

Оценка эффективности функционирования автоматизированных систем с различными процессорными архитектурами

М.С. Гнутов, email: monk666@bk.ru¹
В.Н. Титаренко, email: peresl77@mail.ru¹
А.Б. Сизоненко, email: siz_al@mail.ru¹

¹ Краснодарское высшее военное орденов Жукова и Октябрьской Революции Краснознаменное училище имени генерала армии С.М. Штеменко

***Аннотация.** Представлены основные процессорно - архитектурные концепции. Подробно рассмотрен подход VLIW, как альтернатива суперскалярным машинам для организации параллелизма на уровне инструкций. Опытным путем измерена производительность автоматизированных систем с разными архитектурами процессоров. Проведена опытная оценка эффективности ЭВМ на базе Эльбрус 8С и Intel Core-i7 Перечислены преимущества, недостатки и будущие перспективы технологии VLIW. Предложены комплексные меры повышения эффективности работы разрабатываемого программного обеспечения для систем, основанных на процессорах российского производства «Эльбрус».*

***Ключевые слова:** архитектура процессора, параллелизм, RISC, VLIW, Эльбрус, Intel.*

Введение

Тенденции развития информационных технологий, жесткая конкуренция на рынке компьютерных инноваций приводят к растущим потребностям приложений и, как следствие, необходимости увеличения производительности автоматизированных систем. Также в целях защиты внутреннего рынка Российской Федерации, развития национальной экономики, поддержки российских организаций, осуществляющих свою деятельность в сфере информационных технологий в 2015 году правительством Российской Федерации были введены ограничения на допуск иностранного программного обеспечения, используемого для обеспечения государственных и муниципальных нужд, что простимулировало отечественных производителей электроники и программного обеспечения, а также создало более благоприятные

условия для распространения и продвижения национальной продукции внутри страны [1].

Эти факторы оправдывают вложения и инвестиции, направленные в исследования новых и применения уже известных процессорных архитектур в нашей стране.

Особенно перспективным направлением является семейство микропроцессоров «Эльбрус» основанное на архитектуре VLIW, разрабатываемых российской компанией МЦСТ при участии ИНЭУМ.

1. Основные виды процессорных архитектур

Слово «архитектура» можно перевести с греческого языка как «искусство строить». Фактически это расположение частей и элементов в общей структуре. Архитектура процессора с точки зрения:

- программиста – это совместимость с определенным кластером команд (работающим, например, на Intel x86-64), их систем адресации, организации регистровой памяти (и др.) и способа конечной реализации программы;

- непосредственно аппаратной конфигурации – некие определенные свойства, качества и внутренняя конструкция рассматриваемого семейства процессоров.

В настоящее время существует множество видов процессоров и базовых архитектур. Рассмотрим основные виды.

CISC (англ. Complex Instruction Set Computer — «компьютер с полным набором команд»). Примеры архитектуры: x86 (IA-32) и x86_64 (AMD64).

RISC (англ. Reduced Instruction Set Computer — «компьютер с сокращённым набором команд»). Примеры архитектуры: PowerPC, серия ARM (ARM7, ARM9, ARM11, Cortex).

VLIW (англ. Very Long Instruction Word — «очень длинная машинная команда»). Примеры архитектуры: Intel Itanium, Эльбрус.

MISC (англ. Minimal Instruction Set Computer — «компьютер с минимальным набором команд»). Используется в IoT-сегменте, роутерах.

OISC (One Instruction Set Computer — «архитектура с единственной инструкцией»). Примеры архитектуры: BitBitJump, ByteByteJump, SUBLEQ.

TTA (Transport triggered architecture) TTA является разновидностью OISC. Примеры архитектуры: MOVE Project

Некоторые из указанных выше архитектур сильно разнятся, некоторые схожи, некоторые со временем объединяются. Еще в 80х годах популярная CISC архитектура начала приобретать некоторые

RISK черты. Сейчас в современных процессорах x86-64 и x86 внутри стоит RISC ядро и используются микрокод.

На протяжении многих лет распространенная архитектура x86, x86-64 выигрывала не только за счет совокупности показателей цена-удобство-популярность. Дело в совместимости - x86_64 всё ещё лидер, только как наследник x86. Так как старые программы работают только на x86, то и новые системы должны быть совместимыми для работы привычного программного обеспечения

Развитие архитектуры RISC в современных реалиях оказалось ограниченным из-за парадигмы машины фон Неймана [2], в которой программы реализованы с допущением, что инструкции должны выполняться последовательно, в том же порядке, в каком они указаны в коде. То есть аппаратное отсутствие параллелизма? Нет, в суперскалярных процессорах, позволяющих выполнять сразу несколько команд за такт (путем использования нескольких конвейеров) для распознавания зависимостей между машинными инструкциями применяется сложный аппаратный планировщик. Однако, при росте количества функциональных модулей, в геометрической прогрессии растет и размер этого предсказателя переходов, занимая всё свободное место на кристалле процессора. Это ограничивает развитие ядер на супербыстрых RISC архитектурах.

