

Эффективность процессоров ARM для расчетов классической молекулярной динамики*

В.П. Никольский¹, В.В. Стегайлов^{2,1}

Национальный исследовательский университет “Высшая школа экономики”¹,
Объединенный институт высоких температур РАН²

Суперкомпьютерные вычисления экзафлопсной эры будут неизбежно ограничены энергоэффективностью. Сегодня в качестве возможных кандидатов для этих целей рассматриваются различные микропроцессорные архитектуры. Недавно микропроцессоры с архитектурой ARM в своем развитии достигли момента, когда уже можно серьезно обсуждать их применение для высокопроизводительных вычислений. В данной работе представлен анализ эффективности последних версий ARM микропроцессоров и их производительности для задач классической молекулярной динамики.

1. Введение

Развитие вычислительной техники последних десятилетий проходило в рамках известного закона Мура, согласно которому количество транзисторов, размещаемых на кристалле интегральной схемы, и их производительность удваиваются каждые полтора-два года. В настоящее время развитие элементной базы подошло к физическим пределам, и дальнейший рост производительности высокопроизводительных вычислительных систем все больше и больше основывается не на росте производительности отдельных интегральных схем, а на создании суперкомпьютеров, объединяющих в одно целое колоссальное число вычислительных элементов [1]. Согласно существующим оценкам суперкомпьютеры экзафлопсной эры будут состоять из миллионов вычислительных элементов. Для создания подобных систем первоочередную важность приобретают вопросы энергопотребления. Акценты в развитии аппаратного обеспечения смещаются с задач создания быстрых вычислительных элементов на задачи создания энергоэффективных вычислительных элементов, из которых можно собирать сверхбольшие системы, и на задачи разработки соответствующего интерконнекта. Характерным примером служит история развития систем серии IBM Blue Gene, первая из которых была основана на заведомо слабом процессоре с тактовой частотой всего 700 МГц [2].

Одним из важных факторов развития вычислительной техники последнего десятилетия является использование в области высокопроизводительных вычислений GPU ускорителей, первоначально ориентированных на массовый сегмент рынка микроэлектроники (ускорители для видеоигр). Привлекательное соотношение “цена-производительность” обеспечило стремительный рост популярности подобного аппаратного обеспечения в области высокопроизводительных вычислений.

Другая сегодняшняя тенденция подобного рода — массовое использование процессоров ARM в маломасштабной вычислительной технике и, особенно, в современных смартфонах и планшетах [3]. Потребности рынка обуславливают рост производительности процессоров ARM при сохранении высокого уровня энергоэффективности. В последнем семействе процессоров ARM Cortex-A имеется возможность использования специального модуля для вычислений с плавающей точкой (VFP), что обуславливает их применимость для широкого спектра задач в области высокопроизводительных вычислений [4]. В данной работе мы анализируем производительность нескольких примеров процессоров семейства ARM Cortex-A, как с точки зрения пиковых характеристик и теста типа Linpack, так и для задач класси-

*Работа была поддержана Программой фундаментальных исследований НИУ ВШЭ.

ческой молекулярной динамики (МД).

На сегодняшний день задачи МД моделирования занимают важное место в числе приложений высокопроизводительных вычислений [5–7]. Подобные модели рассматривались в качестве одного из главных приоритетов при разработке суперкомпьютеров семейства IBM Blue Gene. Для развития алгоритмов параллельного решения математических задач на новейших суперкомпьютерах Департамент энергетики США (DOE) в 2006 г. существенно расширил программу Innovative and Novel Computational Impact on Theory and Experiment (INCITE), в рамках которой распределяются значительные гранты вычислительного времени. На задачи классической молекулярной динамики в рамках этой программы приходится примерно 1/5 часть, как по числу проектов, так и по выделенному вычислительному времени [2].

Наряду с разработкой теоретических основ метода МД моделирования [8] интенсивно растет число его применений в современных многомасштабных моделях в физике, химии, биологии, материаловедении и других областях, существенным образом основанных на описании процессов на атомистическом уровне. При этом даже достигнутый на сегодня рекордный размер моделей в триллионы частиц [9] соответствует, например, для металла при нормальной плотности объема всего в несколько мкм³. Задача увеличения максимальных доступных времен молекулярно-динамических расчетов еще сложнее [10]. Разработка подобных вычислительных методов неразрывно связана с прогрессом в суперкомпьютерных технологиях.

