

НОВЕЙШИЕ ПРОЦЕССОРЫ SYNOPSYS ДЛЯ ВСТРАИВАЕМЫХ ПРИЛОЖЕНИЙ – БОЛЕЕ ВЫСОКАЯ ПРОИЗВОДИТЕЛЬНОСТЬ И ГИБКОСТЬ

Т.Халфхилл

Разработчикам кристаллов хорошо известны лицензируемые IP-блоки компании Synopsys: интерфейсы, аналоговые блоки, встраиваемая память, логика. Кроме того, Synopsys разрабатывает собственное семейство процессоров, которые носят название ARC. Далеко не все знают, что ежегодно выпускаются более 1,5 млрд. кристаллов, в состав которых входят процессорные ядра ARC, то есть по числу поставляемых по всему миру процессорных ядер Synopsys уступает лишь ARM. Всего на сегодняшний день компания Synopsys заключила более 190 лицензионных соглашений на использование своих процессорных ядер, в том числе, с такими компаниями, как Aptina, Broadcom, Fujitsu, Infineon, SanDisk, SiliconMotion и CSR/Zoran. В октябре 2014 года компания Synopsys представила новую серию лицензируемых процессорных ядер для встраиваемых приложений на базе Linux – ARC HS38. Обсудим особенности новых процессоров и сравним их с младшими членами семейства высокоскоростных процессоров ARC HS – ARC HS34 и ARC HS36.

ОБЗОР СЕМЕЙСТВА ARC HS

Процессоры семейства ARC HS так же, как и процессоры ARC EM, основаны на 32-разрядной системе команд ARCv2, но обеспечивают более высокую производительность, не ухудшая при этом энергоэффективность кристалла и не увеличивая его площадь. Предназначенные для более производительных встраиваемых приложений ядра ARC HS программно совместимы с ядрами семейства ARC EM.

Первыми членами семейства ARC HS стали две серии процессорных ядер – ARC HS34 и ARC HS36 (табл.1). Процессоры ARC HS34, которые для хранения команд и данных используют тесно связанную память (Closely

Coupled Memory – CCM), предназначены в первую очередь для систем управления в режиме реального времени, например, в таких приложениях, как твердотельные накопители (SSD), сетевые устройства хранения данных (NAS), домашние сети и шлюзы, а также мобильные устройства. Процессоры ARC HS36, которые могут хранить команды и данные либо в CCM, либо в кэш-памяти, предназначены главным образом для более производительных встраиваемых приложений, в том числе в цифровых камерах, цифровом ТВ, телеприставках, автомобильных системах управления и Интернете вещей.

Процессорное ядро ARC HS38 ориентировано на встраиваемые приложения, работающие под

управлением операционных систем с поддержкой виртуальной памяти, таких как Linux. Разработчики кристаллов могут создавать одно-, двух- или четырехядерные конфигурации с блоком управления памятью (memory-management unit – MMU) и быстрой кэш-памятью второго уровня (L2-кэшем). Моделирование показало, что ядро ARC HS38 обеспечивает частоту тактового сигнала в наихудших условиях на уровне 1,6 ГГц при использовании 28-нм КМОП-технологии, например, процесса NPM компании TSMC с использованием подзатворного диэлектрика с высокой диэлектрической постоянной (high-k metal-gate – НКМГ). Этот технологический процесс использует большинство перспективных заказчиков, с которыми сотрудничает компания Synopsys. Следует сказать, что сегодня лишь в немногих встраиваемых системах требуются рабочие частоты на уровне 1,6 ГГц – тем самым обеспечен достаточный запас для будущих разработок.

