Лекция 19

20 апреля

Метрики производительности кэша

- Коэффициент промахов
  - Отношение количества промахов к общему числу обращений (промахи / обращения) = 1 – коэффициент попаданий
  - Характерные показатели (в процентах):
    - 3-10% для L1
    - Может быть достаточно малым (< 1%) для L2, зависит от размера и т.д.
- Время попадания
  - Длительность извлечения данных из кэша
    - Включает время определения того, есть ли требуемые данные в кэше
  - Характерные показатели:
    - 1-2 тактов для L1
    - 5-20 тактов для L2
- Накладные расходы при промахе
  - Дополнительное время из-за промаха
    - Обычно 50-200 тактов для обращения к памяти

Что означают перечисленные показатели?

- Гигантская разница по времени промахов и попаданий
  - Два порядка, для L1 и оперативной памяти
- Пример: 99% попаданий вдвое более эффективно чем 97%
  - Характеристики:
    время попадания 1 такт
    накладные расходы при промахе 100 тактов
  - Среднее время доступа к элементу кэша:
  - 97% попаданий: 1 такт + 0.03 * 100 тактов = 4 такта
  - 99% попаданий: 1 такт + 0.01 * 100 тактов = 2 такта
- Поэтому “коэффициент неудач” используется вместо “коэффициента попаданий”

Дружественный к кэшу код

- В первую очередь улучшать часто работающий код
  - Тела вложенных циклов, часто вызывающиеся функции
- Минимизировать промахи при обращении к кэшу в теле вложенного цикла
  - Повторяющеюся обращения к одним и тем же переменным (временная локальность)
  - Проход по массиву с шагом 1 (пространственная локальность)

Оценка производительности

- Численная характеристика относительной производительности
  - Всего компьютера в целом
  - Отдельных компонент
  - Генерируемого компилятором кода
- Классификация
  - Синтетические / основанные на реальных приложениях
  - Микробенчмарк
- Индустриальные бенчмарки
  - SPEC
  - LINPACK
    - Решает систему линейных алгебраических уравнений Ax = b

