



С.В. Золотарев, А.Н. Рыбаков (ЗАО "РТСофт")

Нынешний год обещает стать началом массового применения многоядерных систем, а через два года они станут (по оценке компании Intel) основной платформой в большинстве сегментов рынка - серверах, настольных компьютерах и ноутбуках, а в перспективе – и во встраиваемых системах. В статье рассматриваются возможности и проблемы использования многоядерных систем с точки зрения программного обеспечения.

Многоядерные системы – это путь к повсеместному использованию параллельных вычислений. Как известно, наиболее распространенным подходом является распараллеливание "потока команд" и (или) "потока данных". Распараллеливание данных – это применение одной операции сразу к нескольким элементам массива данных. Параллелизм задач предусматривает разбиение вычислительного процесса на несколько самостоятельных подзадач (процессов, потоков), каждая из которых выполняется на своем ядре (процессоре). Многоядерные системы (в соответствии с классификацией Флинна) можно отнести к так называемым системам класса MIMD (Multiple Instruction – Multiple Data), в которых несколько программных ветвей выполняются одновременно и независимо друг от друга, но в определенные моменты времени обмениваются данными. В статье в основном рассматриваются вопросы, относящиеся к проблематике ПО, возникающей при распараллеливании задач.

Необходимость массового перехода к параллельным вычислениям, обусловленная широким внедрением многоядерных систем, подразумевает существенное изменение стиля программирования приложений. Еще совсем недавно правильным считалось так называемое структурное программирование, в котором даже операция перехода (goto) рассматривалась как "неприемлемая". Переход на многоядерные системы потребует от разработчиков ПО использования совсем уже немислимых (с точки зрения структурного программирования) вещей таких, как создание параллельных потоков, порождение и обработка асинхронных событий и др. Иными словами, новая аппаратная архитектура (многоядерные процессоры) обуславливает смену и программной парадигмы: переход от последовательного стиля программирования к параллельному.

Закон Амдала, или что можно ожидать от перехода на многоядерные системы?

Многоядерные процессоры – это многопроцессорная система на уровне кристалла, одной из целей которой является стремление повысить эффективность вы-

числительной системы в целом. У компании Intel такой подход получил название CMP (chip-level multiprocessing). Как и для любой многопроцессорной системы, можно оценить получаемую эффективность решения на основе закона Амдала (Gene Myron Amdahl, 1967 г.) [1], который гласит, что эффективность S (прирост производительности) системы зависит от двух параметров (числа процессоров N и доли последовательных операций в программе c) и выражается формулой:

$$S \leq 1/(c + (1 - c)/N).$$

Крайние случаи в значениях c соответствуют полностью параллельным программам ($c = 0$) и полностью последовательным ($c = 1$). Если всего лишь 1/10 часть программы выполняется последовательно, то ускорения в 10 раз получить в принципе невозможно вне зависимости от числа используемых процессоров (ядер). Важное следствие из закона Амдала: максимальный прирост производительности (в N раз при N процессорах) в реальности недостижим, так как воз-

можен лишь в случае, когда последовательно исполняемая часть программы равна нулю, что на практике невозможно. Другим важным следствием закона Амдала является вывод о том, что чем меньше доля последовательно исполняемой части программы, тем больше прирост производительности (рис. 1).

Что же мы имеем на практике? В настоящее время только небольшая часть ПО рассчитана на выполнение на многоядерных процессорах, что подтверждается результата-

ми тестирования с помощью как синтетических тестов, так и тестов для конкретных классов приложений [2]. При тестировании реальный рост производительности дают только те программы, которые имеют оптимизацию под многопоточность такие, как Adobe Premiere Pro 1.5 и 3DMax. Очень важны для повышения производительности приложений разработка и внедрение драйверов устройств, поддерживающих многопоточность. Хорошей новостью является информация о том, что компании ATI и nVidia планируют внедрить поддержку многопоточности в драйвера для видеокарт.

