

Архитектура современных ВУ / Архитектура ЭВМ и системное ПО Лекции 4-5. Центральный процессор

П. А. Макаров



октябрь 2024 г.

1. Общее представление о регистрах и системе команд
2. Архитектура и микроархитектура процессора
3. Основные классы архитектур процессоров
4. Процессоры семейства x86
5. Технологии повышения производительности процессоров

Общее представление о регистрах и системе команд

Типы регистров процессора

- ▶ Регистры общего назначения (GPR — General Purpose Registers).
- ▶ Регистр команды (IR — Instruction Register).
- ▶ Регистр или счётчик адреса команды (IP — Instruction Pointer).
- ▶ Регистр адреса (операнда).
- ▶ Регистр числа.
- ▶ Регистр результата.
- ▶ Сумматор. Аккумулятор.
- ▶ Регистр состояния (SR — Status Register) или регистр флагов (FLAGS).

Общее представление о регистрах и системе команд

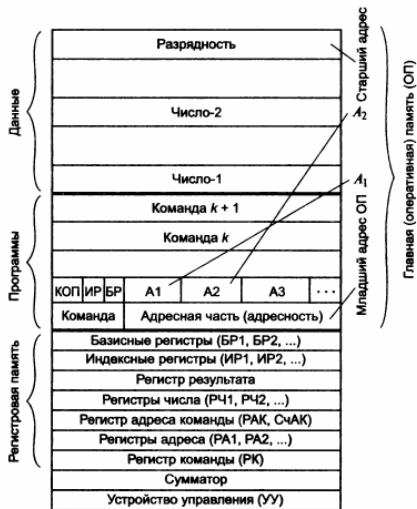


Рис. 1: Структура простейшего центрального устройства ВМ

Общее представление о регистрах и системе команд

Цикл выполнения короткой команды

1. В соответствии с содержимым IP УУ извлекает из ОП очередную команду и помещает её в IR. Некоторые команды УУ обрабатывает самостоятельно, без привлечения АЛУ (пример: `jmp 2478`).
2. Осуществляется расшифровка (декодирование) команды.
3. Адреса A1, A2 и др. помещаются в регистры адреса.
4. Если в команде указаны индексный или базовый регистры, то их содержимое используется для модификации регистра адреса.
5. По значениям регистра адреса осуществляется чтение чисел (строк) и помещение их в регистр числа.
6. Выполнение операции и помещение результата в регистр результата.
7. Запись результата по одному из адресов.
8. Увеличение содержимого IP.

Общее представление о регистрах и системе команд

Система команд

Определение 1

*Полный набор всех команд или инструкций, которые может выполнить процессор называется **системой команд**.*

Типы команд современных процессоров

1. Привилегированные.
2. Чувствительные.
3. Обычные.

Команды классифицируются:

- ▶ по функциям (выполняемым операциям),
- ▶ направлению приёма-передачи информации,
- ▶ адресности.

Общее представление о регистрах и системе команд

Понятия, связанные с системой команд

Команды представляются двоичными числами, в состав которых могут включаться следующие составляющие, называемые полями:

1. код команды;
2. формат команды, если она имеет переменный формат;
3. адреса операндов и результата, которые могут делиться на поля способа адресации и собственно адреса.

Определение 2

Организация командного слова, определяющая количество и размеры её полей, называется форматом команды, а её внутреннее строение — структурой команды.

Определение 3

Если все команды допускают все способы адресации, то такая система команд называется симметричной.

Общее представление о регистрах и системе команд

Основные признаки классификации и типы команд процессоров — начало

- ▶ Команды пересылки: mov.
- ▶ Команды ввода/вывода: in, out.
- ▶ Команды обработки данных.
 - ▶ Короткие операции (один такт):
 - ▶ Логические: not, and, or, xor, test;
 - ▶ Арифметические: neg, add, inc, sub, dec, mul, div, cmp;
 - ▶ Команды сдвига: shr, ror, rcr.
 - ▶ Длинные операции (несколько тактов):
 - ▶ Сложение/вычитание с плавающей точкой;
 - ▶ Умножение/деление с фикс. и плав. точкой.
- ▶ Операции управления.
 - ▶ Безусловный переход (ветвление): jmp.
 - ▶ Вызов подпрограммы: call.
 - ▶ Условные переходы: je, jne, jg, jl.
 - ▶ Команды организации программных циклов: loop.
 - ▶ Команды прерывания: int.
 - ▶ Команды изменения признаков.
 - ▶ Команды управления процессором.

Общее представление о регистрах и системе команд

Основные признаки классификации и типы команд процессоров — продолжение

- ▶ Тип выборки и пересылок данных:
 - ▶ регистр — регистр;
 - ▶ регистр — память;
 - ▶ память — память.
- ▶ Адресация
 - ▶ Непосредственная.
 - ▶ Регистровая.
 - ▶ Прямая.
 - ▶ Относительная.
 - ▶ Косвенно-регистровая.
 - ▶ Косвенно-регистровая со смещением.
 - ▶ Косвенно-регистровая с индексированием и смещением.

Общее представление о регистрах и системе команд

Основные признаки классификации и типы команд процессоров — завершение

▶ Адресность

- ▶ Одноадресные. `OpCODE A1`
- ▶ Двухадресные. `OpCODE A1 A2`
- ▶ Трёхадресные. `OpCODE A1 A2 A3`
- ▶ Безадресные. `OpCODE data`
- ▶ Комбинированные. `OpCODE addr data`

▶ Другие признаки

- ▶ Операции с фиксированной точкой.
- ▶ Операции с плавающей точкой.
- ▶ Десятичная арифметика.
- ▶ Символьная обработка.
- ▶ Обработка чисел большой длины.
- ▶ Векторные операции.
- ▶ Команды индексной арифметики.