Производительность одного ядра процессора ARC HS38 на частоте 1,6 ГГц составляет 3100 DMIPS (Dhrystone MIPS) или 5600 CoreMark (3,5 CoreMark/МГц). Таким образом, агрегированная производительность четырех-ядерного процессора на базе ARC HS38 достигает 12400 MIPS

или 22400 CoreMarks. При минимальной конфигурации ядра ARC HS38 потребление энергии составляет всего 0,036 мВт/МГц (58 мВт при частоте 1,6 ГГц). При этом, площадь кремния, которую занимает ядро, равна 0,2 мм², а плотность кода – на 18% выше, чем у процессоров на базе системы команд ARCV1.

ОСОБЕННОСТИ АРХИТЕКТУРЫ ПРОЦЕССОРОВ ARC HS

Поскольку все ядра ARC HS базируются на той же системе команд ARCV2, что и ядра ARC EM, их фундаментальные свойства схожи. Эта 32-разрядная RISC-архитектура использует как классические 32-разрядные команды, так и 16-разрядные команды для повышения плотности кода во встраиваемых приложениях с малым объемом памяти.

Отличительной особенностью всех процессорных ядер ARC является их широкие возможности конфигурирования и расширения системы команд. Модифицируя свойства и добавляя новые команды, регистры и другие опции, разработчики кристаллов могут подстраивать эти ядра под специфические задачи. Например, пользователь может создать массив

Таблица 1. Семейство процессоров ARC HS

	ARC HS38	ARC HS36	ARC HS34
Версии процессора	Одно-, двух-, четырехядерный		
Частота процессора (макс.)	1,6 ГГц (мин); 2,2 ГГц (тип.)		
Блок управления памятью	Стандартно	–	–
Адресация физической памяти	Конфигурируемая, 32-40-разрядная	32-разрядная	32-разрядная
Блок защиты памяти	–	Опционально	Опционально
Когерентный кэш для симметричной мультипроцессорной обработки	Опционально	Опционально	–
Управление питанием, масштабирование напряжения питания / частоты	Опционально	Опционально	Опционально
L1-кэш	0–64 Кбайт	0–64 Кбайт	–
L1-ССМ	0–16 Мбайт	0–16 Мбайт	0–16 Мбайт
L2-кэш (совместно используемый)	0–8 Мбайт	0–8 Мбайт	–
Модуль обработки операций с плавающей запятой	Опционально 32/64 разряда	Опционально 32/64 разряда	Опционально 32/64 разряда
Малый трассировочный модуль SmaRT, отладка в режиме реального времени	Опционально	Опционально	Опционально

из 32-разрядных регистров на триггерах или ячейках памяти. С помощью регистров расширения (extension core registers) процессоры ARC HS способны поддерживать до 60 регистров ядра.

Разработчики могут добавлять таймеры, прерывания, умножители, аппаратные делители, а также конфигурировать шину памяти для адресации с прямым и обратным порядком битов. Возможность конфигурирования позволяет разработчикам изменять определенные свойства ядра с целью оптимизации его характеристик, размеров и энергопотребления под конкретное приложение. В новейшей серии процессоров ARC HS38 реализованы еще более широкие возможности конфигурирования (рис.1).

Ядра ARC HS поддерживают расширение системы команд. Например, разработчики могут добавить отдельно лицензируемый модуль обработки операций с плавающей запятой, который соответствует требованиям стандарта IEEE-754.2008 и поддерживает данные с одинарной и двойной точностью. Большинство операций являются полностью конвейеризированными, что позволяет процессору выполнять команды с плавающей запятой последовательно в каждом тактовом цикле. Команды деления и извлечения квадратного корня выполняются за несколько тактов, но при этом они не блокируют выполнение других команд.

Пользователи могут вводить собственные команды, которые процессор дешифрирует и выполняет точно так же, как команды стандартного набора. Новые команды, добавленные с помощью инструмента конфигурирования ARChitect, становятся частью совместной среды разработки, которая включает в себя командный симулятор nSIM (со средней производительностью 475 MIPS в турбо-режиме), а также инструмент xCAM, который обеспечивает 100-процентное цикловое моделирование. Для разработки ПО компания Synopsys предлагает оптимизированную версию набора инструментов ARC MetaWare, который включает в себя компилятор C/C++, ассемблер и отладчик. В качестве альтернативы программисты могут использовать инструментальные средства с открытым исходным кодом GNU-C.