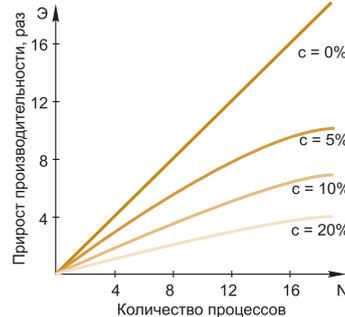


Рис. 1. Прирост производительности в зависимости от числа процессоров и доли последовательных операций

В чем особенности перехода от последовательных к параллельным вычислениям в многоядерных системах?

При переходе с одноядерных процессоров на многоядерные системы приходится принимать во внимание проблему последовательного выполнения. Что это означает применительно к многоядерной архитектуре? В многоядерной системе выполнение считается последовательным, когда одно или более ядер в какой-то момент не могут выполнять код параллельно с другими ядрами. Эта ситуация может возникнуть по разным причинам [3]. Назовем некоторые из них: *ситуации блокировки при доступе к ресурсам, необходимость синхронизации процессов или потоков на различных ядрах, поддержание когерентности кэш-памяти и неравномерность загрузки.*

Блокировки возникают из-за невозможности одновременного доступа приложений на разных ядрах к таким ресурсам, как жесткий диск, некоторые устройства ввода/вывода, прикладные данные в определенных ситуациях (например, в момент "сборки мусора").

Очень часто параллельные процессы, выполняемые на разных ядрах, должны синхронизироваться в определенные моменты времени. Например, приложение на одном из ядер должно использовать промежуточные данные, которые получаются приложением (потоком, процессом) на другом ядре. Первое приложение не может продолжать работу до тех пор, пока не получит эти данные, то есть должно находиться в состоянии ожидания до получения этих данных. В этой ситуации появляются накладные расходы на синхронизацию приложений (процессов, потоков), выполняемых на разных ядрах. Это в свою очередь ведет к понижению эффективности от использования параллельной работы, что находит отражение в так называемом сетевом законе Амдала.

При переходе на многоядерную архитектуру возникает необходимость поддержания когерентности (согласованности) кэш-памяти для всех ядер при использовании разделяемой памяти (shared memory). Поясним проблему. Для ускорения доступа к разделяемой памяти может использоваться кэш-память. Теперь в дополнение к поддержанию когерентности между основной памятью и кэш-памятью каждого отдельного ядра (с помощью методов write through, write back и пр.) необходимо в каждый момент времени поддерживать когерентность основной памяти и кэш-памяти всех ядер, использующих эту разделяемую память при любых операциях чтения-записи. Как правило, эта проблема решается на аппаратном уровне.

В связи с обсуждением проблемы последовательных и параллельных вычислений на многоядерных системах хотелось бы упомянуть об исследованиях компании Intel, направленных на реализацию метода динамического регулирования интенсивности выполнения инструкций (energy per instruction, EPI) в зависимости от степени параллелизма выполняемого ПО [4]. Сейчас впервые исследователи корпорации Intel опытным путем показали эффективность регу-

лирования тактовой частоты в асимметричной мультипроцессорной системе в зависимости от уровня активности вычислительных ядер.

Инструментальные средства для многоядерных систем

Компиляторы

Совершенно очевидно, что для получения максимальной выгоды от использования многоядерной архитектуры требуется поддержка на уровне компилятора. Одной из первых это поняла компания Intel, выпустив в 2005 г. компилятор версии 9.0 для языков программирования C++ и Фортран для ОС Linux и Windows. Этот компилятор позволяет максимальным образом использовать возможности технологии Hyper-Threading и многоядерных процессоров. Компилятор Intel версии 9.0 включает опции автопараллелизма, которые автоматически находят в приложениях возможности для создания множества параллельных потоков и полностью поддерживают реализацию архитектуры OpenMP 2.5.

У Microsoft в Visual C++ 2005 также реализована поддержка параллельной многопоточной обработки путем поддержки стандарта OpenMP. Для этого в Visual C++ 2005 используется либо параметр компилятора /openmp, либо установка в Configuration Properties, C/C++, Language значения OpenMP Support.

