

УДК 004.318

Кожин А.С.

ЗАО «МЦСТ»

Анализ методов объединения процессорных ядер на основе кэша третьего уровня

На сегодняшний день одним из наиболее приоритетных направлений повышения вычислительной мощности микропроцессоров считается проектирование многоядерных систем, объединяющих несколько универсальных и специализированных ядер на одном кристалле. При таком подходе первостепенной задачей является разработка подсистемы памяти, которая обеспечит доступ к необходимым данным с наименьшей задержкой. В современных многоядерных микропроцессорах проблема уменьшения времени доступа в память и поддержки межъядерной когерентности чаще всего решается с помощью использования кэша третьего уровня (L3-кэша), общего для всех ядер.

При разработке общего кэша третьего уровня в первую очередь необходимо решить, должен ли он содержать в себе все строки из кэшей верхних уровней всех ядер (inclusive) или нет (non-inclusive). Третий вариант, в котором в L3-кэше хранятся только те данные, которых нет в ядрах, (exclusive) считается менее эффективным решением вопроса объединения ядер.

Особенность общего (inclusive) L3-кэша в том, что все строки данных, которые есть в кэшах верхнего уровня, должны содержаться и в нем, а при вытеснении строки из него должна вытесняться и из кэшей всех ядер. Одновременно с этим уменьшается общий объём доступной кэш памяти, который становится равным объёму L3-кэша. Влияние этого недостатка можно уменьшить, по возможности выбирая для вытеснения данные, которых нет в ядрах на текущий момент, и используя метод предварительного вычеркивания данных только из ядер. В то же время указанная особенность позволяет существенно ускорить обмен данными между ядрами и уменьшить поток когерентных сообщений в системе. Для этого в каждой строке L3-кэша помимо данных и тэгов следует хранить ее вектор возможных владельцев, куда заносятся номера всех ядер, запросивших себе эту строку. Данная схема успешно используется фирмой Intel в процессорах с архитектурой “Nehalem” [1].

В системах на основе non-inclusive кэша третьего уровня нет требования на взаимное расположение данных в кэшах верхних и нижнего уровней, что позволяет

избежать взаимных вытеснений и увеличивает общий объем кэш памяти. Но при этом значительно возрастают поток когерентных сообщений в системе и время доступа в L3-кэш, так как в этом случае при каждом обращении необходимо разослать соответствующие сообщения во все ядра, даже если они эти данные никогда не запрашивали. Уменьшить влияние этой особенности можно с помощью когерентных фильтров на основе копий тэгов кэшей верхних уровней, что не всегда реализуемо ввиду архитектурных особенностей микропроцессоров и значительных накладных расходов, или с помощью вектора ядер-владельцев, как было реализовано в процессорах с архитектурой “Magny-Cours” фирмы AMD [2].

По результатам проведенного анализа и архитектурных особенностей для будущего 4-х ядерного микропроцессора было выбрано решение на основе общего inclusive кэша третьего уровня и разработан протокол когерентности, учитывающий вектор ядер-владельцев.

Литература

1. *Trent Rolf* Cache Organization and Memory Management of the Intel Nehalem Computer Architecture. – University of Utah Computer Engineering, CS 6810 Final Project, 2009.
2. *Pat Conway, Nathan Kalyanasundharam, Gregg Donley, Kevin Lepak, Bill Hughes* Cache Hierarchy and Memory Subsystem of the AMD Opteron Processor. – IEEE Micro, 2010, vol. 30, no. 2, pp. 16-29.