Общее представление о регистрах и системе команд

Классификация архитектур систем команд по типам памяти

Архитектуры системы команд по типам памяти

- ▶ стековая,
- ▶ аккумуляторная,
- ▶ регистровая (Load — Store),
- ▶ регистр — память,
- ▶ память — память.

Общее представление о регистрах и системе команд

Пример выполнения операции $C = A + B$ для разных архитектур систем команд

Стековая

push A

push B

add

pop C

Общее представление о регистрах и системе команд

Пример выполнения операции $C = A + B$ для разных архитектур систем команд

Стековая

```
push A  
push B  
add  
pop C
```

Аккумуляторная

```
load A  
add B  
store C
```

Общее представление о регистрах и системе команд

Пример выполнения операции $C = A + B$ для разных архитектур систем команд

Стековая

```
push A  
push B  
add  
pop C
```

Аккумуляторная

```
load A  
add B  
store C
```

Регистровая

```
load R1, A  
load R2, B  
add R3, R1, R2  
store R3, C
```

Общее представление о регистрах и системе команд

Пример выполнения операции $C = A + B$ для разных архитектур систем команд

Стековая

```
push A  
push B  
add  
pop C
```

Аккумуляторная

```
load A  
add B  
store C
```

Регистровая

```
load R1, A  
load R2, B  
add R3, R1, R2  
store R3, C
```

Регистр — память

```
load R1, A  
add R2, R1, B  
store R2, C
```

Общее представление о регистрах и системе команд

Преимущества регистровой архитектуры

Во всех компьютерах, разработанных после 1980 года, используется регистровая архитектура Load — Store. Этому способствовали две причины:

1. быстрый доступ к регистрам,
2. большая эффективность для компиляторов.

Пример 1

$$A \cdot B - B \cdot C + A \cdot D. \quad (1)$$

Архитектурой регистр — регистр обладают процессоры Alpha, ARM, MIPS, PowerPC, SPARC.

Архитектура регистр — память характерна для процессоров Intel 80x86, Motorola MC680x0.

Общее представление о регистрах и системе команд

Порядок байт и выравнивание информации

Данные в памяти могут располагаться (byte order)

- ▶ от младшего к старшему — Little Endian (SPARC, Motorola 68000),
- ▶ от старшего к младшему — Big Endian (x86, VAX),
- ▶ в переключаемом порядке — Bi-Endian (PowerPC, ARM, IA-64),
- ▶ в смешанном порядке.

Как правило, информация в памяти должна быть выровнена (Aligned), т. е. слово должно иметь чётный адрес, двойное слово — кратный четырём и т. д. В некоторых процессорах допускается невыровненное расположение информации, что сопровождается её чтением за большее число тактов, а аппаратная составляющая усложняется.

Определение 4

Под архитектурой центрального процессора понимается его программная модель, то есть основные программно-видимые свойства. К основным программно-видимым свойствам относят:

- ▶ набор регистров,
- ▶ систему команд,
- ▶ механизм обработки прерываний.

Пример 2

Процессоры семейства x86 — представители CISC-архитектуры, т. к. по сложности системы команд им нет равных, при этом базовых архитектурных регистров — мало.

Определение 5

Микроархитектура — это конкретная внутренняя реализация данной программной модели. Для одной и той же архитектуры, например, IA-32, применяются существенно различающиеся микроархитектурные реализации.

Начиная с процессоров семейства x86 шестого поколения (Intel Pentium Pro и AMD K5) в микроархитектуре центрального процессора применяется RISC-ядро, исполняющее микрооперации, на которые раскладываются сложные инструкции x86.

Пример 3

Компьютеры Power Macintosh фирмы Apple основаны на процессорах архитектуры PowerPC (PPC), разработанной в 1991 году альянсом компаний Apple, IBM и Motorola (AIM). Power Mac обеспечивает ту же прикладную производительность, что и IBM PC, но на более низких тактовых частотах центрального процессора.

Основные классы архитектур процессоров

CISC (Complex Instruction Set Computer)

Типичным примером CISC являются процессоры Intel x86. Они выполняют более 200 команд разной степени сложности, которые имеют размер от 1 до 15 байт, и обеспечивают более десяти различных способов адресации.

Такое многообразие выполняемых команд и способов адресации позволяет программисту реализовать наиболее эффективные алгоритмы решения различных задач. Однако при этом существенно усложняется структура процессора, особенно его устройства управления, что приводит к увеличению размеров и стоимости кристалла, снижению производительности.

Архитектура CISC характеризуется так же малым числом регистров процессора, что порождает конфликты по данным при конвейеризации вычислений.

Основные классы архитектур процессоров

RISC (Reduced Instruction Set Computer)

Отличается использованием ограниченного набора команд фиксированного формата. Современные RISC-процессоры обычно реализуют около 100 команд, имеющих фиксированный формат длиной 4 байта. Также значительно сокращается число используемых способов адресации. При этом для сокращения количества обращений к памяти RISC-процессоры имеют увеличенный объём внутренней регистровой памяти — от 32 до нескольких сотен регистров. В результате процессор на 20–30% реже обращается к оперативной памяти, что также повышает скорость обработки данных. Упростилась топология процессора, выполняемого в виде одной интегральной схемы, сократились сроки её разработки, она стала дешевле.