Во второй части описаны аппаратные и программные средства, используемые для тестирования производительности ARM-систем. В третьей части обсуждается реальная производительность на тесте RGBenchMM и пиковая производительность ARM-ядер. В четвертой части воедино сводятся данные тестовой молекулярно-динамической задачи для различных типов процессорных ядер, включая наши результаты для ARM Cortex-A5. В пятой части мы проводим сравнение энергопотребления серверов на процессорах Intel и ARM при решении тестовой молекулярно-динамической задачи.

2. Аппаратное обеспечение и методы тестирования

Архитектура ARM — это микропроцессорная архитектура с сокращенным набором команд (Advanced RISC Machine). Архитектура одного семейства ARM процессоров может подразделяться на различные типы ядер. В данной работе мы рассматриваем семейство Cortex-A и ядра Cortex-A5, Cortex-A9 и Cortex-A15. Данные типы ядер не соответствуют строго определенным микросхемам, а представляют собой шаблоны с набором опций, по лицензии на которые микропроцессоры выпускаются различными производителями. В некоторых случаях шаблоны подвергаются определенным видоизменениям (например, ядра Scorpion и Krait). Рассматриваемые типы ядер включают модуль для вычислений с плавающей точкой (VFPv3 или VFPv4).

Основные результаты работы были получены на ODROID-C1. ODROID-C1 — это компактный миникомпьютер производства фирмы Hardkernel, который содержит четырехъядерный процессор Amlogic S805 Cortex-A5 с модулями VFPv4 на каждом ядре, а также графическим ускорителем Mali-450 MP2, который нами не использовался. Процессор функционирует на частоте 1.5 ГГц с 1 гигабайтом памяти DDR3 SDRAM. Устройство имеет слоты eMMC и microSD, 4 USB порта и 10/100/1000 Mbps Ethernet с портом RJ-45. Питание подключается через специальный адаптер. Монитор подключается посредством microHDMI порта. ODROID-C1 использовался нами или под управлением ОС Linux Ubuntu 14.04.1 LTS (odroid 3.10.67-55 #1 SMP) с легковесной графической средой LXDE, или под управлением ОС Android версии 4.4.2. Для тестов производительности других типов ядер ARM Cortex использовались несколько различных смартфонов под управлением ОС Android. Анализ энергопотребления миникомпьютера ODROID-C1 проводился с помощью цифрового ватт-

метра “Smart Power” производства фирмы Hardkernel.

Кроме того, для анализа привлекались опубликованные ранее результаты тестов производительности и энергопотребления.

Тесты производительности микропроцессоров имеют продолжительную историю развития [11]. В области научных расчетов и математического моделирования de facto стандартом для описания производительности являются единицы Флопс (число операций с плавающей точкой в секунду). Измерение данной характеристики обычно связывается с тестом Linpack. Однако процессоры с архитектурой ARM первоначально ориентировались на целочисленные операции (см. [12] и ссылки в ней) их пиковая производительность в единицах Флопс не декларируется производителем (но, как показано ниже, может быть оценена по результатам специальных тестов).

Для тестов производительности алгоритмов классической молекулярной динамики использовался пакет LAMMPS и модель Леннард-Джонсовской жидкости (32 тыс. атомов при плотности $0.8442\sigma^{-3}$, обрезка Леннард-Джонсовского потенциала на расстоянии 2.5σ (в среднем 55 соседей на атом), 100 шагов по времени в NVE схеме интегрирования).

Для компиляции LAMMPS в ОС Linux использовался компилятор GCC версии 4.9. Для компиляции использовались ключи -mcpu=cortex-a5 -mfpu=vfpv4-d16 -mfloat-abi=softfp (отличия по производительности от случая -mfloat-abi=hard оказались минимальными). Использовалась последовательная компиляция для запуска на одном ядре Cortex-A5 (без многопоточности).