ВЫСОКОПРОИЗВОДИТЕЛЬНЫЙ КОНВЕЙЕР СЕМЕЙСТВА ARC HS

Несмотря на принципиальную схожесть с другими процессорами на основе ISA ARCV2, ядра ARC HS обладают некоторыми новыми свойствами, которые значительно повышают быстродействие. В первую очередь, речь идет о высокопроизводительной микроархитектуре на базе полностью обновленного 10-ступенчатого конвейера команд (рис.2). Он позволяет достичь

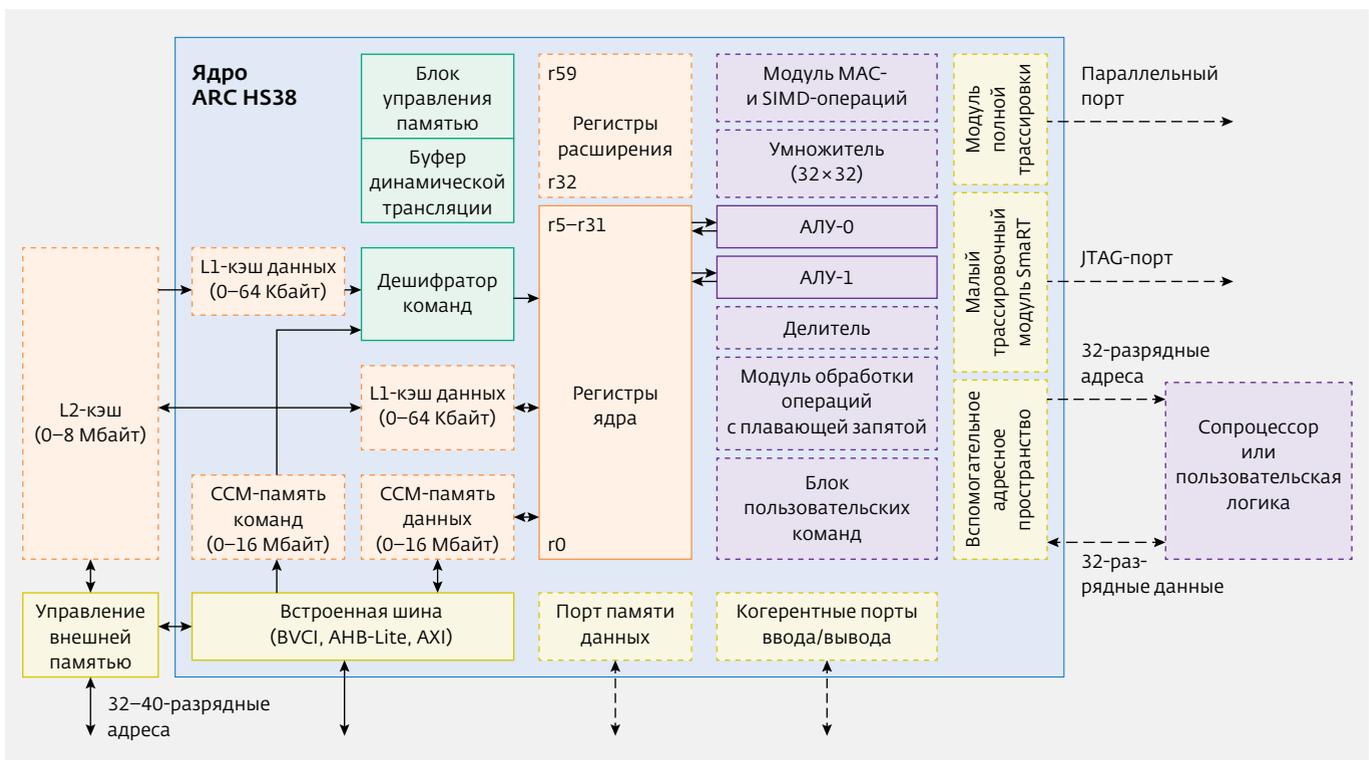


Рис.1. Блок-схема ядра ARC HS38 (пунктирными линиями обозначены опциональные или конфигурируемые пользователем функции)

намного более высоких рабочих частот при любом используемом технологическом процессе, чем простой трехступенчатый конвейер процессоров семейства ARC EM.

