

## ЭВМ Лаба 2

### 1. На чем базируется классификация Флинна?

Самая ранняя и наиболее известная классификация архитектур вычислительных систем была предложена Майклом Флинном в 1966 году. Классификация базируется на понятии потока, под которым понимается последовательность элементов (команд или данных) обрабатываемая процессором. На основе числа потоков команд и потоков данных Флинн выделяет четыре класса архитектур ВС:

1. SISD (single instruction stream / single data stream) – одиночный поток команд и одиночный поток данных. К этому классу относятся однопроцессорные однопоточные машины фон- Неймановского типа. В таких машинах есть только один поток команд, все команды обрабатываются последовательно друг за другом и каждая команда инициирует одну операцию с одним потоком данных. При этом для увеличения скорости обработки команд и скорости выполнения арифметических операций может использоваться конвейерная обработка. Данный класс включает в себя ЭВМ с CISC и RISC архитектурой, а также ЭВМ с суперскалярной обработкой данных.

2. SIMD (single instruction stream / multiple data stream) – одиночный поток команд и множественный поток данных. В архитектурах подобного рода сохраняется один поток команд, включающий, в отличие от предыдущего класса, векторные команды. Это позволяет выполнять одну арифметическую операцию сразу над многими данными – элементами вектора.

3. MISD (multiple instruction stream / single data stream) – множественный поток команд и одиночный поток данных. Определение подразумевает наличие в архитектуре многих процессоров, обрабатывающих один и тот же поток данных. В настоящее время считается, что данный класс пуст.

4. MIMD (multiple instruction stream / multiple data stream) – множественный поток команд и множественный поток данных. Этот класс предполагает, что в вычислительной системе есть несколько процессоров, которые (как правило, асинхронно) выполняют разные команды над разными данными.

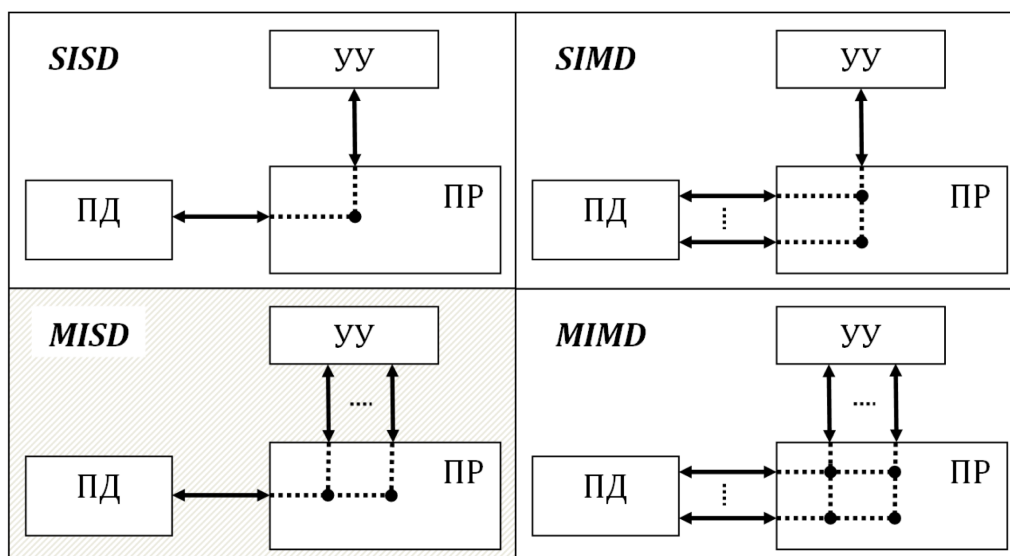


Рис. 1.1. Классификация Флинна:

ПР – процессорные элементы (один или несколько); УУ – устройство управления; ПД – память данных

Предложенная схема классификации (рис. 1.1) и сейчас является самой применяемой при начальной характеристике той или иной вычислительной системы. Если говорится, что компьютер принадлежит классу SIMD или MIMD, то сразу становится понятным базовый принцип его работы.

