X86 — Lurkmore

«Архитектура x86 — это победа маркетинга над здравым смыслом. »

— Старожил кремниевой долины

x86 (aka i80x86, IA-32 и даже x86-64 aka amd64/intel64) — довольно распространенная архитектура процессоров для персональных компьютеров и серверов, благодаря которым ты, да-да, лично ты читаешь этот текст. По совместительству — самая популярная архитектура процессоров для ПеКа, также абсолютный рекордсмен по архитектурным излишествам, их рудиментам и связанным с ними аппаратно-программным изыскам. В последнее время у x86 свои ниши отвоёвывают две совсем уж бессмысленные и беспощадные платформы — ARM и MIPS, которые моложе сабжа всего на несколько лет.

За аффтарством Дани

Детальный обзор процессоров или все точки над И - Обзор

Благодаря своей давней истории и стремлению «Интела» к сохранению совместимости, сабжевый процессор являет нам ярчайший пример свалки исторического мусора, напоминая тем самым ДНК. Начнём с того, что x86 является чуть ли не единственным современным CISC'ом. А это означает множество режимов адресации, избыточную, неортогональную и не единообразную систему команд, совершенно ебанутую систему префиксов команд и смехотворное количество регистров общего назначения (с последним пунктом в X86-64 дела обстоят ~~несколько~~ в два раза лучше).

Исторический очерк. Луркмор образовательный.

Предыстория

Семидесятые. Компания Intel молода, быстро развивается, а после выпуска нескольких своих первых процессоров, таки запиливает в 1974 году настоящий хит того времени — Intel 8080, который имел шину адреса в 16 бит и без особых ухищрений мог адресовать 64 кб, загружая по 8 бит за одно обращение к памяти. Немного спустя его рождение привело к явлению миру и городу первого (персонального) микрокомпьютера Altair 8800 с первым же коммерческим программным продуктом полуподвальной, еще малоизвестной тогда конторки Micro-Soft, состоящей из нескольких друзей-студентов. Ага.

Вскоре после начала производства 8080, группа работников Intel уволилась и создала свой православный аналог 8080, с дополнительными командами и регистрами, полностью бинарно совместимый с 8080. Звался он Z80 и был популярен в домашних компьютерах аж до начала 90-х, успешно конкурируя с MOS 6502.

Прогрессивная совковая промышленность, отвечая Интелу, затрещала, но выдержала, родив из своих недр микросхему КР580ВМ80А с тактовой частотой аж в 2.5 МГц. ИЧСХ, под микроскопом сбоку от точки подвеса первой ноги можно лицезреть остатки интеловской маркировки — i80. По мнению совкодрочеров, такой проеб особенно обиден, ибо в почтовых ящиках, НИИ и на оборонных заводах была проделана работа по всесторонней модернизации исходного проца, да так успешно, что новый продукт был лишь функциональным аналогом оригинала. Впрочем, серийное производство освоили лишь в 1977 году, но через год уже вышел 8086 — собственно, первый x86 процессор.

Появление

Род x86 начинается с процессора 8086, который представлял собой 16-разрядный процессор с адресным пространством памяти в 1Мб. Адресная шина в 20 бит при этом использовалась специфически: программисту выдавался на руки виртуальный 32-битный адрес, разрубленный на две половины — 16 бит на сегмент, по биту через жопу сдвинутые на 4 разряда влево плюс 16 бит на смещение в сегменте формируют полный адрес. Выражаясь языком Си, `real_address=(segment_register<<4)+address_register`. 16-разрядное ОЗУ для 8086 стоило дорого, потому в тираж кроме него пошел и наполовину восьмибитный 8088, загружавший 16-битное слово за два обращения к памяти.

Фокус с громоздкой адресацией придумали дабы 8086 умел адресовать больше памяти — расово новоанглийский LSI-11 при переползании с родных 16 адресных бит на 22 к концу своей истории потребовал переписывания или в лучшем случае перекомпиляции софта, а в своей 32-разрядной (AKA VAX) ипостаси вообще утратил совместимость с предком, тогда как x86 пережил переход на 24-, а затем и на 32-битные адреса совершенно безболезненно. При этом для сохранения совместимости, на которую в «Интеле» и по сей день яростно дрочат, с уже имеющимся софтом для 8086, он исполнялся его через «окошко» в те же самые 64 кб с последующим «наращиванием» в CS:IP первой цифры, что сдвигало «окошко» на 1/256 страницы.

8086 был несовместим с 8080 как по выводам, так и по системе команд, что не помешало японской

компании NEC выпустить свой процессор v20, который таки был совместим с 8086, но умел выполнять и код для 8080. Применялся он в расово японском компьютере PC-88.

Начало восьмидесятых. IBM между делом разрабатывает настольный персональный компьютер. Ведущий инженер проекта Дон Эстридж хочет использовать собственную МежДелМашевскую разработку — ROMP, но узнав, что боевые слоупоки из процессорного отдела опять кормят его завтраками, поменял процессор на 68000-й мотороллер — тот самый, на котором через пять лет был построен «Мак». Однако у мотороллеров на тот момент (1980 год) были неиллюзорные проблемы с производством, да и обвязка на 68000 стоила как самолёт, поэтому разработчики опять передумали, и остановились на интеловском 8088, который был дёшев, производился массово (IBM даже настояла на продаже «Интелом» лицензии на его производство AMD), да вдобавок прекрасно работал с копеечной 8-битной обвязкой от 8085. Заслоупоченный же ROMP, когда его наконец доделали, превратился в знаменитую POWER-архитектуру — ту самую, на которую потом перешли в Apple вместо 680x0.

