СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ

Лекция №1

Основные понятия
Устройство компьютерных систем
Системы счисления
Введение в Assembler

ivlievn@gmai.com Ивлиев Н.А.

Самара, 2018

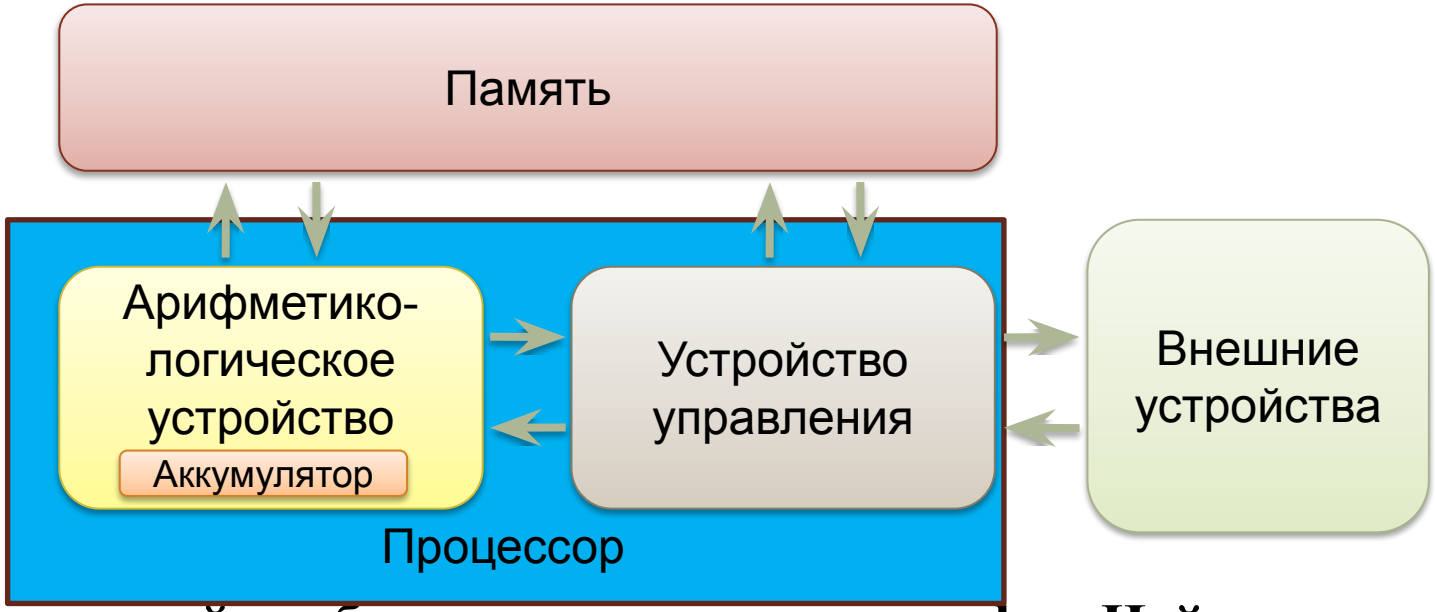
ОСНОВНЫЕ ПОНЯТИЯ

- Программное обеспечение (ПО) — совокупность программ компьютерных систем, необходимых для их эксплуатации.
- Системное программное обеспечение (System Software) - совокупность программ и программных комплексов для обеспечения работы компьютерных систем.

Системное программное обеспечение предназначено для:

- создания операционной среды функционирования других программ;
- автоматизации разработки (создания) новых программ;
- обеспечения надежной и эффективной работы самой компьютерной системы;
- проведения диагностики и профилактики аппаратуры;
- выполнения вспомогательных технологических процессов (копирование, архивирование, восстановление файлов программ и баз данных и т.д.).

УСТРОЙСТВО КОМПЬЮТЕРНЫХ СИСТЕМ



- отличительной особенностью **архитектуры фон Неймана** является то, что инструкции и данные хранятся в одной и той же памяти;
- указанная особенность позволяет производит загрузку и выгрузку управляющих программ в произвольное место памяти процессора, которая в этой структуре не разделяется на память программ и память данных. Любой участок памяти может служить как памятью программ, так и памятью данных.