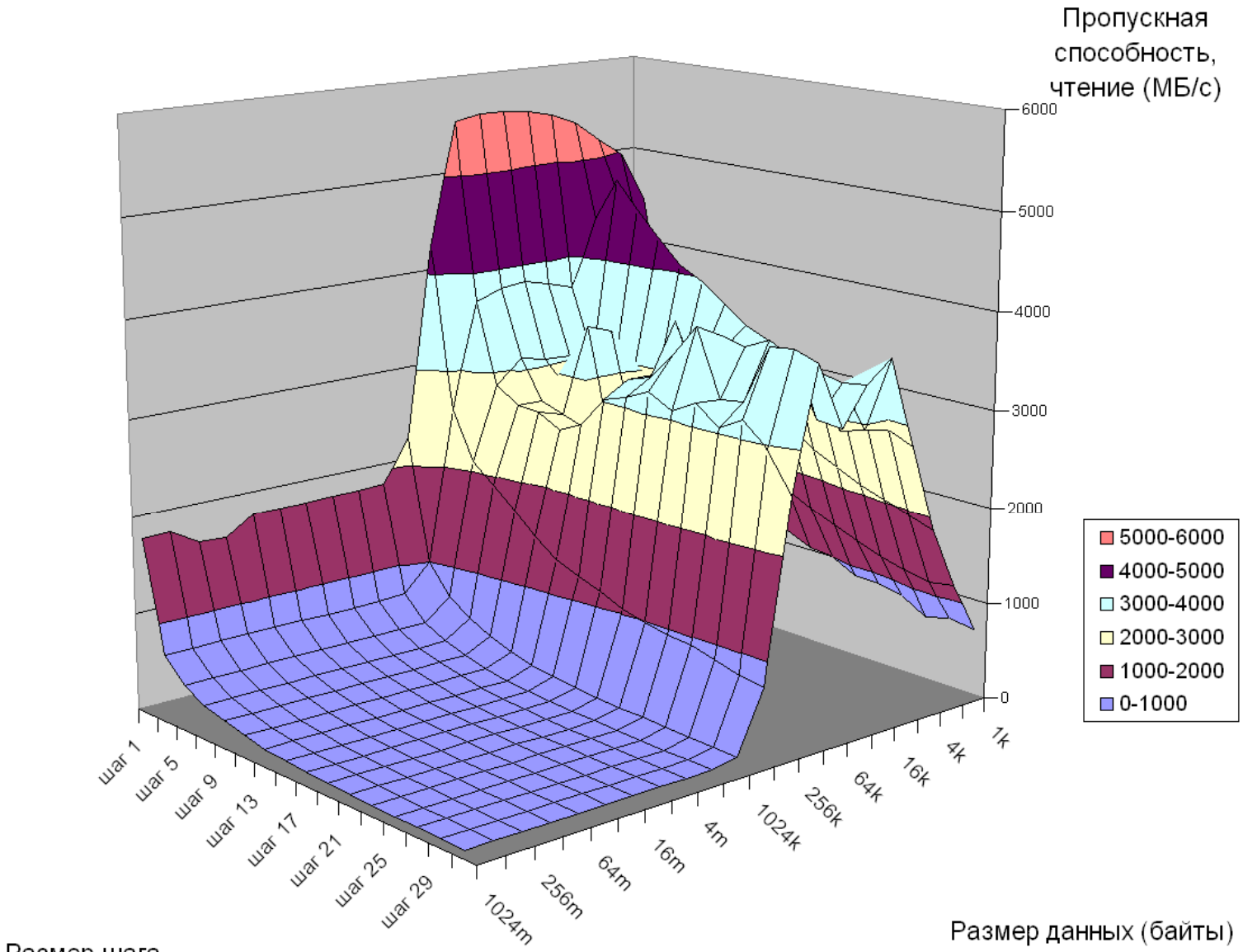
Измерение времени

- В Pentium появился регистр 64-разрядный регистр TSC, подсчитывающий количество выполнившихся тактов
- Инструкция `rdtsc` считывает значение регистра TSC и заносит его в `EDX:EAX`
- `rdtsc` может быть недоступна пользователям на некоторых системах

```
section .rodata
 format db ‘0x%08X 0x%08X’, 10, 0

section .text
global CMAIN
CMAIN:
   ...
   rdtsc
   mov     dword [esp + 8], eax
   mov     dword [esp + 4], edx
   mov     dword [esp], format
   call    printf
   ...
```

Оценка производительности памяти: синтетический бенчмарк (контрольная задача)

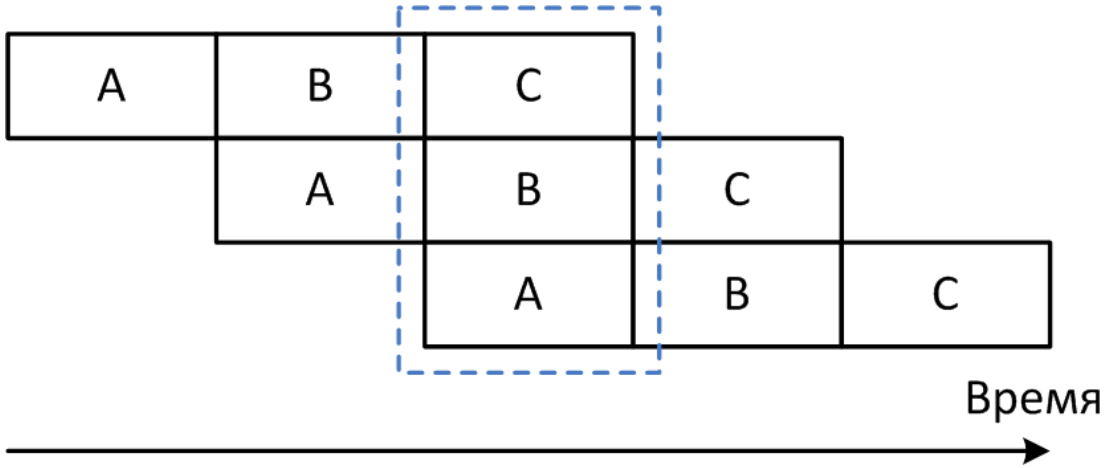
```
/* Оценочная функция */
void test(int elems, int stride) {
    int i, result = 0;
    volatile int sink;

    for (i = 0; i < elems; i += stride)
        result += data[i];
    sink = result;
}

/*
   Запуск test(elems, stride) и вычисление пропускной способности
   при чтении (МБ/с)
*/
double run(int size, int stride, double Mhz) {
    double cycles;
    int elems = size / sizeof(int);

    test(elems, stride);                    /* разогрев кэша */
    cycles = fcyc2(test, elems, stride, 0);  /* вызываем test(elems,stride) */
    return (size / stride) / (cycles / Mhz); /* переводим кол-во циклов в МБ/с */
}
```