При подходе VLIW можно назначить все планирование программному компилятору, который должен выискать в программе независимые инструкции, собрать их вместе в очень длинные слова-инструкции, после чего отправить на одновременное исполнение функциональными модулями, количество которых (в идеале) равно количеству операций в этой инструкции. Аппаратно VLIW-процессор состоит из нескольких простых функциональных модулей, подключенных к шине процессора, нескольких регистров и блоков кэш-памяти. Максимальная скорость обработки определяется только количеством и внутренними составом самих функциональных модулей.

2. Сравнение производительности ЭВМ «Эльбрус 8С» и «Intel Core-i7» на основе синтетических тестов

В рамках научно-исследовательской работы, выполняемой в декабре 2020 года на базе военного инновационного технополиса «ЭРА» в г. Анапа, был проведен ряд сравнительных исследований автоматизированных систем с различными процессорными архитектурами. Для оценки производительности архитектур VLIW и x86-64 были использованы вычислительные машины со следующими характеристиками:

Таблица 1

Характеристики ЭВМ

Процессор	Эльбрус 8С 1,3 Ghz	Intel Core-i7 4570U 1,7 Ghz
Процессорная архитектура	VLIW	X86-64
Количество ядер/потоков	8/8	2/4
Видеоадаптер	Radeon HD 7450 1GB	Intel HD Graphics 4600
Операционная система	Astra Linux Special Edition «Ленинград» 8.1	Astra Linux Special Edition «Смоленск» 1.6

Характеристики твердотельных накопителей и оперативной памяти в таблице 1 не представлены, из-за их малого влияния на результат тестирования ЭВМ.

Для тестирования производительности рабочих станций стенда на базе процессора «Эльбрус-8С» и его сравнения с ЭВМ на базе процессора Intel решено использовать следующие синтетические тесты: coremark; busspeed; dhrystone; linpack; memspeed; mpmflops; whetstone.

Для тестирования производительности ЭВМ стенда было принято решение использовать следующие прикладные программы: postgresql; openssl; apache2; php 7.0; 7-zip; LibreOffice.

Использование данного программного обеспечения позволяет оценить, каким образом будет вести себя ЭВМ при повседневной работе пользователя или при использовании данного ЭВМ в качестве сервера приложений [3].

1. Результаты сравнительного тестирования программой «mpmflops»

Результаты тестирования ЭВМ «Эльбрус» бенчмарком mpmflops было решено разбить по размеру используемых массивов. Обобщённые результаты тестирования представлены на рисунках 1-3.

В данном тесте процессор «Эльбрус 8С» демонстрирует преимущество своей архитектуры. При тестировании с использованием массива меньшей длины процессор «Эльбрус 8С» обгоняет процессор Intel в 3 раза (при использовании режима тестирования с 2 и 8 операндами). При увеличении размера массива, процессор Intel догоняет процессор «Эльбрус 8С». При тестировании ЭВМ «Эльбрус» в режиме 32 операций на малых объёмах данных, его преимущество по сравнению с Intel вырастает до 5 раз. Из этого следует, что для увеличения производительности работы программы на процессоре

«Эльбрус 8С» требуется группировать операции и тщательно подбирать размер обрабатываемых данных одного прохода.



Рис. 1. Результаты теста «trmfloors» с использованием размера массива 102400 элементов



Рис. 2. Результаты теста «trmfloors» с использованием размера массива 1024000 элементов



Рис. 3. Результаты теста «trmfloors» с использованием размера массива 10240000 элементов

2. Результаты сравнительного тестирования с помощью архиватора «7zip»

В данном тесте процессор «Эльбрус 8С» оказывается быстрее процессора Intel. Это можно объяснить большим количеством ядер и возможностью оптимизации под особенности архитектуры VLIW. В задаче архивации процессор «Эльбрус» оказался быстрее на 10%, в задаче разархивирования - на 30%. Из этого можно сделать вывод, что работа с алгоритмами сжатия рабочая станция «Эльбрус» работает достаточно эффективно несмотря на малую частоту.