ARM — Advanced RISC Machine, лицензиатами которой являются множество известных компаний (AMD, Apple, Analog Devices, Atmel, Xilinx, Cirrus Logic, Intel, Marvell, NXP, STMicroelectronics, Samsung, LG, MediaTek, Qualcomm, Sony, Texas Instruments, Nvidia, Freescale, Миландр, HiSilicon, Байкал электроникс).

Основные классы архитектур процессоров

VLIW (Very Large Instruction Word)

Её особенностью является использование очень длинных команд (256 бит и более), отдельные поля которых содержат коды, обеспечивающие выполнение различных операций.

Архитектура VLIW является развитием RISC-архитектуры применительно к суперконвейерным структурам с управлением выполнением команд на основе потока данных, т. е. не в порядке их расположения в программе, а по мере готовности операндов и освобождения соответствующих операционных блоков.

Специальный компилятор планирования VLIW перед выполнением прикладной программы проводит её анализ и по множеству ветвей последовательности операций определяет группу команд, которые могут выполняться параллельно. Каждая такая группа образует одну сверхдлинную команду.

Процессоры типа VLIW выпускают фирмы Transmeta, Intel и Hewlett-Packard. К VLIW-типу можно отнести и процессоры семейства Эльбрус.

Основные классы архитектур процессоров

EPIC (Explicitly Parallel Instruction Computing)

Архитектура EPIC — совместная разработка фирм Intel и Hewlett-Packard, развивающая VLIW-архитектуру. Архитектура EPIC реализована фирмой Intel в 64-разрядном процессоре Itanium и получила название IA-64. В этом процессоре имеется по 120 64-битных регистров общего назначения и 80-битных регистров для данных с плавающей точкой, а также 64-битный регистр предикатов. Команды по 3 штуки упаковываются компилятором в связки (Bundle) длиной 128 бит.

Дополнительно к архитектуре VLIW связка имеет специальное поле — шаблон, где указывается зависимость или её отсутствие команд друг от друга в связке и между связками. Благодаря указанным в шаблоне зависимостям между связками, процессору с N группами функциональных блоков (одна группа содержит три блока) будет соответствовать связка из $3N$ команд.

Основные классы архитектур процессоров

MISC (Minimal Instruction Set Computer)

Процессор MISC (Minimal Instruction Set Computer) работает с минимальным набором длинных команд и характеризуется небольшим набором чаще всего встречающихся команд. Вместе с этим, в данной архитектуре активно применяется выполнение групп команд за один цикл работы процессора (при этом несколько команд “сливаются” в сверхдлинные командные слова). Порядок выполнения команд при этом распределяется так, чтобы в максимальной степени загрузить маршруты, по которым проходят потоки данных. Таким образом, архитектура MISC объединяет вместе суперскалярную (многопоточную) и VLIW концепции. Компоненты процессора просты и работают с высокими скоростями. Ещё одной особенностью архитектуры MISC является использование стековой организации.

Основные классы архитектур процессоров

MIPS (Microprocessor without Interlocked Pipeline Stages)

MIPS (без блокировок в конвейере) — система команд и микропроцессорных архитектур, разработанных компанией MIPS Computer Systems (в 2018 году приобретена корпорацией Wave Computing) в соответствии с концепцией проектирования процессоров RISC (то есть для процессоров с упрощенным набором команд). Ранние модели процессора имели 32-битное машинное слово, позднее появились его 64-битные версии. Существует множество модификаций процессора, включая MIPS I, MIPS II, MIPS III, MIPS IV, MIPS V, MIPS32 и MIPS64, из них действующими являются последние две. Различные реализации MIPS используются в основном во встраиваемых системах, например, в смартфонах, маршрутизаторах, шлюзах, а также до 2010-х годов широко применялись в игровых консолях.

Основные классы архитектур процессоров

Комбинированные системы команд

CISC — RISC, CISC — RISC — VLIW являются попытками воспользоваться преимуществами объединённых систем, например, более удобным программированием, используя сложные команды CISC при создании компиляторов и ОС, а RISC-команды как внутренний инструмент обработки информации на более простой аппаратуре устройства управления. При этом исходные сложные команды во время дешифровки заменяются одной, двумя или несколькими RISC-командами.

Подобная система впервые появилась в процессоре Intel Pentium Pro и продолжает использоваться в архитектуре IA-32 (Intel Architecture). В процессорах AMD K8, Intel Pentium 4 можно увидеть уже и элементы VLIW-архитектуры, поскольку на одной из ступеней конвейера RISC-команды объединяются в тройки, запускаемые на исполнение в функциональные блоки.

Процессоры семейства x86

Предшественники x86 — Intel 8008

Первый 8-разрядный микропроцессор (1972 г.). Этот процессор содержал 3 500 транзисторов, работал на частоте 500 кГц при длительности машинного цикла 20 мкс (10 периодов задающего генератора) и в отличие от предшественников имел архитектуру принстонского типа (компьютер с единой памятью для команд и данных). В нём допускалось применение комбинаций постоянной и оперативной памяти. Значительные изменения (кроме увеличения разрядности) произошли и в регистровом файле. Из-за ограниченных возможностей применяемой технологии в качестве блока регистров общего назначения была применена динамическая память. Большинство команд микропроцессор выполнял за 1–3 машинных цикла. Для работы с медленно действующими устройствами был введен сигнал готовности (*ready*). Система команд содержала 65 инструкций и отличалась значительным количеством команд условного перехода, логических команд и команд сдвигов. Процессор мог адресовать память объемом 16 Кбайт. Однако объем и организация стека остались прежними, — реализация операций со стеком возлагалась на программиста. Узкий интерфейс связи с «внешним миром» требовал применения около 20 схем средней интеграции для сопряжения процессора с памятью и устройствами ввода-вывода.

