



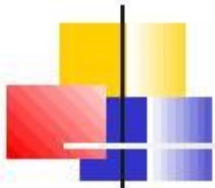
# Современные микропроцессоры

Продолжение 2

«ЭКСПРЕСС-АМБ»

- 1 Что у нас на орбите?
- 2 Взлет характеристик
- 3 Из чего это делается?
- 4 Почему радиация?
- 5 Классы для СБИС
- 6 Процессор процессору рознь
- 7 Память и как с ней бороться?
- 8 Суперскалярность**
- 9 Бренды и мы

# История ЭВМ



## Основные даты

---

- **1951г.** - Первая серийная ЭВМ (ЮНИВАК).
- **1964г.** - Появление интегральных схем.
- **1965г.** - Первый мини-компьютер.
- **1970-е г.** - Создание больших интегральных схем.
- **1977г.** - Первый микрокомпьютер Возняка и Джобса, выпущенный фирмой APPLE
- **1980г.** - Создан центральный процессор на одном кремниевом кристалле.
- **1980-е г.** - Появились сверхбольшие интегральные схемы.

# История ЭВМ

Вычислительные системы, сети и телекоммуникации

## История развития вычислительной техники, информационных технологий

Поколение	Элементная база	Быстродействие, оп/с	Программное обеспечение	Применение	Примеры
1-е (1946 - 1959)	Электронные лампы	10 - 20 тыс.	Машинные языки	Расчетные задачи	ЭНИАК (США), МЭСМ (СССР)
2-е (1960 - 1969)	Полупроводники	100 - 500 тыс.	Алгоритмические языки, диспетчерские системы, пакетный режим	Инженерные, научные, экономические задачи	IBM 701 (США), БЭСМ-6, БЭСМ-4 (СССР)
3-е (1970 - 1979)	Интегральные микросхемы	Порядка 1 млн.	Операционные системы, режим разделения времени	АСУ, САПР, научно-технические задачи	IBM 360 (США), ЕС 1030, 1060 (СССР)
4-е (1980 - настоящее время)	БИС, микропроцессоры	Десятки и сотни млн.	Базы и банки данных	Управление, коммуникации, АРМ, обработка текстов, графика	ПЭВМ, серверы
5-е (1990 - настоящее время)	СБИС	Единицы и десятки млрд.	ЭВМ как вычислительная система, интеллектуальные возможности, обработка знаний	Все возможные области	Рабочие станции, ноутбуки, интеллектуальные контроллеры

# История ЭВМ

Первым этапом затронувшим период с 40-х по конец 50-х годов, было создание процессоров с использованием электромеханических реле, ферритовых сердечников (устройств памяти) и вакуумных ламп. Они устанавливались в специальные разъёмы на модулях, собранных в стойки. Большое количество таких стоек, соединённых проводниками, в сумме представляли процессор. Отличительной особенностью была низкая надёжность, низкое быстродействие и большое тепловыделение.

## Поколения компьютеров

I. 1945 – 1955

электронно-вакуумные лампы



II. 1955 – 1965

транзисторы



III. 1965 – 1980

интегральные микросхемы



IV. с 1980 по ...

большие и сверхбольшие интегральные схемы (БИС и СБИС)



# История ЭВМ

Вторым этапом, с середины 50-х до середины 60-х, стало внедрение транзисторов. Транзисторы монтировались уже на близкие к современным по виду платам, устанавливаемым в стойки. Как и ранее, в среднем процессор состоял из нескольких таких стоек. Возросло быстродействие, повысилась надёжность, уменьшилось энергопотребление.

## Поколения компьютеров

I. 1945 – 1955

электронно-вакуумные лампы



II. 1955 – 1965

транзисторы



III. 1965 – 1980

интегральные микросхемы



IV. с 1980 по ...

большие и сверхбольшие интегральные схемы (БИС и СБИС)



# История ЭВМ

Третьим этапом, наступившим в середине 60-х годов, стало использование микросхем. Первоначально использовались микросхемы низкой степени интеграции, содержащие простые транзисторные и резисторные сборки, затем по мере развития технологии стали использоваться микросхемы, реализующие отдельные

элементы  
цифровой схемотехники  
(сначала элементарные  
ключи и логические  
элементы).

