УДК 621.382.2/.3 Дата подачи статьи: 03.05.17
DOI: 10.15827/0236-235X.119.345-352 2017. Т. 30. № 3. С. 345–352

ВЫСОКОПРОИЗВОДИТЕЛЬНЫЙ МИКРОПРОЦЕССОР 1890ВМ118 С АРХИТЕКТУРОЙ КОМДИВ ДЛЯ СОЗДАНИЯ ДОВЕРЕННЫХ СИСТЕМ

С.И. Аряшев, к.т.н., зав. отделением, aserg@cs.niisi.ras.ru;
С.Г. Бобков, директор, bobkov@cs.niisi.ras.ru;
П.С. Зубковский, зав. отделом, zubkovsky@niisi.ras.ru;
С.А. Морев, младший научный сотрудник, morev@cs.niisi.ras.ru;
Б.Ю. Рогаткин, научный сотрудник, boris240@cs.niisi.ras.ru
(Федеральный научный центр Научно-исследовательский институт системных исследований РАН (ФНЦ НИИСИ РАН), Нахимовский просп., 36, корп. 1, г. Москва, 117218, Россия)

В статье рассматриваются особенности разработки высокопроизводительного микропроцессора для создания доверенных систем. Производительность микропроцессора определяется производительностью ядра или количеством одновременно выполняющихся операций и временем доступа к памяти. Возможность использования микропроцессора для создания доверенных систем основывается на использовании в его составе блоков и узлов собственной разработки.

Производительность ядра микропроцессора определяется в основном тремя характеристиками: тактовой частотой микропроцессора, частотой следования инструкций и количеством операций, выполняемых одной инструкцией. Для микропроцессора 1890ВМ118 эти характеристики в большинстве случаев оптимизируются по параметру соотношения производительности и потребляемой мощности. Повышение тактовой частоты достигается за счет использования заказного проектирования критичных для быстродействия блоков и оптимизации длины конвейера. Частота следования инструкций увеличивается путем использования таких аппаратных решений, как суперскалярное исполнение инструкций, предсказание переходов и предварительная подкачка данных в кэш-памяти. Реализация арифметического сопроцессора, ориентированного на задачи цифровой обработки сигналов, позволяет повысить число операций, выполняемых одной командой. Повышение производительности подсистемы памяти рассматривается в статье с точки зрения симметричного доступа к памяти для двухъядерного микропроцессора. Описан подход к реализации когерентности в кэш-памяти процессорных ядер.

Особое внимание уделено средствам повышения безопасности в микропроцессоре, которые предназначены для создания на его основе доверенных систем. Рассмотрены аппаратные решения для доверенной загрузки операционной системы и изолированного доступа к памяти. Для обеспечения доверенной загрузки предлагается использовать накристальное постоянное запоминающее устройство и однократно программируемую память, содержащие безопасный начальный загрузчик, а также ключи для проверки подписей операционной системы. К средствам изолированного доступа относится рассмотренный в статье контроллер доступа к памяти, реализованный в микропроцессоре. Предложены перспективные направления повышения безопасности систем на кристалле для создания доверенных систем на основе микропроцессоров разработки ФНЦ НИИСИ РАН.

Ключевые слова: *доверенные системы, система на кристалле, архитектура микропроцессора, сопроцессор.*

Одной из основных проблем функционирования современных высокопроизводительных вычислительных систем является защита от несанкционированного доступа. Несанкционированный доступ к вычислительной системе возможен из-за наличия в микропроцессоре неописанных аппаратных средств – закладок, которые могут быть заложены как при проектировании, так и в процессе производства. К несанкционированному доступу могут привести отсутствие защиты программы загрузчика, а также различного рода атаки на ПО, не контролируемые аппаратурой. Растущие требования к увеличению производительности ведут к усложнению микропроцессоров, а в результате к многократному усложнению поиска возможных ошибок или закладок в чужом проекте. Решение проблемы – в использовании микропроцессоров и ПО собственной разработки со встроенными средствами повышения безопасности. Из-за сложности проектов возникают самые различные ошибки в современных микропроцессорах, включая микропроцессоры мировых лидеров. Собственное ПО и полное владение проектом микропроцессора в большинстве случаев позволяют парировать эти ошибки программными средствами.