Отечественный производитель отметился и тут, и в той стране имел известность в определенных кругах как КР1810ВМ86, что кагбэ намекает на 8086, но запил оказался немного не торт, и почти ниасилил.

Развитие

Развитием 8086 стал процессор 80186, не достигший значимых успехов на рынке ПеКа, хотя и бывший длительное время весьма популярным для встраиваемой электроники, в связи с чем на рынок были выпущены SoC-версии данного процессора. Архитектурных отличий нет, так, несколько новых команд и немного другой механизм обработки прерываний.

Следующим был 80286. Несмотря на всю его неуклюжесть, для своего времени (1982 г.) являлся значительным шагом вперед благодаря ~~появлению~~ преждевременным родам защищенного режима. При возможности подключения к шине 16 мегабайт, мог адресовать в защищенном режиме гигабайт памяти, что приходилось делать весьма нетривиальным способом. Вернуться же из защищенного режима было невозможно без перезагрузки, поэтому толком использовать больше 1 мб реально умел лишь Microsoft Xenix, и, возможно, Minix (однозадачная unix-подобная ОС, написанная Танненбаумом для демонстрации студентам). Вообще, конечно, были и маньяки, умудрявшиеся на AT 16 MHz и фряху поднять, но пруфлинков на это не осталось. Хотя один релкомовский сервер именно на таком железе в лапки попадался и довольно долго проработал. Это *можно было* сделать, но это была вещь в себе.

Новое революционное достижение Intel — 32-разрядный процессор 80386, родоначальник архитектуры IA-32. Имел полноценное страничное MMU для организации виртуальной памяти. Адресуемое пространство памяти — 4Gb, а с перезагрузкой сегментных регистров — чуть меньше 8 Gb, но это не важно, поскольку адресных линий у i386DX было всего 32, а у облегчённого SX — и вовсе 24, как у 286. Умопомрачительная величина для эпохи MS-DOS. Архитектуру оценили сразу и массово, уже вскоре появились DOS-оболочки (DESQview, Windows 2.0 for 386) и полноценные ОС (OS/2, BSD, Linux[1]…) с поддержкой 32-битной архитектуры. Значимых изменений архитектуры в целом больше не будет вплоть до настоящего времени, если не считать таковым внедрение 64-битного расширения в 2003 году.

На этом революции заканчиваются и начинается эволюция. В 80486 появилась трансляция внешних CISC инструкций во внутренние RISC, была добавлена встроенная кэш-память, а в модели **DX** — и встроенный сопроцессор. В Pentium — конвейерная архитектура с предсказанием ветвлений. Начинается внедрение расширений — MMX, SSE, SSE2, SSE3, Тысячи их! В Pentium Pro появилась возможность адресовать посредством PAE до 64GB памяти и встроенная поддержка многопроцессорности. В Pentium II интеловцы решили отказаться от многоногих сокетов в пользу слотов. В Pentium III они же поняли, что сфейлили и вернулись обратно к сокетам. Немного опередил время Pentium III Tualatin, имея производительность выше, чем у Pentium IV и энергопотребление ниже, чем у Pentium III Coppermine, но был незаслуженно отброшен маркетологами в гонке за 2 ГГц, но потом наработки из него вернулись в Core2 и Atom.

AMD тоже не зевала: в 2000 году её процессор Athlon преодолел планку в гигагерц раньше интеловских поделий, а годом спустя компания выпустила спецификацию AMD64. Несколько позже Intel, эпично сфейлившись со своим Itanium, который был плохо совместим с x86, подхватила эту концепцию как EM64T, впоследствии переименовав ее в x86-64. Этот стандарт привнёс в архитектуру 64-битные расширения, попутно выпилив часть рудиментов, а также, как потом выяснилось, и полезных фич. Это может означать полную победу x86: если ей и предрекали смерть вместе с 32-разрядыми вычислениями, то теперь она фактически обрела новую жизнь, на горе конкурентам и здравому смыслу. Первые 64-разрядные процессоры появились в 2003 году.

Кроме Intel и AMD, производством x86-процессоров для ПК занимались UMC, IDT, VIA, Cyrix, Texas Instruments и ещё более десятка компаний, но к настоящему времени одни из них безнадёжно отстали от лидеров и бросили это дело, а другие остановились на узкоспециализированных решениях не для массового рынка.

В 2006 году предпоследний оплот не-интеловских десктопов, Apple, забил на PowerPC к великой скорби PowerPC-фагов, и стали выпускать основанные на интеловских процессорах десктопы и ноутбуки. В первую очередь потому, что очередное поколение PPC, при всей его производительности оказалось настолько неиллюзорной кофеваркой, что на десктопные маки пришлось запиливать едва ли не первую в мире официальную искароппки систему жидкостного охлаждения. О том, чтобы ставить эдакую хуиту в

ноутбуки или компактные машины типа iMac’ов не могло быть и речи, поэтому вместо процессоров G5 мобильные и компактные Маки комплектовались камнями G4, а то и G3, через что стали неиллюзорно делать сасай у Glorious PC Master Race. Учитывая же, что к тому моменту основную прибыль Эпплу приносили именно ноуты, результат немного предсказуем. Так что теперь на этот ваш Мак можно невозбранно заливать самую передовую в мире версию Windows, а на все остальные унылые компы, хоть и не без изъебств, но таки заливать священную Макось на радость нищебродам и копирастам в назидание. История, однако, циклична и теперь яблоки перелезают на ARM, заявив мол, что новые интелы херово тестируются и клепаются абизянками.