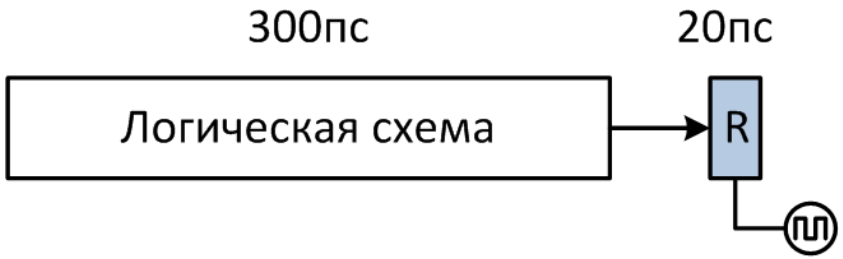
Intel(R) Xeon(TM) CPU 2.80GHz 2x2

Конвейер

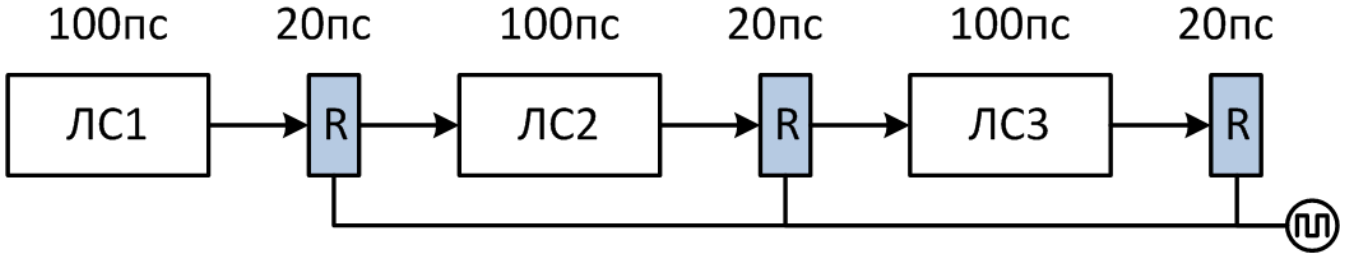
- Общая для различный предметных областей методика
- Длительность обработки неизменна или несколько увеличивается
- Увеличение пропускной способности