Особенно остро вопрос безопасности стоит в банковской сфере, промышленности и на транспорте. Хакерские атаки могут привести к существенным денежным потерям, экологическим бедствиям или к прекращению штатного функционирования транспорта и промышленных предприятий. Для такого рода применений создаются доверенные вычислительные системы. В США критерий оценки доверенных компьютерных систем определен стандартом Министерства обороны США (*Department of Defense Trusted Computer System Evaluation Criteria, TCSEC*, DoD 5200.28-STD, December 26, 1985), более известным как «Оранжевая книга». Данный стандарт получил международное признание и оказал сильное влияние на последующие разработки в области информационной безопасности. Невозможно создать абсолютно безопасную от проникновения (доверенную) систему, поэтому в стандарте предложено оценивать

лишь ее степень доверия. Под доверенной системой понимается система, использующая аппаратные и программные средства для обеспечения одновременной обработки информации разных категорий секретности группой пользователей без нарушения прав доступа. В стандарте заложены уровень гарантированности, подотчетность, доверенная вычислительная база, монитор обращений, ядро безопасности, периметр безопасности. Безопасность и доверенность оцениваются по уровню управления доступом к информации и обеспечению конфиденциальности и целостности.

В России в настоящее время отсутствуют критерии оценки доверенных компьютерных систем. Доверенные вычислительные системы предполагают использование только отечественных компонент, включая элементную базу и ПО. Постановлением правительства РФ от 09.08.2016 г. № 764 введены понятия интегральной схемы первого уровня, когда микросхема разработана и произведена налоговым резидентом стран-членов Евразийского экономического союза, и интегральной схемы второго уровня, когда проектирование и разработка, а также испытания интегральной схемы произведены этим же резидентом. При отсутствии других критериев оценки доверенности ее степень можно также определять по степени «отечественности».

Основным компонентом доверенной системы является микропроцессор. В ФНЦ НИИСИ РАН разработка микропроцессоров является одним из основных направлений, в институте разрабатываются и серийно выпускаются 64-разрядные микропроцессоры с архитектурой КОМДИВ64, подобной архитектуре MIPS64 [1].

Производительность микропроцессора определяется производительностью ядра или количеством одновременно выполняющихся операций и временем доступа к памяти. В настоящее время доступ к памяти для большинства задач становится узким местом и в наибольшей степени определяет производительность. Существуют два способа повышения производительности:

– архитектурные решения, позволяющие увеличить количество одновременно выполняемых инструкций и скрыть простои, связанные с промахами в кэш-памяти;

– оптимизация подсистемы памяти.

В свою очередь, для первого способа используются следующие пути увеличения производительности:

– внеочередное исполнение инструкций (*Out-of-Order*, OoO);

– мультитредовая технология (*Multithreading*);

– использование многоядерности.

В современных микропроцессорах развивается технология переупорядочения не зависящих друг от друга инструкций для максимального повышения эффективности распараллеливания. Такие процессоры называются процессорами с внеочередным исполнением инструкций (out-of-order processors). Техника переупорядочения инструкций замечательна тем, что резко ослабляет негативные эффекты от медленной оперативной памяти и от наличия зависимых цепочек инструкций. Для таких процессоров рост производительности при наличии двух одновременно выполняемых инструкций составляет 1,8–1,9 раза, для трех – 2,5–2,8 раза. Однако такие процессоры существенно усложняются и резко возрастает потребление ядра.

Другим решением повышения производительности является использование мультитредовой технологии, аппаратно поддерживающей эффективное выполнение нескольких тредов (thread) для каждого ядра. Тред – это минимальный аппаратный функциональный блок микропроцессора, поддерживаемый операционной системой и использующий общие ресурсы с другими тредами (кэш-память, регистры).

Для встраиваемых применений более эффективным по показателю «производительность/потребляемая мощность» является увеличение числа ядер.

С точки зрения оптимизации подсистемы памяти используются многоуровневые кэш-памяти, симметричный доступ к общей памяти, механизмы предвыборок и многоуровневая буферизация [2].

В разрабатываемом высокопроизводительном микропроцессоре 1890ВМ118 для повышения производительности и возможности создания доверенных систем применена многоядерность и оптимизирована подсистема памяти.

Данный процессор является двухъядерным и создан по проектной норме 28 нм. Оптимизация архитектуры ядра микропроцессоров осуществлялась по параметру «производительность/потребляемая мощность».

Важнейшим параметром является производительность вычислительных систем. В работе [3] производительность представлена как $\frac{N_{\text{инст}}}{\text{Программа}} \times \frac{N_{\text{цик}}}{N_{\text{инст}}} \times \frac{T_{\text{прог}}}{N_{\text{цик}}} = \frac{T_{\text{прог}}}{\text{Программа}} =$ время работы CPU, где $N_{\text{инст}}$ – число выполненных инструкций за время работы микропроцессора (CPU); $N_{\text{цик}}$ – число циклов микропроцессора при выполнении программы; $T_{\text{прог}}$ – время выполнения программы.

Как видим, производительность зависит от трех характеристик – тактовой частоты, частоты следования инструкций и количества операций, выполняемых за одну инструкцию. Более того, время работы CPU в равной степени зависит от каждой из них: увеличение на 10 % одной дает общий прирост также на 10 %.