Известные костыли и прочие особенности

IME (Intel Management Engine)

Intel решила не упускать модный тренд глобальной слежки. И, чтобы не мелочиться, встраивает бэкдор с 2008 года в каждый чипсет, для облегчения работы одминов, ага. И не просто бэкдор, а процессорное ядро, ОЗУ и ПЗУ, из которого выполняется зашифрованный код. У параноиков, конечно, от этого сначала засвербило в одном месте, а потом зачесались и руки, в результате чего выяснилось следующее:

1. На некоторых материнках можно подсунуть мусор вместо кода IME, но будет срабатывать защита и комп будет вырубаться, не предложив сохраниться.
2. Ядро IME имеет доступ ко всей доступной памяти и устройствам, включая сеть.
3. IME работает даже тогда, когда компьютер выключен, но вставлен в розетку.
4. Нельзя просто взять и выключить IME.
5. Сетевой трафик на IME абсолютно не виден хосту тк вырезается на входе сетевого порта модулем IME.

AMD, чтобы не отставать от лидера, также добавила в свои чипсеты фичу AMD Secure Technology, но сделала её отключаемой.

Флаг защищенного режима

Первый архитектурный геморрой. Имел место в процессоре 80286. Достаточно эпичен. Заключается в том, что установкой этого флага процессор переводится в режим, в котором доступны все его новые фички (включая доступ к памяти до 16 Mb), но при этом он теряет совместимость с 8086, а сбросить этот флаг можно только полным ресетом процессора. А поскольку ко времени появления компьютеров PC AT с этим процессором существовало более 9000 программ под MS-DOS, которая была гвоздями прибита к 8086, то новые возможности процессора оставались невостребованными. Для решения проблемы был придуман костыль: для возврата в реальный режим выполнялся сброс процессора через контроллер клавиатуры, что само по себе было операцией небыстрой. Поэтому использовать память свыше 1 Мб было можно только для виртуальных дисков, кеш-буферов и временных хранилищ[2]. Windows и OS/2, работая в защищенном режиме, пользовались этим костылем для обработки прерываний и запуска программ DOS. Начиная с процессора 80386 флаг защищенного режима сбрасывать было уже можно, но костыль с контроллером клавиатуры таскают и до сей поры, дабы обеспечить совместимость. Также в 80386 появился режим виртуального 8086, что тоже в какой-то степени решало проблему совместимости с DOS.

Адресная линия A20

Второй по значимости костыль. Дело в том, что ряд нерадивых программистов использовали тот факт, что адреса в сегменте FFFFh процессором 8086 заворачивались на первые 64k памяти.

technobabble
Дело в том, что ещё при проектировании 8086 процессора для совместимости, а точнее для простоты портирования старого 8-битного кода с 8080 и Z80, физический 20-битный адрес было решено от программиста спрятать. Взамен ему выдали виртуальный 32-битный адрес, состоящий из 2-х 16-битных полей -- сегмента (точнее *номера* сегмента, см. ниже, ещё точнее - сегмента с 16-битной гранулярностью, а не *номера-дескриптора*) и смещения. Если программист со всем своим кодом, данными и стеком влезал в 64K памяти, то он мог писать по-старинке, если же ему требовалось больше -- вот тут-то и начиналось веселье. Дело в том, что на обе половины адреса было отведено по своему отдельному набору регистров, и при трансляции виртуального адреса в физический процессор брал сперва содержимое сегментного регистра, умножал его на 16 (т.н. "гранулярность памяти"), и прибавлял к полученной величине содержимое регистра смещения. Но поскольку сегментный регистр мог содержать от 0h до FFFFh, то максимальным значением адреса при таком алгоритме получалось 10FFEFh, что, разумеется, превышало максимально адресуемые FFFFFh. Поэтому старший бит адреса попросту обрезался, что и приводило к тому, что у любого сегмента с номером, большим F000h, старшие адреса начинали заворачиваться на сегмент 0000h.

В 80286, имевшем 24-битовый физический адрес, старшие биты больше не обрезались, и поэтому такие адреса указывали в расширенную память, а в результате такие хитровыебанные программы на этом процессоре не работали. Решили эту проблему, сделав 20-ю адресную линию отключаемой, заведя

управление ею на вышеупомянутый контроллер клавиатуры. Костыль также таскают по сей день[3], хотя уже и о программах-то тех, наверное, давно забыли, как и о самой MS-DOS вспоминают лишь ностальгисты да олдфаги. Область расширенной памяти, адресуемая в реальном режиме (размером в 65520 байт), получила название High Memory Area (HMA), которую, впоследствии, MS-DOS использовала для размещения своего ядра, естественно, при этом потеряв совместимость с A20-зависимыми программами (тем не менее, для их запуска можно было грузить DOS в обычные нижние адреса памяти). Тут следует сделать важное замечание для людей, которые не совсем понимают нафига всё это сделано. Отключать A20 придумали в **ibm** действительно для совместимости. Если забыть о том, что линии можно(нужно?) отключать и ебанутых долбоёбах, которые использовали memory wrap, то нововведение дескрипторов сегмента не кажется такой плохой идеей, так как вместе с защищенным режимом уже в 286 позволило говорить о какой-никакой многозадачности. Также следует не забывать что в компьютерах 21-го века, управление 20-й адресной линией производится как правило более быстрыми способами, чем через контроллер клавиатуры (опция в BIOS «Gate a20 option», где normal — использование контроллера клавиатуры, а Fast — уже что-то реализуемое через сам чипсет)

Аппаратная мультизадачность

В процессорах начиная с 80286 был введен механизм аппаратного переключения контекстов задач, но практически не использовался разработчиками по причине неуклюжей реализации, и поэтому из x86-64 был выпилен[4][5].

