработчик может конфигурировать все устройства с помощью нескольких диалоговых окон.

Управление проектами (**Project Management**) стало еще проще: появилась возможность создавать из связанных файлов группы.

Функция **Project Targets** позволяет создавать несколько программ в одном проекте. Например, разработчику необходимо иметь две версии программы, одна из которых использу-

ется в качестве тестовой, а вторая — в качестве рабочей. В каждой из этих программ в рамках одного проекта может быть настроен свой собственный набор инструментов. Диалоговое окно опции проекта (**Project Options**) позволяет установить необходимый набор настроек для всех файлов программы, группы файлов или для отдельного файла. Опции включают в себя: выбор микроконтроллера и адресного пространства, определение файлов, генерируемых ассемблером, компилятором и компоновщиком, а также выбор средств отладки и программирования.

μ Vision отслеживает те места в тексте программы, где определяются перемен-

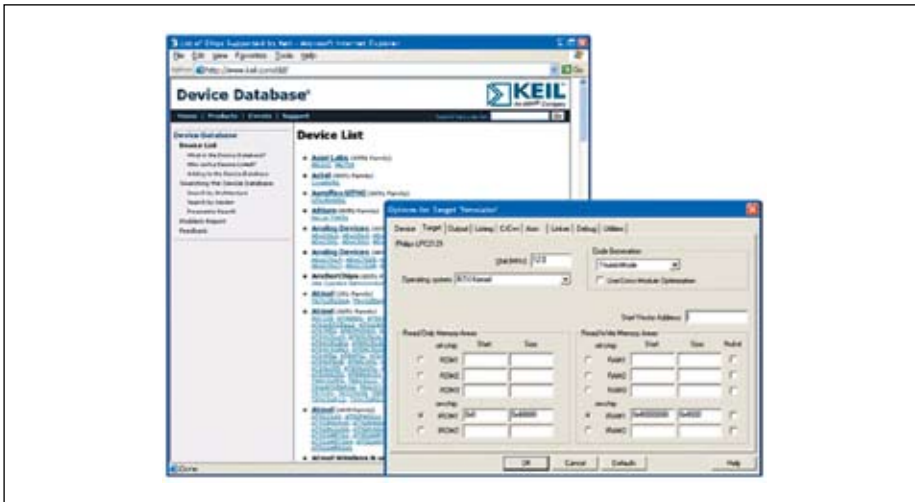


Рис. 2. База данных устройств (Device Database)

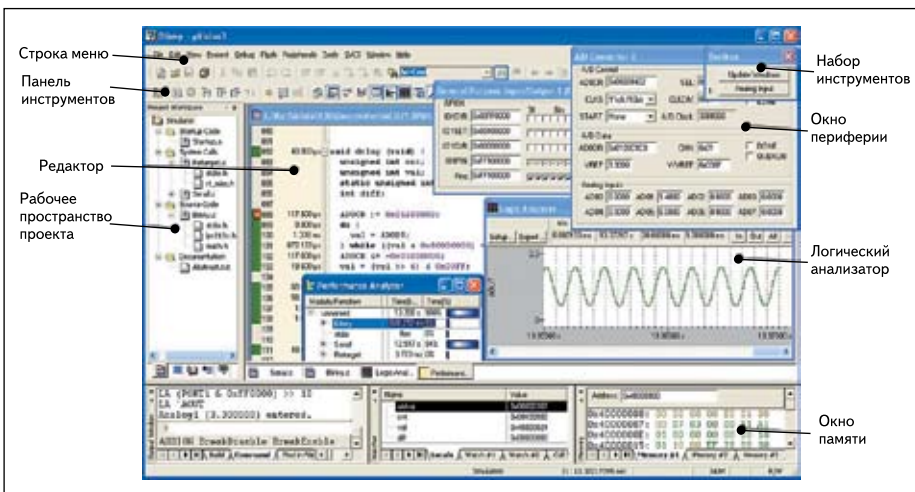


Рис. 3. Keil μVision3

формацию можно использовать для определения того, на какую часть программы затрачивается больше всего времени и какая часть программы нуждается в дальнейшей оптимизации.

В 2003 году Keil выпустила улучшенный μVision3 (рис. 3). К старому программному пакету добавились: выделение ошибок, шаблоны текстов программ и навигация по функциям. Также были введены функции логического анализатора и мастера конфигураций. В окне логического анализатора (**Logic Analyzer**) записываются значения переменных, состояния входов/выходов периферии. Значения этих сигналов могут быть представлены в трех различных форматах: биты, в которых записываются состояния входов/выходов в виде логической единицы или нуля; состояние входов/выходов, которое показывает динамику изменения сигналов; аналоговый сигнал, представляющий график изменения состояния входов/выходов.