Частота микропроцессора зависит от технологии изготовления, количество инструкций за такт – от архитектуры микропроцессора, а количество

операций в инструкции – от системы команд и технологии компиляции. Каждый из этих параметров сложно изменить изолированно от остальных, поскольку основные технологии, определяющие каждую характеристику, взаимозависимы: длина конвейера влияет на частоту, количество инструкций, выполняемых за такт, влияет на длину конвейера. Рассмотрим увеличение значений всех трех компонент на примере микропроцессора 1890ВМ118.

Тактовая частота. Как было сказано выше, частота во многом определяется технологией производства. Расчет частоты функционирования микропроцессора 1890ВМ118 при нормальной температуре окружающей среды для технологии КМОП 28 нм процесс HPC+ показал возможность достижения частоты 1,5 ГГц. Однако для худшего случая (температура – 60 ºС или +125 ºС), минимального напряжения питания и худшего процесса без принятия специальных мер частота микропроцессора падает до 700 МГц. Для повышения частоты разработана методика, позволившая почти удвоить частоту. На первом шаге определяются наиболее критичные к задержкам блоки. В ядре микропроцессора это блок трансляции виртуальных адресов в физические (TLB), регистровый файл и кэш-память первого уровня. Соответственно блоки TLB, регистровые файлы и кэш-память первого уровня спроектированы полностью заказным образом. Общепризнанной методикой повышения частоты микропроцессора является удлинение конвейера. В простейших микроконтроллерах глубина конвейера составляла 1–3 стадии. Для некоторых современных микропроцессоров глубина конвейера достигает 40. Однако при прерываниях и переходах конвейер разрушается и производительность соответственно падает. Особенно это критично для встраиваемых применений. Именно поэтому в ядрах КОМДИВ64 используется минимально возможная глубина конвейера. Для рассматриваемого ядра 1890ВМ118 глубина конвейера составляет 7 стадий. Сбалансированный 7-стадийный конвейер позволяет достичь частоты функционирования ядра 1,5 ГГц.

С точки зрения подсистемы памяти на частоту микропроцессора оказывает влияние кэш-память второго уровня. Большие площади памяти приводят к заметным паразитным значениям и соответствующим задержкам. Оптимальным вариантом с точки зрения времени доступа и размера для 28 нм КМОП технологического процесса определен размер 512 Кб стандартной кэш-памяти второго уровня для 7-стадийного конвейера, не влияющий на быстродействие микропроцессора.

Следующей мерой повышения частоты микропроцессора является использование методики проектирования, рассмотренной в работах [4, 5]. Синтез микросхемы начинается с начального этапа разработки модели микросхемы, что позволяет на раннем этапе определять наиболее узкие места проекта, выделять блоки разного быстродействия и оптимизировать модель микросхемы. Создаются специальные скрипты САПР для обнаружения наиболее критичных путей, разрабатываются дополнительные библиотечные элементы к существующей библиотеке, позволяющие увеличить быстродействие проекта. При необходимости осуществляется ручная расстановка и/или разводка наиболее критичных по быстродействию блоков в соответствии с разработанной методикой.

Частота следования инструкций. Микропроцессор 1890ВМ118 является суперскалярным RISC-микропроцессором. Суперскалярность – это способность выполнения нескольких инструкций за один такт. В микропроцессоре 1890ВМ118 возможно одновременное выполнение двух инструкций. Исследования показали, что дальнейшее увеличение числа одновременно выполняемых инструкций приводит к заметному усложнению и увеличению потребляемой мощности, в то же время производительность растет нелинейно.

Для двух одновременно выполняемых инструкций производительность ядра 1890ВМ118 растет в 1,2–1,5 раза в зависимости от задачи, для трех инструкций – в 1,5–1,8 раза. Более эффективным является создание многоядерной системы на кристалле.

Возможность одновременного выполнения инструкций во многом поддерживается механизмом предсказания переходов и спекулятивным выполнением инструкций [6]. Суть метода заключается в прогнозировании направления ветвления программы и начале выполнения предполагаемых операций до подтверждения условия перехода. Повышение эффективности данного метода актуально для любого современного RISC-микропроцессора. Это объясняется тем, что в обычных программах порядка 20 % инструкций являются командами ветвления, которые в простейших случаях реализации блока выборки инструкций приводят к остановке конвейера. Базовые подходы к механизму предсказания ветвления одинаковые для всех RISC-микропроцессоров – это статическое и динамическое предсказания ветвления. Точность предсказания переходов в современных процессорах превышает 90 %. В ранних микропроцессорах КОМДИВ64 использовалось статическое предсказание ветвления. Исследования показали, что оптимальным с точки зрения сложности (площади и потребляемой мощности) и достижения вероятности предсказания (порядка 90 %) является динамическое предсказание с ограничением очереди предсказаний четырьмя записями и алгоритмом выбора предсказания по двухуровневой схеме. Этот вариант реализован в ядре 1890ВМ118.