С ноября 2005 г. OpenMP поддерживается компилятором gcc (<http://gcc.gnu.org/>) для языков программирования C, C++ и Фортран95 с помощью опции компилятора — fopenmp. Эта поддержка реализована в рамках проекта GOMP (GNU OpenMP, <http://gcc.gnu.org/projects/gomp>). Следует упомянуть также набор компиляторов EKOPath компании PathScale (www.pathscale.com) для 64-разрядных компьютерных систем на базе Linux. EKOPath оптимизирован для архитектур AMD64 и EM64T и поддерживает OpenMP 2.0.

Программные отладчики

Программисты хорошо знают о том, как трудно отлаживать многопоточные приложения. Типичной является ситуация, когда при аварийном завершении программы требуется проанализировать стек вызовов функций во всех потоках, но отладчик показывает только стек потока, на котором произошло аварийное завершение программы. Например, известно, что стандартные средства отладчика gdb плохо приспособлены для отладки многопоточных приложений. Поставщики ОС разрабатывают специальные версии отладчика gdb, ориентированные на использование в конкретной ОС путем включения в ее ядро дополнительных возможностей поддержки отладки многопоточных приложений. Одной из таких реализаций является TotalView компании Etnus — отладчик для платформ на базе Linux, Unix и LynxOS. Отладчик TotalView поддерживает многопоточность, MPI, OpenMP, языки программирования C/C++ и Фортран, а также смешанные коды с использованием различных языков программирования.

Очень полезным средством для оптимизации и отладки параллельных программ является пакет Intel Threading Tools, предназначенный для диагностики ошибок и анализа производительности многопоточных приложений, использующих модели потоков Win32 и OpenMP:

- обнаружение взаимных блокировок (deadlocks) и условий гонок (race conditions) между потоками с помощью Intel Thread Checker;
- использование Intel Thread Checker и компиляторов Intel для локализации проблем на уровне исходного кода;
- анализ эффективности и поиск путей повышения производительности OpenMP-программ с помощью Thread Profiler.

Аппаратные отладчики

Для работы с виртуальными машинами от аппаратного отладчика требуется поддержка ряда специальных функций (в частности, возможность видеть, к какой виртуальной машине относятся те или иные процессы и нити). Такими возможностями обладает аппаратный отладчик TRACE32 фирмы Lauterbach (www.lauterbach.com). Благодаря полной поддержке встроенных аппаратных блоков управления памятью, реализованной в TRACE32, пользователь может одновременно отлаживать несколько процессов в нескольких виртуальных машинах. Можно отлаживать даже два варианта одного и того же процесса, запущенного в разных виртуальных машинах. В частности, компания Lauterbach объявила о выпуске ПО интегрированной поддержки ядра (kernel awareness) для ОС LynxOS-178 компании LynuxWorks. Чтобы иметь доступ ко всем функциям TRACE32, не нужно производить никаких изменений в прикладных программах или ядре (применять патчи, перехватчики, инструментальные "довески" и др.). В результате отлаживается именно то приложение, которое будет присутствовать в конечном продукте, что очень важно для его сертификации.

Следует упомянуть и другие аппаратные отладчики, поддерживающие отладку многоядерных конфигураций для различных целевых платформ и ОС: Green Hills Probe и SuperTrace от компании Green Hills (www.ghs.com/products/probe.html, www.ghs.com/products/supertraceprobe.html), WindPower ICE компании Wind River (http://www.autosoft.com.cn/auto_english/files/product/tool/windpower_ice.pdf), RealView ICE от компании ARM (<http://www.arm.com/products/DevTools/RVI.html>).

Стандарты взаимодействия прикладных программ для многоядерных систем: MPI, OpenMP и POSIX

При разработке параллельных программ в рамках модели параллелизма задач используются специализированные библиотеки и системы параллельного программирования такие, как PVM, LAM, CHMP и др. Наметилось три основных подхода к реализации

Ключ, которым пользуются, всегда блещит.

Бенджамин Франклин

этих систем, отличающихся методами взаимодействия параллельных задач. Один подход строится на основе концепции обмена сообщениями, второй — на использовании разделяемой памяти. Третий подход на основе стандарта POSIX объединяет оба этих подхода.