**2.К какому классу по Флинну относится ЭВМ фон-неймановского типа?**  
ЭВМ фон Неймана относится к SISD

**3.К какому по Флинну относятся векторно-конвейерные и векторно-параллельные ВС?**

SIMD (single instruction stream / multiple data stream) – одиночный поток команд и множественный поток данных. В архитектурах подобного рода сохраняется один поток команд, включающий, в отличие от предыдущего класса, векторные команды. Это позволяет выполнять одну арифметическую операцию сразу над многими данными – элементами вектора.

**4.Классификация ВС по типу строения оперативной памяти.**

По типу строения оперативной памяти системы разделяются на:

Системы с общей (разделяемой) памятью;

Системы с распределенной памятью;

Системы с физически распределенной, а логически общедоступной памятью (гибридные системы).

В вычислительных системах с общей памятью (Common Memory Systems или Shared Memory Systems) значение, записанное в память одним из процессоров, напрямую доступно для другого процессора. Общая память обычно имеет высокую пропускную способность памяти (bandwidth) и низкую латентность памяти (latency) при передачи информации между процессорами, но при условии, что не происходит одновременного обращения нескольких процессоров к одному и тому же элементу памяти.

Доступ разных процессоров ВС к общей памяти осуществляется, как правило, за одинаковое время. Поэтому такая память называется еще UMA памятью (Unified Memory Access) - памятью с одинаковым временем доступа. Системы с общей памятью называются также сильносвязанными вычислительными системами.

В вычислительных системах с распределенной памятью (Distributed Memory Systems) каждый процессор имеет свою локальную память с локальным адресным пространством. Для систем с распределенной памятью характерно наличие большого числа быстрых каналов передачи данных, которые связывают отдельные части этой памяти с отдельными процессорами. Обмен информацией между частями распределенной памяти осуществляется обычно относительно медленно. Системы с распределенной памятью называются также слабосвязанными вычислительными системами.

Вычислительные системы с гибридной памятью - (Non-Uniform Memory Access Systems) имеют память, которая физически распределена по различным частям системы, но логически разделяема (образует единое адресное пространство). Такая память называется еще логически общей (разделяемой) памятью (logically shared memory). В отличие от UMA-систем, в NUMA-системах время доступа к различным частям оперативной памяти различно.

### 5. Детализация класса MIMD.

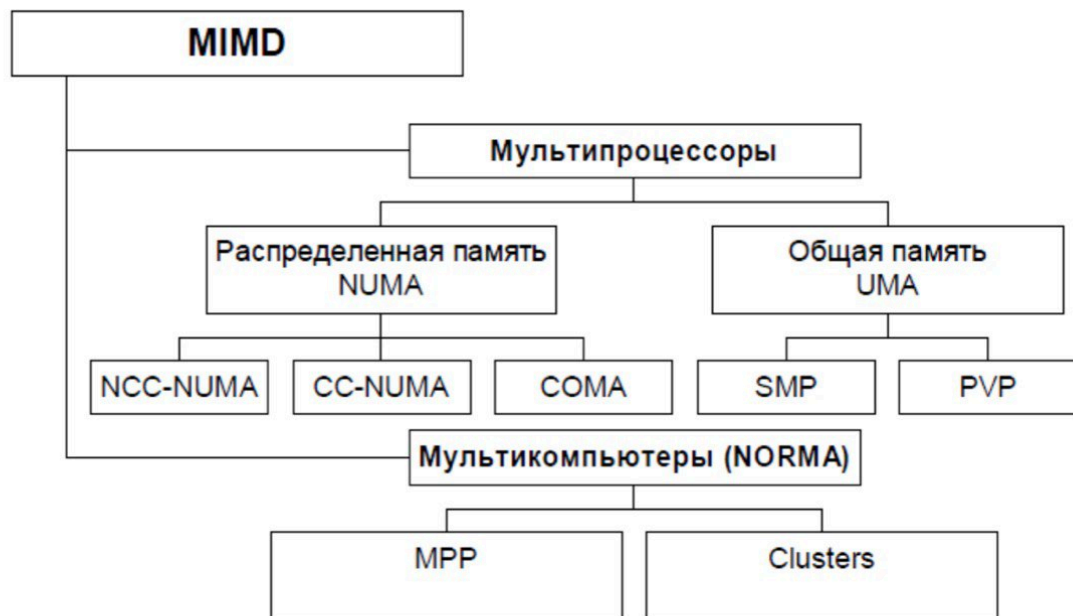


Рис. 5. Детализация класса MIMD

6. Какой принцип распараллеливания используется при распараллеливании арифметических выражений?