Процессоры семейства x86

Предшественники x86 — Zilog Z80

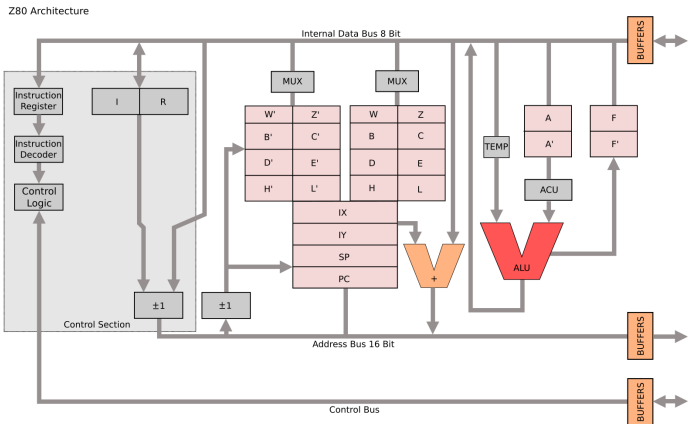


Рис. 2: Архитектура процессора Zilog Z80

Процессоры семейства x86

Предшественники x86 — Motorola MC6800

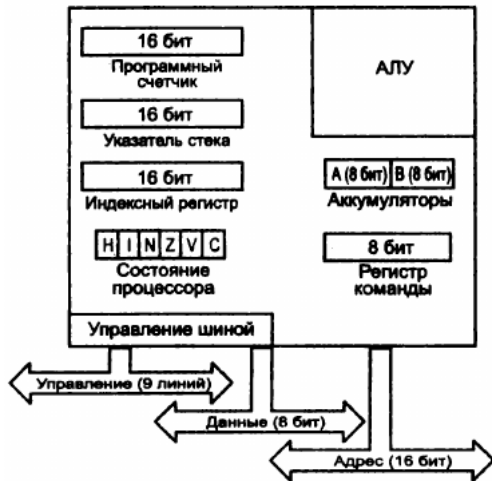


Рис. 3: Архитектура процессора Motorola MC6800

Процессоры семейства x86

Предшественники x86 — Intel 8080

Intel 8080 Architecture

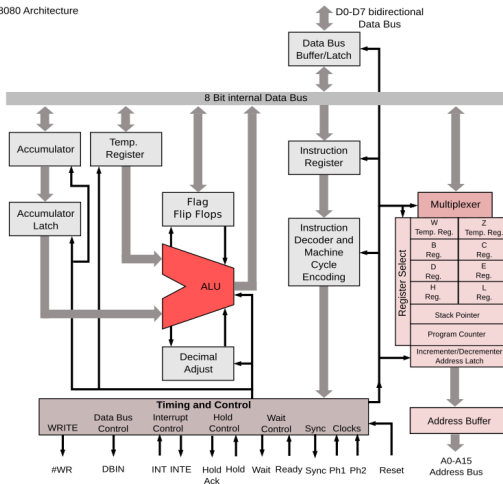


Рис. 4: Архитектура процессора Intel 8080

Процессоры семейства x86

Первое поколение x86 — 8086 и 8088

Содержал 29 000 транзисторов. Высокое быстродействие элементов обеспечило тактовую частоту 5 МГц, а 16-разрядная архитектура и машинный цикл 200 нс — производительность, превышающую аналогичный параметр 8080 на порядок. Структура микропроцессора была полностью перестроена, он как бы был разделен на два одновременно работающих функциональных блока. Это операционный блок (EU — Execution Unit) и блок интерфейса (BIO — Bus Interface Unit). В результате исполнение одной команды совмещалось во времени с выборкой следующей команды или данных из памяти. В ЦП появился небольшой буфер команд, что давало дополнительную экономию времени при обращениях к памяти.

Адресация 1 Мбайт оперативной памяти (благодаря 20 адресным линиям) и её сегментация могут быть отнесены к одним из наиболее существенных достижений. Сегментация памяти и большое число уровней прерываний были ориентированы на работу системы в многозадачном режиме. Однако механизм защиты памяти пока реализован не был.

Процессоры семейства x86

Архитектура и микроархитектура базового процессора x86

Определение 6

Базовый микропроцессор семейства x86 — это упрощённая схема процессора, основными устройствами которого являются 16-разрядные регистры и АЛУ процессора Intel 80286.

- ▶ Адресное пространство — 16 Мбайт (24-разрядная ША).
- ▶ Поддержка виртуальной памяти (ёмкостью до 1 ГБ).
- ▶ Аппаратная мультизадачность.
- ▶ Повышенное быстродействие (4, 8 МГц, клоны — 10, 12.5, 16 и 20 МГц).
- ▶ Встроенная система управления памятью и средства её защиты.
- ▶ Дополнение системы команд 16 новыми инструкциями.
- ▶ Размещение на одном кристалле контроллера прерываний и DMA, а также таймера и системного генератора.