Организация конвейерных вычислений

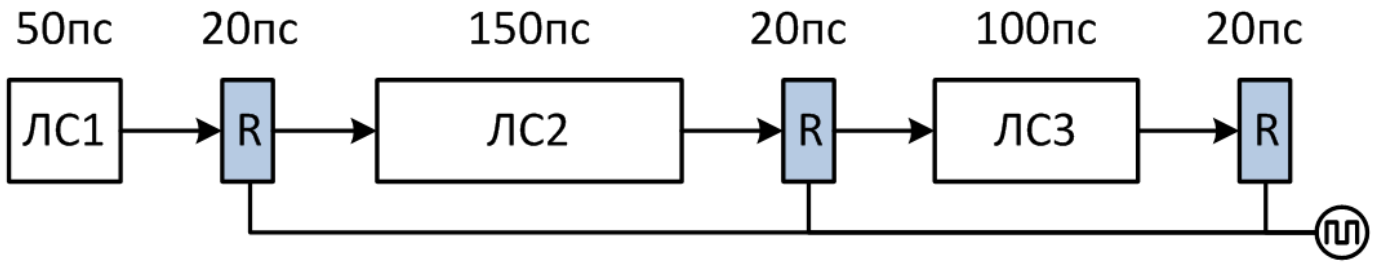


Задержка 320пс
Пропускная способность 3.12 GIPS



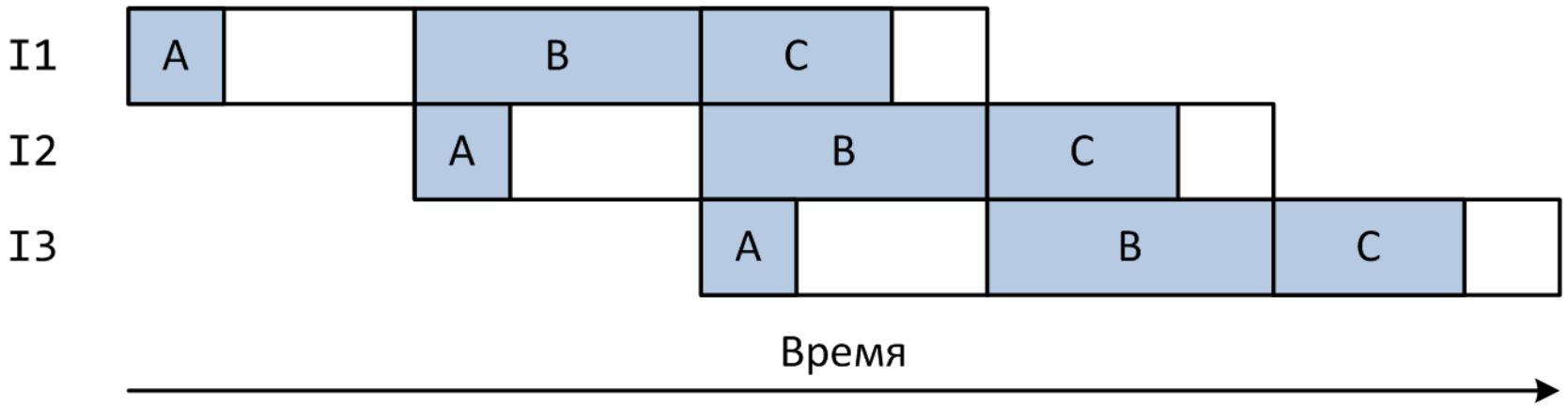
Задержка 360пс
Пропускная способность 8.33 GIPS

Неоднородность ступеней конвейера



Задержка 510пс

Пропускная способность 5.88 GIPS



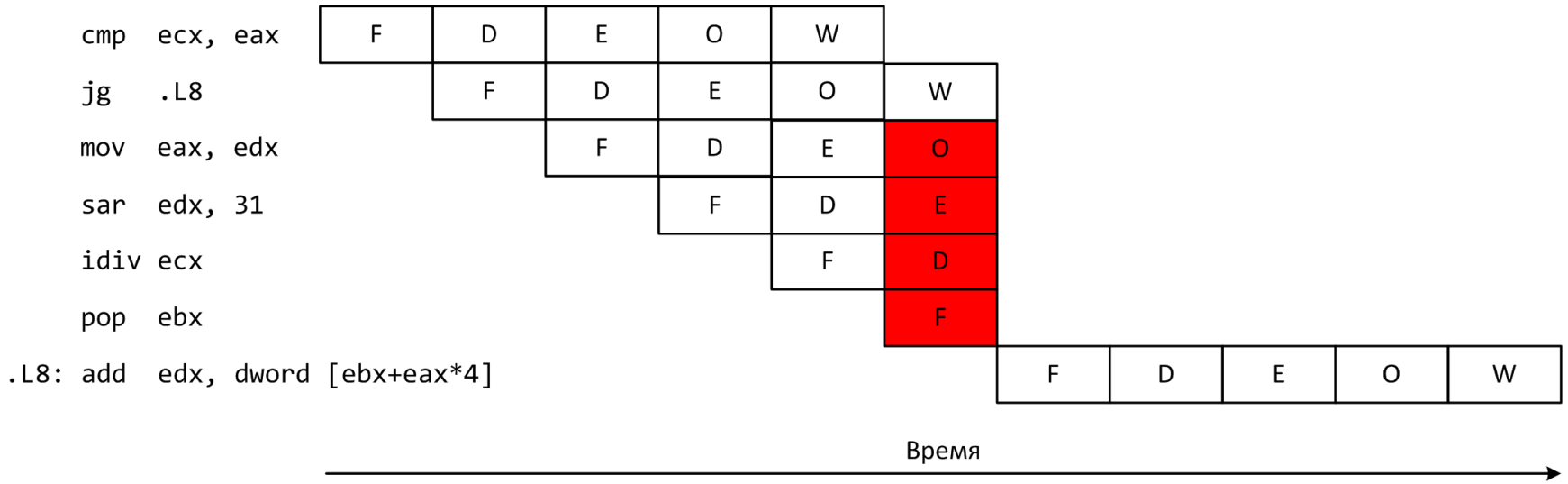
Упрощенная схема работы конвейера

1. (F) Извлечение инструкции из памяти
2. (D) Декодирование, обновление EIP
3. (E) Извлечение данных
4. (O) Непосредственное выполнение операции
5. (W) Запись результата