Разработанный компанией Synopsys конвейер позволяет достичь наивысшей производительности, сохраняя при этом простоту и энергоэффективность исполнения скалярных операций. Он поддерживает динамическое предсказание переходов, раннее детектирование неверно предсказанных переходов, обеспечивает отсутствие внутренних конфликтов при выполнении простых целочисленных команд и произвольный переход к загрузке, делению, выполнению операций с плавающей запятой и пользовательским командам. Максимальное количество ожидаемых обработок команд конфигурируется пользователем и может достигать восьми.

Предусмотрены команды, которые способны загружать и хранить 64-разрядные слова, исключая тем самым необходимость исполнения последовательных 32-разрядных команд загрузки и хранения данных. Фактически компилятор ARC MetaWare автоматически использует команды загрузки/хранения слов двойной длины, когда обнаруживает последовательные 32-разрядные операции. Такие команды, в частности, полезны для быстрого копирования регистров в стек или из стека при переключении контекста. Имеется еще одна команда, которая сохраняет или восстанавливает состояние процессора, – эффективный вызов подпрограммы, который использует загрузку и хранение слов двойной длины. Для еще более быстрого переключения контекстов пользователи

могут вводить второй массив регистров, благодаря чему исключается необходимость каждый раз сохранять и восстанавливать регистры.

Процессоры ARC HS также поддерживают невыровненные команды загрузки и сохранения с нарушением границ памяти и строк кэша. Более того, при этом не расходуются дополнительные тактовые циклы – невыровненные данные обрабатываются так же быстро, как и выровненные.

После выравнивания команд на 3-й ступени конвейер передает их дешифратору 4-й ступени (см. рис.2). На 5-й ступени процессор выбирает операнды данных из массива регистров. Специальная обходная логика этой ступени может передать операнд из любой предыдущей ступени, если предварительно было вычислено его предполагаемое значение. Если входные операнды не доступны, обходная логика может задержать конвейер до появления требуемых данных. (Процессоры ARC HS могут выдерживать до четырех непопаданий в кэш данных без задержки конвейера.)

Команды исполняются на ступенях 6, 7, 8 и 9. Если операнды одноцикловой операции в АЛУ становятся доступны на 5-й ступени (либо из массива регистров, либо из обходной логики), то "раннее АЛУ" (ALU-0) будет вычислять результат на 6-й ступени. Если один или более операндов еще не доступны на 5-й ступени и операция может быть отложена для "позднего АЛУ" (ALU-1) на 9-й ступени, то операция передается на эту ступень без вычислений. Команды получают любые нужные им операнды по мере их передвижения к "позднему АЛУ" через дополнительные обходные