«Мастер» конфигураций (**Configuration Wizard**) позволяет конфигурировать файлы инициализации ассемблера, C-компилятора или отладчика, что упрощает настройку специфических опций программы (рис. 4). Эта функция применяется в большинстве шаблонных приложений, встроенных в интегрированную среду разработки μVision. «Мастер» конфигураций прост в использовании: как только разработчик изменяет любой параметр в диалоговом окне, это изменение тут же находит свое отражение в тексте рабочей программы. Таким образом, настройки, сделанные «мастером» конфигураций, могут быть достаточно просто перенесены в другой проект подстановкой нужных значений в диалоговые окна «мастера» конфигураций. Это упрощает повторное использование уже готовых шаблонов в новых проектах и помогает быстро приступить к разработке собственного проекта.

На протяжении многих лет клиенты Keil высказывали свои предложения и замеча-

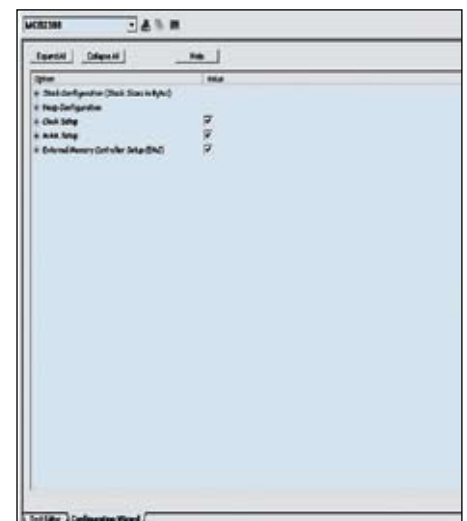


Рис. 4. «Мастер» конфигураций

ные, а также те места, где они используют. Обзорщик исходных файлов (**Source Browser**) собирает и показывает информацию обо всех символах в программе. Кроме того, в окне **Source Browser** можно увидеть всю статистику по используемым в коде программы переменным, а также быстро определить количество использований того или иного символа в исходном коде.

В отладчике окно **Watch Window** позволяет отслеживать текущие значения переменных, а окно **Memory Window** отражает различные области памяти. Окно **Serial Window** позволяет использовать симулятор UART с несколькими окнами для устройств с последовательными интерфейсами.

Встроенный симулятор поддерживает полную симуляцию работы устройства, включая инструкции центрального процессора, системы памяти и периферии микроконтроллера. После того как разработчик выбирает устройство из базы данных **Device Database**, μVision автоматически конфигурирует систему для работы с выбранным микроконтроллером. При помощи логической и временной

симуляции можно проверить приложение до его запуска на реальном оборудовании. Симулятор также позволяет опробовать различные аварийные ситуации, которые сложно отладить непосредственно на оборудовании.

В окне анализатора эффективности кода (**Code Coverage**) можно увидеть процент запущенных в приложении инструкций. Анализатор эффективности кода (**Code Coverage**) помогает осуществить контроль тестирования программы, разделив ее части на запущенные и незапущенные в ходе выполнения проверки. Таким образом, меняя стратегию тестирования, разработчик может убедиться в том, что все части кода были проверены.

Анализатор производительности (**Performance Analyzer**) дает возможность увидеть, сколько времени заняло выполнение той или иной функции или блока программы. Результаты такой проверки представляются в виде графиков, на которых можно оценить количество вызовов функции, время выполнения функции и процент от общего времени выполнения программы. Эту ин-

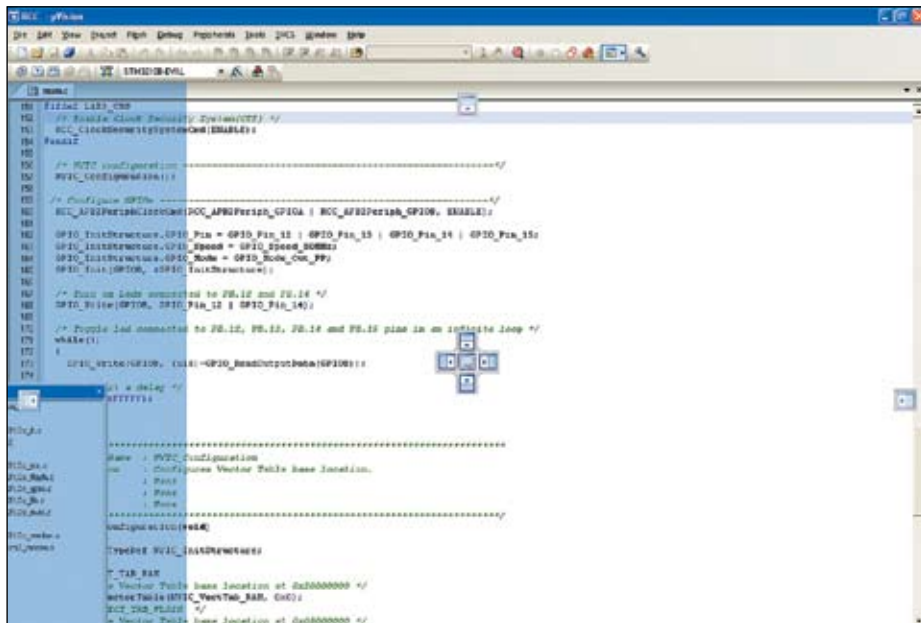


Рис. 5. Keil µVision4. Гибкая система управления окнами

ния относительно того, что они хотели бы улучшить в инструментальных средствах Keil и какие новые функции необходимо добавить. Такая обратная связь позволила внедрить в систему множество инноваций, которые присутствуют в инструментальных средствах до сих пор. Однако стало понятно, что для того, чтобы включить в программный пакет некоторые из этих идей, µVision необходимо серьезно переделать. Кроме того, в процессе изменения дизайна команда разработчиков инструментальных средств сосредоточилась на упрощении интерфейса и внедрении интуитивно понятных функций, которые давно ждали пользователи пакета µVision.