Таблица 1. Результаты теста RGBenchMM на различных устройствах с процессорами ARM

ARM-ядро	R_{max} , МФлопс/ядро	Флоп/такт	Источник
Cortex-A5	284	0.2	ODROID-C1
Cortex-A8	66	0.1	[16] (Google Nexus S)
Cortex-A9	372	0.3	Google Nexus 7
Cortex-A9	460, 614	0.4	Samsung Galaxy S II X, Note II
Cortex-A9	352 - 655	0.2 - 0.4	[16] (8 устройств)
Scorpion	575, 588	0.4	Samsung Galaxy S Plus, S II
Scorpion	446 - 772	0.4 - 0.5	[16] (4 устройства)
Krait	613 - 896	0.4 - 0.5	[16] (14 устройств)
Krait 400	1073	0.5	Sony Xperia Z2
Krait 400	1038 - 1197	0.5	[16] (11 устройств)
Cortex-A15	1125	0.7	Samsung Nexus 10
Cortex-A15	1164 - 1502	0.6 - 0.7	[16] (4 устройства)

3. Оценка пиковой производительности ARM-ядер

В отличие от процессоров таких фирм, как, например, Intel, AMD и IBM, разработчики и производители процессоров ARM не публикуют данные по пиковой производительности R_{peak} в единицах Флопс. Согласно существующим исследованиям [13] ядро Cortex-A9 использует 1-2 такта на выполнение одной операции с плавающей точкой (Флопа), а ядро Cortex-A15 выполняет все типы подобных операций за 1 такт микропроцессора, включая операции fused multiply-add (FMA) или fused multiply-accumulate (FMAC).

Для ОС Android доступно тестовое приложение RGBenchMM [14], основанное на реализации процедуры типа DGEMM с возможностью распараллеливания на 2 и 4 потока. Данное приложение написано на C++ в Android NDK и лишено присущей Java программам в ОС Android заниженной производительности, в частности, исполняемый код в ключевых операциях использует специальные инструкции FMAC [15].

В таблице 1 представлены данные по абсолютной производительности R_{max} в МФлопс на одно ядро и по относительной производительности (по отношению к тактовой частоте процессора) в единицах Флоп на один такт процессора.

Из таблицы видно, что, в отличие от приведенных выше данных [13] по пиковой производительности полученных в специализированных тестах, производительность рассматриваемых ядер Cortex-A9 и Cortex-A15 на приближенном к реальной задаче тесте RGBenchMM примерно в 2 раза меньше. Данное отличие связано с тем, что в реальных вычислениях значительное число тактов процессора уходит на загрузку данных из памяти в регистры. Мы видим, что подобные эффекты еще сильнее выражены для более “слабых” ядер Cortex-A5 и A8. Исходя из приведенных данных, представляется разумным считать пиковую производительность ядра Cortex-A5 равной 2-3 тактам на Флоп с учетом операций FMAC и 4-6 тактам на Флоп без учета FMAC.

