

RISC-V – ПРОЦЕССОРНЫЕ ТЕХНОЛОГИИ БЕЗ ОГРАНИЧЕНИЙ

Якушкин С. И., Владимиров К. И., Пузикова В. В.

ООО «Синтакор»

Москва, Россия

E-mail: {sergey.yakushkin, konstantin.vladimirov}@syntacore.com

В работе рассматривается открытая процессорная архитектура RISC-V, этапы её развития и стандартизации, ключевые факторы успеха. Обсуждаются понятия системы команд, модульности и расширяемости, профилей и платформ. Проводится анализ мировых трендов и актуальных продуктов на основе архитектуры RISC-V для различных приложений. Кроме того, рассматривается поддержка инструментов разработки и программного обеспечения, а также задачи разработки открытых стандартов для систем с открытой экосистемой приложений и программного обеспечения.

ВВЕДЕНИЕ

Проектирование и разработка современных микропроцессорных ядер – трудоемкая задача, требующая нетривиальных инструментов и серьёзных инвестиций. Сложность разработки новых поколений процессорных архитектур постоянно возрастает из-за увеличивающихся требований производительности, а также по причине того, что большинство распространённых систем команд проприетарны и не расширяемы, т.е. обладают как патентными, так и техническими ограничениями на переиспользование. Это приводит к закреплению различных неудачных решений в экосистеме и сдерживанию развития микроархитектуры и программного обеспечения [1]. Решением перечисленных проблем являются новые принципы построения системы команд и экосистемы разработки, а также переход к свободным и открытым архитектурам [2].

I. ОТКРЫТАЯ ПРОЦЕССОРНАЯ АРХИТЕКТУРА

RISC-V – свободная и открытая процессорная архитектура. Её основой является модульная и расширяемая система команд [3], развиваемая открытым сообществом. Система команд RISC-V спроектирована для широкого спектра оборудования от смарт-карт до высокопроизводительных серверов и ускорителей. Базовый набор команд очень минималистичен, поэтому его легко реализовать в любых решениях. При этом в кодировке оставлено место для стандартных и пользовательских расширений, предназначенных для достижения максимальной эффективности на высокопроизводительных платформах.

Базовый набор команд – это обязательная часть системы команд. Доступно несколько вариантов этого набора. Так, RV64I является основным набором инструкций для работы с 64-битными данными и адресами. Существуют альтернативные варианты базового набора – RV32E, RV32I, RV64E, RV128I. В наборе RV32I присутствует всего лишь 47 инструкций.

Среди основных стандартных расширений можно отметить расширение M для умножения и

деления целых чисел, F и D – для чисел с плавающей точкой одинарной и двойной точности соответственно, V – для векторных команд, B – для битовых операций. Новые стандартизованные расширения помечаются буквами Z, S или H в начале названия. Например, Ztso определяет последовательную консистентность памяти как расширение и т.д. Таким образом, могут существовать RISC-V системы с поддержкой и без поддержки модели памяти TSO (Total Store Ordering).

Дополнительно определяются понятия профилей и платформ. Стандартные профили задают список расширений системы команд обязательных в выбранной области применения. Например, профили RVA20U64 и RVA20S64 для прикладных процессоров объединяют все расширения, которые требуются для загрузки Linux и работы стека приложений на 64-битной архитектуре. Профили обеспечивают гибкость стандарта без фрагментации экосистемы. Стандартные платформы основаны на профилях и дополнительно определяют системные требования программного обеспечения к аппаратному обеспечению, процессу загрузки, управлению питанием и другим аспектам.

II. ЭТАПЫ РАЗВИТИЯ СТАНДАРТОВ И ЭКОСИСТЕМЫ

Проектирование системы команд началось в академической среде приблизительно в 2010 г. Работы включали в себя эксперименты и исследование существующих архитектур, формальное определение RISC-V, создание тестовых SnK. Международный консорциум RISC-V был создан в 2015 г. Уже к 2017 году на рынке появились встроенные ядра для коммуникаций и управления питанием, началось активное развитие программного обеспечения. Основные стандарты были приняты к 2019 году, что позволило создавать микроконтроллерные ядра и SnK для IoT приложений.

На примере линейки микроконтроллерных ядер Syntacore (SCR1, SCR3, SCR4 и SCR6), представленной на рис. 1, можно отметить конкурентные уровни производительности RISC-V, доступ-

ные решения с открытым кодом [4], широкий набор инструментов программирования и библиотек, возможность строить специализированные многопроцессорные системы. Выпущены продукты для широкого диапазона приложений, включающие как микроконтроллеры с низким энергопотреблением, так и высокопроизводительные машины для ЦОД. На китайском рынке можно отметить платежные терминалы Alipay на основе RISC-V, и чип LTE Cat.1 для IoT от China Mobile. По прогнозам Semico Research к 2025 году микроконтроллерные ядра RISC-V займут более 28% мирового рынка IoT.