Команда LOADALL

Недокументированная команда, имевшаяся в процессоре 80286, которую использовали для обращения к памяти выше 1Мб в реальном режиме. Использовалась драйвером HIMEM.SYS. Говорят, что был выигрыш в скорости, по сравнению с переключением через клавиатурный контроллер. Других преимуществ нет. Маньяки из IBM даже пытались реализовать через нее что-то наподобие виртуальной машины, но сфейлили из-за невысокой скорости работы.

«Расширенный» реальный режим

Также известный как Unreal mode (по аналогии с реальным режимом, real mode). Очень интересный костыль, связанный с наличием т. н. «теневой части» сегментного регистра aka дескрипторного кеша. Впервые появился в 80386, с возможностью возвращаться из защищённого режима в реальный. Нехитрый трюк позволяет модифицировать реальный режим так, что в нём становится можно обращаться ко всем четырем гигабайтам адресного пространства. Недокументированная, но вполне полезная и юзабельная в некоторых ситуациях фича. То есть, *была* полезной, когда кто-то ещё использовал для чего-то полезного реальный режим. Сейчас эта фича используется в основном в BIOS-ах, и то только из-за лени программистов.

Turbo-режим (и соответствующая кнопка)

Некоторые криворукие разработчики использовали частоту процессора для тайминга в своих приложениях, для совместимости с ними был запилен такой костыль, понижающий частоту ЦП до 4.77 мГц (как 8088). Лулз еще в том, что название кнопки противоположно ее действию. Особенность не самого процессора x86, а скорее самой платформы. Использовалась начиная с 286 и порой аж до 486 и первых Pentium, позже была предана анафеме, хотя кнопку можно было встретить и на более новых корпусах. Программные реализации применяются и ныне — ограничение скорости ЦП есть в DOSBox и VirtualBox.

SMM (System Management Mode)

Несмотря на «4-уровневую» систему колец защиты (от наиболее высокого 0-го уровня, где, по идее, работает менеджер Операционной системы, который имеет наиболее высокий приоритет до 3-го, наименее привилегированного), имеется еще один «надуровень» который имеет… еще более высокие привилегии! Точнее, это программный код который вообще никому не виден и который (когда процессор передает ему управление по сигналу #SMI) имеет полный и неограниченный доступ абсолютно ко всем ресурсам системы.

Более того, в архитектуре предусмотрен специальный бит, и если обработчик этого режима (SMI Handler) установит его, то OS даже если и сильно захочет, не сможет не то что блокировать его, но и даже прочитать!

Вообще, всё задумывалось как лучше — SMI-обработчик это на самом деле часть BIOS и управление ему передаётся по сигналу от чипсета при наступлении каких-то событий (например, запись в некоторые порты или сигнал о перегреве) совершенно прозрачно для операционной системы, за исключением потраченного на выполнение этого обработчика времени. То есть, BIOS обеспечивает поддержку или эмуляцию некоторых девайсов чипсета, освобождая ОС от необходимости делать это самой, что даёт нам шансы в старости снова увидеть DOS (да-да, там в 90% случаев было примерно так же: Onboard BIOS Extensions эмулировали сферовакуумное оборудование, а софт работал прямо с этим «оборудованием» без

дров).

Хорошо? Хорошо, но не совсем: во-первых, для SMM нужна поддержка в железе: в чипсете и в процессоре (аж целый специальный режим).

Во-вторых, SMI-обработчики беспардонно отнимают время у задач операционной системы, никак не прерываемы, не откладываемы и приходят ВНЕЗАПНО

В-третьих, факт наличия SMM, именуемого иногда Ring −2, вызывает небеспочвенные опасения у специалистов касательно появления руткитов и прочих троянов, которые не будут брезговать добавлять в BIOS свой резидентный модуль — имея с этого профит в виде повышенной живучести и скрытности, так как выпилить их можно будет только перешивкой биоса, причем — только на программаторе. И совершенно ненапрасно, поскольку первый такой V2P был написан ещё в 1998-ом… И очень интересно глючил систему, «зануляя» по байту с конца массива флэшки за каждую перезагрузку. Отчего порты на матери поражённой такой заразой начинали друг за другом довольно экзотично отмирать даже после того, как антивирь вычищал код-инсталлятор.

В-четвёртых, мысль сумрачных гениев из компании Интел не стоит на месте, и SMM-у ещё в середине девяностых придумали более продвинутую замену: ACPI. Идея ACPI в том, что BIOS всё равно предлагает обработчики для событий чипсета, но в виде аккуратно сложенного в доступную операционной системе табличку байткода. ОС содержит интерпретатор байткода и при необходимости вызывает эти обработчики в своём контексте когда и как хочет. Все довольны. SMM можно выкидывать на помойку.