Наиболее известным представителем первой группы является спецификация MPI (Message Passing Interface) для языков программирования С и Фортран, первый вариант которой был разработан в 1994 г. Спецификация MPI включает около 200 функций и реализована для множества компиляторов и ОС. Одной из наиболее распространенных реализаций MPI является библиотека MPICH. Кроме того, на рынке представлены несколько коммерческих реализаций MPI, например MPI/Pro производства компании Verari Systems Software (www.verari.com). MPI/Pro оптимизирует время работы параллельных приложений и обеспечивает их масштабируемость, осуществляя балансировку параметров производительности и использования ресурсов. Компания Verari предлагает версии MPI/Pro для различных ОС, включая Windows, Linux, Mac OS X, LynxOS, а также таких интерконнектов, как Gigabit Ethernet, Myrinet и InfiniBand.

Представителем второй группы является спецификация OpenMP (Open specifications for Multi-Processing). Первая версия спецификации OpenMP (www.openmp.org) появилась в 1997 г. и предназначалась для языка программирования Фортран. У истоков OpenMP стояли такие известные компании, как IBM, Intel, Sun и Hewlett-Packard. В 1998 г. появились варианты OpenMP для языков C/C++, и к настоящему моменту последней является версия 2.5. Поддержка спецификации OpenMP есть во всех компиляторах Intel (начиная с шестой версии), в Microsoft C/C++ (начиная с Visual Studio 2005), в GCC. Очень интересной является сама концепция OpenMP — это набор специальных директив компилятору (pragma), библиотечных (API) функций и переменных среды. Наиболее оригинальны здесь pragma-директивы компилятору. Они используются для обозначения областей (regions) в коде и могут выполняться параллельно. Компилятор, поддерживающий OpenMP, преобразует исходный код и вставляет соответствующие вызовы функций для выполнения этих областей кода параллельно.

Третьим широко используемым стандартом, который применяется для реализации параллельных вычислений, является POSIX (Portable Operating System interface for unIX — переносимый интерфейс ОС на уровне исходных текстов). Первое описание POSIX (www.pasc.org) было опубликовано в 1986 г. Основная спецификация разработана как IEEE 1003.1 и одобрена как международный стандарт ISO/IEC 9945-1:1990. С точки зрения организации параллельных вычислений наибольший интерес представляют три части стандарта: 1003.1a (OS Definition), 1003.1b

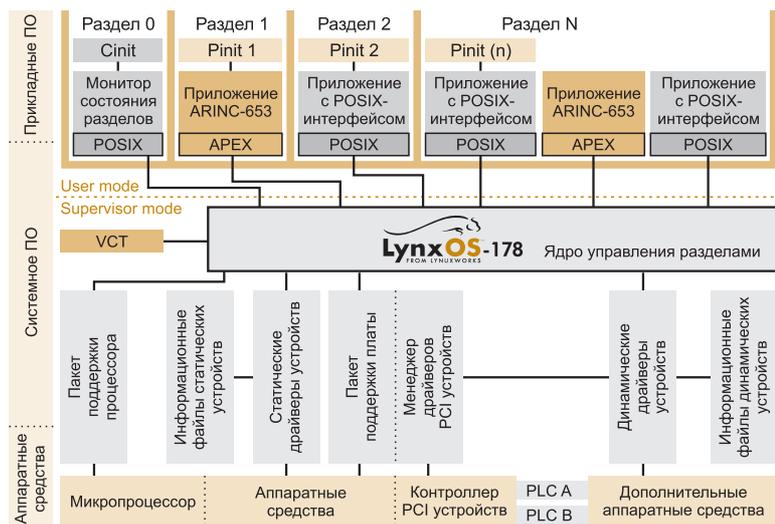


Рис. 2. Реализация ARINC-653 в LynxOS-178

(Realtime Extensions) и 1003.1c (Threads). В рамках POSIX возможно реализовать параллельные вычисления как на основе обмена сообщениями аналогично MPI, так и на основе разделяемой памяти, как в OpenMP. Естественно, в POSIX возможна и любая комбинация этих методов. В настоящее время в наибольшей степени стандарту POSIX соответствуют (сертифицированы) две ОС РВ: LynxOS компании LynuxWorks (www.linuxworks.com) и Integrity компании Green Hills (www.ghs.com).

