

Развитие и особенности современных и винтажных архитектур

Д. В. Луцив

Кафедра системного программирования СПбГУ



CS220 (231000)

Содержание

- 1 Особенности RISC и CISC
 - CISC
 - RISC
 - Перспективы RISC и CISC
- 2 Основные современные принципы
 - Оптимизация
 - Машины с параллельными вычислителями
- 3 Стековые машины
- 4 Виртуальные машины
 - Интерпретирующие и компилирующие трансляторы
 - Виртуальные машины
 - Примеры
 - JIT/AOT
- 5 Пример — процессоры Intel 80x86
 - 1970-е – 1980-е
 - 1990-е – 2000-е

CISC

CISC — Complicated/Complex Instruction Set Computer.
60е–70е годы — рост популярности и гротескности CISC.
DEC PDP — несколько сотен команд и несколько сотен
способов задания операндов.

Достоинства и недостатки CISC

- Удобные команды семантически связаны с языками высокого уровня
- Небольшое количество *сильно умных* команд
- из этого всего - много больших микропрограмм =>
 - тормоза
 - большое дорогое МикроПЗУ

RISC

RISC — Reduced Instruction Set Computer.

В основном, с 1980х.

- Более низкоуровневые и примитивные команды
- Выполняются, в основном, за одинаковое количество тактов (1 или 2)
- Команды для арифметики только на регистрах, для памяти - отдельные

Перспективы RISC и CISC

CISC жив и будет жить, так как:

- совместим
- ускоряется, содержа ядро RISC

Типичный современный компьютер

- Работа синхронизируется несколькими тактовыми генераторами
- Команды, насколько возможно, выполняются аппаратным обеспечением
- Выполнение команд, насколько возможно, распараллеливается
- Простой машинный код
- К памяти обращаются отдельные команды
- Большая внутренняя регистровая память

Конвейер

- 1 Выборка команды
- 2 Декодирование команды
- 3 Выборка операндов
- 4 Декодирование операндов
- 5 Выполнение

На конвейере одновременно несколько команд, разные блоки процессора не простаивают.

Опасности на конвейере

- Чем длиннее конвейер, тем меньше блоки \Rightarrow выше тактовая частота выборки команд
- Тем страшнее условный переход, при котором приходится сбросить конвейер
- На RISC после команды условного перехода может выполняться еще одна – две (т.н. *delay slot*) — конвейер с большой инерцией. Это надо учитывать в компиляторе.

Условные переходы пришлось научиться предсказывать.

Кэш

- Ассоциативная \Rightarrow дорогая и горячая память
- Часто многоуровневый
- Политики записи write-back и write-through
- Политика кэширования отличается для буферов устройств

Классификация

- SISD
- SIMD
- MISD
- MIMD

Векторные

- Классический SIMD
- Много одинаковых процессоров
- Одинаковый код
- Разные данные — маленькая память у каждого процессора
- Первый — ILLIAC4
- Типичный современный класс — GPU

Суперскалярные

- Из памяти выбирается сразу несколько команд
- Они помещаются на несколько конвейеров
- У каждого конвейера собственные АЛУ
- Параллельно выполняются команды:
 - Стоящие рядом
 - Независимые
 - При неясных прогнозах на переход, по возможности, из разных веток
- М.б. один конвейер с несколькими АЛУ, либо несколько, но каждому по одному АЛУ
- Альтернатива — VLIW
- Почти MIMD

Мультипроцессоры и мультикомпьютеры

- Мультипроцессоры - много гомогенных процессоров в одной машине
- Мультикомпьютеры - гомогенные компьютеры в быстрой (и не очень, зависит от задач) сети

Достоинства и недостатки

Достоинства

- красиво и просто
- хорошо компилируется код

Недостатки

- ухищрения для хранения вершины стека внутри процессора
- плохо распараллеливаются

Калькуляторы

- США, 1970-е ▶ HP-25, ▶ HP-65
- СССР, 1980-е ▶ Электроника МК-61, ▶ Электроника МК-52

