

УДК 629.78.018

Т.С. НИКИТИНА, И.Б. ТУРКИН

*Национальный аэрокосмический университет им. Н.Е. Жуковского "ХАИ", Украина***МЕТОД ПЛАНИРОВАНИЯ МУЛЬТИМЕДИЙНЫХ ЗАДАЧ В ОПЕРАЦИОННЫХ СИСТЕМАХ РЕАЛЬНОГО ВРЕМЕНИ ДЛЯ ПЛАТФОРМ НА ОСНОВЕ МНОГОЯДЕРНЫХ ПРОЦЕССОРОВ**

Рассматриваются особенности архитектуры многоядерных процессоров, новые требования к программному обеспечению, использующих все преимущества систем с параллельными вычислениями. Проведен анализ проблемы планирования мультимедийных задач реального времени в системах на основе многоядерных процессоров. Рассмотрен динамический метод планирования мультимедийных задач для систем на основе многоядерных процессоров.

Ключевые слова: мультимедийные приложения, многоядерный процессор, алгоритм планирования, периодические задачи, реальное время.

Введение

На современном этапе переход к многоядерным процессорам (МП) становится основным направлением повышения производительности вычислительных систем. Можно сказать, что идея построения многоядерных микропроцессоров является развитием идеи кластеров, но в данном случае дублируется целиком процессорное ядро.

Другим предшественником многоядерного подхода можно считать технологию Intel – HyperThreading, где также есть небольшое дублирование аппаратуры и использование двух потоков инструкций, использующих общее ядро. Многоядерный процессор имеет два или больше "исполнительных ядер".

Операционная система рассматривает каждое из исполнительных ядер, как дискретный процессор со всеми необходимыми вычислительными ресурсами. Поэтому многоядерная архитектура процессора, при поддержке соответствующего программного обеспечения, осуществляет полностью параллельное выполнение нескольких программных потоков. К 2009 году все ведущие разработчики микропроцессоров создали многоядерные процессоры.

Многоядерная архитектура процессоров диктует новые требования к ПО, вызванное необходимости переходить на новые технологии программирования, использующих все преимущества систем с параллельными вычислениями.

Даже уже известные методы многопоточкового программирования не могут в полной мере использоваться в силу аппаратных отличий МП.

Анализ публикаций показывает, что на совре-

менном этапе средства разработки параллельного ПО для систем на основе МП только начинают развиваться, а также требуют от программистов понимания ряда сложных методик и условий в области параллельных вычислений.

Среди основных отличий МП от многопроцессорных систем – наличие общего ресурса в виде кэш-памяти L2, разделяемой ядрами процессора [1] (рис. 1, а).

Такую базовую архитектуру МП поддерживают компании Intel, AMD, IBM, Sun Microsystems. Также известна архитектура МП Intel Xeon 7000, которая поддерживают технологию Hyper-Threading, благодаря чему способны выполнять 4 вычислительных потока одновременно на 2 ядрах.

Кроме того, они оснащены кэш-памятью третьего уровня объемом 16 МБ, используемой обоими ядрами (рис. 1, б). Процессор UltraSparc T1, содержит 8 ядер, собранных в общую серверную SMP-архитектуру (рис. 1, в).

Проблема общей кэш-памяти является центральной в задаче организации эффективных параллельных вычислений на основе МП, поскольку существует возможность конфликтов во время доступа к общей кэш-памяти [2].

Рассмотрим суть проблемы доступа к общей кэш-памяти на примере. Пусть четыре отдельные задачи параллельно выполняются на 4 ядрах процессора и им выделен 1 квант времени. Как видно на рис. 2, задачи не получают равномерный доступ к кэш-памяти.

Планировщик как ранее не может спрогнозировать равноправное планирование, так как кэш уже не является отдельным ресурсом, что приводит к блокировке одних задач другими.