Мультипроцессорные конфигурации для многоядерных систем и их поддержка на уровне ОС

Многоядерные процессоры требуют от производителей ОС поддержки различных архитектур многопроцессорной обработки таких, как SMP (симметричная MP), AMP (ассиметричная MP) и других. Лидером в этом процессе является компания QNX Software Systems, которая объявила о выпуске комплекта разработчика QNX Momentics Multi-Core Edition – полного набора инструментов разработки и миграции ПО на новое поколение многоядерных аппаратных решений, включая процессоры BCM12xx и BCM14xx компании Broadcom, процессор MPC8641D компании Freescale и многоядерные процессоры компании Intel. Комплект разработчика QNX Momentics Multi-Core Edition будет доступен в первом квартале 2006 г. В этом комплекте будет поддерживаться несколько моделей многопроцессорности для многоядерных архитектур:

- AMP (asymmetric, ассиметричная) – для обеспечения полного управления и отказоустойчивости;
- SMP (symmetric, симметричная) – для максимальной степени параллелизма и масштабируемости;
- BMP (bound, исключительная) – для быстрого осуществления миграции кода и снижения сложности разработки.

QNX Momentics Multi-Core Edition предлагает пользователю широкие возможности при выборе модели многопроцессорности в зависимости от требуемой

функциональности (http://www.qnx.com/news/pr_1760_2.html).

Серьезную поддержку многоядерных систем на базе процессоров Intel, AMD64 и SPARC имеет ОС Solaris 10 компании Sun Microsystems. Например, встроенная система виртуализации и обеспечения защиты информации под общим названием Solaris Containers позволяет системному администратору организовать в рамках единой ОС несколько виртуальных системных разделов, называемых зонами. Каждой такой зоне можно назначить контейнер – набор локализованных системных ресурсов. Контейнеры могут являться основой для управления ресурсами на уровне ядер. Реализованный в рамках Solaris 10 набор функций так называемого прогнозируемого самовосстановления (Predictive Self-Healing)

позволяет автоматически определять сбои в работе ядер и переводить их в пассивный режим без влияния на работу всех остальных ядер процессора.

Поддержка многоядерных систем реализована в различных дистрибутивах Linux, например, в Red Hat Enterprise Linux 4 и Novell/SUSE LINUX Enterprise Server 9. Реализованы SMP и NUMA (Non-Uniform Memory Architecture) модели многопроцессорности. При конфигурировании ядра Linux можно задавать признаки, определяющие работу планировщика, возможность совместного использования кэша и др.

Наконец, надо упомянуть о поддержке многоядерных систем семейством ОС Windows. Детальная информация о поддержке различными версиями Windows многопроцессорной и многоядерной технологии приведена на сайте http://vpr.ocs.ru/microsoft_windows_x64_edition.html.

Многоуровневая виртуализация

Появление многоядерных процессоров даст мощный дополнительный толчок массовому внедрению технологий виртуализации на различных уровнях. В последнее время даже в рамках одноядерных и однопроцессорных систем быстро развиваются технологии и средства виртуализации. Назовем некоторые из разработанных и используемых подходов.

- *Стандарт ARINC-653* (Avionics Application Software Standard Interface) разработан компанией ARINC (Aeronautical Radio, Inc., www.arinc.com) в 1997 г. Стандарт вводит концепцию изолированных разделов на основе универсального программного интерфейса APEX (Application/Executive) между ОС и прикладным ПО (<https://www.arinc.com/cf/store/documentlist.cfm>). Требования интерфейса между прикладным ПО и сервисами ОС определяются таким образом, чтобы разрешить прикладному ПО контролировать диспетчеризацию, связь и состояние внутренних обрабатываемых элементов. В 2003 г. принята новая редакция. ARINC 653 в качестве одного из основных требований

Dual Core – многоядерная революция!

Будущее
принадлежит
процессорам
с несколькими
ядрами.



Самая полная в России линия высокопроизводительных встраиваемых платформ на основе Intel Core Duo.