Но не тут-то было. Несмотря на моральное устарение, SMM жив даже в последних моделях x86 процессоров и чипсетов. Помереть SMM-у мешает тот факт, что через него реализовано несколько других уродливых костылей, из которых самый главный — эмуляция PS/2 клавиатуры через USB, и вообще эмуляция доисторических клавиатурных портов, которые, помимо своего прямого назначения, позволяют невозбранно ресетить систему и используются для этого многими олдфажными ОС-ями. Алсо, во многих системах SMM управляет кулером процессора, без чего может наступить Пиздец. Все эти костыли опциональны; то есть, если ОС поддерживает ACPI, она при загрузке отключает SMM и дальше делает всё сама через ACPI. ACPI поддерживают все выпущенные за последний десяток лет мейнстримные ОСи (винды, линух, фрибсдя, макось, и другие). Но убрать костыль, как все уже догадались, не позволяет желание сохранить обратную совместимость с доисторическим софтом, а также предпочтением мобильных камушков — управлять собственным кулером самостоятельно, игноря эту директиву (большинство буков Lenovo, Samsung-и почти все, словивший эпичный пиздец на этом эффекте Fujitsu-Siemens и т. п.).

Длинный режим

Упомянутое выше архитектурное расширение до 64 бит представляет собой еще один режим работы, несовместимый с реальным режимом, но немного совместимый с защищенным. В этом режиме доступны шестнадцать 64-битных регистров общего назначения и Over 9000 адресного пространства.

Примечателен тем, что разработчиком этой архитектуры была AMD, в то время как Intel ~~скопиниздила~~ переделала технологию[6] (кое-что не доделав), изменив неполиткорректное вражеское название AMD64 на нейтральное EM64T. Сей факт несколько нетипичен, так как крайне редко Intel что-то так дословно копипастила у AMD, обычно все было наоборот. Скопипиздили, кстати, довольно коряво, и первые 64-битные зионы, например, бодро рапортовали о 40 битах физического адреса, имея всего 36. Однако ты, Анон, должен помнить — никто просто так ничего без последствий не пиздит. Вся загогулина есть в том, что существует соглашение о перекрёстном лицензировании технологий, по условиям которого эти две конторки обязаны(!!!) делиться подобными технологиями. Отак-то!

Несмотря на свою молодость, даже этот режим уже пару раз перепилили по всё той же набившей оскомину причине: совместимость с быдлокодом. Дело в том, что так увлекшись избавлением от атавизма сегментной адресации, инженегры AMD заодно выпилили две древнючие команды: LAHF и SAHF, использовавшиеся в основном для анализа флагов состояния не менее древнючего куска кремния по имени 8087(fstsw ax+sahf). Эти команды существовали со времён дедушки 8086, и занимались пересылкой нижнего байта регистра флагов в аккумулятор и обратно. Начиная с Пня-2, эту команду перестали указывать в документации, тщетно надеясь, что быдло-погромисты забудут этот уродливый костыль, и будут пользоваться богоугодными pushfd+pop reg/push reg+popfd, работающими на новых процах с нулевым начислением обращений в память благодаря хитрожопой штуке «очередь записи», но не тут-то было. Оборзевшие от безнаказанности x86-фаги продолжали совать эти команды везде, до куда добирались, в частности — в софт для виртуализации. Ну а дальше приключилась стандартная история для x86-архитектуры: вместо того, чтобы показать быдлокодерам писю, в архитектуру вставили очередной костыль. Да, мой юный друг, на ранних Athlon 64 и соответствующих им интелах невозможно было запустить 64-битного гостя даже если хост был тоже 64-битный. Такие дела.

VEX префикс

Революционная идея — выпилить старые костыли методом запила новых — пришла на ум инженерам Intel после очередного сеанса раскуривания какого-то нового типа веществ. Однажды, когда в очередной раз стало подходить к концу пространство опкодов, интелевцы задумались: до коих пор, мать твою, нам

городить мелкие костыли, не пора ли запилить такой, чтобы хватило лет этак на десяток? И им пришла в голову гениальная мысль, суть которой в следующем. Давно известно, что размер опкода в x86 — всего один байт. Ну так вот исторически сложилось. И возможных инструкций можно закодировать всего 256. Поначалу (8086) этого хватало даже с лихвой: можно было, не боясь исчерпать опкодовое пространство, для самых частоиспользуемых операций кодировать индекс регистра непосредственно в коде инструкции: ведь команда занимает всего один байт, а память в те времена была по цене золота. Также среди этих кодов существовали так называемые префиксы, сами не кодирующие никакой операции, но немного (или много) меняющие смысл следующего опкода, грубо говоря, рассматривались с ним как единый целый опкод. И таким, и только таким способом можно было расширять однобайтное пространство. О первом глобальном расширении задумались при разработке 80286: тогда в префикс превратили команду «POP CS» за номером 15, доставив баттхёрта некоторым авторам вирусов, активно её использовавшим. Пространство расширилось еще на 255 кодов. Пиздец же начался с эпохи великого и ужасного SSE, когда новые команды стали расти как грибы после дождя. В дело пошло переопределение префиксов REPNE/REPE и OPSIZE. Появились трехбайтовые команды с длинной цепочкой префиксов (и постфиксов). Когда амдшники создавали свой AMD64, они в порыве энтузиазма расправились с однобайтовыми INC/DEC, превратив их в префикс REX. Код стал состоять из префикс-байтов чуть менее, чем наполовину. И вот, свершилось. Вместо этой цепочки переопределенных однобайтных префиксов решили сделать один многобайтный универсальный. В общем — разумное решение. Но! Где взять для него опкод? Ведь однобайтовое пространство уже давно занято. Но ведь гений костылестроения Intel не знает границ! Решили просто: взяли две, уже не совсем нужные в 10-х годах XXI века команды LES и LDS, вспомнив, что они, помимо всего прочего имеют еще и байт-описатель адресации и не могут использовать регистровый операнд. Дырка найдена! Теперь берём команду LES или LDS, кодируем регистровый операнд, а остальные биты — в нашем распоряжении. Выпиливаем все лишние префиксы и ставим вместо них VEX. Можно даже уподобиться RISC'ам и закодировать трех-, четырёх-, и пятиоперандные команды, битов в префиксе хватит (он бывает двух- и трех-байтовым). Правда, некоторые биты приходится делать инверсными, ибо иначе получится LDS или LES, но разве это костыль против такой революции?

