

Исследование потенциала механизма улучшенной обработки условных переходов типа “if-then” для суперскалярного микропроцессора

О. И. Ладин^{1,2}, А. Ю. Сивцов²

¹Московский физико-технический институт (государственный университет)

²АО «Интел А/О»

Высокая эффективность использования функциональных блоков микропроцессора суперскалярной архитектуры обеспечивается работой модуля предсказания переходов. Следовательно, одним из существенных ограничений производительности микропроцессоров суперскалярной микроархитектуры являются ошибки предсказаний этого модуля. При неверном предсказании возникает необходимость удалить из машины уже загруженные команды и загрузить правильные, следующие за переходом.

В общем случае исполняемые команды можно разделить на зависящие по управлению от перехода команды (Control Dependent, CD) и не зависящие (Control Independent, CI), которые в свою очередь можно разделить на зависимые по данным от CD команд (Control Independent Data Dependent, CIDD) и не зависимые по данным (Control Independent Data Independent, CIDI). Исполнение CIDI-команд абсолютно не зависит от результата перехода [2]. В настоящий момент предложены техники, использующие факт независимости CI команд [2][3][4] для отмены их удаления и повторной загрузки в процессор при неверном предсказании перехода.

В данной работе исследуется подобный метод, позволяющей избежать повторной загрузки CI-команд и повторного исполнения CIDI-команд при исполнении переходов типа “if-then”, в которых точка схождения путей исполнения есть адрес перехода (рис. 1). Такие переходы могут быть легко детектированы оборудованием. С другой стороны, для них существует только одна ветвь CD-команд, а путь на провал содержит в себе все команды пути на взятие, что упрощает их обработку.

Увеличения производительности предлагается достичь в случае неверного предсказания на провал “if-then” перехода. В этом случае нет необходимости перезагружать CI-команды, которые уже содержатся в неправильно предсказанном пути исполнения. Также нет необходимости перенаправлять конвейер загрузки новых инструкций, так как правильный путь исполнения не содержит новых команд, которые должны были бы быть загружены взамен имеющимся. Единственным действием остаётся предоставление правильных данных CIDD-командам, которые зависят от отмененных CD-команд.

В программный потактовый симулятор микропроцессора архитектуры Intel64 было внедрено динамическое детектирование “if-then” переходов путём поиска точки схождения (Reconvergence Point, RP) [3] и обработка согласно предложенной схеме. Зависимость CIDD-команд от CD-команд является причиной ожидания первых разрешения перехода, а, значит, может негативно сказаться на производительности в случае верного предсказания перехода. В связи с этим, техника применяется исключительно для труднопредсказуемых переходов.

При моделировании было установлено, что на заданных тестах 3% переходов являются труднопредсказуемыми, и к 14% от их числа может быть применена предложенная оптимизация. Из результатов моделирования следует, что средний прирост производительности при внедрении подобной техники в существующий процессор достигает +1,5%.

Литература

1. *Chen-Yong Cher, T. N. Vijaykumar* Skipper: A Microarchitecture For Exploiting Control-flow Independence // 34th Annual International Symposium on Microarchitecture – 2001 – pp. 4-15.
2. *Jamison D. Collins, Dean M. Tullsen, Hong Wang* Control Flow Optimizations Via Dynamic Reconvergence Prediction // 37th Annual IEEE/ACM International Symposium on Microarchitecture – 2004.
3. *Ahmed S. Al-Zawawi et al.* Transparent Control Independence (TCI) // 34th Annual International Symposium on Computer Architecture – 2007 – pp. 448-459.

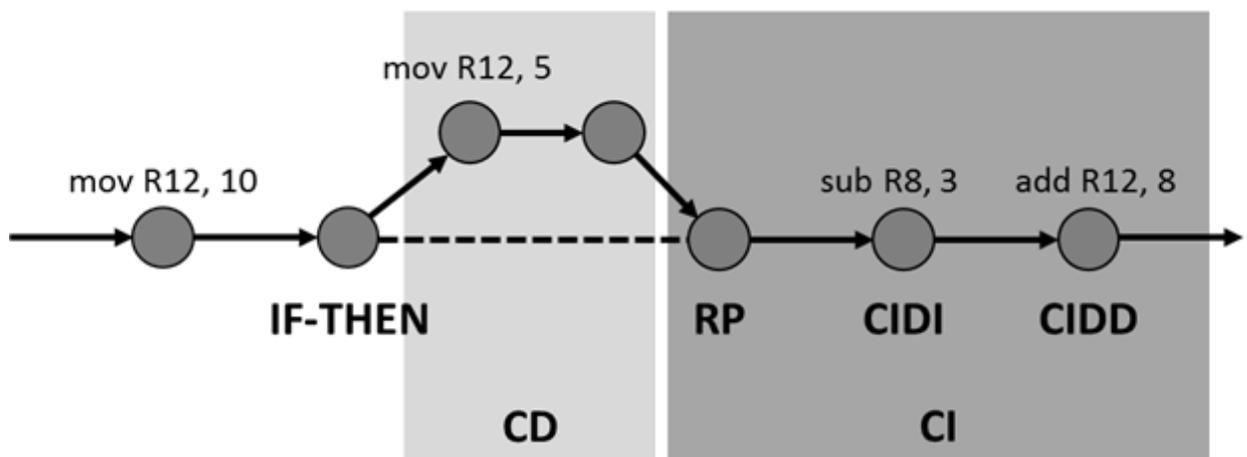


Рис. 1. Схематичное изображение перехода типа “if-then”.