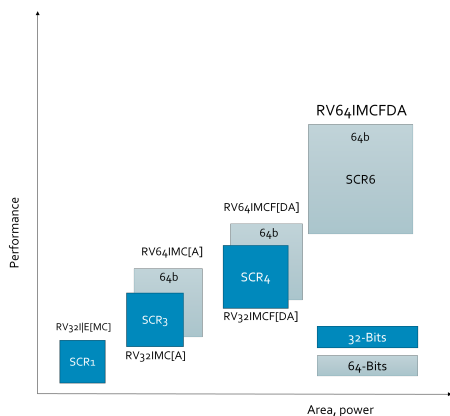


Рис. 1 – Линейка микроконтроллерных ядер SCR

Важной частью развития экосистемы является развитие инструментов разработки. RISC-V поддержан как целевая архитектура в компиляторах GCC и LLVM, в операционных системах Linux и FreeBSD, в системе реального времени Zephyr, в отладчиках GDB и OpenOCD, в большом количестве инструментов разработки. Также доступны модели разного уровня, от функциональных (SAIL, Spike, QEMU) до потактовых, таких как gem5.

III. МОБИЛЬНЫЕ И СЕРВЕРНЫЕ ПРИМЕНЕНИЯ

В 2021 году были приняты стандарты и ратифицированы расширения для многопроцессорных систем с поддержкой Linux, векторных и битовых операций, гипервизора, режима отладки. Отметим, что поддержка Linux имеется и в производимых Syntacore ядрах SCR5, SCR7 и SCR9 (рис. 2), которые используются в системах промышленной автоматизации, медиапроцессорах, а также в процессорах для ИИ.

В 2023 году компания Google объявила о начальной поддержке RISC-V в операционной системе Android. На RISC-V China Summit были представлены серверные процессоры с 64 ядрами SG2042, сервер 8xHS-S1 с 2048 ядрами от PerfXLab, 10 Гбит коммутатор Milk-V Vega, относящиеся к сегменту высокопроизводительных систем [5]. Сообщество активно обсуждает дальнейшее развитие RISC-V: профиль RVA23, серверную и Android платформы, стандарты безопасности

и защиты от атак по сторонним каналам, развитие WorldGuard и IOPMP, разработку Vector 2.0 с 64-битными инструкциями и мн.др.

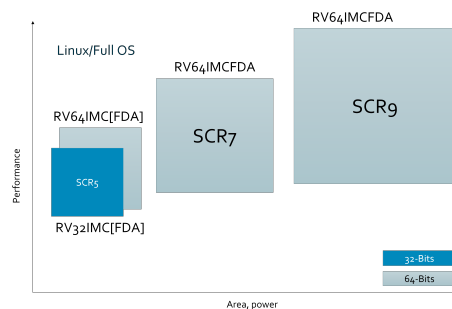


Рис. 2 – Линейка Linux-capable ядер SCR

IV. ЗАКЛЮЧЕНИЕ

Основным фактором успеха RISC-V является открытая кооперация в международном консорциуме, включающем 3500+ участников из 70+ стран. В консорциуме представлено 12 сегментов рынка, 100+ ведущих производителей процессоров (Intel, Qualcomm), 50+ ведущих разработчиков программного обеспечения (Google, Microsoft) и 150+ исследовательских организаций. Сильные региональные сообщества созданы в Китае, Индии, Европе, а с 2022 г. начал свою работу и Российский Альянс RISC-V.

Влияние RISC-V на индустрию можно сравнить с появлением ключевых открытых стандартов и сообществ, таких как USB, PCI и Linux. На данный момент отмечается высокий спрос во всех сегментах и на всех уровнях производительности, производятся существенные инвестиции. Благодаря совместным усилиям большого числа ведущих экспертов и компаний происходит быстрый рост экосистемы и активное развитие технологий. Можно сделать вывод, что успех RISC-V неизбежен не только в сегменте микроконтроллерных ядер, но и в области вычислений общего назначения.

V. СПИСОК ЛИТЕРАТУРЫ

1. Фролов, В. А. Исследование технологии RISC-V / В. А. Фролов, В. А. Галактионов, В. В. Санжаров // Труды Института системного программирования РАН. – 2020. – Т. 32, № 2. – С. 81–98.
2. Программная инфраструктура RISC-V [Электронный ресурс] / Синтакор. – Режим доступа: https://syntacore.com/media/riscv_moscow_2019/riscv_sw_upd.pdf – Дата доступа: 07.11.2023.
3. Design of the RISC-V Instruction Set Architecture: Technical Report / Electrical Engineering and Computer Sciences, University of California at Berkeley; A. Waterman. – Berkeley, 2016. – 104 с. – № UCB/EERC-2016-1.
4. SCR1 RISC-V Core [Electronic resource] / Syntacore. – Mode of access: <https://github.com/syntacore/scr1>. – Date of access: 07.11.2023.
5. RVSC2023 [Electronic resource] / CNRV. – Mode of access: <https://github.com/cnr/rvsc2023>. – Date of access: 07.11.2023.