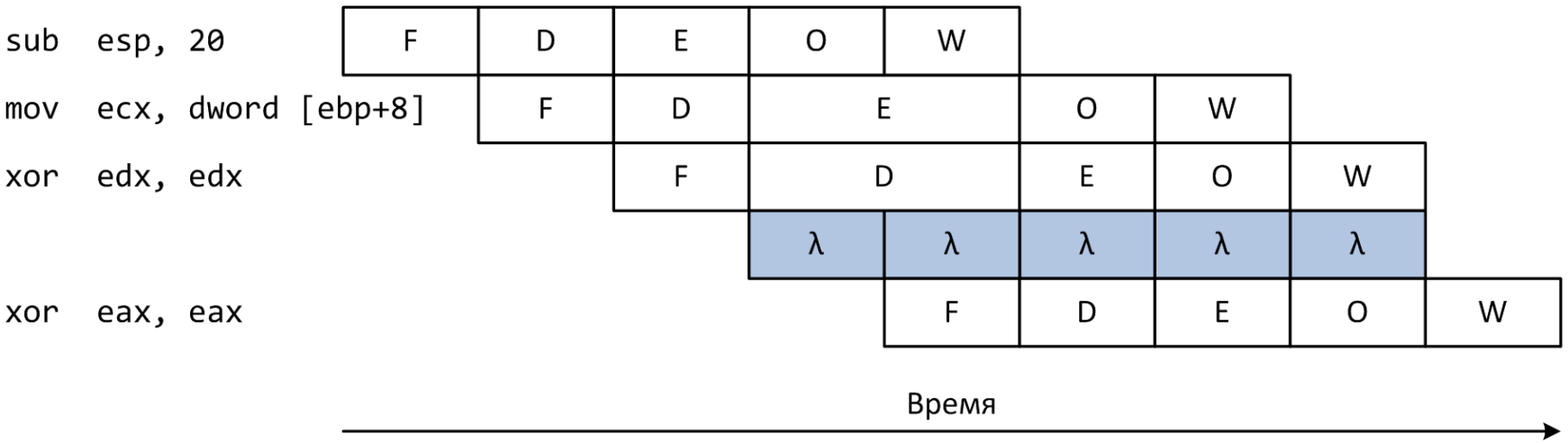
Недостатки конвейерной организации

- Опустошение конвейера
  - Современные процессоры могут содержать конвейеры длины 15 и более
  - Изменение EIP сбрасывает промежуточные результаты выполнения следующих инструкций
- Зависимости по чтению и записи регистров
- Остановки из-за обращения к памяти
- Сброс промежуточных результатов при возникновении исключительной ситуации