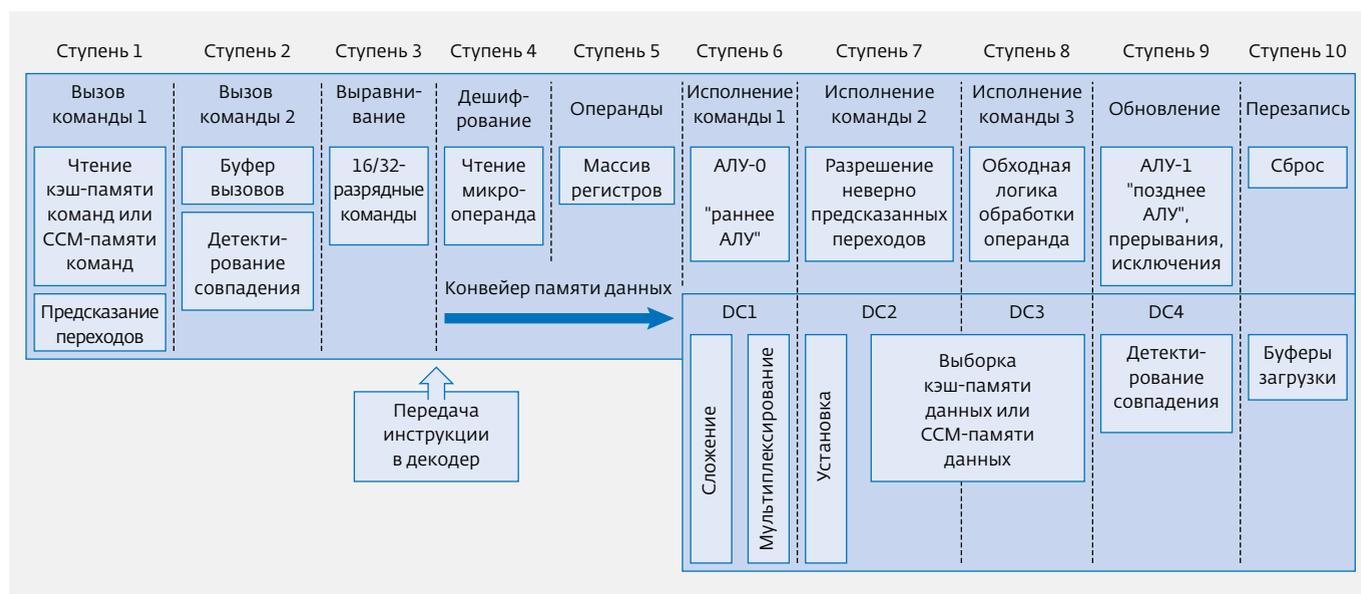


Рис.2. 10-ступенчатый конвейер команд семейства процессоров ARC HS

цепи на ступенях 6, 7 и 8. Команды загрузки/хранения выполняются в собственном 4-ступенчатом конвейере блока загрузки/хранения.

Особенностью конвейера процессора ARC HS является то, что в нем предусмотрено разрешение неверно предсказанных переходов на 7-й ступени и немедленный сброс конвейера без прохождения трех конечных ступеней. Такое раннее выявление неверного предсказания переходов позволяет экономить тактовые циклы.

На 9-й ступени ("позднее АЛУ") могут исполняться простые целочисленные команды и приниматься решения о переходах в зависимости от загруженных данных. Таким образом, большинство команд не требуют предварительной загрузки данных, поскольку данные, вновь появившиеся после 4-й ступени (DC4) конвейера загрузки/хранения, передаются прямо в соответствующие команды на 8-ю ступень или на более ранние стадии. Так как большинство целочисленных инструкций подпадает под эти условия, процессоры ARC HS могут в большинстве случаев "скрывать" задержку доступа к памяти данных и, таким образом, тратить меньше времени на ожидание данных перед выполнением операций в АЛУ.

Если целочисленная команда достигает "раннего АЛУ" на 6-й ступени, в то время как предыдущая команда еще выполняется в "позднем АЛУ" на 9-й ступени, то обе эти команды могут выполняться параллельно. Несмотря на такую возможность, ARC HS не является истинно супер-скалярным процессором, поскольку на каждом такте в скалярный конвейер передается только одна команда.

Кроме того, заметим, что в конвейере загрузки/хранения две из четырех ступеней отведено под выборку данных из кэш-памяти или ССМ. Разделение этих операций с памятью на две ступени смягчает временные ограничения, позволяя использовать однопортовое СОЗУ и обеспечивая возможность достижения более высоких тактовых частот. Дополнительное повышение производительности достигается за счет того, что кэш или ССМ имеют 64-разрядный интерфейс и разбиты на несколько банков. Пользователь может конфигурировать степень ассоциативности кэша (два или четыре канала), а кэш команд использует программу-предсказатель перехода для уменьшения расхода энергии на вызовы излишних строк кэша.