Наиболее известные баги

Двойная сигма

Древняя бага, поражала 80386 еще в те времена, когда они не разделились на 386DX и 386SX. Заключалась в том, что ранние 80386 зависали на 32-битном коде. Лулз заключался в том, что даже Intel могла отличить плохой проц от нормального только после тестирования. Была объявлена программа замены, а дабы не терять PROFIT, Intel оттестировала возвращенные б/у процы и повторно выпустила их в продажу, причем хорошие маркировались расовыми буквами ΣΣ (дабл-сигма), а плохие «16 BIT S/W ONLY» (в те времена на 32-битный код многим было похуй). Нынче и то и другое весьма ценится у коллекционеров пруфлинк.

F00F bug

Nuff side

Одним из эпичнейших фейлов Intel была ошибка в процессоре Пентиум в реализации инструкции lock cmpxchg8b с регистровым аргументом (также известная как F00F bug, по первым байтам команды). Заключалась она в том, что любая пользовательская программа могла запросто завесить всю систему. По идее, любая операция с префиксом lock, не обращающаяся к памяти, бессмысленна, и пень это понимал. Но в случае команды cmpxchg8b он тупил, забывая снять блокировку шины, и подвисал после получения адреса обработчика прерывания, так как операции записи не происходило и опаньки. Для преодоления этой проблемы разработчикам ОС приходилось прибегать к нетривиальным изъебствам. Если у вас есть Pentium с ОС Linux вы можете увидеть при загрузке строки «FOOF bug detected, installing workaround», что означает, что вы обладаете бажным процессором Pentium. Фрюха тоже запускает воркараунд, обнаружив первопень.

Арифметический баг Pentium

> —Сколько инженеров Intel нужно, чтобы заменить лампочку?
>
> — 1.9999367, но это вполне приемлемая точность.

— Народное

Кроме того, первые серии Pentium 60/66 MHz (о, веселые 90-е!) весьма пренебрежительно относились к арифметике — а именно, в некоторых случаях операция деления давала неточный результат. Хотя и утверждалось, что ошибка проявляется в одном случае из 9 миллиардов, Интелу пришлось смириться и массово заменить бажные процессоры (предварительно повыебывавшись: дескать, докажите, что вам нужна такая точность).

Ошибка породила немало лулзов, объясняющих, например, переход от численного именования процессоров (80286, 80386) к именам типа «Pentium» и т. д. Путем нехитрых вычислений можно понять, что номер каждого следующего поколения процессоров получался путем сложения номера предыдущего и числа 100. По логике Intel же 486.0 + 100.0 = 585.999996347. Выпускать процессор 585.999996347 Intel

не захотели и дали ему имя «Pentium».

… А если серьёзней, в те далекие, лихие времена, Intel-подобные камни выпускали все кому не лень, при этом оставляли даже систему именования, увеличивая местами циферки — у AMD был Am486, у Cyrix — Cx486 и так далее. По версии Intel такое поведение сторонних фирм негативно отражалось на доходах — гоп-фирмы как бы пользовались этим[7] и выходили в «плюс», а Intel соответственно, в «минус». Жадные до денег манагеры Intel, думали-думали и придумали — раз нельзя сделать циферки торговой маркой, то давайте наш абортарий будет давать имена высерам наших рабов-инженеров. И понеслось…

Джойреактор знает, что проблема существует и по сей день и прописана в куче стандартов.

Народные названия компьютеров и процессоров

- **IBM PC** — по правде говоря, в тогдашние 80-е годы он почти всегда назывался «Персональный Компьютер», с придыханием, ибо стоил в СССР дороже в два раза, чем новые «Жигули».
- IBM PC XT — Эксти, Иксти, ХаТэ, Икстишка.
- IBM PC PS/2 — ПиЭс Пополам, Писипополам, P.S., Постскриптум, Полупись (по аналогии с полуосью — OS/2).
- IBM PC AT — Эйти, Айти, Айтишка, Эйтишка, А-Тэ, Атэшка.
- 286 — Двойка, Двушка.
- 386 — Тройка, Трешка.
- 486 — Четвёрка.
- **Pentium** — Пень, Пентюх, Пенёк, Первопень, Пент, Пётр.
- Pentium Pro — Пропентюх, Пэ-про, Прошка.
- Pentium II — Пень два, Два пенька, Пэ-два, Тупень, Второпень, Второй пень, Двупень.

  Прим.: В конце 90-х, когда процессор характеризовал уровень компьютера в целом, часто были недопонимания, когда говорили «у меня комп — двойка (тройка, четверка)» — мог иметься в виду как PII (III, 4), так и 286 (386, 486).