## Поколения компьютеров

I. 1945 – 1955

электронно-вакуумные лампы



II. 1955 – 1965

транзисторы



III. 1965 – 1980

интегральные микросхемы



IV. с 1980 по ...

большие и сверхбольшие  
интегральные схемы (БИС и СБИС)



# История ЭВМ

Затем в микросхемах реализовывались более сложные элементы — элементарные регистры, счётчики, сумматоры, позднее появились микросхемы, содержащие функциональные блоки процессора — микропрограммное устройство, арифметико-логическое устройство, регистры, устройства работы с шинами данных и команд.

## Поколения компьютеров

### I. 1945 – 1955

электронно-вакуумные лампы



### II. 1955 – 1965

транзисторы



### III. 1965 – 1980

интегральные микросхемы



### IV. с 1980 по ...

большие и сверхбольшие интегральные схемы (БИС и СБИС)





# История ЭВМ

Четвёртым этапом, в начале 70-х годов, стало создание, благодаря прорыву в технологии создания БИС и СБИС (больших и сверхбольших интегральных схем, соответственно), микропроцессора — микросхемы, на кристалле которой физически были расположены все основные элементы и блоки процессора. Фирма Intel в 1971 году создала первый в мире 4-х разрядный микропроцессор 4004, предназначенный для использования в микрокалькуляторах.

Постепенно практически все процессоры стали выпускаться в формате микропроцессоров.

## Поколения компьютеров

### I. 1945 – 1955

электронно-вакуумные лампы



### II. 1955 – 1965

транзисторы



### III. 1965 – 1980

интегральные микросхемы



### IV. с 1980 по ...

большие и сверхбольшие интегральные схемы (БИС и СБИС)



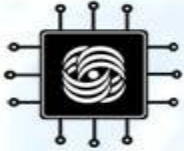
# История ЭВМ

## IV поколение (с 1980 по ...)

---

- компьютеры на больших и сверхбольших интегральных схемах (БИС, СБИС)
- **суперкомпьютеры**
- **персональные** компьютеры
- появление пользователей-**непрофессионалов**, необходимость «дружественного» интерфейса
- более **1 млрд.** операций в секунду
- оперативная памяти – до нескольких **гигабайт**
- **многопроцессорные** системы
- компьютерные **сети**
- **мультимедиа** (графика, анимация, звук)





## Микрочип

- **Сверхбольшая Интегральная Схема (СБИС)**
  - Транзисторы, резисторы, конденсаторы
- **4004 - 2,250 транзисторов**
- **Pentium IV – 42,000,000 транзисторов**
  - Каждый транзистор 0.13 микрон

# История ЭВМ

**5 поколение - ЭВМ на сверхбольших интегральных микросхемах**

**Основные черты:**

**ЭВМ как вычислительная система, интеллектуальные возможности, обработка знаний, наличие нескольких процессоров: центральный и периферийные, общение с ЭВМ на нескольких естественных языках, ввод / вывод графической информации.**

# История архитектур ЭВМ

**Вернемся к началу...**

**Первая попытка создать программируемый компьютер была предпринята в 1941 году в США в Гарвардском университете математиком Говардом Эйксоном. Компьютер строился по заданию и с участием компании IBM.**

**Он имел длину 17 метров, вес 4,5 тонны, содержал в себе 765 тысяч реле и переключателей и 800 км проводов.**

**Он был по сути арифмометром, и мог производить простые действия с 72 числами длиной 23 разряда. На одну операцию сложения тратилось более 3 секунд. Данные вводились с бумажной перфоленты.**

# История архитектур ЭВМ



**Компьютер Марк-1  
1941 год  
США  
Гарвардский университет**



# История архитектур ЭВМ

**Первый компьютер Марк-1 был собран на электромеханических реле.**

Однако электромеханические реле работают весьма медленно и недостаточно надежно. Поэтому, начиная с 1943 г. в США группа специалистов под руководством Джона Мочли и Преспера Экерта начала конструировать компьютер ENIAC (рис.2) на основе электронных ламп. Этот монстр содержал десятки тысяч электронных ламп и релейных переключателей. Созданный ими компьютер работал в тысячу раз быстрее, чем Марк-1. Однако обнаружилось, что большую часть времени этот компьютер простаивал — ведь для задания метода расчетов (программы) в этом компьютере приходилось в течение нескольких часов или даже нескольких дней подсоединять нужным образом провода. А сам расчет после этого мог занять всего лишь несколько минут или даже секунд.