На 10-й ступени команды завершают свою работу и записывают результаты в массив регистров или кэш данных. Загрузка, деление и пользовательские команды могут заканчивать работу в произвольном порядке, соблюдая, при этом, условия исключений. Буфер завершения (graduation buffer) хранит до восьми команд, позволяя процессору выполнять в фоновом

режиме обработку команд загрузки и сложные арифметические операции.

ДОПОЛНИТЕЛЬНЫЕ ВОЗМОЖНОСТИ, УЛУЧШАЮЩИЕ ПРОИЗВОДИТЕЛЬНОСТЬ ПРОЦЕССОРОВ ARC HS

Кроме высокоскоростного 10-ступенчатого конвейера процессор ARC HS обладает и другими возможностями, повышающими производительность. Подобно ядрам ARC EM у ядер ARC HS имеется усовершенствованный

Таблица 2. Сравнение особенностей архитектуры процессоров ARC EM и ARC HS

	Семейство ARC HS	Семейство ARC EM
MAC-команды	Опционально	-
64-разрядный MAC - аккумулятор	Опционально	-
Векторные команды	Опционально	-
Загрузка/хранение слов двойной длины	Опционально	-
Невыровненные операции загрузки/хранения	Опционально	-
Команда с плавающей запятой	Опционально	-
Команда XBFU	Извлечение беззнакового битового поля	-
Отладка состояния внешней остановки	Стандартно	-
Опция повышения плотности кода	Стандартно (разрешены 16-разрядные команды)	Опционально
Опция сканирования битов	Стандартно (разрешены команды с побитовой обработкой)	Опционально
Опция загрузки и сдвига	Стандартно (разрешены 8/16-разрядные команды загрузки и сдвига)	Опционально

делитель по основанию 4, который способен выполнять операции за 4–19 тактовых циклов в зависимости от разрядности данных. Предусмотрен ряд опциональных команд, которые поддерживают цифровую обработку сигналов и SIMD-операции. Компилятор ARC MetaWare автоматически генерирует такие команды, когда обнаруживает эти операции в исходном коде C/C++. Имеются команды, которые могут выполнять несколько вариантов 64-разрядных операций умножения с накоплением (MAC), векторное сложение и векторное вычитание. Векторные операнды могут иметь разрядность 16, 32 или 64.

Еще одной опцией является модуль обработки операций с плавающей запятой (FPU), который аппаратным способом выполняет 32- и/или 64-разрядные операции с плавающей запятой и полностью соответствует спецификации IEEE-754.2008. Кроме того, в отличие от предыдущей версии, в этом модуле реализовано аппаратное извлечение квадратного корня. Хотя оба семейства – ARC EM и ARC HS – базируются на одной и той же системе команд ARCV2, новое семейство имеет ряд дополнительных возможностей, а некоторые опции процессоров ARC EM стали стандартными свойствами ARC HS (табл.2).

Используемые в процессорах семейства ARC HS стандартные шины AXI и AHB-Lite имеют конфигурируемые

32- или 64-разрядные интерфейсы. Шина AXI способна обрабатывать до 11 ожидающих выполнения транзакций памяти, включая одно непопадание в кэш команд, четыре некэшируемые загрузки, четыре некэшируемых сохранения и два повторных заполнения кэша данных. Тактовые частоты шины конфигурируются пользователем и кратны тактовой частоте ядра. Самая высокая тактовая частота соответствует кратности 1:4. Эти частоты можно программно изменять в процессе работы. Задержки шинных транзакций можно уменьшить по сравнению с более старым ядром ARC 700 с помощью организации моста между BVIC-интерфейсом ядра и шинами AXI или AHB-Lite.