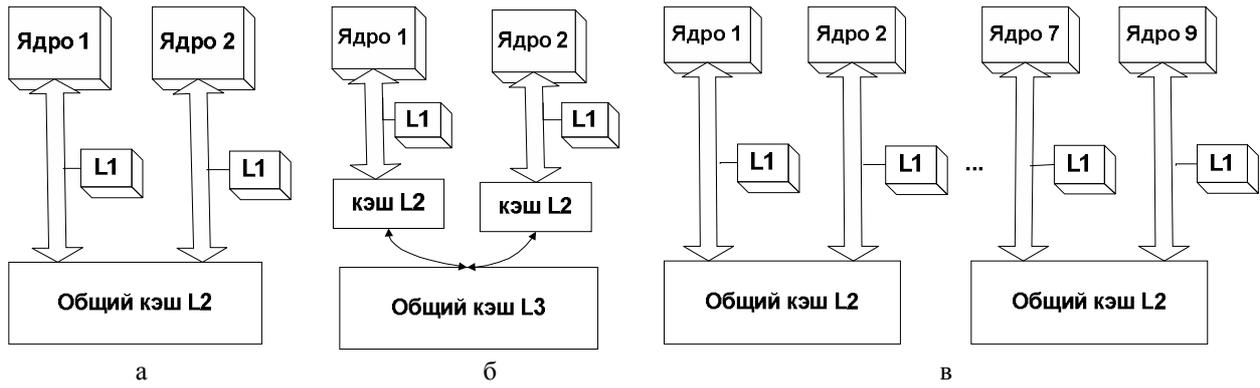


Рис. 1. Кэш-память в различных архитектурах МП:
 а – базовая архитектура МП; б – архитектура МП Intel Xeon 7000; в – МП UltraSparc T1

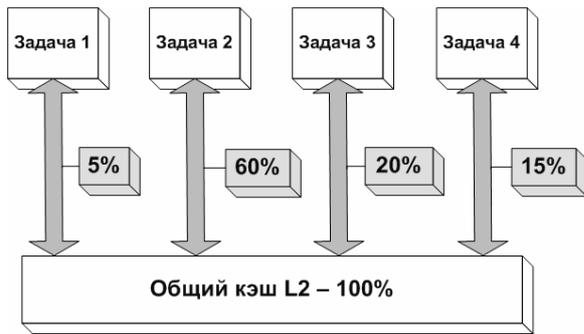


Рис. 2. Пример планирования задач в системе на основе 4-х-ядерного МП

Одно из решений данной проблемы – это эффективная политика планирования на уровне операционной системы (ОС). Уже известны исследования направленные на разработку эффективных методов планирования задач, позволяющих равномерно использовать общий кэш L2 [3 – 5]. Данные методы планирования основаны на приоритетном планировании, однако они не могут быть применимы для класса задач реального времени, когда необходимо также учитывать временные ограничения.

1. Цель исследования

К задачам реального времени (РВ) можно отнести мультимедийные приложения, которые используют предельно высокие скорости передачи данных и требуют воспроизведение в режиме реального времени. Под мультимедийной системой может пониматься мультимедийный сервер (МС) транслирующий видео файлы через интернет различным клиентам. В рамках такого перспективного направления [5] развивается такая область, как видео по заказу (рис 3). Пользователь может заказывать и просматривать видео фильмы, передачи в высоком качестве.

В задачи МС входит кодирование информации и передача ее клиентским процессам, которые в свою очередь декодируют и отображают видео. ПО МС относится к классу многопоточных задач мягкого РВ, когда различные потоки ПО реализуют необходимую функциональность для мультимедиа. Данное ПО накладывает ограничения и на ОС, которая должна обеспечивать таким задач необходимый уровень сервиса.

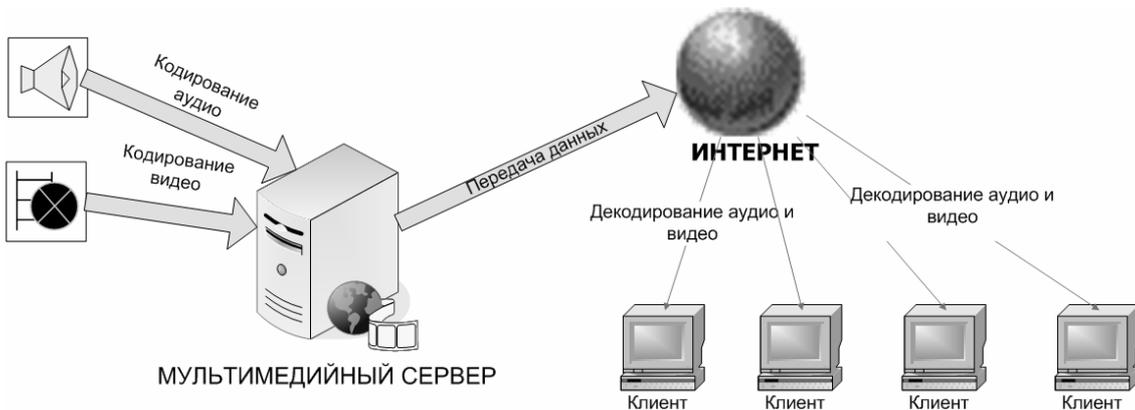


Рис. 3. Видео по заказу, транслируемое через интернет

Целью данного исследования является повышение производительности мультимедийных систем

на основе МП за счет модификации известного метода планирования задач РВ.