# История архитектур ЭВМ

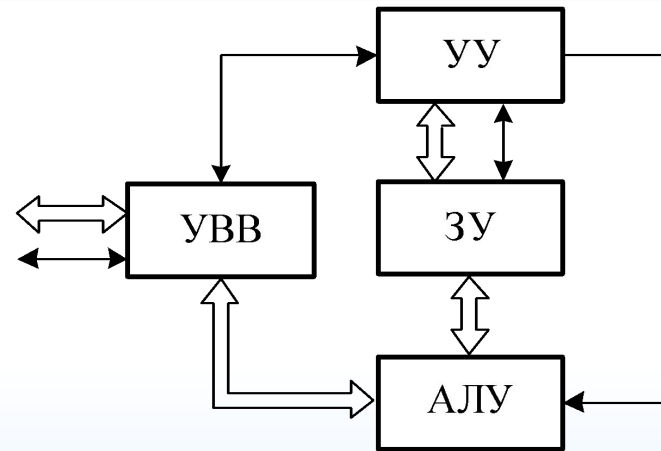
Чтобы упростить и ускорить процесс задания программ, Мочли и Экерт стали конструировать новый компьютер, который мог бы хранить программу в своей памяти. В 1945 г. К работе был привлечен знаменитый математик Джон фон Нейман, который подготовил доклад об этом компьютере. Доклад был разослан многим ученым и получил широкую известность.



# История архитектур ЭВМ

Нейман ясно и просто сформулировал общие принципы функционирования компьютеров, т.е. универсальных вычислительных устройств. До сих пор подавляющее большинство компьютеров сделано в соответствии с теми принципами, которые изложил в своем докладе Джон фон Нейман. Первый компьютер, в котором были воплощены принципы фон Неймана, был построен в 1949 г. английским исследователем Морисом Уилксом.

# История архитектур ЭВМ



В вычислительной машине Дж. Фон Неймана, структура которой представлена на рисунке, ввод-вывод информации осуществляется через арифметико-логическое устройство (АЛУ), поэтому обработка информации прекращается на время ввода-вывода. Где ЗУ - запоминающее устройство, УУ - устройство управления машины, УВВ - устройство ввода-вывода информации.

# История архитектур ЭВМ

Современные ЭВМ строятся на основе принципа программного управления , предполагающего следующее уточнение понятия алгоритма:

Алгоритм - предписание, однозначно задающее процесс преобразования исходной информации в виде последовательности элементарных дискретных шагов, приводящих за конечное число их применений к результату.

Исходные данные, промежуточные и конечные результаты кодируются в двоичной форме и разделяются на единицы или элементы информации, называемые словами.

# История архитектур ЭВМ

Принципы фон Неймана:

- разнотипные слова информации (данные и команды) различаются по способу использования, но не по способам кодирования;
- слова информации размещаются в ячейках памяти и идентифицируются номерами ячеек, называемыми адресами слов.

# История архитектур ЭВМ

Дальнейшее увеличение производительности вычислительных машин осуществлялось путем совмещения этапов выполнения операций и опережающего просмотра команд.

По мере совершенствования методов опережающего просмотра постепенно был осуществлен ***метод просмотра нескольких команд и данных с соответствующей выборкой из памяти и предварительной подготовкой к выполнению операций.***

# Современные ЭВМ

Современные микропроцессоры представляют весьма сложные устройства. Их работа может существенно отличаться от описанной выше работы элементарного микропроцессора, и для их рассмотрения разработаны специальные термины и понятия.

# Современные ЭВМ

В понятие **архитектуры микропроцессора**

входит совокупность некоторых характеристик:

- 1) структура МП (совокупность компонентов и связи между ними), в простейшем случае – регистровая модель (или стековая модель – с конвейером);
- 2) способы представления и форматы данных МП;
- 3) способы обращения к доступным элементам структуры МП (регистрам, памяти, УВВ);
- 4) система команд МП;
- 5) характеристики обменных сигналов и управляющих слов МП;
- 6) реакция на внешние сигналы (система прерываний).

