## Приветствую читателей курса "Архитектура современных ЭВМ"!

Материал курса будет передаваться регулярно через старосту курса. В середине октября и конце ноября запланировано наше общение собрание (через ZOOM) для взаимных вопросов и ответов.

Каждому студенту будет передано персональное задание для подготовки к экзамену. Экзамен будет проводиться последовательно отдельно для каждого студента (через ZOOM) за примерно 20 минут.

В любое время мне можно задавать вопросы по почте: **tom11@bk.ru**

**Профессор Томилин Александр Николаевич**

## Введение

Целью курса "Архитектура современных ЭВМ**"** является рассмотрение способов организации параллелизма обработки информации на различных уровнях вычислительных систем. Будут рассмотрены концептуальные (архитектурные) решения такой организации и структурные решения, позволяющие реализовать предложенные концепции. Следует отметить, что области сущностей вычислительных систем, покрываемые понятиями “архитектура” и “структура”, заметно пересекаются, что приводит к неоднозначному использованию этих понятий в описаниях.

Понятие архитектуры параллельной вычислительной системы является широким, поскольку под архитектурой понимается: общая организация параллельной обработки данных, используемая в системе, особенности системы команд процессоров, организация памяти, топология связи между процессорами.

Попытки систематизировать всё множество архитектур предпринимались, начиная с 60-х годов прошлого века, и продолжаются сейчас.

Часто используют два понимания термина “архитектура” – в ”узком” и в “широком” аспекте.

С термином “архитектура” в узком аспекте связываются все те сущности вычислительной системы, которые определяют особенности программирования для неё, в первую очередь, система команд, выполняемых процессорами.

С термином “архитектура” в широком аспекте связывается разделение вычислительной системы на уровни с определением интерфейсов между ними (например, уровни организации аппаратуры, функций системного программного обеспечения – операционных систем и систем программирования, прикладного программного обеспечения).

В 1966 году М. Флинном (M. Flynn) был предложен подход к классификации архитектур вычислительных систем, основанный на понятии последовательностей (потоков) команд и данных (под потоками данных в основном понимаются отдельные наборы данных, обрабатываемых в задачах (процессах).

В классификации Флинна описываются четыре архитектурных класса:

OKOD (SISD) - одиночный поток команд и одиночный поток данных;

OKMD (SIMD) - одиночный поток команд и множественные потоки данных;

MKOD (MISD) - множественные потоки команд и одиночный поток данных;

MKMD (MIMD) – множественные потоки команд и множественные потоки данных.

**OKOD**. К этому классу относятся компьютерные системы, которые имеют один процессор, способный в каждый момент обрабатывать один поток последовательно исполняемых команд (инструкций) над данными задачи.

**OKMD.** В этих системах одна и та же команда параллельно выполняется над разными данными.

К OKMD – систем относят и "векторные" компьютеры. Команды векторных компьютеров оперируют массивами (векторами) данных подобно тому, как команды “скалярных” машин оперируют отдельными элементами данных.

**MKOD.** Теоретически в этом классе машин многие потоки команд должно выполняться над единственным потоком данных. Реальных машин этого класса создано не было.

**MKMD.** Эти машины параллельно выполняют несколько потоков команд над различными потоками данных. Этими потоками команд могут быть программы ветвей (параллельных процессов) одной задачи или программы разных задач.

Подробная организация параллелизма в вычислительных системах классов OKOD, OKMD и MKMD будет являться предметом рассмотрения в представляемом курсе.

### 

### Внутрипроцессорный параллелизм

В процессе совершенствования процессоров вычислительных машин было пройдено много этапов развития внутрипроцессорного параллелизма:

* организация параллельной работы основных устройств – арифметико-логического устройства (АЛУ), устройства управления (УУ), оперативной памяти (ОП) за счет введения устройств промежуточного хранения обрабатываемой информации – возникновение конвейера исполняемых команд в процессоре;
* разделение АЛУ на функциональные устройства (ФУ) выполнения отдельных операций (например, сложения, умножения, деления) и организация параллельной работы ФУ;
* организация конвейерного выполнения операций в ФУ;
* организация совмещенного выполнения нескольких потоков команд в обрабатывающем узле процессора за счёт чередования выполнения команд из разных потоков для эффективной загрузки ФУ;
* организация параллельного выполнения нескольких потоков команд в обрабатывающем узле процессора за счёт параллельного использования в нём нескольких комплектов оборудования (“многоядерность”)

**Параллелизм работы основных устройств ЭВМ.**

В самых первых ЭВМ основные устройства – ОП, УУ, АЛУ работали последовательно.

Вначале осуществлялся выбор текущей команды программы из оперативной памяти. Устройства УУ и АЛУ простаивали.

Затем выполнялась дешифрация в УУ выбранной команды и определялись адреса нахождения в ОП операндов, необходимых для ее выполнения операции в АЛУ (или определялся адрес в ОП команды, на которую следует передать управление). ОП и АЛУ простаивали.

На следующем этапе снова работала ОП – выполнялся выбор операндов или первой команды из указанной новой последовательности команд. УУ и АЛУ простаивали.

Затем с использованием выбранных операндов операция, указанная в команде, выполнялась в АЛУ (если команда не являлась командой передачи управления). ОП и УУ простаивали.

Последним этапом было осуществление записи в ОП результата выполненной в АЛУ операции (если запись предусматривалась в команде). УУ и АЛУ простаивали.

После осуществления указанных этапов выполнения команды начинался аналогичный цикл действий для выполнения следующей команды. Простои в работе основных устройств ЭВМ были весьма значительными.

Введение двух новых структурных решений позволило ликвидировать простои в работе основных устройств ЭВМ и обеспечить их параллельную работу.

Этими решениями были:

* организация параллельной работы нескольких блоков оперативной памяти (“расслоение” памяти);
* организация буферного хранения информации, передаваемой между основными устройствами ЭВМ.

Одновременно существенному уменьшению количества команд в программе и числа обращений к оперативной памяти послужило:

* введение в устройстве управления индексных регистров, используемых для автоматической модификации адресов операндов (например, для работы с массивами данных в программных циклах);
* организация автоматического повторного выбора из буферных устройств промежуточного хранения информации (без обращения к оперативной памяти) часто используемых команд и данных – организация “кэш-памятей”..

Уменьшению числа обращений к оперативной памяти способствовало также введение для часто используемых данных регистров, непосредственно указываемых в командах (регистров “общего назначения”).

Реализация параллелизма работы основных устройств ЭВМ удачно осуществлена в отечественной универсальной ЭВМ БЭСМ-6 (1967г.), созданной в Институте точной механики и вычислительной техники (ИТМ и ВТ) под руководством академика Сергея Алексеевича Лебедева.

Образовавшийся в БЭСМ-6 “конвейер команд” - параллельное выполнение нескольких команд на разных стадиях их исполнения (от выборки из ОП до выполнения в АЛУ и записи результата выполнения в ОП) позволил в 4 раза увеличить быстродействие машины по сравнению с использованием последовательной работы её основных устройств при той же элементной базе и конструкции ЭВМ.

Использование в командах “прямо адресуемых” регистров было реализовано в ЭВМ IBM-360 и затем в производимых в СССР и странах Совета экономической взаимопомощи (СЭВ) ЭВМ “единой системы” (ЕС ЭВМ).

**Структурная схема БЭСМ-6 и описание параллелизма работы основных устройств этой машины будут представлены в следующем материале.**

**Этапы "доэлектронной" и "допараллельной электронной" организации вычислительных машин отражены в прилагаемой Презентации 1**