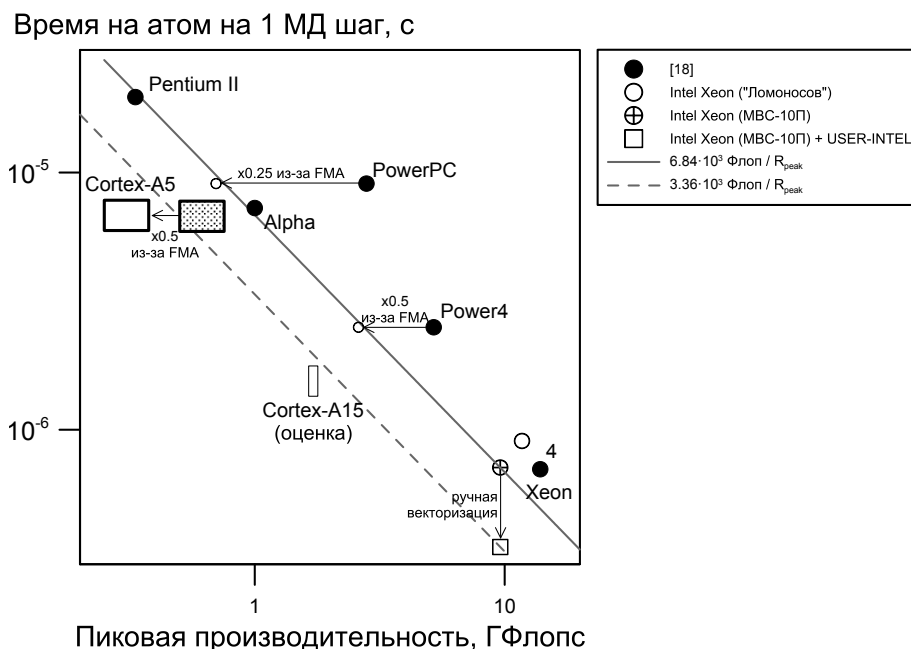


Рис. 1. Сравнение быстродействия пакета LAMMPS для модели Леннард-Джонсовской жидкости на различных процессорах (последовательный код без многопоточности).

4. Сравнение ядер по эффективности расчетов LAMMPS

С точки зрения конечного потребителя аппаратного обеспечения для высокопроизводительных вычислений различные типы процессоров отличаются производительностью R_{peak} в единицах Флопс. Вообще говоря, конечный потребитель ожидает, что время решения задачи будет уменьшаться как

$$t \sim \frac{1}{R_{peak}}. \quad (1)$$

В случае параллельных вычислений возможность уменьшения времени расчета ограничена коммуникацией. Однако и в случае последовательных вычислений соотношение (1) может не выполняться для конкретных задач при переходе от одного процессора к другому из-за разной скорости загрузки данных, других факторов (кэш, компилятор, ОС), а также из-за использования операций специального вида, как, например, FMA [17].

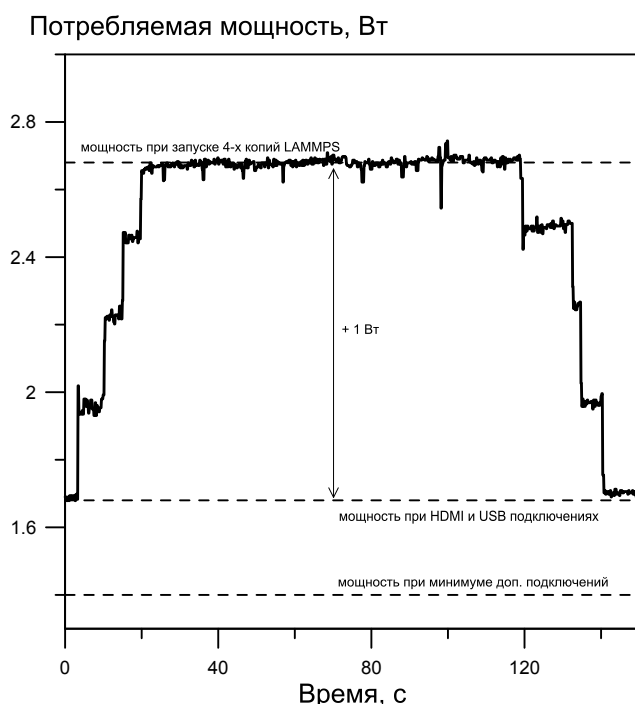


Рис. 2. Потребляемая миникомпьютером ODR0ID-C1 мощность при запуске 4-х копий LAMMPS с тестовой МД задачей. Запуск проводился при подключении монитора по microHDMI и клавиатуры и мыши по USB. Показан уровень энергопотребления незагруженной системы без указанных внешних подключений.

Для сопоставления различных микропроцессоров (и гетерогенных вычислительных элементов) между собой необходимо найти разумный “общий знаменатель”. Эту роль, естественным образом, может играть пиковая производительность R_{peak} [17]. На рис. 1 показаны времена расчета с помощью LAMMPS на 1 атом и на 1 МД шаг для модели Леннард-Джонсовской жидкости. Приведены данные тестов для 1 ядра (черные кружки) с сайта LAMMPS [18] для процессоров Intel Pentium II Over-Drive 333 МГц, DEC Alpha 500 МГц, PowerPC 440 700 МГц, Power4 1.3 ГГц и Intel Xeon 3.47 ГГц. В данном случае время расчета определяется устройством ядра микропроцессора и способностью компилятора создать эффективный исполняемый код. Все точки (кроме квадрата) соответствуют одному и тому же программному коду LAMMPS на C++, но разным типам и версиям компиляторов. Видно, что результаты для процессоров Intel и DEC хорошо соответствуют соотношению (1). Если

значения пиковой производительности для процессоров IBM отмасштабировать, чтобы исключить из R_{peak} операции FMA, не использующиеся в классическом МД алгоритме, то соответствующие точки также прекрасно ложатся на общую зависимость $6.84 \cdot 10^3$ Флоп/ R_{peak} (сплошная прямая на рис. 1).