Опустошение конвейера

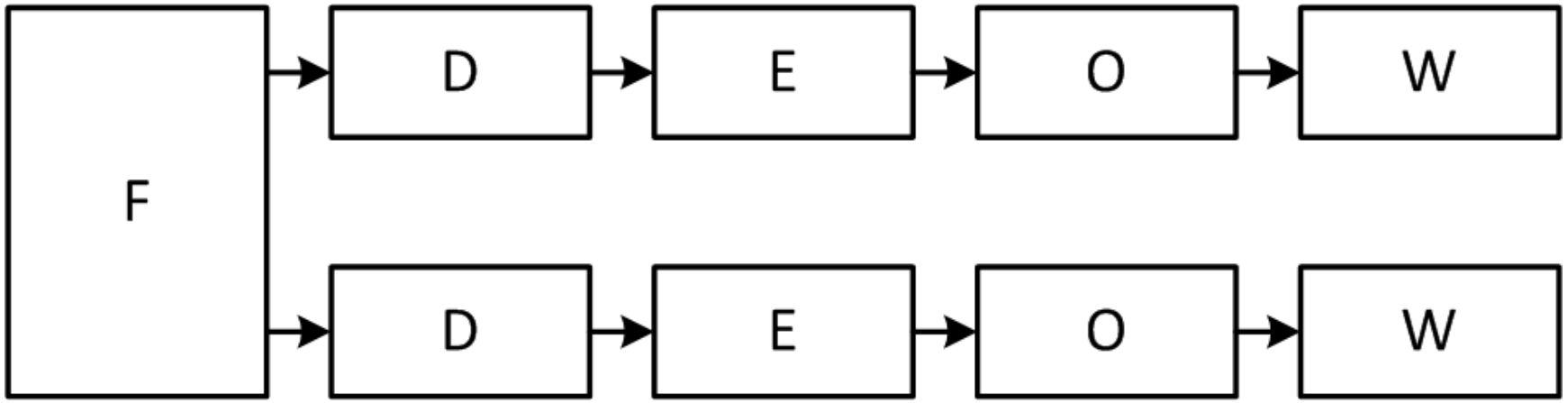


Приостановка конвейера

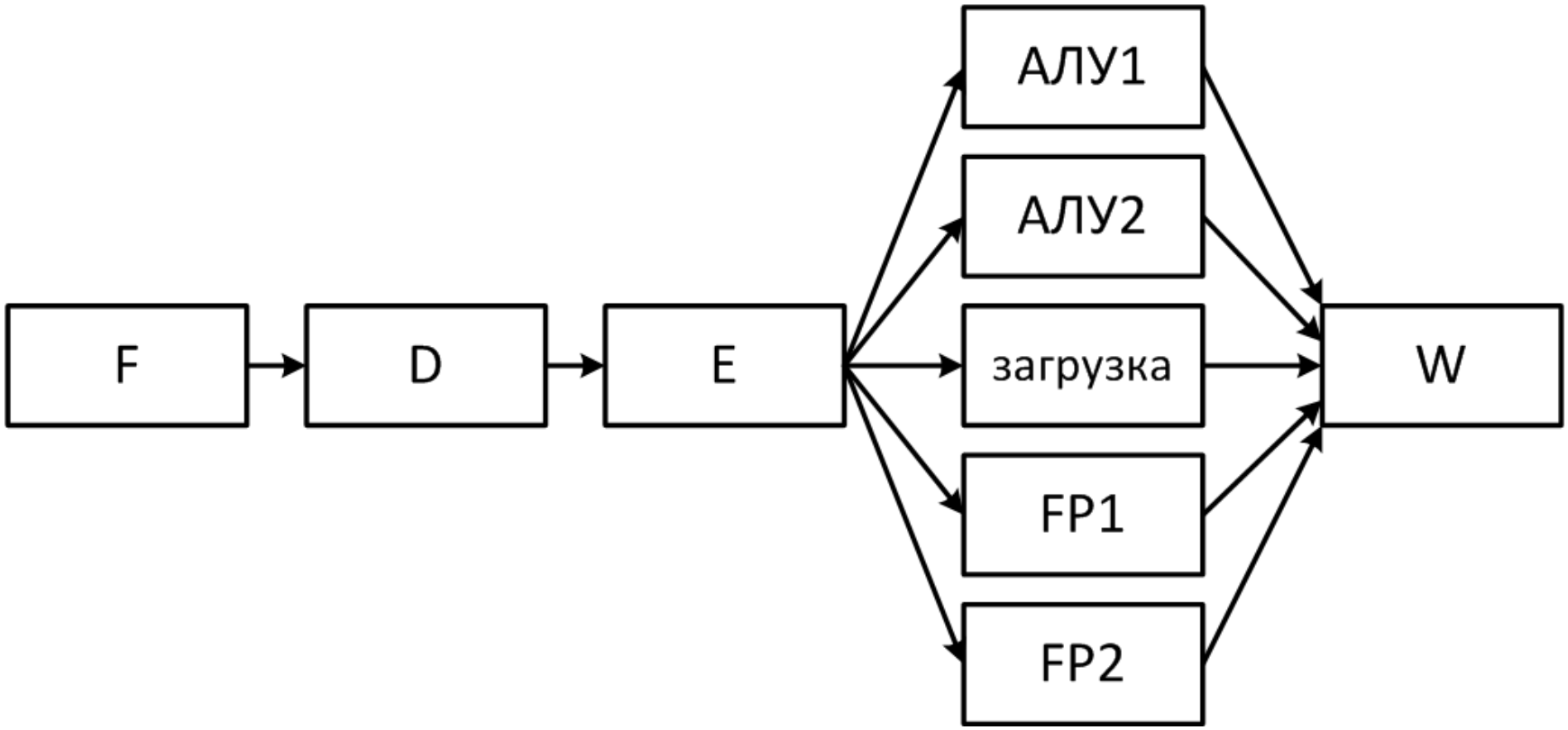


Двойной конвейер

- Двойной конвейер с общим этапом выборки команд
- Впервые появился на Pentium: введены u- и v-конвейеры
- Возможности конвейеров неравноценны



Суперскалярная архитектура



CISC и RISC архитектуры

- Исторически CISC предшествовал RISC
  - VAX/CISC
  - MIPS, SPARC/RISC
- Принципы RISC-архитектуры
  - Команды выполняются непосредственно на аппаратуре
  - Глубокий конвейер
    - Легкость декодирования команды – фиксированный формат
  - С памятью работают только две команды: загрузка и выгрузка
  - Большое количество регистров общего назначения
- Начиная с 486 реализован гибридный подход: CISC-процессор с RISC-ядром