# Современные ЭВМ

В ходе эволюционного развития архитектур процессоров в состав системы команд вводились и, в силу преемственности программного обеспечения, закреплялись сложные команды, которые по мнению разработчиков соответствовали решаемым задачам. Оптимизация программы была равнозначна минимизации времени исполнения.

Команды бывают разных типов: "регистр, регистр -> регистр", "память, память -> память", "регистр -> память" и др.

Сложные команды модифицируют содержимое групп регистров и ячеек памяти, и для их реализации при приемлемых затратах оборудования, как правило, применяется микропрограммирование



# Современные ЭВМ

## Два типа команд микропроцессора

- Команды называются скалярными, если входные операнды и результат являются числами (скалярами).
- Команды называются векторными, если входные операнды и, возможно, результат являются вектором (массивом) чисел, а для преобразования данных массива (вектора) используется одна векторная команда.

Векторные команды нужны для обработки данных движения объекта, изображения, звука, 3-D моделирования и других задач компьютерной графики

# Современные ЭВМ

Примером векторной команды служит команда, при выполнении которой:

- умножаются два очередных элемента двух массивов,
- далее произведение суммируется с содержимым некоторого заданного регистра,
- после чего модифицируются адреса памяти для доступа к двум очередным элементам массивов.

Указанная последовательность действий повторяется заданное число раз по счетчику, определенному в теле команды.

# Современные ЭВМ

Само появление векторных команд обусловлено стремлением ускорить обработку массивов данных за счет исключения затрат времени на выборку и дешифрацию команд обработки, одинаковых для всех компонент входных массивов.

Однако использование векторных команд требует подготовки программистом векторизованного кода программ, что, вообще говоря, эквивалентно разработке параллельных программ.

# Современные ЭВМ

При сохранении последовательных программ для ускорения обработки применяются

**суперскалярные процессоры**, в которых за счет параллельной работы функциональных устройств процессора в одном такте вырабатывается несколько скалярных результатов.

# Современные ЭВМ

Суперскалярность — архитектура вычислительного ядра, использующая несколько декодеров команд, которые могут нагружать работой множество исполнительных блоков. Планирование исполнения потока команд является динамическим и осуществляется самим вычислительным ядром.

Если в процессе работы команды, обрабатываемые конвейером, не противоречат друг другу, и одна не зависит от результата другой, то такое устройство может осуществить параллельное выполнение команд. В суперскалярных системах решение о запуске инструкции на исполнение принимает сам вычислительный модуль, что требует много ресурсов

# Современные ЭВМ

В более поздних системах, таких как Эльбрус-3 и Itanium, используется статпланирование, то есть параллельные инструкции объединяются компилятором в длинную команду, в которой все инструкции заведомо параллельные (архитектура VLIW).

Процессоры, поддерживающие суперскалярность:

- Pentium, AMD Duron, AMD ATHLON и другие более поздние процессоры с архитектурой x86
- Последние процессоры с архитектурами SPARC, ARM, MIPS
- Эльбрус-1,2

# Современные ЭВМ

Стремление использовать присущий большинству программ естественный параллелизм вычисления целочисленных адресных выражений и собственно обработки данных в формате с плавающей точкой привело к появлению разнесенных архитектур **(decoupled architecture)**.

Естественный параллелизм вычислений связан с независимым выполнением большинства последовательных операций до разветвления программы. Такие команды можно выбрать из памяти заранее и заранее подготовить для них данные, а затем всю эту группу независимых команд очень быстро обработать на конвейерном процессоре.