Показаны результаты тестов [17] на суперкомпьютере МГУ им. М. В. Ломоносова “Ломоносов” без векторизации кода (открытый кружок) и на суперкомпьютере МСЦ РАН МВС-10П без векторизации (кружок с крестом) и с векторизацией кода (квадрат) с использованием модуля USER-INTEL. В этом случае LAMMPS был скомпилирован Intel C++ с учетом ручной векторизации алгоритма подсчета сил, ориентированного на микроархитектуру Intel Xeon (пакет USER-INTEL в LAMMPS). Результатом подобной оптимизации кода является ускорение расчетов в 2 раза, соответствующее $3.36 \cdot 10^3$ Флоп/ R_{peak} (пунктирная прямая на рис. 1).

В отличие от процессоров других фирм для ARM Cortex-A5 значение R_{peak} не декларируется производителем. Прямоугольники на рис. 1 показывают результаты тестов LAMMPS на ARM Cortex-A5 с учетом не полной определенности его пиковой производительности, с учетом поправки на операции FMA и с учетом отличия времен расчета для случая 1-ой копии LAMMPS, запущенной на процессоре (5.95 мкс/атом/шаг), и случая 4-х одновременно работающих копий на 4-х ядрах ARM Cortex-A5 (7.8 мкс/атом/шаг).

Видно, что архитектура Cortex-A5 совместно с компилятором GCC 4.9 обеспечивает высокую эффективность использования аппаратных ресурсов для операций с плавающей точкой. При скромных абсолютных значениях производительности соотношение времени расчета и пиковой производительности ядра процессора находится на уровне результатов Intel Xeon с ручной векторизацией. Однако в случае Cortex-A5 для ее достижения не потребовалось доработка программного кода.

Для ядра Cortex-A15 значение пиковой производительности хорошо определено [13]. Оценку времени расчета LAMMPS на системе, аналогичной Samsung Nexus 10, можно сделать по результатам для ODROID-C1, исходя из их соотношения R_{max} (см. таблицу 1). Рис. 1 показывает, что для Cortex-A15 можно ожидать производительности LAMMPS несколько большей, чем у процессора IBM Power4.

Таблица 2. Потребление энергии при МД расчете Леннард-Джонсовской жидкости (на 1 атом на 1 МД шаг интегрирования). Приведены значения потребляемой мощности сервера/миникомпьютера при полной загрузке всех 4-х ядер. Используются литературные данные по энергопотреблению серверов на базе процессоров Intel Xeon с архитектурами Ivy Bridge и Haswell.

Система	Потребляемая мощность, Вт	Время на атом на шаг, мкс	Энергия на атом на шаг, мкДж
Ivy Bridge Server	316 [19]	0.053	16.8
Ivy Bridge Server	451.8 [20]	0.029	13.1
Haswell Server	462.9 [20]	0.027	12.5
ODROID-C1	2.4	1.75	4.2
Аналог Samsung Nexus 10	~ 8 [21]	0.4	3.2

5. Сравнение энергоэффективности

Сравнение различных вычислительных систем по энергоэффективности сопряжено с существенными трудностями. Зачастую делаются попытки характеризовать энергопотребление процессора. Но оно существенно зависит от режима его работы. По-видимому, разумно рассматривать энергопотребление всей системы (сервера, миникомпьютера, смартфона), не выделяя энергопотребление самого процессора. В данной работе мы провели оценки энергоэффективности работы различных серверов, исходя из условия из полной загрузки МД расчетом Леннард-Джонсовской жидкости.

Рис. 2 показывает как меняется энергопотребление ODROID-C1 при последовательном запуске 4-х независимых расчетов тестовой МД задачи. По приведенным данным уровень энергопотребления при полной загрузке можно оценить как 2.4 Вт (вычитая мощность на HDMI и USB подключения).