Количество операций, выполняемых одной инструкцией. Существенное повышение производительности выделенных задач достигается за счет создания специализированных сопроцессоров.

В микропроцессор 1890ВМ118 встроен сопроцессор CPV, ориентированный на обработку изображений и сигналов [7]. Вычислительный блок векторного сопроцессора позволяет выполнять операции одинарной и двойной точности над векторами шириной 128 разрядов. Пиковая производительность достигается при использовании команд комплексного умножения с прибавлением и вычитанием третьего операнда («бабочка Фурье») и составляет за такт 10 вещественных операций двойной точности или 20 вещественных операций одинарной точности. Сопроцессор CPV может программироваться на языках Си и ассемблера.

Проведенные исследования производительности прототипа микропроцессора 1890ВМ118 показали, что по производительности он приближается к современному Out-of-Order микропроцессору XLP316 фирмы Broadcom.

Для оценки использовались тесты CoreMark и SciMark2. Тест CoreMark был разработан консорциумом EEMBC (Embedded Microprocessor Benchmark Consortium). Тест CoreMark вычисляет производительность ядра микропроцессора, выполняя относительно простой код, однако этот код использует общие практически для всех приложений структуры данных и алгоритмы. Для теста выбрана такая реализация, чтобы все вычисления (и их итоговые значения) производились непосредственно в процессе выполнения программы, что предотвращает исключение кода при оптимизации в процессе компиляции. Результаты работы всех тестов сверяются с эталонными.

В состав теста CoreMark входят различные вычислительные алгоритмы, характерные для микропроцессоров для встраиваемых систем, в том числе

– *операции с матрицами*, такие как умножение матрицы на константу, вектор или другую матрицу; также производится проверка данных в итоговой матрице;

– *сортировка списков*; применяется для определения того, как быстро процессор может получить данные из памяти для сканирования списка; если списки больше доступной кэш-памяти, то при их обработке проверяется и эффективность подсистемы кэш-памяти;

– *операции переключения конечного автомата*; конечный автомат тестирует входную строку, чтобы определить, являются ли входные данные числом, и принимает состояние invalid, если это не число; для проверки правильности операции в тесте CoreMark подсчитывается, сколько раз было пройдено каждое состояние;

– *подсчет контрольных сумм*; осуществляется подсчет 16-битного циклического избыточного кода (CRC) на основе данных, содержащихся в элементах списка.

Результат теста составил 3 309 итераций в секунду для микропроцессора XLP316 и 2 705 итераций в секунду для прототипа микропроцессора 1890ВМ118. Оба микропроцессора функционировали на частоте 1 ГГц.

Еще одним репрезентативным набором тестов производительности микропроцессоров является тест SciMark2. Это мощный однопоточный тест (интенсивные научные вычисления), показывающий производительность одного ядра микропроцессора в мегафлопсах (Mflops). Тесты написаны на языке C и входят в состав Phoronix Test Suite.

Тесты производительности SciMark2 разработаны Национальным институтом стандартизации США (NIST) для измерения производительности математических вычислений и являются общепризнанным пакетом.

Пакет состоит из пяти вычислительных ядер.

- *Fast Fourier Transform* (FFT), выполняющее одномерное прямое преобразование Фурье для 2^12 (2^20) комплексных чисел. Это ядро использует комплексную арифметику, перестановки битов, неконстантные ссылки на память и тригонометрические функции. Первая часть ядра производит обращение битов (без измерения), вторая – требуемые N log(N) вычислений.
- *Jacobi Successive Over-relaxation* (SOR). Это метод последовательной сверхрелаксации Якоби, реализующий типичные паттерны доступа к памяти, свойственные для конечно-разностных методов, таких как решение двухмерного уравнения Лапласа с ограничениями Дирихле. Ядро работает на сетках размером 100×100 и 1 000×1 000 соответственно.
- *Monte Carlo integration*, вычисляющее приближенное значение числа Пи оценкой площади четверти круга методом Монте-Карло. Алгоритм выбирает случайные точки на квадрате и проверяет их принадлежность кругу радиуса 1. Ядро использует генераторы случайных чисел, синхронизованные методы, вкладывание функций.
- *Sparse matrix multiply* – умножение разреженных матриц, использующее неструктурированные матрицы в сжатом формате. Это ядро использует косвенную адресацию и нерегулярный доступ к памяти.
- *Dense LU matrix factorization*, находящее LU-разложение плотной матрицы. Алгоритм использует линейные ядра (BLAS) и операции над плотными матрицами.