2. Сравнительный анализ

Для эффективного планирования задач РВ, к которым относятся мультимедийные приложения, в планировщике ОС должен быть реализован необходимый алгоритм планирования, поддерживающий РВ. Наиболее известными методами планирования для задач мягкого РВ считаются Earliest deadline first (EDF) и Rate Monotonic (RM). Алгоритм RM, назначает каждому процессу фиксированный приоритет согласно его периоду и времени выполнения. Во время работы алгоритма планировщик всегда запускается готовый к работе процесс с наивысшим приоритетом, прерывая при необходимости работающий процесс.

В алгоритмах с динамическим назначением приоритетов приоритет задач может изменяться во время выполнения. В алгоритме EDF самый высокий приоритет получает задача с ближайшим крайним сроком выполнения. Таким образом, RM решает проблему планирования периодических задач, статическим методом назначения приоритетов, а алгоритм EDF – динамическим.

Для планирования мультимедийных задач наибольший интерес представляет алгоритм EDF [6, 7], который позволяет гарантировать максимальную 100% загрузку системы, обеспечивая эффективное планирование, тогда как алгоритм RM гарантирует соблюдение сроков выполнения лишь при 69% загрузки [5].

3. Общая концепция динамического метода планирования

Пусть ПО МС является многопоточными в системе на основе МП и основной задачей данного исследования, является метод планирования позволяющий равномерно распределять кэш-память между потоками (задачами), и гарантировать выполнение в режиме РВ. Равномерно распределять кэш-память между задачами на аппаратном уровне не возможно, однако можно эффективно управлять им на программном уровне ОС.

В данном исследовании предлагается ввести дополнительный параметр для задач, как требование к кэш-памяти. Вычисление данного параметра является достаточно сложной задачей, однако уже известны алгоритмы его вычисления [8, 9].

Таким образом, эффективная политика планирования будет осуществляться на основе динамического алгоритма EDF, где будет учитываться такой параметр, как требования к кэш-памяти для каждой отдельной задачи.

Рассмотрим пример, когда МС, организован в виде многопоточных задач (multithreaded tasks –

MTTs), где каждый МТТ состоит из периодических задач, которые имеют общий период, но различное время выполнения. Каждая ММТ обладает общим рабочим пространством – областью памяти.

Существует два варианта планирования подобных задач:

– когда, многопоточная задача запускается на одном ядре, в то время другие ядра обрабатывают другие задачи. То есть такое планирование, по сути, ничем не отличается от системы с одним процессором, так как потоки обрабатываются последовательно. А значит, каждая задача при параллельной обработке на ядрах обращается к различным блокам памяти.

– когда, потоки одной задачи выполняются параллельно на ядрах процессора. Такая схема является более естественной для многопоточных задач и это ускоряет выполнение ПО. Данная схема является более интересной, так как параллельные потоки используют одну область памяти, что значительно снижает трафик к кэш-памяти.

Пусть некоторые потоки разных ММТs вызывают трафик к кэш-памяти, т.е. имеют высокое требования к кэш-памяти, назовем их *L2T-задачами*. Значит совместное планирование L2T-задач, может привести к значительным промахам доступа к кэш-памяти.

Рассмотрим пример планирования 4-х задач {A, B, C, D} на 3 ядрах процессора, где задачи {B, C} являются потоками одного ММТ. Пусть задачи A и B, в момент времени 0 обладают наивысшим приоритетом и они также являются L2T-задачами, т.е. при их совместном планировании возникнет трафик к общей кэш-памяти. Если же спланировать задачу C вместо A, то мы избежим задержки при планировании. К тому же, важным моментом является то, что задачи C, D являются частью одного ММТ, что значительно ускорит их выполнение, так как велика вероятность работы с одними данными, которые и будут находиться в кэш-памяти.

Задачи будут выполняться быстрее, хотя данное планирование может негативно повлиять на соблюдение сроков выполнения. Однако такое управление приоритетами в рамках алгоритма EDF возможно, как показывают следующие исследования, где изложены принципы управления приоритетами. Именно динамическая схема назначения приоритетов дает возможность соблюдать временные ограничения для задач и учитывать факторы многопоточности приложений и разделяемой кэш-памяти в МП.