По данным работы [21] можно определить значение потребляемой мощности при полной загрузке миникомпьютера ODROID XU+E, основанного на 4-х ядрах Cortex-A15 (наличие также ядер Cortex-A7 приводит к тому, что данная оценка может быть завышена). Имеются литературные данные [19, 20] по энергопотреблению серверов, основанных на процессорах Intel Xeon с архитектурой Ivy Bridge и Haswell. В результате, оказалось возможным оценить количество энергии, требующееся для расчета 1 МД шага на 1 атом для рассматриваемой МД модели простой жидкости (таблица 2). Время расчета для них оценено по универсальной зависимости (рис. 1) и их пиковой производительности без учета FMA (128 ГФлопс для E5-2650 v2, 240 ГФлопс для E5-2690 v2 и 249.6 ГФлопс для E5-2690 v3). Время расчета для ODROID-C1 при использовании всех ядер соответствует рис. 1 (7 мкс/4 ядра). Также приведена оценка для системы с 4-х ядерным процессором архитектуры Cortex-A15.

Таблица 2 свидетельствует о том, что расчеты МД задач на системах с процессорами ARM Cortex-A требуют примерно в 5 раз меньше энергии, чем расчеты на последних моделях процессоров Intel.

6. Заключение

Проанализированы результаты теста RGBenchMM, аналогичного тесту Linpack, для различных систем на основе процессоров с ядрами ARM Cortex-A, включая смартфоны и миникомпьютер ODROID-C1. Сделана оценка пиковой производительности ядра Cortex-A5 в ODROID-C1.

Результаты расчетов с помощью кода LAMMPS тестовой МД задачи проанализированы в контексте производительности других типов процессоров и метрики “время расчета – пиковая производительность”. Показана высокая эффективность использования аппаратных возможностей операций с плавающей точкой в архитектуре Cortex-A в комбинации с компилятором GCC 4.9. Скорость расчета LAMMPS на ядрах ARM Cortex-A в 2 раза выше, чем на процессорах других архитектур с той же пиковой производительностью, причем при отсутствии необходимости по доработке программного кода.

Проведены измерения энергопотребления миникомпьютера ODROID-C1. Сделанные оценки показывают 5-ти кратное преимущество по энергопотреблению систем на основе ядер ARM Cortex-A на рассматриваемой тестовой МД задаче по сравнению с архитектурами Intel Ivy Bridge и Haswell.

Литература

1. Sadovnichy V., Tikhonravov A., Voevodin V., Opanasenko V. “Lomonosov”: supercomputing at Moscow State University // Contemporary High Performance Computing: From Petascale toward Exascale: Vetter J.S. Ed. CRC Press: Boca Raton, FL, 2013. С. 283–307.