Численные значения производительности, полученные после запуска пяти вычислительных тестов, усредняются, в итоге выдается одна оценка производительности, которая называется Composite Score.

Результат теста для микропроцессора XLP316 составил 181,7 Мфлопса, для прототипа микропроцессора 1890ВМ118 – 149,5 Мфлопса. Сравнение производилось при частоте функционирования 1 ГГц.

Микропроцессор 1890ВМ118 является системой на кристалле. Интеграция функций микропро-

цессора и системного контроллера на одном кристалле и создание систем на кристалле приводят не только к повышению надежности системы и уменьшению ее габаритов в силу сокращения числа компонент, но и к увеличению производительности всей системы.

Следующим фактором повышения производительности вычислительных систем является организация подсистемы памяти в целом. Особенно важна задача повышения скорости обмена данными с ОЗУ для задач с большими объемами данных, не позволяющими эффективно использовать кэш-память. В таких случаях ускорение возможно прежде всего за счет введения предвыборок, увеличения частоты и ширины памяти. В двухъядерном микропроцессоре 1890ВМ118 реализуется 64-разрядный канал памяти с частотой до 1000/2000 МГц. Два процессорных ядра и общий системный контроллер объединены по схеме с симметричным доступом к общей памяти ОЗУ (архитектура SMP). Отключение средств когерентного доступа к памяти дает возможность использовать двухъядерный микропроцессор в режиме асимметричного доступа к памяти (архитектура AMP) с разделением ее между ядрами. Это позволяет расширить область применения микропроцессора: при некотором снижении надежности добиться максимального увеличения производительности или повышения надежности вычислительной системы в целом за счет аппаратного разделения памяти.

Архитектура SMP предъявляет ряд требований к разрабатываемой системе, направленных на обеспечение целостности данных при симметричном доступе к памяти в многоядерной системе и взаимодействие процессорных ядер. К таким требованиям можно отнести когерентность кэш-памяти процессорных ядер, поддержку атомарных операций чтение–модификация–запись, межпроцессорные прерывания.

В разрабатываемой системе на кристалле общий системный контроллер обеспечивает доступ к ОЗУ для каждого микропроцессорного ядра, а также подключение периферийных устройств. Для обмена данными между ОЗУ, микропроцессорными ядрами и контроллером периферийных устройств используется шина AMBA AXI.

Проблема обеспечения целостности данных при обмене данными внутри системы на кристалле может возникнуть в том случае, когда один из участников обмена данными получает доступ к не самой последней копии данных. Причиной этому могут стать кэшируемые обращения для записи и чтения от обоих ядер к данным в ОЗУ по одинаковому адресу (к одной строке кэш-памяти), а также обращение от периферийного устройства к области памяти, копия которой уже содержится в кэш-памяти одного или обоих ядер.

Таким образом, разработку протокола когерентности для системы на кристалле можно условно разделить на две задачи – реализация средств обеспечения когерентного обмена данными между ядрами и реализация когерентного ввода/вывода для периферийных устройств.

В качестве протокола когерентности для межпроцессорного обмена широко используется протокол MOESI. По сравнению с базовым протоколом MSI он позволяет уменьшить число обращений в кэш-память другого ядра. Кроме того, передача модифицированной строки кэш-памяти может осуществляться напрямую между ядрами, а не путем записи/чтения строки в ОЗУ. Обеспечение когерентного ввода/вывода для периферийных устройств достигается за счет проверки наличия строки с адресом, по которому происходит обращение от периферийного устройства, в кэш-памяти одного или обоих ядер. Запрос на прослушивание (snooping) кэш-памяти содержит физический адрес обращения, признак записи/чтения и признак того, что объем записываемых данных меньше ширины строки кэш-памяти. Ответ на прослушивание формируется по окончании соответствующего действия. При промахе запроса на прослушивание в кэш ответ периферийному устройству выдается без выполнения дополнительных действий.

Существенным недостатком прослушивания ядер при обращении в ОЗУ от периферийного устройства является потеря производительности как для ядер – из-за постоянных обращений в кэш-память, так и для обмена данными между устройством и ОЗУ из-за ожидания ответа из кэш-памяти для каждого сегмента данных размером со строку кэш-памяти.

Решением этой проблемы может стать наличие в системном контроллере блока, называемого директорией и содержащего информацию о наличии в кэш-памяти каждого ядра строки с адресом обращения. Любой запрос строки от периферийного устройства на запись или чтение поступает в директорию. Физический адрес запроса проверяется на совпадение с адресами, хранящимися в директории. При выявлении совпадения запрос на прослушивание отправляется в то ядро, в кэш-памяти которого обнаружен адрес запроса от периферийного устройства. Если совпадение не обнаружено, запрос на прослушивание в кэш-память ядер не отправляется, а периферийное устройство получает разрешение на продолжение передачи данных. Такая директория снижает нагрузку на кэш-память ядра, поскольку в нее попадают только запросы на прослушивание по адресам, наличие которых подтверждено директорией. Кроме того, ответ об отсутствии строки с адресом обращения от директории приходит в периферийное устройство быстрее, чем от кэш-памяти, сокращая таким образом время, затрачиваемое на прослушивание.