ОСНОВНЫЕ ПОНЯТИЯ

- **Микропроцессор –** процессор, т.е. устройство, отвечающее за выполнение операций, записанных в машинном коде, представляет из себя сверхбольшую интегральную схему, реализованную в едином полупроводниковом кристалле.

CPU - Central Processing Unit – выполняет математические действия над числами, находящимися в ***основной памяти*** компьютера или специальных внутренних ячейках – ***регистрах*** и дополнительные специальные действия.

Выполняемые микропроцессором **команды** предусматривают арифметические действия, логические операции, передачу управления (условную и безусловную) и перемещение данных (между регистрами, памятью, портами ввода-вывода).

ОСНОВНЫЕ ФУНКЦИИ МИКРОПРОЦЕССОРА

- чтение и дешифрация команд из основной памяти;
- чтение данных из оперативной памяти (ОП) и регистров адаптеров внешних устройств (ВУ);
- обработка данных и их запись в ОП и регистры адаптеров ВУ;
- выработка управляющих сигналов для всех прочих узлов и блоков ПК.
- прием и обработка запросов и команд от адаптеров на обслуживание ВУ;

ОСНОВНЫЕ ПОНЯТИЯ

- Микропроцессоры характеризуются **тактовой частотой**, **разрядностью** или длиной слова (внутренней и внешней), **архитектурой** и **набором команд**.
- **Архитектура микропроцессора** определяет необходимые **регистры, стеки, систему адресации**, а также **типы** обрабатываемых процессором **данных**.
- **Степень интеграции** определяется размерами кристалла и количеством реализованных в нём **транзисторов**.

ОСНОВНЫЕ ПОНЯТИЯ

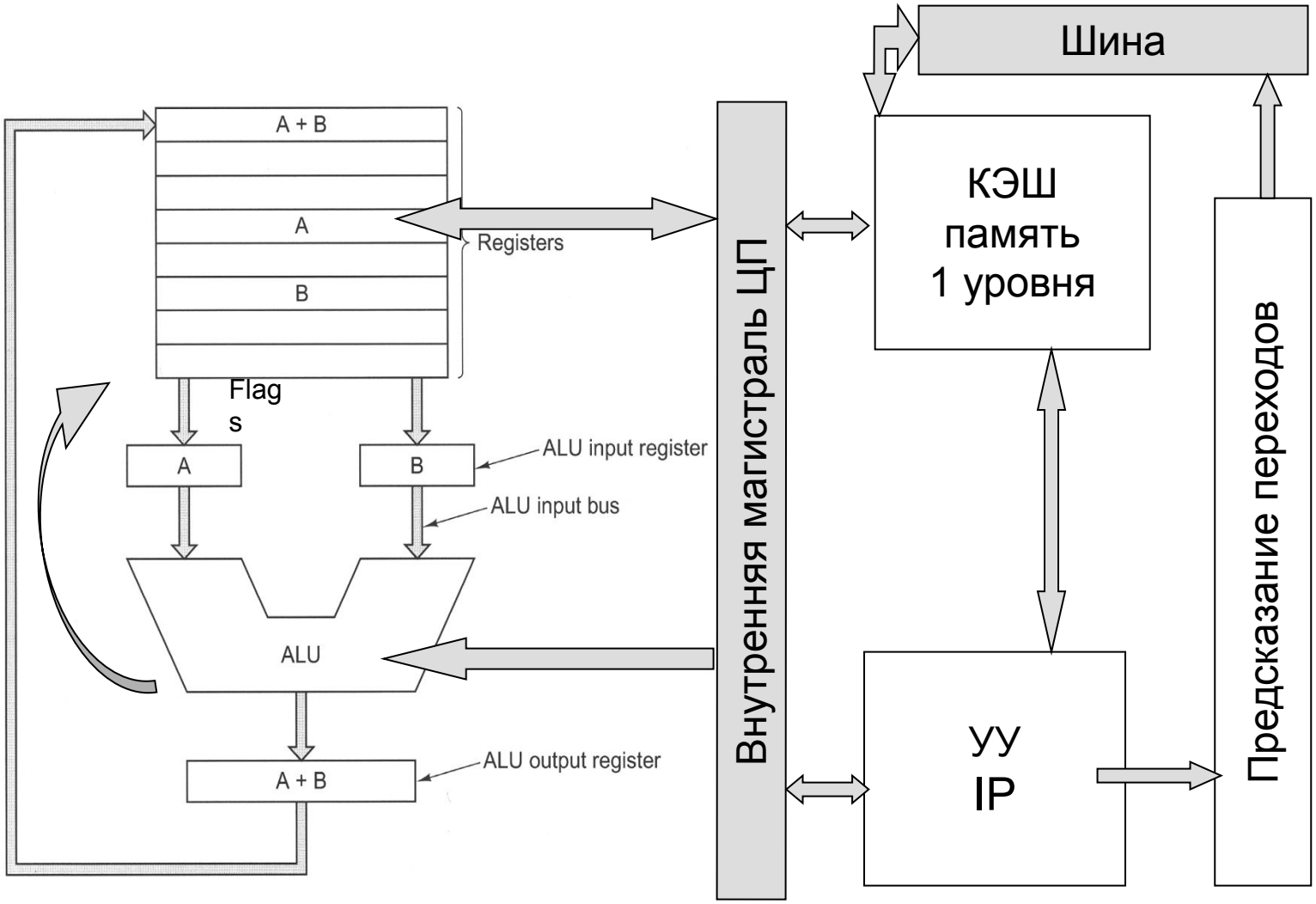
- **Тактовая частота –** характеризует производительность подсистемы (процессора, памяти и пр.), то есть количество выполняемых операций в секунду.