Процессоры семейства x86

Архитектура базового процессора x86

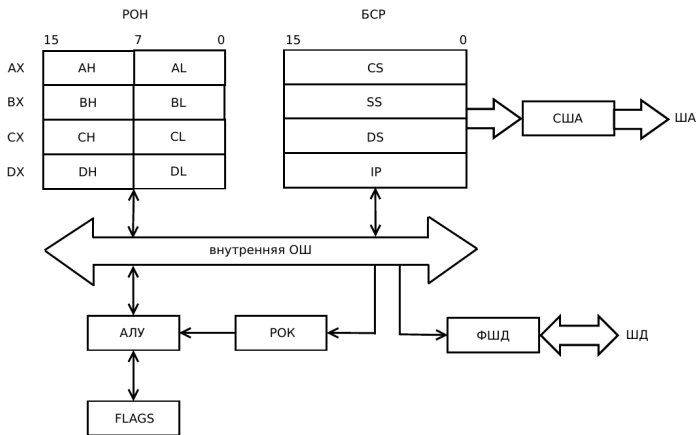


Рис. 5: Упрощённая схема процессора Intel 80286

Процессоры семейства x86

Регистры базового процессора

| Обозначение | Название | Старший байт | Младший байт |
|-------------|---------------------|--------------|--------------|
| AX | Accumulator | AH | AL |
| BX | Base | BH | BL |
| CX | Counter | CH | CL |
| DX | Data | DH | DL |
| SP | Stack Pointer | - | - |
| BP | Base Pointer | - | - |
| SI | Source Index | - | - |
| DI | Destination Index | - | - |
| CS | Code Segment | - | - |
| DS | Data Segment | - | - |
| SS | Stack Segment | - | - |
| ES | Extra Segment | - | - |
| IP | Instruction Pointer | - | - |

Таблица 1: Регистры процессора 80286

Процессоры семейства x86

Регистр флагов базового процессора

| | | | | | | | | | | | | | | | | |
|-------|----|----|----|----|----|----|----|----|----|----|---|----|---|----|---|----|
| Биты | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Флаги | - | - | - | - | OF | DF | IF | TF | SF | ZF | - | AF | - | PF | - | CF |

Таблица 2: Регистр флагов процессора 80286

Флаги условий

- ▶ CF — Carry Flag.
- ▶ OF — Overflow Flag.
- ▶ ZF — Zero Flag.
- ▶ SF — Sign Flag.
- ▶ PF — Parity Flag.
- ▶ AF — Auxiliary carry Flag.

Флаги состояний

- ▶ DF — Direction Flag.
- ▶ IF — Interrupt Flag.
- ▶ TF — Trap Flag.

Процессоры семейства x86

Режимы работы базового процессора

Режимы работы процессора Intel 80286

1. Real Address Mode (в нём процессор 80286 действует аналогично 8086, что обеспечивает обратную совместимость с ПО, написанным для 8086, а также возможность работы под управлением ОС MS DOS).
2. Protected Virtual Address Mode (в этом режиме реализуется режим виртуальной памяти, аппаратная мультизадачность и адресация к большему пространству памяти).

В 1984 г. процессоры i80286 были использованы в компьютерах IBM PC AT.

ОС MS DOS может работать только в реальном режиме.

Другие ОС, например, OS/2 (предложенная фирмой IBM альтернатива DOS) и UNIX (XENIX и AIX), могут использовать защищённый режим.

Процессоры семейства x86

Архитектура IA-32 (x86-32)

Главные особенности процессоров Intel 80386SX и 80386DX

1. 32-разрядный ввод-вывод и адресация памяти (реальная — до 4 Гбайт, и до 64 Тбайт виртуальной памяти).
2. Рабочая тактовая частота — 33 МГц.
3. Аппаратные системы управления памятью и защиты.
4. Средства работы с виртуальной памятью со страничной организацией (MMU = SU + PU).
5. Использование кэш-памяти.

Режимы работы процессора Intel 80386

1. Real Address Mode.
2. Protected Virtual Address Mode.
3. Virtual 8086 Real Mode.

Процессоры семейства x86

Регистры IA-32

| Имя | Номер | Назначение |
|-------------------|-------|--|
| Регистры данных | | |
| EAX | 000 | Выделенный сумматор, используемый во всех основных вычислениях |
| ECX | 001 | Универсальный счётчик циклов |
| EDX | 010 | Регистр данных, дополняющий сумматор |
| EBX | 011 | Буферное запоминающее устройство; в 16-битном режиме — указатель |
| Адресные регистры | | |
| ESP | 100 | Указатель стека |
| EBP | 101 | Указатель базы. Используется для хранения адреса текущего фрейма стека. Иногда применяется в качестве буфера |
| ESI | 110 | Индекс источника |
| EDI | 111 | Индекс назначения |

Таблица 3: Регистры общего назначения IA-32

Системные регистры

- ▶ сегментные (segment),
- ▶ контрольные (control),
- ▶ отладочные (debug),
- ▶ тестовые (test).

Технологии повышения производительности процессоров

Операции над вещественными числами (с плавающей точкой)

FPU (Floating Point Unit — блок чисел/операций с плавающей точкой) или NPX (Numeric Processor eXtension — числовое расширение процессора).

Начиная с 80486 восемь регистров математического сопроцессора ($ST(0) - ST(7)$) встраивают в ЦП. Каждый из этих регистров имеет разрядность 80 бит и хранит числа в формате расширенной точности стандарта IEEE 754 (15 бит — экспонента, 64 — мантисса).

Эти регистры доступны в стековом порядке. Имена (номера) регистров устанавливаются относительно вершины стека ($ST(0)$ — вершина стека, $ST(1)$ — следующий регистр ниже вершины стека, $ST(2)$ — второй после вершины стека и т. д.).

Технологии повышения производительности процессоров

Обработка команды или цикл процессора

Может быть разделена на несколько основных этапов (микрокоманд), которых как минимум пять (выборка, декодирование, чтение исходных данных, выполнение, запись результата). Каждая короткая операция требует для своего выполнения времени, равного такту генератора процессора (tick of the internal clock).