Важным свойством разрабатываемых микропроцессоров является возможность создания доверенных систем на их основе.

Отечественные микропроцессоры для ответственного применения должны поддерживать организацию доверенной среды исполнения, гарантирующей заложенную устойчивость к попыткам взлома всей системы. Если принять критерий оценки эффективности такой защищенности системы за время и количество условных единиц, которые готов потратить злоумышленник на преодоление защиты, то ресурс системы можно считать условно надежным при условии, что система готова противостоять взлому достаточное количество времени, в течение которого злоумышленник откажется от попыток взлома. При проектировании микропроцессора 1890ВМ118 особое внимание было уделено повышению безопасности, в основу чего легли архитектурные решения и принимаемые еще на стадии проектирования, и используемые в ПО. На основе микропроцессора 1890ВМ118 становится возможным комплексный подход к построению системы, устойчивой к взлому, позволяющей развернуть программно-аппаратный комплекс доверенной среды исполнения, эффективность организации которой ограничивается лишь расчетным списком уязвимостей и потенциальных атак, противостоять которым способна система. В микропроцессорах реализованы такие аппаратные решения для повышения безопасности систем, как организация доверенной загрузки операционной системы, использование второго ядра для контроля безопасности, политика изолированного доступа к памяти. Рассмотрим эти аппаратные решения.

Доверенная загрузка микропроцессора становится возможной при использовании накристального ПЗУ, в которое разработчиком аппаратуры записываются флаги конфигурации доверенной загрузки и хэш-ключи проверки загружаемой операционной системы. В накристальное ПЗУ на этапе производства микросхемы записывается загрузчик первого уровня, который проверяет цифровую подпись загрузчика второго уровня, находящегося в основной памяти, и дает разрешение на его загрузку. Схема загрузки операционной системы при этом примерно следующая.

После включения питания или по сигналу сброса (Reset) микросхемы микропроцессор начинает исполнять инструкции по адресам, отображаемым на накристальное ПЗУ.

Из накристального ПЗУ запускается загрузчик первого уровня, который копирует необходимый код программы в ОЗУ микропроцессора.

Код, скопированный в ОЗУ, выполняет проверку цифровой подписи загрузчика второго уровня, в случае успешной проверки загружается операционная система.

Накристальное ПЗУ (ROM) представляет собой классическое энергонезависимое ПЗУ, которое создается на производстве в процессе выращивания основного кристалла. Данные в ПЗУ также записываются на стадии производства и после этого не могут быть перезаписаны. В процессе работы системы на кристалле ПЗУ доступно только для чтения данных. Ввиду того, что данные в память записываются при производстве, несанкционированная запись в нее невозможна. В отличие от отдельных микросхем ПЗУ, которые интегрируются на системную плату отдельно, невозможен и подлог накристальной ПЗУ, что гарантирует достоверность содержащихся в ПЗУ данных.

В совокупности с накристальным ПЗУ для доверительной загрузки операционной системы может использоваться однократно программируемая память XPM (Extra Permanent Memory).

Память XPM, построенная в базисе стандартной КМОП-логики, является однократно записываемой. Незапрограммированная память содержит в себе логические «0». В основе физики процесса записи в ячейку памяти логической «1» лежит принцип пробоя подзатворного окисла транзистора высоким напряжением, получаемым схемой накачки напряжения или отдельной линией, выведенной на высокое напряжение. Запатентованный NVM (non-volatile memory) механизм гарантирует полную защищенность данных. В связи с этим данная память может находить применение в области защиты информации. В системах на кристалле она может использоваться для хранения важных данных, прошивок и т.п.

Второе ядро микропроцессора может использоваться как монитор безопасности (гипервизор), осуществлять перехват подозрительной активности на основе метрик (временных, поведенческих), обнаружение недопустимой конфигурации устройств, прочистку кэш-памяти и сохранение точек восстановления после сбоя.

Изолированный доступ к памяти для периферийных устройств гибко настраивается с помощью блока защиты совместного доступа к ней. Данный блок (MPU) позволяет на аппаратном уровне реализовать политику изолированного доступа к памяти системы. Использование MPU в микропроцессоре 1890ВМ118 делает их устойчивыми, надежными и в некоторых случаях более безопасными, препятствуя задачам приложения получить нежелательный доступ к стеку или области памяти, или повредить стек и память данных, используемые другими задачами. MPU позволяет избежать распространенных атак, направленных на порчу конфигурации или критических данных, принадлежащих другому процессу.