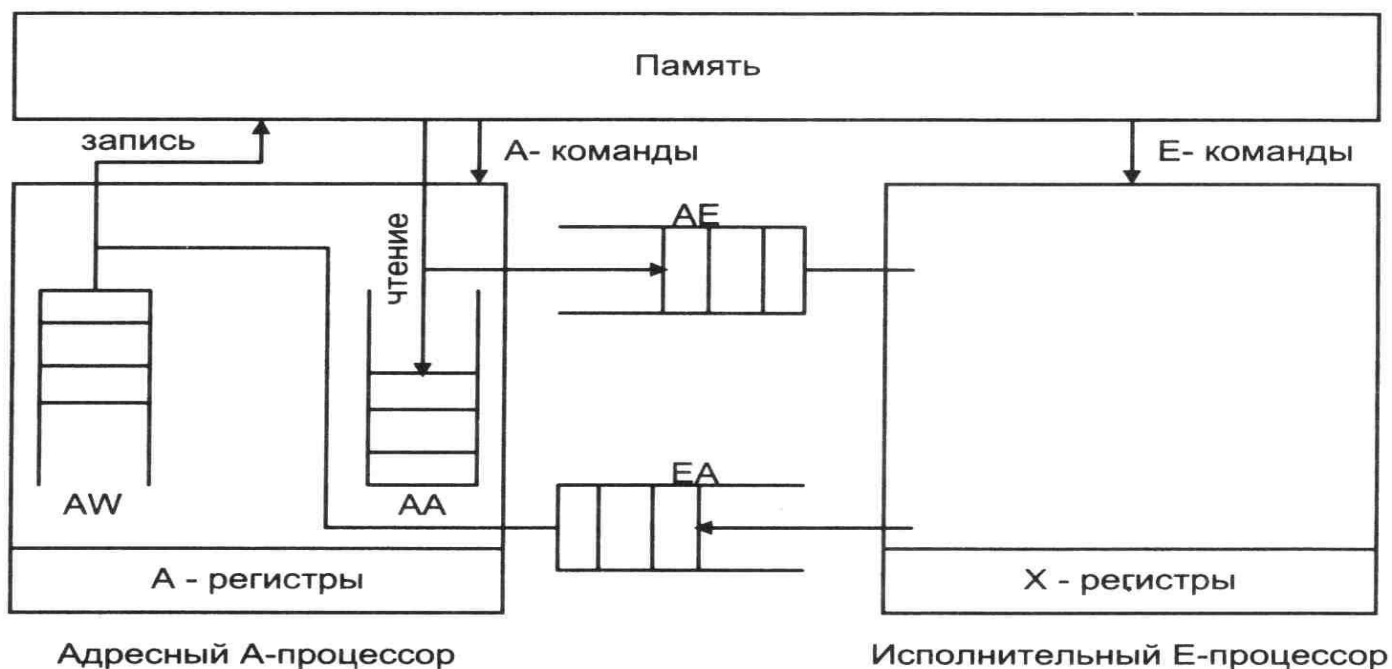
# Современные ЭВМ

Особенно, конечно, такая возможность ускорения вычислений проявляется при работе с данными 3-D графики и подобными им, но и при простых вычислениях можно добиться значительного ускорения за счет предварительного считывания большинства данных из памяти. Это дает возможность устранения основного противоречия между большой производительностью процессора и в десятки раз более медленной скоростью выборки из памяти.