В качестве альтернативы использования шин AXI или AHB-Lite для подключения устройств разработчики могут ввести опциональную 32-разрядную шину, которая обеспечивает доступ к ядру процессора за один тактовый цикл. Еще одна конфигурируемая опция – блок защиты памяти (лицензируемый отдельно), который предохраняет критические области памяти от случайного и неавторизованного доступа.

В ядрах семейства ARC HS реализованы те же усовершенствованные функции прерываний, которые впервые появились в семействе ARC EM. Эти процессоры поддерживают до 240 конфигурируемых

пользователем прерываний с 16 уровнями приоритета. Упростить программу обработки прерываний позволяет возможность автоматического сохранения и восстановления содержимого регистров в стеке при входе в эту программу и при выходе из нее. Программисты могут устанавливать, какие регистры подлежат сохранению. Как уже отмечалось выше, ускоренное переключение контекстов возможно с помощью введения второго массива регистров при конфигурировании ядра. В этом случае процессор может переключаться между разными контекстами без сохранения и восстановления регистров – он просто переходит на другой массив регистров.

БОЛЕЕ ПРОСТАЯ РАЗРАБОТКА И ВЕРИФИКАЦИЯ

Пакет разработчика MetaWare представляет собой полный интегрированный набор средств разработки для поддержки семейства ARC HS и других процессоров ARC. Он включает в себя оптимизированный компилятор C/C++, который обеспечивает высокую плотность кода, отладчик и симулятор nSIM. Инструменты MetaWare поддерживают все опции семейства ARC HS, включая трассировку и отладку в режиме реального времени.

Семейство HS – первые процессоры ARC, которые обеспечивают в процессе отладки полную трассировку снаружи кристалла в режиме реального времени (RTT). Для поддержки этой функции на кристалле требуется наличие параллельного трассировочного порта, поскольку последовательный JTAG-порт не имеет достаточной пропускной способности для передачи большого объема данных. Пользователи могут выбрать, сохранять ли данные трассировки в системной памяти или в памяти тестового устройства.

Пользователи могут также настроить трассировку таким образом, чтобы отслеживать данные в определенных границах и от различных источников, например, от программного счетчика, регистров ядра, интерфейсов памяти и вспомогательного интерфейса. Управлять параметрами трассировки помогают также программируемые триггеры и фильтры. Трассировка в режиме реального времени реализована в одноядерных или многоядерных (до 4-х ядер) системах и соответствует стандарту Nexus 5001. Процесс трассировки не влияет на работу процессора.

В качестве альтернативы, разработчики кристаллов, которым не нужны расширенные возможности отладки, могут использовать более компактный модуль трассировки, который называется малым модулем трассировки в режиме реального времени SmART (Small Real-Time Trace). Он использует стандартный двухпроводной порт JTAG (типа JTAG или JTAG7) и позволяет пользователям устанавливать переменное число точек обработки (от 2 до 64). Возможности JTAG-порта в семействе ARC HS те же, что и у семейства ARC EM.

Кроме того, для обоих семейств процессоров общими являются два симулятора Synopsys. Более быстрый из них – симулятор набора команд nSIM, потактовая точность которого превышает 95%. В типовой системе разработки он моделирует процессор ARC HS с частотой около 30 МГц. Отладчик ARC MetaWare полностью поддерживает этот симулятор, который также совместим с виртуальными платформами Virtualizer от Synopsys и OSCI SystemC. Если для профилирования, оптимизации или верификации требуется потактовая точность не ниже 100%, разработчики могут использовать симулятор Synopsys xCAM.

* * *

Во второй части статьи речь пойдет о технологии симметричной мультипроцессорной обработки, опциональном L2-кэше, блоке управления памятью, методах энергосбережения и других особенностях новых процессоров семейства ARCHS. Кроме того, мы сравним процессоры ARC HS38 с его основными конкурентами. ●