Блок имеет следующие возможности:

– определение границ окна доступа устройства к системной памяти;

– определение типа операции доступа к окну, чтение и/или запись;

– регистры для контроля ошибок доступа устройств к памяти;

– формирование прерывания при ошибке доступа к окну.

За каждым «мастер»-устройством закреплено свое окно доступа в системную память, окно доступно по чтению и/или по записи. Если в конфигурационном регистре окна биты доступа не установлены, адресное окно недоступно. При обращении к памяти проверяются начальный и конечный адреса окна, тип операции (чтение/запись). Если адрес обращения не принадлежит границам окна или тип операции запрещен, выставляется флаг ошибки окна, адрес и атрибуты обращения записываются в управляющие регистры и вызывается прерывание. После конфигурации управляющих регистров MPU возможна однократная запись бита, запрещающего последующие изменения конфигурационных регистров MPU.

Можно выделить два основных перспективных направления повышения безопасности систем на кристалле на базе отечественных микропроцессоров: на основе средств виртуализации и на основе доверенной среды исполнения. Концепция безопасности на основе виртуализированной системы базируется на основе минимизации потенциально уязвимых участков кода путем изоляции недоверенной операционной системы в виртуальной машине с помощью программно-аппаратных средств. Кроме свойств изолированности доверенной среды, виртуализация позволяет осуществлять клонирование, резервное копирование и откат виртуальной машины, это повышает надежность системы, что также актуально в серверных приложениях. Оба направления взаимосвязаны и находятся в плоскости разработки аппаратно-программных средств. Разработка двух направлений имеет пересекающиеся части в области аппаратных средств защиты памяти и ядра микропроцессора, так, большинство современных архитектур микропроцессоров поддерживают режим виртуализации операционной системы. Аппаратная поддержка инфраструктуры доверенного режима исполнения приложений без поддержки виртуализации позволяет относительно упростить архитектуру защищенной системы, однако не избавляет от программно-аппаратной части, что в универсальном случае представляет собой монитор событий, реализованный на основном или отдельном микропроцессорном ядре.

Все описанные аппаратные решения для повышения производительности и безопасности применяются в блоках микропроцессоров, разрабатываемых в ФНЦ НИИСИ РАН, что позволяет строить на их основе доверенные высокопроизводительные системы.

Литература

1. Бобков С.Г. Импортозамещение элементной базы вычислительных систем // Вестн. РАН. 2014. Т. 84. № 11. С. 1010–1016.
2. Аряшев С.И., Бычков К.С. Оптимизация механизма предварительного считывания в кэш-памяти второго уровня. Проблемы разработки перспективных микро- и наноэлектронных систем // Сб. науч. тр. М.: Изд-во ИППМ РАН, 2016. Ч. II. С. 274–279.
3. Patterson D.A., Hennessy J.L. Computer architecture: a quantitative approach. 4th ed. The Morgan Kaufmann Series in Computer Architecture and Design. Elsevier Sc. Kindle Ed., 2011, 676 p.
4. Бобков С.Г. Высокопроизводительные вычислительные системы. М.: Изд-во НИИСИ РАН, 2014. 299 с.
5. Бобков С.Г., Евлампиев Б.Е. Анализ САПР микроэлектроники и выделение методов достижения предельного быстродействия для СБИС со сложной структурой класса микросхем графического контроллера // Инструментальные средства программирования: сб. стат.; [под ред. акад. РАН В.Б. Бетелина]. М.: Изд-во НИИСИ РАН, 2006. С. 112–128.
6. Барских М.Е., Бобков С.Г. Исследование влияния динамического предсказания ветвлений на производительность перспективных микропроцессоров НИИСИ РАН // Информационные технологии. 2015. № 10. С. 736–742.
7. Бобков С.Г., Аряшев С.И., Барских М.Е., Зубковский П.С., Ивасюк Е.В. Высокопроизводительные расширения архитектуры универсальных микропроцессоров для ускорения инженерных расчетов // Информационные технологии. 2014. № 6. С. 27–37.

Software & Systems
DOI: 10.15827/0236-235X.119.345-352

Received 03.05.17
2017, vol. 30, no. 3, pp. 345–352

HIGH-PERFORMANCE MICROPROCESSOR 1890BM118 FOR TRUSTED COMPUTING SYSTEMS

S.I. Aryashev [1], *Ph.D. (Engineering), Branch Manager, aserg@cs.niisi.ras.ru,*
S.G. Bobkov [1], *Director, bobkov@cs.niisi.ras.ru*
P.S. Zubkovsky [1], *Head of Department, zubkovsky@niisi.ras.ru*
S.A. Morev [1], *Junior Researcher, morev@cs.niisi.ras.ru*
B.Yu. Rogatkin [1], *Research Associate, boris240@cs.niisi.ras.ru*

[1] *Federal State Institution "Scientific Research Institute for System Analysis of the Russian Academy of Sciences" (SRISA RAS), Nakhimovsky Ave. 36/1, Moscow, 117218, Russian Federation*

Abstract. This article considers the problems of developing a high-performance microprocessor for trusted computing systems.