Рис. 6: Выполнение команд без конвейеризации

Технологии повышения производительности процессоров

Простой конвейер (pipeline)

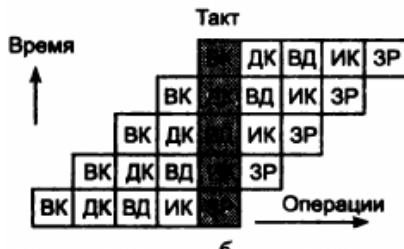


Рис. 7: Пятиступенчатый конвейер

Коэффициент ускорения S в идеальном случае

$$S = \frac{N_c \cdot N_s}{N_c + N_s - 1}. \quad (2)$$

Технологии повышения производительности процессоров

Конфликты в конвейерах и скалярные процессоры

Под конфликтом конвейера понимается невозможность выполнения команды в соответствующем ей такте. При этом в конвейер нельзя загрузить очередную команду и его производительность падает. Выделяют три вида конфликтов:

- ▶ структурные;
- ▶ конфликты по данным;
- ▶ конфликты по управлению.

Определение 7

Скалярный процессор — это процессор с единственным конвейером (к этому типу процессоров относятся, например, все процессоры Intel до 486 включительно).

Технологии повышения производительности процессоров

Суперскалярные процессоры

Определение 8

Суперскалярный процессор имеет более одной линии конвейера. В суперскалярных процессорах используется параллелизм на уровне команд.



Рис. 8: Суперскалярный конвейер из двух линий

Технологии повышения производительности процессоров

Увеличение разрядности систем

Тип ВМ и их разрядность (по состоянию на 1980-е годы)

- ▶ микроЭВМ — 8 разрядов;
- ▶ мини-ЭВМ — 16 разрядов;
- ▶ большие ЭВМ — 32 разряда;
- ▶ сверхбольшие (супер) ЭВМ — 64 разряда.

В процессе развития микропроцессоров рубежи в 16 и 32 разряда (IA-32) были преодолены довольно быстро, а в начале 2000-х г. в процессорах фирм Intel и AMD произошел переход на 64-разрядные архитектуры.

Технологии повышения производительности процессоров IA-64

Связь с IA-32 только по названию, т. к. архитектура IA-64 не совместима непосредственно с набором команд IA-32. Здесь появляется полностью отличный набор команд, а также используются принципы VLIW. Архитектура IA-64 используется линией процессоров Itanium.

Усовершенствования:

- ▶ в 16 раз увеличено количество регистров общего назначения и регистров для чисел с плавающей точкой (теперь по 128);
- ▶ механизм переименования/ротации регистров, чтобы сохранять значения в регистрах при вызове функций.

Технологии повышения производительности процессоров AMD64

Набор команд AMD64 (x86-64) построен на основе IA-32 и обеспечивает наследственность семейства x86. При расширении набора команд компания AMD избавилась от части устаревших команд — «16-bit legacy».

Основные усовершенствования

- ▶ В два раза увеличено количество регистров общего назначения и SSE (теперь по 16);
- ▶ регистры общего назначения (RAX–RDI, R8–R15) теперь действительно **общего назначения** и ничем больше не ограничены.

Пример 4

Архитектура использована в Athlon 64, Athlon 64 X2, Athlon 64 FX, Opteron, Turion 64, Turion 64 X2, Sempron («Palermo», «Manila»).

Технологии повышения производительности процессоров EM64T или Intel 64

EM64T — Extended Memory 64-bit Technology (Yamhill). EM64T совместим с кодами, написанными для AMD64, хотя и имеет ряд недостатков сравнительно с AMD64.

Пример 5

EMT/Intel 64 используется в ЦП архитектуры Intel NetBurst — Xeon («Nocona»), Celeron D («Prescott» и далее), Pentium 4 («Prescott» и далее), Pentium D, Pentium Extreme Edition и архитектуры Intel Core — Xeon («Woodcrest»), Intel Core 2.

Замечание 1

Поскольку AMD64 и EMT64 почти не различаются, для ссылки на них используются нейтральные названия — x86-64, x86_64 (Linux и Apple Mac OS X), x64 (Microsoft и Sun Microsystems).

Технологии повышения производительности процессоров

Регистры общего назначения x86-64 — начало

| 64-битный регистр | младшие 32 бита | младшие 16 бит | |
|----------------------|--------------------|----------------|--------|
| | | Байт 1 | Байт 0 |
| RAX | EAX | AH | AL |
| RBX | EBX | BH | BL |
| RCX | ECX | CH | CL |
| RDX | EDX | DH | DL |
| RSI | ESI | - | SIL |
| RDI | EDI | - | DIL |
| RBP | EBP | - | BPL |
| RSP | ESP | - | SPL |

Таблица 3: “Классические” регистры общего назначения x86-64

Технологии повышения производительности процессоров

Регистры общего назначения x86-64 — завершение

| 64-битный регистр | младшие | | |
|----------------------|---------|--------|-------|
| | 32 бита | 16 бит | 8 бит |
| R8 | R8D | R8W | R8B |
| R9 | R9D | R9W | R9B |
| R10 | R10D | R10W | R10B |
| R11 | R11D | R11W | R11B |
| R12 | R12D | R12W | R12B |
| R13 | R13D | R13W | R13B |
| R14 | R14D | R14W | R14B |
| R15 | R15D | R15W | R15B |

Таблица 4: “Новые” регистры общего назначения x86-64

Технологии повышения производительности процессоров

Векторная обработка (SIMD-команды) — MMX

MMX (MultiMedia eXtension) — архитектура системы команд (57 команд для арифметики с фиксированной точкой), предназначенных для задач мультимедиа, связи и графики. Их общие свойства:

- ▶ небольшая разрядность целочисленных данных (8-разрядные пиксели или 16-разрядный звук);
- ▶ небольшая длина циклов, но большое число их повторений;
- ▶ большой объём вычислений и значительное количество операций умножения и накопления;
- ▶ существенный параллелизм операций в программах.