- Pentium III — Пень три, Пэ-три, Третий пень, Трипень, Тройка, Трёшка, Пятихатка (если 500 МГц).
- Celeron — Селерон, Целерон, Келерон, Целка, Целер, Кселерон, Целик, Селик, Селика, Сельдерей, Цэл; реже: Келерон, Суслерон, Соплерон, Лохотрон, Дохлерон, Карлсон, Селекон, Саурон, Калорон, Затычка для сокета, Заглушка для материнки, Эмулятор процессора.
  - Celeron на ядре Tualatin (P3) — Туалерон, Целерон три.
- Pentium 4 — Четвертый пень, Пэ-четыре, квадропень.
  - Pentium 4 на ядре Prescott — Печка, Духовка, (Пре)Скотина (первые два названия связаны с тем, что процессор неиллюзорно грелся (пример: Pentium 4 630 обладал теплопакетом аж в 80 (sic!) ватт)).
  - Pentium 4 на ядре Cedar Mill — Сидор, Цезарь, Мель.
- Pentium D — Печка Дэ.
- Pentium с Hyper-Threading — Гиперпень
- Xeon — Ксеон, Зеон, Ксенон, Зивон, Неонка.
- Intel Core — Кора, Корка, Корь.
- Intel Core 2 Duo — Кор(е) два дуо, Коре дует, Кора дура, Два дула, Двустволка, Кора дуба, Кора ясеня (Core 2 Duo E7xxx), Дупло, Конура (по названию ядра — *Conroe*), Интел Горо-д-в-а-ду.
- Intel Core 2 Quad — Квад, Квадро, Квадрик, Кор(е) два квад, Квадрат, Кора кедра.
- Intel Core i3 — Кор(е) ай три, и-Зэ, высри, бомж (самый дешёвый и наименее производительный из всех iX).
- Intel Core i5 — Кор(е) ай пять, Айпятый, И-пять, Ай-блять.
- Intel Core i7 — Кор(е) ай семь, И-семь, Айс, Кор топор, Ай сэвэн, Айседьмой, ЙаСемерко.

AMD:

- AMD Duron — Дурон, Дюрон, Лохотрон, Дурень, Дурка, Дурик, Дурдом.
- AMD Sempron — Затычка, Лохотрон, Косарь (цена: 1000 рублей), Тормоз.
- AMD Athlon — Атлон, Аслон, Атхлон, Афлон, Эшлон (sic!); а так же: Печь, Утюг, Утюговый Атлон, Калорифер.
  - Athlon Thunderbird — Громокряк.
- AMD Athlon XP — Атлон Экс Пи, Атлон Ха Пэ, Икс Пи, Атлон Хы Ры; по названию ядра: *Palomino* — Палыч, Паламин, Палпатин; *Thoroughbred* — Срубред, Табурет; *Barton* — Батон, Бартик, Бартон, Батрон, Бартер; *Thorton* — Торт, Тортик, Тортон, Тхортон, Тортер.
- AMD Phenom — Фен, Феном, Пахом, Фенол, Финик, Фенамин, Фенька.
- AMD Opteron — Оптер, Опертон, Оптерон. Серверная версия, малодоступная быдлу, потому сохранила оригинальное название.
- AMD Bulldozer — он же Бульдозер, Трактор, Буль, Ковшик, Кукурузник.
- AMD FX — Эфикс, фуфыкс, фиксик, фикус.
- AMD Ryzen — Восставший из пепла, Рязань, Рызень, Ряженка, Кукурузен, Резина, а за «Княут» уже кое-где банят.

Цитаты

> Отгрохал новый русский особняк. Позвал гостей. Водит, показывает — все в шоке — всё круто. И тут доходят до ванной, а один из гостей спрашивает:
>
> — Чо у тя в ванной за плитка? Какая-то непонтовая! — Да ты знаешь сколько я за нее бабла отвалил?! — Нет, а скока? Хозяин шепнул ему на ухо сумму. Тот, сделав удивленное лицо, говорит: — Фигасе! Платиновая что ли!? А кто делает? — Интел!

— Боян

> Если Вам приспичило иметь PC, покупайте это. Я не буду пытаться отговорить Вас. Я просто отказываюсь рекомендовать изначально ущербные концепции (как аппаратуры, так и программного обеспечения). Я отказываюсь далее способствовать обогащению фирмы Microsoft. Микросхемы Intel убоги по своей сути, и ничего тут поделать невозможно. Такими они и были задуманы. На PC Вы сталкиваетесь с заранее запланированным убожеством каждые несколько лет -то новые программы не годятся для старой аппаратуры, то новая аппаратура не годится для старых прорамм… По какому же заколдованному кругу вы путешествуете, господа пользователи PC? Продолжайте выбрасывать деньги на более быструю аппаратную часть с каждой новой версией ОС и прикладных программ, которые становятся все более медленными и раздутыми. Единственное, на что остается надеятся, так это на то, что ваша техника когда нибудь будет работать, как Амига в 1987 году, да и то вряд ли эти надежды сбудутся…

— Читатель журнала «ZX-Ревю», 1996 год

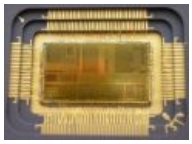
Галерея ЦП

Разик

Двушка

Трёшка

Четвёрка



Гуро

Первопень

Пеньдва. OverDrive — ядро PII под сокет PPro, жуткая редкость, ибо цена как у самолета

Примечания

1. ↑ Линус начал писать лялекс именно на i386, когда заебался разгребать косяки minix. Кстати, поддержка этого процессора была полностью удалена из ядра только в 2012 году [1]
2. ↑ Собственно, поэтому (а также по причине адской дороговизны оперативной памяти в те времена) подавляющее большинство материнских плат для двушек больше намертво запаянного в них одного мегабайта и не содержали.
3. ↑ В последних зионах этот костыль таки выпилили, см. en.w:A20 line
4. ↑ Пруфлинк. TSS теперь нужен только для хранения указателей стеков и карты ввода/вывода, но не для аппаратного переключения задач.