Microprocessor performance is determined by core capacity or a number of stages, which may be in execution simultaneously, and memory access time. Microprocessor applicability to create trusted systems is based on using self-made blocks and nodes.

Microprocessor core performance is determined by three characteristics: clock frequency, operations per instruction, instruction rate. For 1890VM118 microprocessor these characteristics were optimized by performance/power parameter. Clock frequency increasing is achieved using custom development of timing critical blocks and pipeline length optimization. Hardware solutions, such as superscalar instruction execution, branch prediction and preliminary data load in cache memory, increase instruction rate. Implementation of the arithmetic co-processor focused on digital signal processing tasks allows increasing the number of operations per instruction. The paper considers increasing memory subsystem performance in terms of symmetric memory access for a dual-core microprocessor. It also describes the approach to implementing cache coherence in processor cores.

The authors pay special attention to increasing the security level in microprocessor for trusted computing systems. They consider hardware solutions for operating system trusted boot and isolated memory access. To ensure trusted boot the authors suggest using on-chip ROM and one-time-programmable memory containing a secured bootloader and the keys to verify the signatures of an operating system. Isolated access solutions include a memory access controller discussed in the article and implemented inside a microprocessor. The paper proposes prospective solutions for creating trusted systems based on microprocessors by NIISI RAS.

Keywords: trusted systems, System-on-a-Chip, microprocessor architecture, co-processor.

References

1. Bobkov S.G. Import phaseout of computing system hardware components. *Vestn. RAN* [Herald of the Russian Academy of Sciences]. 2014, vol. 84, no. 11, pp. 1010–1016 (in Russ.).
2. Aryashev S.I., Bychkov K.S. Optimization of a preread mechanism in second-level cache. *Problemy razrabotki perspektivnykh mikro- i nanoelektronnykh sistem. Sb. nauch. tr.* [Design Problems of Advanced Micro and Nanoelectronic Systems. Collected Papers]. Moscow, IPPM RAN Publ., 2016, part II, pp. 274–279 (in Russ.).
3. Patterson D.A., Hennessy J.L. *Computer architecture: a quantitative approach*. 4th ed. The Morgan Kaufmann Series in Computer Architecture and Design. Elsevier Sc. Kindle Ed., 2011, 676 p.
4. Bobkov S.G. *Vysokoproizvoditelnye vychislitelnye sistemy* [High-performance Computing Systems]. Moscow, NIISI RAN Publ., 2014, 299 p.
5. Bobkov S.G., Evlampiev B.E. Microelectronics CAD analysis and selecting methods of maximum highspeed response for SBIS with a complex structure of a graphics controller chip class. *Instrumentalnye sredstva programmirovaniya: sb. stat.* [Programming Tools. Collected Papers]. V.B. Betelin (Ed.). Moscow, NIISI RAN Publ., 2006, pp. 112–128 (in Russ.).
6. Barskikh M.E., Bobkov S.G. Investigation of the influence of dynamic branch prediction to perspective microprocessors performance from NIISI RAS. *Informatsionnye tekhnologii* [Information Technologies]. 2015, no. 10, pp. 736–742 (in Russ.).
7. Bobkov S.G., Aryashev S.I., Barskikh M.E., Zubkovsky P.S., Ivasyuk E.V. High-performance Extensions of Microprocessor Architecture for Speeding-up of Scientific and Engineering Calculations. *Informatsionnye tekhnologii* [Information Technologies]. 2014, no. 6, pp. 27–37 (in Russ.).

Примеры библиографического описания статьи

1. Аряшев С.И., Бобков С.Г., Зубковский П.С., Морев С.А., Рогаткин Б.Ю. Высокопроизводительный микропроцессор 1890ВМ118 с архитектурой КОМДИВ для создания доверенных систем // Программные продукты и системы. 2017. Т. 30. № 3. С. 345–352. DOI: 10.15827/0236-235X.119.345-352.

2. Aryashev S.I., Bobkov S.G., Zubkovsky P.S., Morev S.A., Rogatkin B.Yu. High-performance microprocessor 1890BM118 for trusted computing systems. *Programmnye produkty i sistemy* [Software & Systems]. 2017, vol. 30, no. 3, pp. 345–352 (in Russ.). DOI: 10.15827/0236-235X.119.345-352.