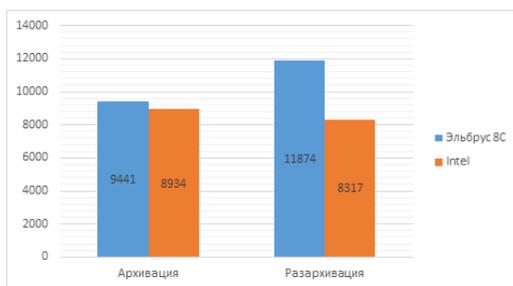


Рис. 4. Результаты тестирования «7zip»

К сожалению остальные тестовые программы и синтетические тесты показали, в целом, отставание российской системы от ее зарубежного аналога. Тестирование прикладных программных продуктов показало, что основной недостаток новой архитектуры VLIW – отсутствие оптимизации уже имеющегося программного кода. Этим объясняется низкая эффективность работы серверов баз данных, интерпретаторов языка, алгоритмов шифрования.

Это происходит, в том числе, из-за самой концепции написания программ. Современные системы требуют современных решений - при написании программ необходимо использовать подход так называемого «параллельного программирования». Это требует определенных усилий и понимания работы компилятора у программиста. Однако, программное обеспечение, оставшееся, как наследство, от одноядерных CISC и RISK систем не дает воспользоваться всей мощью архитектуры VLIW. Совместное использование возможностей компилятора, а также методов представления и преобразования алгоритмов при написании программистом кода даст значительное ускорение работы автоматизированных систем за счет увеличения доли параллельных вычислений и достаточно компактной формы алгоритма.

3. Варианты ускорения вычислительной задачи в автоматизированной системе с параллельной структурой

Любой набор процессов использует конечное количество системных ресурсов. Для каждого элемента множества системных ресурсов имеется ряд факторов, определяющих очередность исполнения того или иного процесса. Зададим алгоритм запуска приоритетных процессов на 11 позиций с помощью графа (см. рис 5).

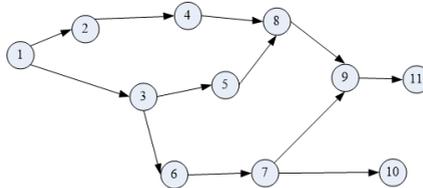


Рис. 5. Граф алгоритма

Данный граф алгоритма можно преобразовать в ярусно – параллельную форму (см. рис 6).

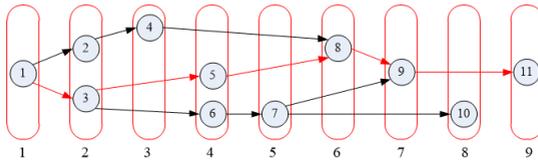


Рис. 6. Граф алгоритма в ярусно – параллельной форме представления

Закон Амдала иллюстрирует, что возможная выгода от использования параллельных вычислений во многом предопределена свойствами применяемых в программах методов и алгоритмов.

Естественно, что максимальное ускорение, которое можно получить при исполнении задач на параллельной вычислительной системе по закону Амдала рассчитывается как

$$S = \frac{1}{p + \frac{1-p}{c}}, \quad (1)$$

- S – коэффициент максимального увеличения скорости работы системы,
- p – доля параллельных операций в рассматриваемой системе,
- c – количество ядер.

При большом количестве ядер графического процессора можно обозначить как

$$S \approx \frac{1}{p}. \quad (2)$$

Т.е. при доле параллельных операций $p=50\%=0,5$, вычисляемых на системе с большим количеством процессоров (или одновременно обрабатываемых параллельных потоков), максимальное ускорение будет всего лишь в 2 раза относительно последовательного варианта расчета. При $p=0,9$ - в 10 раз.

В программном исполнении можно добиться максимального ускорения процесса:

первое – увеличивая ширину на ярусно-параллельной форме (далее – ЯПФ) представления алгоритма и уменьшая ее высоту после расчета высоты канонической ЯПФ (поиск кратчайшего пути – путь 1-3-5-8-9-11 на рисунке 6 и преобразование ЯПФ на рисунке 7). Шириной ЯПФ является количество исполнителей (вычислительных ядер), которые могут быть задействованы. Высота ЯПФ – скорость выполнения алгоритма, т.е. количество ярусов можно представить количеством тактов. Построенная ЯПФ дает максимальную скорость и максимальную «жадность» (максимальное количество исполнителей) без изменения программной структуры самого алгоритма.

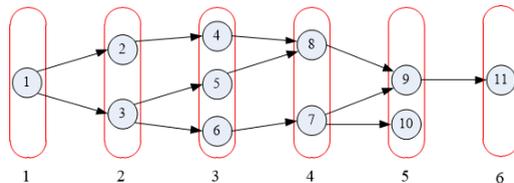


Рис. 7. Преобразованный граф алгоритма в ярусно – параллельной форме представления

и второе – распараллеливанием задач – т.е. преобразовывая отношения между элементами массива. На рисунке 8 сверху схематично изображено шифрование в режиме простой замены, выполненное последовательно за 7 тактов, затем то же самое действие, выполненное параллельно, но за 4 такта.