Преимущества новых платформ от РТСофт:

- Лучшая производительность на ватт потребляемой мощности.
- Существенная экономия средств: запуск нескольких одноядерных приложений на одной платформе.
- Повышенная безопасность благодаря возможности изоляции приложений, исполняемых в разных ядрах.
- Высокая степень интеграции передовых технологий: PCI Express, GB Ethernet, USB 2.0, SATA, 3D графика.

powered by Embedded
Dual Core Processor

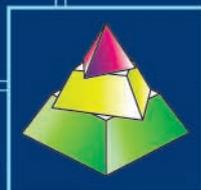
 **kontron**


СРЕДСТВА И СИСТЕМЫ АВТОМАТИЗАЦИИ

105037, Москва, Никитинская 3, 3АО «РТСофт»
Тел.: (495) 742-6828, 967-1505
Факс: (495) 742-6829
e-mail: rtsoft@rtsoft.msk.ru; www.rtsoft.ru



VII международная специализированная выставка
International specialized exhibition



АВТОМАТИЗАЦИЯ 2006 automation

14-17 / 11 / 2006

санкт-петербург
saint petersburg

ТЕМАТИЧЕСКИЕ НАПРАВЛЕНИЯ ВЫСТАВКИ:

- Информационные технологии в промышленности.
- Автоматизация управления предприятием.
- Автоматизация управления производством и производственной инфраструктурой.
- Промышленная автоматизация. АСУ ТП.
- Автоматизация проектно-конструкторской деятельности.

МЕСТО ПРОВЕДЕНИЯ ВЫСТАВКИ:

Петербургский СКК, пр.Ю.Гагарина, 8
метро "Парк Победы"
Т./ф.: (812) 718 35 37, 378 00 67
e-mail: ais@orticon.com
www.farexpo.ru/ais

ЗАКАЗ ПРИГЛАСИТЕЛЬНЫХ БИЛЕТОВ:

www.farexpo.ru/ais/invitation
e-mail: ais1@orticon.com

ОРГАНИЗАТОРЫ:



ВАС ПРИГЛАШАЮТ:



для ОСРВ в авиации вводит архитектуру изолированных виртуальных машин (partition, разделов), как это показано на рис. 2. Стандарт ARINC-653 реализован в различных ОС РВ таких, как LynxOS-178 компании LynuxWorks, VxWorks AE-653 компании WindRiver, Integrity-178 компании Green Hills, CsLeos компании BAE и др. Особенностью реализации концепции виртуальных машин в ARINC-653 является жесткое и заранее определенное квантование времени между виртуальными машинами. Основная цель введения концепции виртуальных машин в ARINC-653 – гарантировать отсутствие общего отказа системы.

- *User Mode Linux (UML)* – Linux в пользовательском режиме. UML – это самый универсальный эмулятор. Он работает не совсем так, как традиционные эмуляторы аппаратного обеспечения, и позволяет создавать виртуальные машины, имеющие оборудование, которого может и не быть на компьютере. Это очень удобно для тестирования различных конфигураций аппаратного обеспечения, потому что не требует физического присутствия аппаратных средств. UML состоит из набора патчей к ядру Linux, которые позволяют запускать другие ОС в консольных окнах. Поэтому каждый пользователь может загружать у себя сколько угодно ОС независимо от других пользователей. Процесс виртуализации настолько полный, что на виртуальных системах можно запускать даже X11. До последнего времени User-Mode Linux можно было применять лишь для устройств с архитектурой x86 и Intel IA-32. Специалисты LynuxWorks изменили UML таким образом, чтобы обеспечить поддержку ядра PowerPC G5. Новая версия User-Mode Linux хороша для перехода на перспективные 64-разрядные архитектуры и новые чипы класса PowerPC 970.

- *Программные среды (системы) виртуальных машин.* Часто программные продукты этого класса называют *эмуляторами ОС*. На сегодняшний день двумя самыми популярными программными средами виртуальных машин являются Microsoft Virtual PC (www.microsoft.com) и группа продуктов VMware (www.vmware.com) компании VMware Inc. Система виртуальных машин позволяет запускать на компьютере одновременно несколько разных ОС и переключаться из одной ОС в другую путем перехода из одного окна в другое без перезапуска компьютера. Суть ПО системы виртуальных машин в том, что на компьютере, работающем под управлением основной (базовой) ОС, создаются один или несколько виртуальных компьютеров, на каждом из которых можно запустить собственную "гостевую" ОС.