Выводы

В данном исследовании были рассмотрены основные архитектуры МП, особенности организа-

ции кэш-памяти, и обоснована необходимость модернизации ПО и ОС для эффективного использования новой аппаратной архитектуры МП.

Более узко была рассмотрена проблема планирования мультимедийных приложений, которые требуют высокой производительности для отображения видео в высоком качестве. Рассмотренный метод планирования ПО МС в системах на основе многоядерных процессоров, требует дальнейшего исследования, с целью практической проверки его эффективности.

Литература

1. Levy M. *Measuring Multicore Performance.* / M. Levy // *Embedded computing.* – 2008. – P. 99-102.
2. Fedorova A. *Cache-Fair Thread Scheduling for Multicore Processors.* / A. Fedorova, M. Seltzer, D. Michael // *ACM Transactions on Computer Systems.* – 2007 – P. 184-215.
3. Anderson J. *Real-time scheduling on multicore platforms.* / J. Anderson, J Calandrino [Электронный ресурс] – Режим доступа: <http://www.cs.unc.edu/~anderson/papers>.

4. Anderson J. *Parallel real-time task scheduling on multicore platforms.* / J. Anderson, Calandrino J // *Proc. of the 27th IEEE Real-Time Systems Symp.* – 2006. – P. 89-100.

5. Dapeng Wu. *Streaming Video over the Internet: Approaches and Directions.* / Wu. Dapeng, Y.T. Hou // *IEEE transactions on circuits and systems for video technology.* – 2001. – P. 103-109.

6. Туркин И.Б. *Анализ алгоритмов планирования задач реального времени для многопроцессорных систем.* / И.Б. Туркин, Т.С. Никитина // *Радиоэлектронные и компьютерные системы.* – № 6 (33). – 2008. – С. 58-61.

7. Liu C.L. *Scheduling algorithms for multiprogramming in a hard real-time environment.* / C.L. Liu, J.W. Layland // *J. Assoc. Comput.* – 1973. – P. 46-61.

8. Fedorova A. *Throughput-oriented scheduling on chip multithreading systems.* / A. Fedorova, M. Seltzer, C. Small, D. Nussbaum // *Technical Report. Division of Engineering and Applied Sciences, Harvard University.* – 2004.

9. Parekh S. *Thread-sensitive scheduling for SMT processors* / S. Parekh, S. Eggers, H. Levy, J. Lo / [Электронный ресурс] – Режим доступа: <http://www.cs.washington.edu/research/smt>.

Поступила в редакцию 12.01.2009

Рецензент: д-р техн. наук, проф., зав. кафедрой В.М. Вартанян, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков, Украина.

МЕТОД ПЛАНУВАННЯ МУЛЬТИМЕДІЙНИХ ЗАВДАНЬ В ОПЕРАЦІЙНИХ СИСТЕМАХ РЕАЛЬНОГО ЧАСУ ДЛЯ ПЛАТФОРМ НА ОСНОВІ БАГАТОЯДЕРНИХ ПРОЦЕСОРІВ

Т.С. Нікітіна, І.Б. Туркін

Розглядаються особливості архітектури багатоядерних процесорів, нові вимоги до програмного забезпечення, що використовують всі переваги систем з паралельними обчисленнями. Зроблено аналіз проблеми планування мультимедійних завдань реального часу в системах на основі багатоядерних процесорів. Розглянутий динамічний метод планування мультимедійних завдань для систем на основі багатоядерних процесорів.

Ключові слова: мультимедійні додатки, багатоядерний процесор, алгоритм планування, періодичні завдання, реальний час.

METHOD OF PLANNING MULTIMEDIA TASKS IN REAL TIME SYSTEMS FOR PLATFORMS ON BASIS OF MULTICORE PROCESSORS

T.S. Nikitina, I.B. Turkin

Explaining the features of architecture multicore processors, new requirements of software, using all advantages of the parallel. The analysis of problem of planning of multimedia tasks on the real time systems on the basis of multicore processors. The dynamic method of planning of multimedia tasks is considered for the systems on the basis of multicore processors.

Key words: multimedia application, multicore processor, scheduling algorithm, periodic task, real time.

Никитина Татьяна Сергеевна – аспирантка каф. 603 Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ», Харьков, Украина, e-mail: Tanya-nt@mail.ru.

Туркин Игорь Борисович – д-р техн. наук, проф., зав. кафедрой 603 Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ», Харьков, Украина.