2. Стегайлов В.В., Норман Г.Э. Проблемы развития суперкомпьютерной отрасли в России: взгляд пользователя высокопроизводительных систем // Программные системы: теория и приложения: электрон. научн. журн. 2014. Т. 5, N. 1(19). С. 111–152. URL: http://psta.psiras.ru/read/psta2014_1_111-152.pdf (дата обращения: 14.06.2015).
3. An Interview with Steve Furber: URL: <http://cacm.acm.org/magazines/2011/5/107684-an-interview-with-steve-furber/fulltext> (дата обращения: 31.07.2015).
4. Mitra G., Johnston B., Rendell A.P., McCreath E., Zhou J. Use of SIMD Vector Operations to Accelerate Application Code Performance on Low-Powered ARM and Intel Platforms // 2013 IEEE 27th International Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW). May 2013. P. 1107–1116. DOI: 10.1109/IPDPSW.2013.207.
5. Янилкин А.В., Жилиев П.А., Куксин А.Ю., Норман Г.Э., Писарев В.В., Стегайлов В.В. Применение суперкомпьютеров для молекулярно-динамического моделирования процессов в конденсированных средах // Вычислительные методы и программирование. 2010. Т. 11, С. 111–116.
6. Жилиев П.А., Стегайлов В.В. Ab initio молекулярная динамика: перспективы использования многопроцессорных и гибридных суперЭВМ // Вычислительные методы и программирование. 2012. Т. 13, С. 37–45.
7. Куксин А.Ю., Ланкин А.В., Морозов И.В., Норман Г.Э., Орехов Н.Д., Писарев В.В., Смирнов Г.С., Стариков С.В., Стегайлов В.В., Тимофеев А.В. ЗАЧЕМ и КАКИЕ нужны суперкомпьютеры эксафлопсного класса? Предсказательное моделирование свойств и многомасштабных процессов в материаловедении // Программные системы: теория и приложения: электрон. научн. журн. 2014. Т. 5, N. 1(19). С. 191–244. URL: http://psta.psiras.ru/read/psta2014_1_191-244.pdf (дата обращения: 14.06.2015).
8. Норман Г.Э., Стегайлов В.В. Стохастическая теория метода классической молекулярной динамики // Математическое моделирование. 2012. Т. 24, N. 6. С. 3–44.
9. Eckhardt W. et al. 591 TFLOPS multi-trillion particles simulation on SuperMUC // Supercomputing. Vol. 7905, Springer Berlin Heidelberg, 2013. P. 1–12.
10. Подрыга В.О., Поляков С.В., Пузырьков Д.В. Суперкомпьютерное молекулярное моделирование термодинамического равновесия в микросистемах газ-металл // Вычислительные методы и программирование. 2015. Т. 16, N. 1. С. 123–138.
11. Curnow H.J., Wichmann B.A. A synthetic benchmark // Computer Journal. 1976. Vol. 19, N. 1. P. 43–49.
12. Козлов-Кононов Д. Процессорные ядра семейства Cortex. Сочетание высокой производительности и низкого энергопотребления // Электроника: НТБ. 2010. N. 8. P. 16–24. URL: <http://www.electronics.ru/journal/article/135> (дата обращения: 31.07.2015).
13. Garg R. Exploring the Floating Point Performance of Modern ARM Processors: URL: <http://www.anandtech.com/show/6971/exploring-the-floating-point-performance-of-modern-arm-processors> (дата обращения: 14.06.2015).
14. Garg R. RgbenchMM - Android Apps on Google Play: URL: <https://play.google.com/store/apps/details?id=org.codedivine.rgbench&hl=en> (дата обращения: 14.06.2015).

15. Garg R. Prelim analysis of RgbenchMM:
URL: <http://codedivine.org/2012/09/25/prelim-analysis-rgbenchmm/> (дата обращения: 14.06.2015).
16. Swedroid — Nordens största Android-community: URL: <http://www.swedroid.se/> (дата обращения: 14.06.2015).
17. Смирнов Г.С., Стегайлов В.В. Эффективность алгоритмов классической молекулярной динамики на суперкомпьютерном аппаратном обеспечении // Математическое моделирование (в печати).
18. One-processor timings on the Lennard-Jones liquid benchmark:
URL: http://lammmps.sandia.gov/bench/lj_one.html (дата обращения: 14.06.2015).
19. Supermicro — Ivy Bridge Based DCO SuperServer Power & Cost Savings:
URL: http://www.supermicro.com/white_paper/white_paper_Ivy-Bridge-Power.pdf (дата обращения: 14.06.2015).
20. Intel Xeon E5-2600 V3 Review: Haswell-EP Redefines Fast:
URL: <http://www.tomshardware.com/reviews/intel-xeon-e5-2600-v3-haswell-ep,3932-9.html> (дата обращения: 14.06.2015).
21. Abdurachmanov D. et al. Explorations of the viability of ARM and Xeon Phi for physics processing // Journal of Physics: Conference Series. 2014. Vol. 513, P. 052008.

Efficiency of ARM processors for classical molecular dynamics calculations

Vsevolod Nikolskiy and Vladimir Stegailov

Keywords: ARM architecture, floating point operations, efficiency, molecular dynamics

Supercomputing of the exascale era is inevitably limited by power efficiency. Nowadays different CPU architectures are considered as possible choices for these purposes. Recently the development of ARM processors has come to the point when their floating point performance can be seriously considered for a range of scientific applications. In this talk we present the analysis of the floating point performance of the latest ARM cores and their efficiency for the algorithms of classical molecular dynamics.