Компьютеры

- США, вторая половина 1970-х – первая половина 1980-х.
 - ▶ LISP-машины
- СССР, 1980-е – 1990-е годы.
Арифметика на стеке, микропрограммирование,
16-битные. Ухищрения для хранения верхушки стека
внутри процессора.
 - Самсон (ЛГУ)
 - Кронос (НГУ)
 - Кронос потом сделали 32-битным
- США, 2010 — ▶ GA144 — 144-ядерный, ориентированный на
стековый язык Forth

Интерпретаторы

Плюсы

- не требуют сборки ПО
- проще в реализации
- больше информации при отладке

Минусы

- ошибки, в основном, на стадии выполнения
- медленно работают
- для быстрой работы требуют откомпилированных внешних библиотек

Компиляторы

Плюсы

- компилирует в быстрый код
- код достаточно автономен (не требует наличия интерпретатора)
- контроль многих ошибок при компиляции

Минусы

- сложнее писать
- долго собирается ПО
- меньше информации для отладки

Минусы VM

- медленнее компилированного кода
- медленно запускаются, особенно при JIT
- вообще довольно здоровые, особенно из-за Framework
- кроме всего прочего требуют компилятора, наследуя почти все минусы компиляции

Плюсы VM

- быстрее интерпретатора
- много информации во время выполнения
- по прежнему поддерживают откомпилированные внешние библиотеки
- поддерживают JIT и AOT
- делают уже откомпилированный код независимым от платформы и системы
- часто поставляются с очень большими библиотеками с одинаковым интерфейсом под разные платформы - Framework
- лучше, чем компилировать, т.к.
 - стековые компиляторы проще
 - для отладки много возможностей

Характерные черты

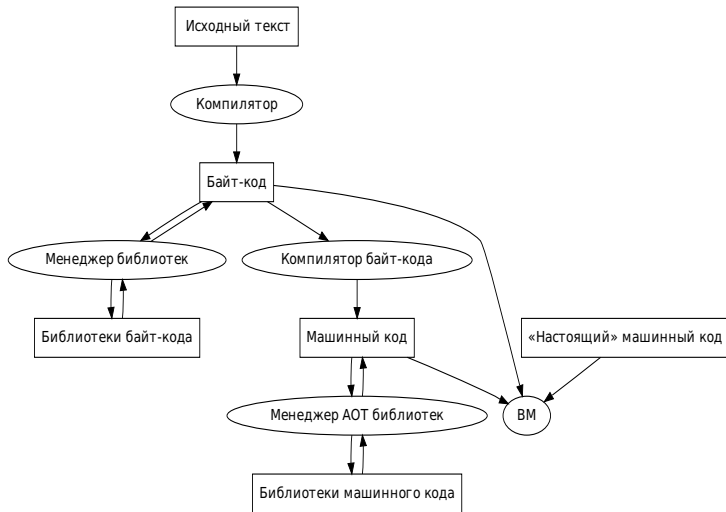
Часто стековые
JIT и AOT комбинируются

Стековые

- Lilith
- AVE
- Perl, Python, другие интерпретаторы
- интерпретаторы конкатенативных языков
- JVM
- .NET

Регистровые

- LLVM (на самом деле — промежуточное представление, а не VM)
- Parrot (Ruby, Perl)



1970-е – 1980-е (I)

- 8086
 - Параллельно с основным (FWAIT) — стековый математический, отдельная микросхема
- 80186
 - Нововведения касались работы с устройствами
- 80286
 - Нововведения касались защиты процессов

1970-е – 1980-е (II)

- 80386
 - Переработаны системы защиты и переключения процессов, отладки, адресации
 - Внутренний математический сопроцессор (опционально)
- 80486
 - Пятиступенчатый конвейер
 - Внутренний и внешний кэши (параметры защиты и адресации из таблиц в памяти теперь берутся быстрее)
 - Улучшенный сопроцессор
 - Тактовые частоты от 16 до 160 МГц — был успешен и долго жил

1990-е – 2000-е

- 80586
 - Суперскалярный
 - 2 конвейера
 - предсказание переходов
 - шина данных 64 бита
 - отдельные кэши для кода и данных
 - встроенный быстрый FPU
 - поддержка симметричного включения нескольких процессоров
- 80586 MMX, и последующие
 - MMX, SSE, SSE2, SSE3, 3DNOW - SIMD
- Параллелизм
 - Многоядерные
 - HT

Вопросы



▶ EDU.DLUCIV.NAME