5. ↑ Надо отметить, что все выпиленное в x86-64 продолжает жить в 16- и 32-битных режимах, усложняя и удорожая процессор. Именно поэтому (тепловыделение, цена) Intel Atom проиграет ARM-у в более чем 9000 рыночных нишах для мобильных процессоров. Ни одна современная 64-битная операционная система для платформы x86 не использует ни реальный режим, ни защищенный режим, ни перечисленные выше костыли (кроме исторически сложившейся системы команд). Все эти костыли и рудименты окончательно сдохнут только вместе с совместимостью с ранними моделями x86 и DOS-ом. Однако на данный момент (2010-й год) ни один из производителей отказаться от такой совместимости наглости (или дурости?) не набрался.
6. ↑ Intel на тот момент уже использовала 64-разрядную архитектуру собственной разработки в опять-же своих ЦП Itanium. Но была она несовместима с x86 чуть более чем полностью.
7. ↑ Условие диверсификации поставщиков CPU для *IBM PC* было обязательной частью сотрудничества Intel'а с голубым гигантом.

Ссылки

- Справочный сайт по x86
- Про дырявые процессоры

Девайс

3dfx Amiga An Hero ASUS EEE Brick Game Dreamcast Ellen Feiss IPad IPhone IPod Kirby Made in China MSX N-Gage NES PSP QRBG121-тян RTX Ru mac S-90 VHS Windows Phone 7 Windows Phone 8 X86 Быдлодевайс Вымышленные приборы ГЛОНАСС Говнозеркалка Детектор Дискета Жарков Защита от дурака Зомбоящик Кактус Квадрокоптер Китайский айфон Консоли КТ315 Лятор Магнитофон Ман Маршрутизатор Машина времени Машина Судного дня Мегапиксель Мобилодрочер Муртазин Навител НЛ-10 Она металась, как стрелка осциллографа Пейджер Планшет Поебень Приборчик Радиолюбитель Резонатор Гельмгольца Рингтон Свистелки и перделки Силумин Спектрум Стиллавин Тёплый ламповый звук Тамагочи Терменвокс Терморектальный криптоанализатор Тупые свитчеры Тяни-толкай Фингербокс Циска Экономители Эльдорадо Юность КП101 Яблочник

Игры

1C 3dfx A challenger appears Action 52 Aion Alignment All your base are belong to us Angry Birds Angry Video Game Nerd Another World Arcanum Assassin's Creed Baldur's Gate Barrens chat BASKA Battletoads Beat 'em up BioWare Bitches and whores Blizzard Blood Brick Game Bridget Carmageddon Chris-chan Civilization Combats.ru Command & Conquer Company of Heroes 2 Contra Copyright Corovaneer Online Counter-Strike Crimsonland Crysis Daggerfall Dance Dance Revolution Dangerous Dave Dark Souls Dead Space Demonophobia Denuvo Deus Ex Diablo Did he drop any good loot? Digger Disciples Doki Doki Literature Club! Doom DOOM: Repercussions of Evil Dopefish DotA Dreamcast Duke Nukem 3D Dune 2 Dungeon Keeper Dungeons and Dragons Dwarf Fortress Earthworm Jim Elasto Mania Elite EVE Online Everquest 2 F-19 Falcon Punch Fallout Fate/stay night Five Nights at Freddy's Flashback FPS GAME OVER Game.exe GameDev.ru GamerSuper Garry's Mod Giant Enemy Crab GoHa.Ru Gothic Granado Espada Grand Theft Auto Guilty Gear Guitar Hero Half-Life Half-life.ru Heroes of Might and Magic Hit-and-run Hitman HL Boom Homeworld I.M. Meen Ice-Pick Lodge IDDQD Immolate Improved! It's dangerous to go alone! Take this. Itpedia Jagged Alliance Kantai Collection Katawa Shoujo Kerbal Space Program Killer Instinct

Числа

1 Guy 1 Jar 101-й километр 10:10 1111 12309 127.0.0.1 128 bit 13 14/88 1500 рублей 16 рублей 1917 1984 2 Girls 1 Cup 2 в 1 2000 2012 год 228 25-й кадр 265

28 героев-панфиловцев 282 статья 3,5 анонимуса 3,62 3605 3730 40 кг хурмы 410 42 640 килобайт 666 7:40 90% женщин — изнасилованы 95% населения — идиоты 9600 бод и все-все-все DotA In 5 Seconds IT'S OVER NINE THOUSAND! Leet Monkey Dust Nokia 3310 X86 Автомобильные номера Большой Пиздец/Предполагаемые даты БОЧ рВФ 260602 Веб 1.0 Веб 2.0 Великая теорема Ферма Восьмидесятые Вячеслав Мальцев Гет Двести двадцать Девяностые ДЕЕ1991ГР Деление на ноль Десятые Днепропетровские маньяки Жертвы пранка Закон Парето Звёздные войны Золотой миллиард Зона 51 Инфа 100% Йобибайт Квадратура круга Код Матан Миллиард расстрелянных лично Сталиным Мне 20 и я бородат Мытищи Нулевые Плюс 1 Полшестого Правило 34 Правило 63 Правило трёх секунд Проблема 2000 Простые числа Пятисемит Рулетка Семь чудес света Слава роботам Сотни нефти Стопицот Сырно Тёмная башня Теория относительности Три обезьяны Тринадцать миллионов педофилов Число Грэма Число Эрдёша Чуров Чуть более, чем наполовину Эльф 80-го уровня

w:X86 en.w:X86