# Современные ЭВМ

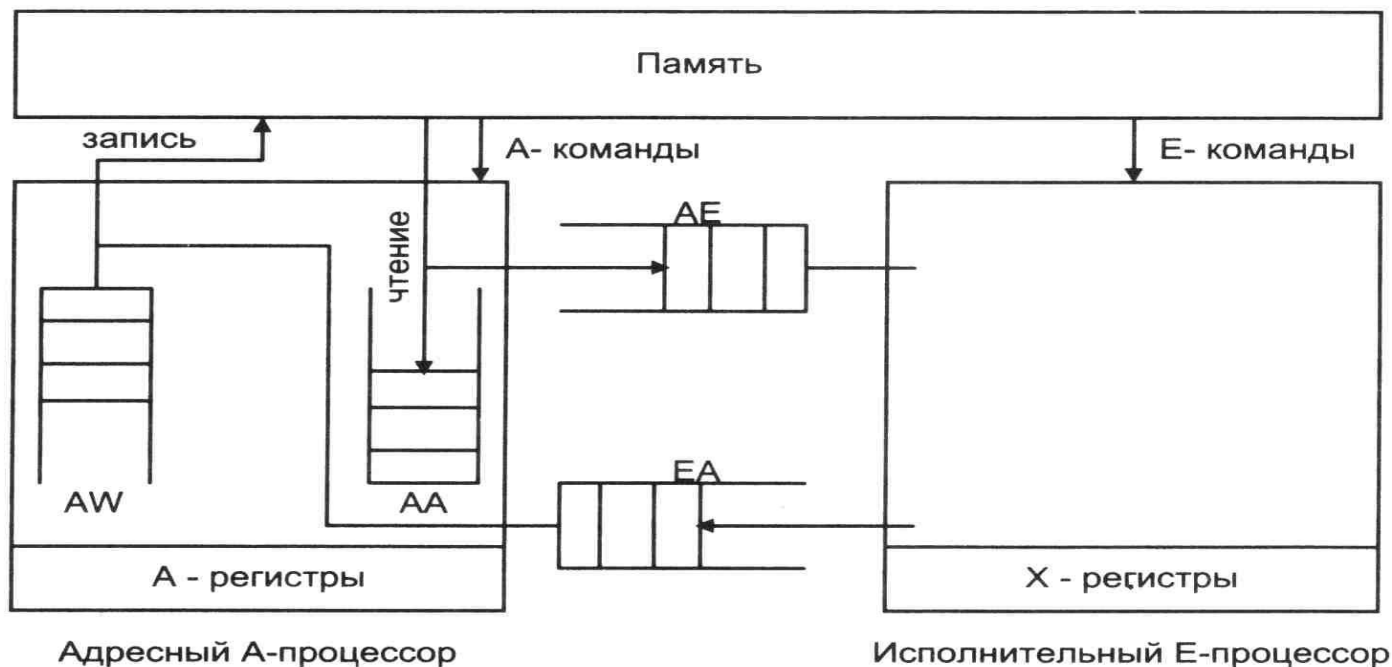
В первом приближении, микропроцессор с разнесенной архитектурой, как показано на рисунке, состоит из двух связанных подпроцессоров, каждый из которых управляется собственным потоком команд.



*Микропроцессор с разнесенной архитектурой*

# Современные ЭВМ

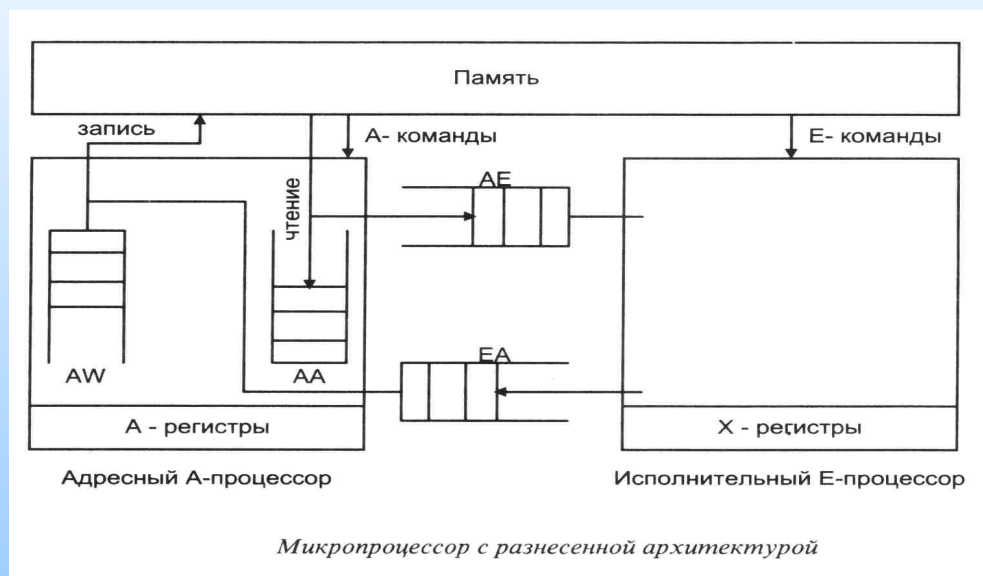
Условно эти подпроцессоры называются адресным А-процессором и исполнительным Е-процессором. А- и Е-процессоры имеют собственные наборы регистров  $AO, A1, \dots$  и  $XO, X1, \dots$ , соответственно и наборы команд.



# Современные ЭВМ

А-процессор выполняет все адресные вычисления и формирует обращения к памяти по чтению и записи. А-процессор является обыкновенным целочисленным процессором, поэтому он способен выполнять произвольные целочисленные преобразования, не связанные с вычислением адресов.

Е-процессор реализует вычисления с фиксированной и с плавающей точкой.