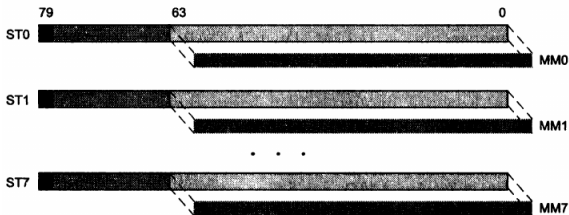


Рис. 9: Совмещение стека регистров FPU с MMX-регистрами

Технологии повышения производительности процессоров

Упакованные форматы данных MMX (знаковые и беззнаковые)

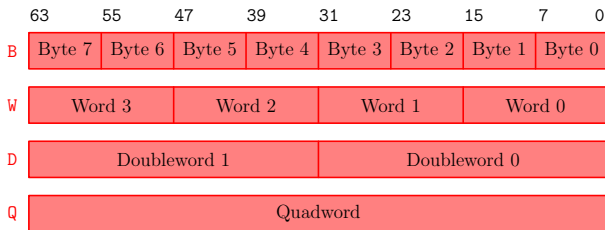


Рис. 10: Упакованные и неупакованные форматы данных MMX

1. Packed byte (B) — восемь упакованных байт;
2. Packed word (W) — четыре упакованных слова;
3. Packed doubleword (D) — два упакованных двойных слова;
4. Quadword (Q) — учетверённое слово.

Технологии повышения производительности процессоров

Синтаксис MMX-команд

instruction [src, dest]

Суффикс, определяющий тип данных и арифметику

- ▶ US — Unsigned Saturation;
- ▶ S или SS — Signed Saturation. Если в суффиксе нет ни S, ни SS, используется циклическая арифметика (Wrap Around);
- ▶ B, W, D, Q — тип данных. Если в суффиксе две из этих букв, то первая соответствует входному операнду, а вторая — выходному.

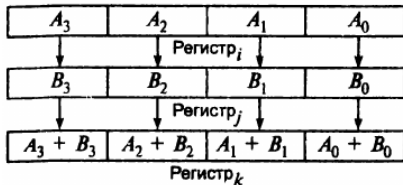


Рис. 11: Команда paddusw — пример векторной операции над упакованными словами

Технологии повышения производительности процессоров 3DNow!

Архитектура 3DNow! впервые реализована в процессорах AMD K6-2 (май 1998 г.). Технология 3DNow! включает 21 дополнительную команду, новые типы данных и использует регистры MMX (MM0 – MM7) для поддержки высокопроизводительной обработки 3D-графики и звука. В то время как архитектура MMX предполагает целочисленную арифметику, векторные команды 3DNow! параллельно обрабатывают две пары 32-разрядных вещественных операндов одинарной точности. Процессор может выполнять две 3DNow! команды за такт и, следовательно, 4 операции над числами с плавающей точкой одновременно. Технология 3DNow! предполагает наличие в ЦП для выполнения векторных MMX и 3DNow! операций нескольких устройств (пара умножителей, сумматоров и т. д.). Все команды 3DNow! имеют длительность исполнения 2 такта и полностью конвейеризированы.

Технологии повышения производительности процессоров

SSE — Streaming SIMD Extension

SSE (или SIMD-FP) — система команд предложенная Intel в 1999 г. в Pentium III (ядро Katmai). Это 70 новых команд, в числе которых выделяются следующие.

- ▶ 50 команд предназначены для повышения эффективности операций над числами с плавающей точкой. С этой целью в ЦП встроены 128-битовые регистры — восемь регистров, названные XMM0–XMM7 (в AMD64 число SSE/XMM регистров было увеличено от 8 до 16). В результате операции с плавающей точкой могут совершаться за один цикл процессора.
- ▶ 12 команд (New Media) дополняют ранее введенные 57 команд MMX для чисел с фиксированной точкой.
- ▶ 8 команд (New Cacheability) повышают производительность кэш-памяти L1 при работе с мультимедийными данными.

SSE — набор команд, которые обрабатывают только значения с плавающей точкой. Т. к. используются более длинные регистры, SSE может упаковать в каждый регистр четыре числа с плавающей точкой. SSE (как и 3DNow!) ограничено только числами одинарной точности.

Технологии повышения производительности процессоров SSE2

SSE2 — введенный с Pentium IV набор команд является существенным развитием SSE, оперирует с теми же самыми регистрами и обратно совместим с SSE процессора Pentium III. В расширении SSE2 операции со 128-битовыми регистрами могут выполняться не только как с четвёрками вещественных чисел одинарной точности, но и как с парами вещественных чисел двойной точности, с шестнадцатью однобайтовыми целыми и пр. SSE2 представляет собой симбиоз MMX и SSE, и позволяет работать с любыми типами данных, вмещающимися в 128-битовые регистры.

Технологии повышения производительности процессоров SSE3

SSE3 — набор команд, также известный как Prescott New Instructions (PNI), является третьей версией команд SSE для IA-32. Intel использует SSE3 с начала 2004 г. в ЦП Pentium IV Prescott. В апреле 2005 г. AMD также включает SSE3 в ЦП Athlon 64 (версия E — ядра Venice и San Diego). SSE3 содержит 13 дополнительных по отношению к SSE2 команд. Самое существенное новшество — «горизонтальная арифметика».

