

Архитектуры IA64 и x86-64 (AMD64)

Д. В. Луцив

Кафедра системного программирования СПбГУ



CS220 (231000)

Содержание

- 1 Общие современные тенденции
 - Общие понятия
- 2 x86-64
- 3 IA 64
 - Регистровый файл
 - Система инструкций
 - ОЗУ

Архитектуры IA64 и AMD64

Архитектура x86 под давлением обратной совместимости быстро развиваться уже не может. На данный момент новые разрабатываемые архитектуры всё чаще склоняются к имитации x86 с потерей производительности там, где это необходимо и к штатным режимам работы, не совместимым с x86.

TLB

Для хранения отображения адресных пространств в физические адреса используются Translation Lookaside Buffer'ы. Это и в более старых (x86) процессорах было, но там они выглядели, как «кэш таблицы страниц». Страницы обычно уже не по 4КиБ, а более 1МиБ (4 МиБ, например).

Регионы

Регионы являются аналогами сегментов, но для процесса они расположены в едином линейном адресном пространстве. Это просто отрезки адресного пространства.

Так, при переключении процессов переключается память, в которую отображаются регионы кода и данных процесса, однако регионы ядра и библиотек системы остаются прежними. При переключении процессов и потоков переключаются регионы, в которые отображается стек.

Если адресное пространство очень велико, то можно зафиксировать регионы. Физическая память всё равно будет выделяться страничной системой лениво.

Виртуализация и дополнительный уровень привилегий

Добавляется возможность загружать сразу несколько операционных систем и переключаться между ними. Специально для этого вводится термин «уровень привилегий -1 ».

В i80386 было 4 уровня: 0, 1, 2, 3.

Такими темпами теперь привилегии могут расти до $-\infty$ (смех за кадром). Например, если мы захотим загрузить несколько виртуализаторов.

Виртуализация и дополнительный уровень привилегий

Добавляется возможность загружать сразу несколько операционных систем и переключаться между ними. Специально для этого вводится термин «уровень привилегий -1».

В i80386 было 4 уровня: 0, 1, 2, 3.

Такими темпами теперь привилегии могут расти до $-\infty$ (смех за кадром). Например, если мы захотим загрузить несколько виртуализаторов.

Регистровый файл

- Те же регистры, но расширенные до 64 с префиксом R, например, RIP.
- Ещё 8 новых под номерами.
- Медиа-регистры (SIMD)
 - 7 64-битных
 - 16 128-битных
- Передача первых параметров процедур подразумевается в регистрах, остальных — на стеке

Система инструкций

Без принципиальных изменений, но с возможностью использовать 64 бита и с относительной адресацией, отталкиваясь от RIP.

Модель памяти I

- 64-битный режим реализует поддержку расширенных регистров через новую группу префиксов команд REX.
- 64-битный режим обеспечивает адресацию данных относительно 64-битного регистра RIP. x86 архитектура обеспечивала адресацию относительно IP регистра только в командах передачи управления. RIP-относительная адресация повышает эффективность позиционно-независимого кода и кода, который адресует глобальные данные.
- Сегментная адресация выключается (что очень затруднило работу виртуальных машин, запускающихся, как пользовательский процесс).
- Ещё можно использовать FS и GS в качестве смещений. Почему именно их, не совсем ясно. Но почему бы и нет...

Модель памяти II

- Память делится на регион системы и регион процесса.
- Регионы разных процессов и потоков могут содержать страницы, отображаемые в одни и те же области ОЗУ.
- Например, при переключениях между разными программами надо поменять всё, при переключениях между разными экземплярами одной и той же программы — данные и стек (код не трогать), при переключении между потоками — только стек.
- Системный регион считается в дополнительном коде в «отрицательных» адресах. Это позволяет не менять адрес его верхней границы (всегда -1) при любом объёме памяти.

Ссылки

▶ [Статья с описанием](#)

▶ [Википедия](#)

Регистровый файл

Состоит из статической и стековой части.

- Статическая часть — большое количество 64-битных регистров.
- Стековая — набор с относительной адресацией, который при переполнении может выгружаться на стек в ОЗУ а при опустошении — загружаться оттуда. Используется для организации фреймов процедур.
- Есть предикативные регистры для условного выполнения кода

Математический сопроцессор работает без стека, тоже с собственными регистрами. Числа с плавающей запятой до 96 битов.

Система инструкций

- Инструкции системы VLIW, параллелизм явный, до 6 независимых команд в инструкции
- Предикативные регистры устанавливаются командами сравнения и могут проверяться перед выполнением инструкций
- Инструкции для обмена данными с памятью отдельные (в них никакой арифметики, только загрузка/выгрузка)
- Без оптимизации по зависимым-независимым инструкциям работает очень медленно. С оптимизацией — достаточно быстро.

Модель адресации ОЗУ

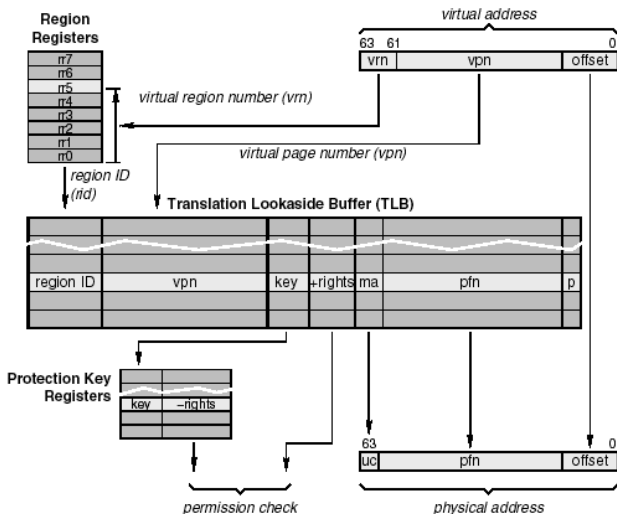


Иллюстрация из [статья](#).

Пояснения

- TLB и PCR разреженные.
- Права запрета (-rights) приоритетнее.
- Возможно включить (для отдельных регионов) автоматическое выделение новых страниц при промахе TLB. ОС выделяет физическую память большими блоками, и процессор сам добавляет её в таблицы страниц. При этом обработчик невыделенной страницы вызывается реже.
- Для диапазона адресов страницы часто применяется термин Page Frame.
- Размер страницы может настраиваться программно.

▶ Ещё статья

▶ Но рынок решил иначе