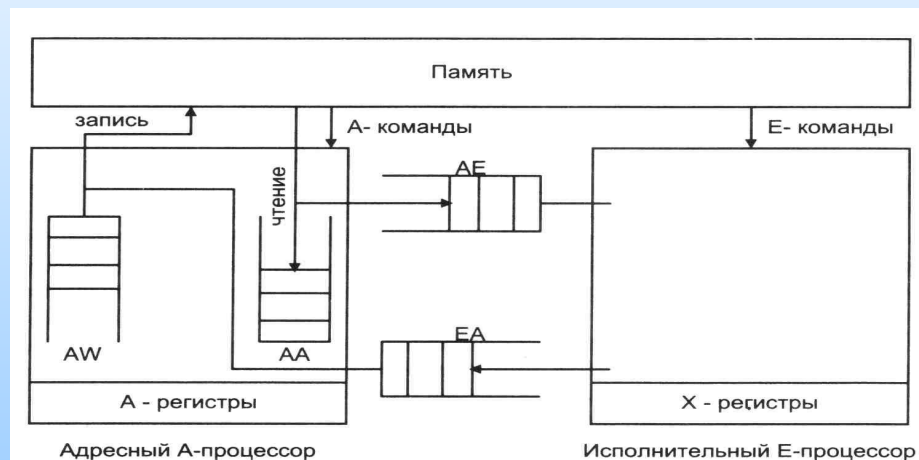


# Современные ЭВМ

Данные, извлекаемые из памяти, используются либо в А-процессоре, будучи помещенными в FIFO очередь АА, либо помещаются в FIFO очередь, называемую АЕ очередью, для отсылки в Е-процессор.

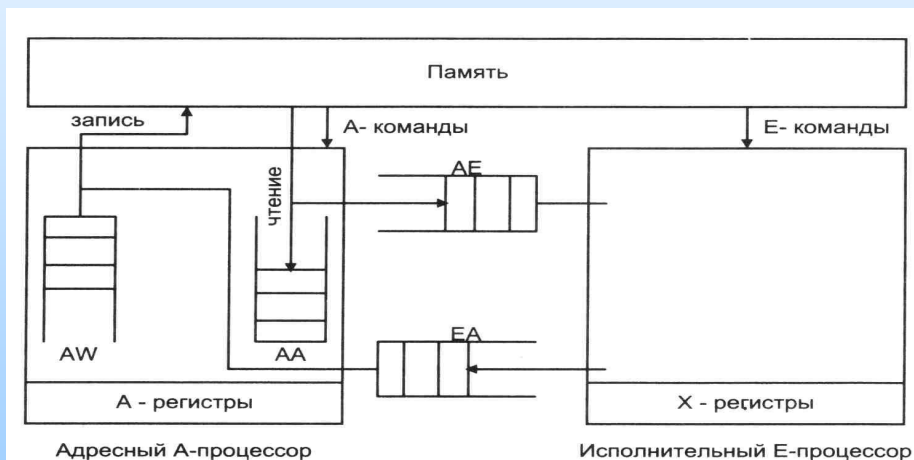
Когда Е-процессору требуются данные из памяти, он берет их из очереди АЕ. Если очередь пуста, то Е-процессор задерживается до поступления данных, что решает вопросы синхронизации работы А и Е-процессоров.

Если Е-процессор выработал данное, которое должно быть отправлено в память, то он помещает его в FIFO очередь ЕА.