Если на этапе преобразования ЯПФ назначение на процессорные вычислители может совершаться средствами операционной системы и (или) существующим специальным программным обеспечением с определенным набором библиотек, то на данном этапе необходима

работа программиста для преобразования программного кода выполняемого алгоритма.

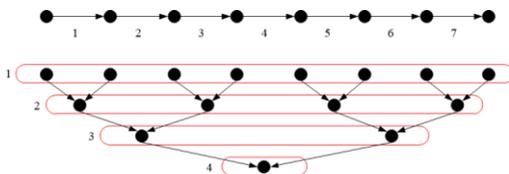


Рис. 8. Эквивалентное преобразование - шифрование в режиме простой замены

Заключение

Сложным многопроцессорным системам, входящих в состав современных комплексов средств автоматизации сегодня приходится одновременно обрабатывать множество различных задач. Многопроцессорность и многопоточность на аппаратном уровне, совмещенная с параллелизмом, реализованном на программном уровне с применением алгоритмов преобразования, позволит отечественным процессорам, основанном на архитектуре VLIW со временем занять одно из лидирующих мест на рынке информационных технологий.

На данный момент тактовая частота процессора Эльбрус ниже, чем у процессоров Intel, но за счёт организации параллелизма и хорошей оптимизации кода, разница в частоте не будет решающим фактором. По пиковой производительности отечественный процессор уже сейчас обгоняет серверный Intel Xeon E5-2609, а по количеству выполняемых команд за один такт превосходит практически любого конкурента

На основе проведенных исследований можно сделать вывод, что в целом автоматизированная система, основанная на процессоре «Эльбрус» и операционной системы Astra Linux «Ленинград» 8.1 уже сейчас способна в ряде задач заменить аналогичную на основе ЭВМ с процессором Intel и ОС Windows. ЭВМ с процессором «Эльбрус» обеспечивает большую надёжность от хакерских атак по сравнению с ЭВМ на основе Intel в связи с использованием собственной защищённой архитектуры и отсутствием известных аппаратных уязвимостей. Операционная система Astra Linux Special Edition «Ленинград» также способна обеспечить защищённость данных на программном уровне с использованием как встроенных СЗИ, так и с СЗИ, поставляемыми сторонними разработчиками. Дополнительная система защиты parsec данной ОС гарантирует защиту от несанкционированного доступа и возможности повышения прав пользователя, а мандатные метки и метки

конфиденциальности обеспечивают защищённость от распространённых атак на аналогичные системы семейства Linux.

Список литературы

1. Постановление правительства РФ от 16.11. 2015 N 1236 «Об установлении запрета на допуск программного обеспечения, происходящего из иностранных государств, для целей осуществления закупок для обеспечения государственных и муниципальных нужд» //Собрание законодательства РФ. – 2015. – №. 47.

2. Sohi, G. S. Multiscalar processors / G. S. Sohi, S. E. Breach, T. N. Vijaykumar // In 25 Years ISCA: Retrospectives and Reprints. – 1998. – P. 521–532.

3. Мишин, А. А. Исследование и сравнительная оценка эффективности архитектуры "Эльбрус" // Инновационные, информационные и коммуникационные технологии. – 2019. – №. 1. – С. 91– 95.

4. ФСТЭК подписал указ об использовании отечественного ПО на объектах критической информационной инфраструктуры [Электронный ресурс] – Режим доступа: <https://www.securitylab.ru/news/512136.php>

5. Приходько, Д. И. Экспертный анализ недостатков и методы их преодоления для архитектуры VLIW // Научные достижения и открытия современной молодежи. – 2018. – С. 14 – 15.

6. Мельников, М. О. Аппаратные уязвимости микропроцессоров архитектуры X86/X86_64 IAARM / М. О.Мельников, Е. В. Игонина // Фундаментально-прикладные проблемы безопасности, живучести, надежности, устойчивости и эффективности систем. – 2019. – С. 128 – 133.

7. Ким, А.К. Микропроцессоры и Вычислительные комплексы семейства «Эльбрус»: учебное пособие / А. К. Ким, В. И. Перекатов, С. Г. Ермаков; СПб. : Питер, 2013. –272с.

8. Большое тестирование процессоров различных архитектур [Электронный ресурс] – Режим доступа: https://habr.com/ru/company/icl_services/blog/501588/