Системы с одной и той же тактовой частотой могут иметь различную производительность, так как на выполнение одной операции разным системам может требоваться различное количество тактов (обычно от долей такта до десятков тактов), а кроме того, системы, использующие конвейерную и параллельную обработку, могут на одних и тех же тактах выполнять одновременно несколько операций.

- **Разрядность –** (условно) количество разрядов (импульсов),которое может обработать процессор за один такт.

Разрядность процессора связана прежде всего с разрядностью регистров процессора, а также с разрядностью шины данных**.**

ВНУТРЕННЯЯ СТРУКТУРА ПРОЦЕССОРА



ПРОЦЕСС ВЫПОЛНЕНИЯ ПРОГРАММЫ

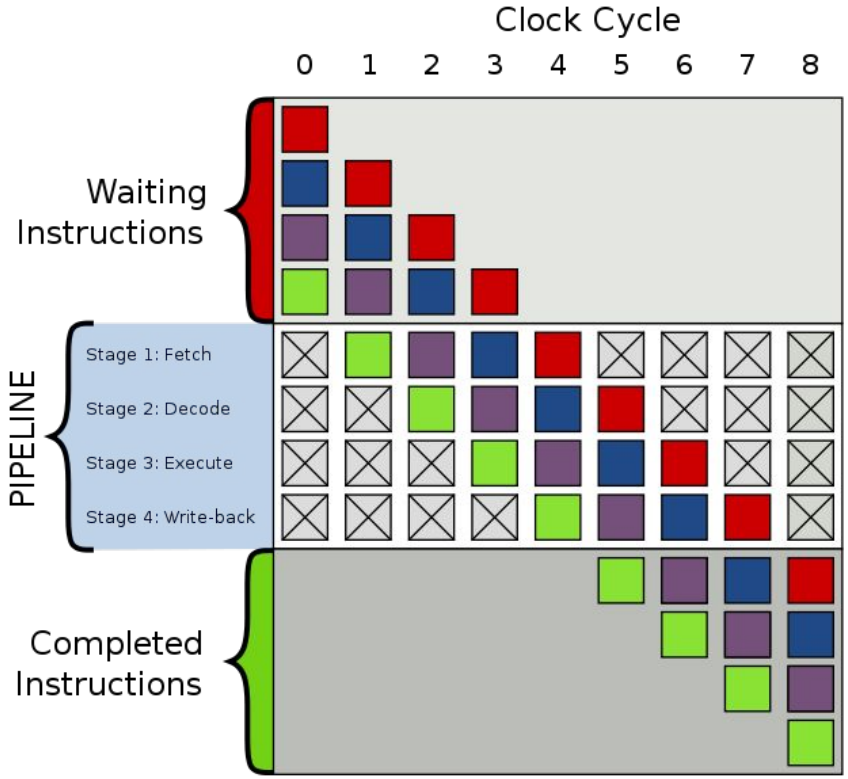
Этапы цикла выполнения:

1. Процессор выставляет число, хранящееся в регистре счётчика команд, на шину адреса и отдаёт памяти команду чтения.
2. Выставленное число является для памяти адресом; память, получив адрес и команду чтения, выставляет содержимое, хранящееся по этому адресу, на шину данных и сообщает о готовности.
3. Процессор получает число с шины данных, интерпретирует его как команду (машинную инструкцию) из своей системы команд и исполняет её.
4. Если последняя команда не является командой перехода, процессор увеличивает на единицу (в предположении, что длина каждой команды равна единице) число, хранящееся в счётчике команд, в результате там образуется адрес следующей команды.

Данный цикл выполняется неизменно, и именно он называется **процессом** (откуда и произошло название устройства).

КОНВЕЙЕР ПРОЦЕССОРА

- **Конвейер** — это способ организации вычислений, используемый в современных процессорах и контроллерах с целью ускорения выполнения инструкций (увеличения числа инструкций, выполняемых в единицу времени).



1. Получение – Fetch
2. Раскодирование – *Decode*
3. Выполнение – *Execute*
4. Запись результата – *Write-back*

УСТРОЙСТВО КОМПЬЮТЕРНЫХ СИСТЕМ

Гарвардская архитектура отличается от архитектуры фон Неймана тем, что программный код и данные хранятся в разной памяти.

- В гарвардская архитектуре невозможны многие методы программирования (например, невозможно динамически перераспределять память между программным кодом и данными);
- гарвардская архитектура позволяет более эффективно выполнять работу в случае ограниченных ресурсов, поэтому она часто применяется во встраиваемых системах.

КЛАССЫ АРХИТЕКТУР МИКРОПРОЦЕССОРОВ

- **CISC архитектура (Complex Instruction Set Computer) –** исходная архитектуры, обладает полным набором инструкций.
- **RISC архитектура (Reduced Instruction Set Computers) –** сравнительно небольшой (сокращённый) набор наиболее используемых команд определённый в результате статистического анализа большого числа программ для основных областей применения.
- Арифметику **RISC** процессоров отличает высокая степень дробления конвейера.
  - Этот прием позволяет увеличить тактовую частоту (значит, и производительность) компьютера; чем более элементарные действия выполняются в каждой фазе работы конвейера, тем выше частота его работы.
- **RISC** - процессоры в 2 - 4 раза быстрее имеющих ту же тактовую частоту **CISC** - процессоров с обычной системой команд и высокой производительностью, несмотря на больший объем программ, на (30 %).

ПАРАЛЛЕЛЬНАЯ АРХИТЕКТУРА

- Архитектура фон Неймана обладает тем недостатком, что она последовательная. Какой бы огромный массив данных ни требовалось обработать, каждый его байт должен будет пройти через центральный процессор, даже если над всеми байтами требуется провести одну и ту же операцию.
- Для преодоления этого недостатка предлагались и предлагаются архитектуры процессоров, которые называются параллельными.
- Возможными вариантами параллельной архитектуры могут служить следующие способы обработки данных :
- **SISD** — один поток команд, один поток данных;
- **SIMD** — один поток команд, много потоков данных;
- **MISD** — много потоков команд, один поток данных;
- **MIMD** — много потоков команд, много потоков данных.