VMWare Workstation позволяет запускать несколько ОС Microsoft Windows, Linux и Novell NetWare. Реализована полноценная поддержка сети, портируемость окружений и гибкий подход к работе с созданными окружениями.

Проект *Virtual PC*, изначально разрабатываемый компанией Connectix, в 2003 г. был куплен компанией Microsoft. К сожалению, после этого Virtual PC ли-

шился поддержки "гостевых" Unix-подобных систем (в том числе Linux) и оказался полностью ориентирован только на установку Windows-систем на другие платформы.

- *Технология виртуализации Intel (VT)* является компонентой многоядерной технологии и *обеспечивает поддержку виртуализации на аппаратном уровне* [5]. По своей сути VT – это аппаратная и платформенная поддержка виртуализации. В отличие от известных программных виртуальных машин, VT обеспечивает поддержку виртуальных машин на уровне процессора, повышая как надежность и производительность работы приложений, так и общую безопасность.

Архитектурой VT поддерживаются два принципиальных класса ПО: монитор виртуальной машины (VMM) и "гостевое" ПО. Поддержка виртуализации на уровне процессора обеспечивается новым режимом работы чипа, которая, в терминологии Intel, называется работой в режиме VMX. Режим VMX поддерживается 10 новыми командами: VMPTLDR, VMPTLST, VMPCLEAR, VMPCREAD, VMPCWRITE, VMPCCALL, VMPLAUNCH, VMPCRESUME, VMPCXOFF и VMPCXON. Выделяются два типа работы в режиме VMX: VMX root operation и VMX non-root operation. Как правило, монитор виртуальной машины VMM работает в режиме root, а "гостевое" ПО – в режиме non-root. Технология виртуализации Intel будет поддерживаться различными производителями ОС, например компаниями-поставщиками Linux – RedHat, SuSe и MontaVista, а также в других программных средствах виртуализации, например в VMWare.

Проблемы лицензирования ПО для многоядерных процессоров

Появление многоядерных процессоров вызвало у пользователей опасение, что им придется платить в многократном (пропорциональном числу ядер) размере за лицензии ПО, и в том числе за ОС. В частности, такую позицию занимает компания Oracle. Хотя в последнее время (в конце декабря 2005 г.) произошли изменения в лучшую сторону, стоимость лицензий Oracle на многоядерные процессоры AMD и Intel рассчитывается с умножением числа ядер на коэффициент 0,5, а для чипов Sun UltraSparc T1 (так называемых Niagara) на 0,25. При этом для многоядерных процессоров других производителей (в том числе IBM) коэффициент остался прежним – 0,75.

Однако Intel придерживается другого мнения: так как многоядерные процессоры – это развитие технологии Intel HT, то и лицензирование ПО для многоядерных процессоров должно проводиться по принципам, которые уже были согласованы и применены к технологии HT. К настоящему времени компании Microsoft, Sun, IBM, а также различные поставщики Linux обещали, что для работы на многоядерных процессорах будет достаточно одной лицензии на их программные продукты. В апреле 2005 г. IBM изменила свою политику цен и теперь считает двухъядер-

ные процессоры AMD и Intel одним процессором, однако продолжает рассматривать каждое ядро своих собственных процессоров Power как отдельный процессор. Дополнительную информацию и подтверждение пользователи должны будут получать у поставщиков своих ОС, БД и другого ПО.

Заключение

Насколько востребованными и эффективными будут новые многоядерные системы в различных сегментах рынка? Нет сомнений, что многое в распространении многоядерных систем будет зависеть от реализации ПО: если оно будет удобным и понятным, многие не прочь будут воспользоваться полезными во многих случаях вычислительными ресурсами современных многоядерных процессоров.

Хорошо понимая сложность стоящих задач в связи с массовым переходом на многоядерные решения, основные поставщики аппаратных и программных средств организовали ассоциацию Multicore (www.multicore-association.org), целью которой является разработка промышленных стандартов для многоядерных систем. Сейчас в рамках ассоциации работают над четырь-

мя самостоятельными, но взаимосвязанными стандартами для многоядерных систем: Resource Management (RAPI), Communication API (CAPI), Debug API и Transparent Interprocess Communication (TIPC). Проведены рабочие встречи, в которых приняли участие крупнейшие ИТ-компании такие, как Xilinx, Express Logic, Wind River, Freescale, ARC Int., MIPS Technologies, Synopsys, PolyCore Software и др.