Взяв за основу идеи и предложения конечных пользователей, Keil разработала µVision4 — программный пакет с новыми функциями и возможностями, среди них: гибкая система управления окнами (**Flexible Window Management System**), система просмотра (**System Viewer**) и многопроектное рабочее пространство (**Multi-project Workspace**).

В предыдущих версиях µVision конфигурация окон в рабочем пространстве была фиксирована, то есть окна имели определенный размер. В µVision4 представлена гибкая система управления окнами (**Flexible Window Management System**), которая позволяет разработчикам полностью контролировать положение и размер всех окон проекта (рис. 5).

С помощью нового пользовательского интерфейса можно лучше организовать рабочее пространство, он более удобен при большом количестве одновременно открытых окон. Таким образом, среда представляет собой мощный инструмент для разработки приложений без лишних деталей. Гибкая система управления окнами также позволя-

ет открыть одновременно несколько окон одного типа. Например, теперь можно открыть окна состояния памяти и просматривать в каждом из них различные диапазоны памяти или значения в памяти в различных представлениях: в шестнадцатеричном формате, в формате с плавающей запятой и т. д. Или же открыть два различных фрагмента кода в окнах редактора. Окна могут быть размещены в любом месте, даже вне окна µVision, например как отдельное окно операционной системы. Каждый раз при перемещении окон специальный помощник указывает все возможные варианты перемещения того или иного окна.

Большинство интегрированных систем разработки могут отслеживать значения регистров периферийных устройств. Система просмотра µVision4 (**System Viewer**) отображает те регистры, в которые процессор может записать какое-либо значение, для того чтобы разработчик имел возможность контролировать функции работы с периферией. Система просмотра позволяет программисту использовать расширенные возможности управления просмотром этих значений: теперь в окно просмотра можно включать не только регистры с определенным именем (например, ADout), но и содержание самого регистра вместо его значения или состояния (например, «вызов прерывания» вместо «1»). Благодаря этому стало значительно проще определять, в каком состоянии находится то или иное периферийное устройство.

В µVision4 разработчик имеет возможность сохранять настройки, наиболее подходящие для текущего тестирования (функция **Debug Restore Views**). Например, когда тестируется часть кода, описывающая работу UART, в окне системы просмотра можно наблюдать состояние всех регистров UART, а логический

анализатор автоматически записывает сигналы UART. Если же происходит отладка CAN-интерфейса, в окне логического анализатора можно просматривать только сигналы, относящиеся к CAN, без отображения сигналов UART. Эти настройки могут быть сохранены как для текущего проекта, так и для всех проектов вообще, что позволяет использовать одни и те же настройки при отладке тех же интерфейсов в других проектах.

При разработке приложения часто возникает необходимость работать с несколькими проектами одновременно, например, когда приложение состоит из нескольких частей: загрузчика и перегружаемых частей программ. В µVision4 появилась возможность открывать несколько различных проектов внутри одного рабочего пространства. В окне проектов (**Project Window**) показаны все проекты, открытые в данном рабочем пространстве, а активный проект выделен. Большинство операций, вызываемых из меню, будет относиться к активному проекту, а операция **Batch Build** скомпилирует все открытые проекты.

На данный момент версия Keil µVision3 ежедневно используется тысячами разработчиков встраиваемых приложений по всему миру для создания программ для микроконтроллеров 8051, 166 или ARM. Интегрированная среда разработки µVision4 спроектирована для того, чтобы повысить производительность труда. Благодаря гибкой системе управления окнами разработчики теперь могут создать под свои задачи визуальное пространство оболочки, а возможность работать с несколькими проектами одновременно помогает при разработке сложных приложений. Система просмотра позволяет в любой момент отслеживать состояние регистров периферии, а функция **Debug Restore Views** делает более удобной анализ работы программы.

Бета-версию µVision4 можно свободно скачать по ссылке <https://www.keil.com/demo/uvision4.asp>. В одном файле вместе с бета-версией находится и руководство по ее использованию. Установка µVision4 осуществляется как add-on на уже установленный программный пакет с µVision3. При этом функциональность оригинального ПО никоим образом не меняется. После инсталляции бета-версии для управления проектом можно использовать как µVision4, так и µVision3. ■

Литература

1. C51 Product Manuals — <http://www.keil.com/c51/man/c51.htm>
2. C166 Product Manuals — <http://www.keil.com/c166/man/c166.htm>
3. ARM Product Manuals — <http://www.keil.com/arm/man/arm.htm>
4. www.keil.com
5. www.arm.com
6. www.microcontroller.ru