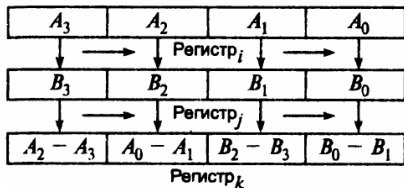


Рис. 12: SSE3-инструкция `hsubps` — Horizontal-SUBtract-Packed-Single — горизонтальное вычитание упакованных слов одинарной длины

Технологии повышения производительности процессоров

Динамическое исполнение (dynamic execution technology)

Определение 9

Динамическое исполнение — технология обработки данных процессором, обеспечивающая более эффективную работу процессора за счет манипулирования данными, а не просто линейного исполнения списка инструкций.

- ▶ Предсказание ветвлений.
- ▶ Внеочередное выполнение.
- ▶ Переименование (ротация) регистров (register rename).
- ▶ Выполнение по предположению (спекулятивное исполнение):
 - ▶ предикация (predication);
 - ▶ опережающее чтение данных (speculative loading);

Эти возможности осуществляются комбинированно — при компиляции и выполнении программы.

Технологии повышения производительности процессоров

Предикация

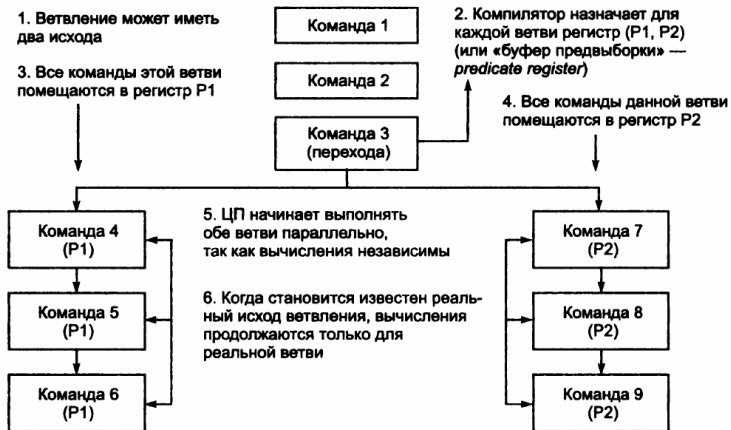


Рис. 13: Предикация всех альтернативных ветвей оператора условного перехода

Технологии повышения производительности процессоров

Опережающее чтение



Рис. 14: Опережающее считывание данных в регистры ЦП из памяти

Технологии повышения производительности процессоров

Множественное декодирование команд

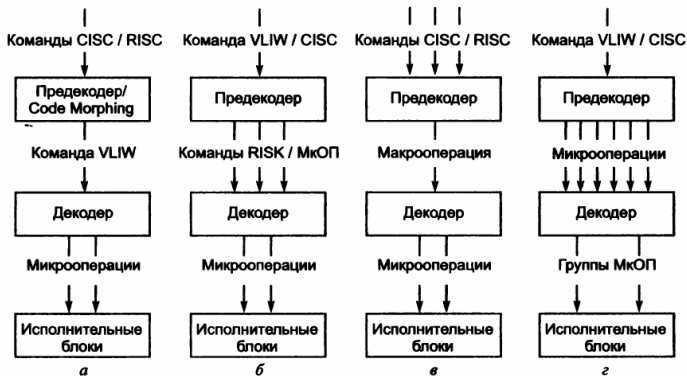


Рис. 15: Множественные декодеры и смежные технологии: а — преобразование CISC/RISC в VLIW; б — преобразование VLIW/CISC в RISC; в — макрослияние; г — микрослияние

Технологии повышения производительности процессоров

Технология Hyper-Threading

Разделение времени на аппаратном уровне, разбивая физический процессор на два логических процессора, каждый из которых использует основные аппаратные ресурсы — ядро, кэш-память, шины, исполнительное устройство. Благодаря этому многопроцессная ОС использует один физический процессор как два логических, и формирует одновременно два потока команд.

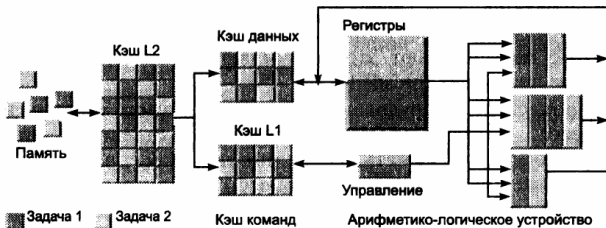
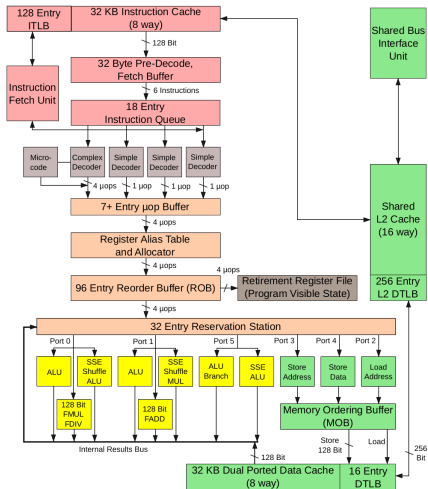


Рис. 16: Технология Hyper-Threading (HT)

Технологии повышения производительности процессоров

Многоядерная архитектура



Intel Core 2 Architecture

Рис. 17: Архитектура Intel Core 2