Ближайшее время покажет, насколько успешным будет развитие и внедрение многоядерных систем.

Список литературы

1. *Седых В.* Бой не в своей категории // Экспресс-Электроника. 2005. № 7.
2. *Митрофанов А.* Двухъядерные процессоры. <http://www.3dnews.ru/cpu/dualcore-cpu/index03.htm>.
3. *Todd Brian.* Putting multicore processing in context: Part one. Embedded.com, February 2006.
4. *Murali Annavaram, Ed Grochowski, John Shen.* Mitigating Amdahl's Law Through EPI Throttling. www.princeton.edu/~jdonald/research/cmp/annavaram_mitigating.pdf.
5. *Narendar B. Sahgal, Dion Rodgers.* Understanding Intel® Virtualization Technology (VT). http://download.microsoft.com/download/9/8/f/98f3fe47-dfc3-4e74-92a3-088782200fe7/TWAR05015_WinHEC05.ppt.

Золотарев Сергей Викторович – канд. техн. наук, руководитель направления системного ПО,
Рыбаков Алексей Николаевич – технический директор ЗАО "РТСофт".

Контактный телефон: (495) 742-68-28, 967-15-05.
[Http://www.rtssoft.ru](http://www.rtssoft.ru) E-mail: pr@rtssoft.ru

МОНИТОРИНГ РАБОТЫ SCADA-СИСТЕМ ДЛЯ QNX ЧЕРЕЗ INTERNET

П.Н. Кабанов, А.В. Фрейдман (Научный центр "Науцилус")

Проблемы системной интеграции, требующие взаимодействия систем, работающих в ОС РВ QNX, с удаленными системами под управлением других ОС, прежде всего со SCADA-системами под MS Windows, успешно решаются с помощью новых разработок Научного центра "Науцилус", использующих протоколы OPC DA/HDA/AE и XML. Удачным примером подобного рода может служить интеграция системы диспетчерского управления магистральным нефтепроводом "Уса-Ухта-Ярославль" (г.Ухта), использующей SCADA-пакет RealFlex для QNX, с центральной диспетчерской компании Транснефть в Москве, работающей со SCADA-пакетом Genesis32 для MS Windows.

Введение

SCADA-системы, функционирующие под управлением ОС РВ QNX, развиваются уже довольно длительное время [1]. Привлекательность таких систем состоит в малом времени отклика, предельной производительности, высокой надежности, устойчивости и малом потреблении ресурсов. Важной чертой таких решений является возможность сочетать в одном алгоритме функций мониторинга и управления РВ.

Отличительные характеристики и преимущества SCADA-пакета Phocus 4, как наиболее развитой SCADA-системы для QNX4 и QNX6, подробно описаны в [2].

Чтобы обеспечить взаимодействие QNX-систем сбора данных и диспетчерского управления с многочисленными управляющими и производственными системами, использующие другие ОС, в Научном центре "Науцилус" были разработаны различные средства сетевой передачи данных, использующие

протоколы высокого уровня. Уже довольно давно был создан ODBC Bridge, обеспечивающий передачу данных между SCADA-пакетом RealFlex и реляционными БД под управлением MS Windows, которые широко используются в системах информационного управления производством.

Затем в компании Науцилус были разработаны OPC серверы для QNX систем, поддерживающие протокол OPC Data Access (DA) и обеспечивающие сетевую передачу данных РВ между системами QNX и MS Windows через Ethernet-TCP/IP. Таким образом, возможно через любой OPC-клиент в MS Windows запросить и получить данные из Phocus, RealFlex, ISaGRAF/ISaGRAF Pro и из любого другого QNX приложения, имеющего OPC DA Server.

Недавно OPC DA серверы SCADA-систем для QNX были дополнены серверами OPC HDA и OPC AE, поддерживающими передачу данных предыстории, а также тревог и событий. Кроме того, было со-