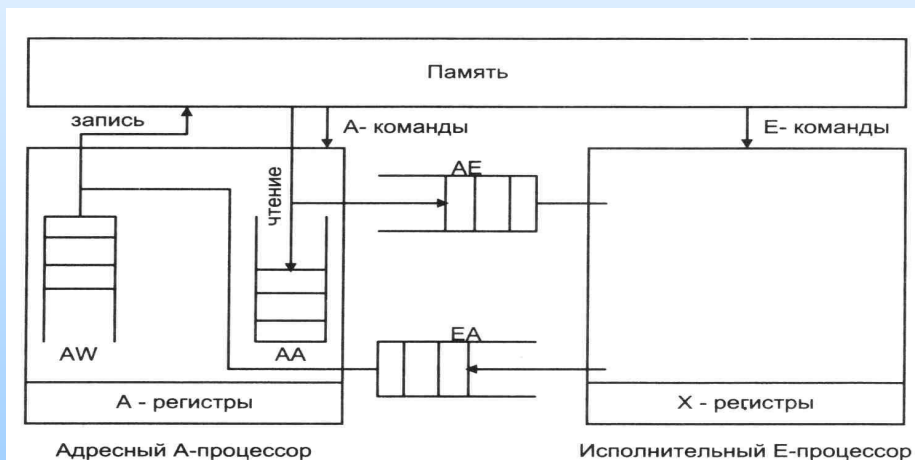
# Современные ЭВМ

При записи данных в память после вычисления адреса А-процессор сразу отправляет адрес в FIFO очередь AW адресов записи в память, не дожидаясь, пока данные поступят в очередь EA. А-процессор группирует пары, выбирая первые элементы очередей EA и AW и отправляя эти пары в память. Естественно, если одна из очередей или обе пусты, то отсылка в память приостанавливается.



# Современные ЭВМ

Разнесенная архитектура позволяет достигать при скалярной обработке производительности, характерной для векторных процессоров, за счет предвыборки данных из памяти и автоматической развертки нескольких последовательных витков цикла в А-процессоре. Проблемы расщепления программы на программы для А- и Е-процессоров решаются на уровне компилятора или специальным блоком-расщепителем.



Микропроцессор с разнесенной архитектурой

## Два типа архитектур микропроцессоров

Анализ кода программ, генерируемого компиляторами языков высокого уровня, показал, что практически используется только ограниченный набор простых команд форматов "регистр, регистр -> регистр" и "регистр <-> память".

Компиляторы не в состоянии эффективно использовать сложные команды. Это наблюдение способствовало формированию концепции процессоров с сокращенным набором команд, так называемых RISC-процессоров.

## **Два типа архитектур микропроцессоров**

Другим обстоятельством, фактически приведшим к появлению RISC-процессоров, было развитие архитектуры конвейерных процессоров типа Cray. В этих процессорах используются отдельные наборы команд для работы с памятью и отдельные наборы команд для преобразования информации в регистрах процессора.



## **Два типа архитектур микропроцессоров**

Каждая такая команда единообразно разбивается на небольшое количество этапов с одинаковым временем исполнения (выборка команды, дешифрация команды, исполнение, запись результата), что позволяет построить эффективный конвейер процессора, способный каждый такт выдавать результат исполнения очередной команды.

## Два типа архитектур микропроцессоров

После обособления RISC-процессоров в отдельный класс, процессоры с традиционными наборами команд стали называться CISC-процессорами с полным набором команд. Как правило, в этих процессорах команды имеют много разных форматов и требуют для своего представления различного числа ячеек памяти.

Это обуславливает определение типа команды в ходе ее дешифрации при исполнении, что усложняет устройство управления процессора и препятствует повышению тактовой частоты до уровня, достижимого в RISC-процессорах на той же элементной базе.

## Классификация архитектур микропроцессоров


В настоящее время на основе разработок компаний NexGen и AMD, подхваченных компанией Intel, предпринята попытка решить проблему повышения производительности в рамках архитектуры x86. Эти компании в последних разработках, сохраняя преемственность по системе команд с CISC-микропроцессорами семейства x86, создали новые устройства с использованием элементов RISC-архитектуры.

## Классификация архитектур микропроцессоров

Примером такого подхода могут служить микропроцессоры Nx586 (NexGen), K5, K6 (AMD), Pentium (Intel), использующие концепцию разделенной (decoupled) архитектуры и RISC ядра. В микропроцессор встраивается аппаратный транслятор, превращающий команды x86, в команды RISC-процессора. При этом одна команда x86 может породить до четырех команд RISC-процессора. Исполнение команд происходит как в развитом суперскалярном процессоре

**В следующей лекции более подробно будут рассмотрены следующие темы:**

- Разновидности микропроцессоров**
  - микроконтроллеры;**
  - сигнальные и медийные процессоры;**
  - универсальные микропроцессоры.**
- Основные производители универсальных микропроцессоров.**
- Основные типы микропроцессоров, применяемые в космосе.**



**Спасибо за  
ВНИМАНИЕ**