7. Почему арифметические выражения относят к классу задач с неявным параллелизмом?

8. Что такое ЯПФ?

9. Какие основные свойства арифметических операций используются при преобразовании арифметических выражений в эквивалентные?

10. Какие характеристики параллельности Вы знаете?

11. Чем характеризуется эффективность схемы при оптимальной загрузке процессоров?

12. В чем основной смысл леммы Брента?

13. UMA & NUMA & COMA

**UMA** – архитектура с однородным доступом к памяти. В машинах UMA каждый процессор имеет одно и то же время доступа к памяти.

**NUMA** – архитектура с неоднородным доступом к памяти.

**COMA** – архитектура с доступом только к кэш-памяти.

## UMA

В основе самых простых мультипроцессоров лежит одна шина. Если процессору нужно считать информацию из памяти, он проверяет, свободна ли шина.

Если шина занята, процессор ждет. При наличии двух или трех процессоров доступ к шине вполне управляем. При наличии большего числа процессоров (32, 64 и т.д.) производительность системы полностью ограничивается пропускной способностью шины, а большинство процессоров будут простаивать.

Для разрешения этой проблемы необходимо добавить кэш-память к каждому процессору. Поскольку теперь слова можно получать из кэш-памяти, то и движения в шине будет меньше и система сможет поддерживать большее число процессоров. Следующая разработка – каждый процессор имеет не только кэш-память, но и свою локальную память, к которой он получает доступ через локальную шину.

Каждый элемент кэш-памяти может находиться в одном из четырех состояний (протокол MESI):

1. Invalid – элемент кэш-памяти содержит недействительные данные.
2. Shared – несколько кэшей могут содержать данную строку, основная память обновлена.
3. Exclusive – никакой другой кэш не содержит эту строку, основная память обновлена.
4. Modified – элемент действителен, основная память недействительна, копий элемента не существует.

### UMA-мультипроцессоры с перекрестной коммутацией

Имеются 4 адресные шины, которые используются для отслеживания строк в кэш-памяти. Каждая шина использует  $\frac{1}{4}$  часть физического адресного пространства. Для выбора шины используется 2 адресных бита.

### UMA-мультипроцессоры с многоступенчатой коммутацией

Коммутатор содержит два входа и два выхода. Сообщения, приходящие на один из входов могут переключаться на любой выход. Коммутаторы 2 x 2 можно компоновать различными способами и получать многоступенчатые сети.

## NUMA

Мультипроцессоры NUMA (Non Uniform Memory Access – с неоднородным доступом к памяти). Как и мультипроцессоры UMA, они обеспечивают единое адресное пространство для всех процессоров, но в отличие от машин UMA, доступ к локальным модулям памяти происходит быстрее, чем к удаленным. Следовательно, все программы UMA будут работать без изменений на машинах NUMA, но производительность будет хуже, чем на машинах UMA с той же тактовой частотой.

Машины NUMA три ключевые характеристики, которыми все они обладают и которые в совокупности отличают их от других мультипроцессоров.

1. Существует одно адресное пространство, видимое для всех процессоров.
2. Доступ к удаленной памяти осуществляется с использованием команд **load** и **store**.
3. Доступ к удаленной памяти происходит медленнее, чем доступ к локальной памяти.

Мультипроцессор Stanford DASH.

Мультипроцессор Sequent NUMA-Q.

NUMA-мультипроцессор Sun Fire E25K.

## **СОМА**

Основная память каждого процессора используется как кэш-память. Такая машина называется СОМА (Cache Only Memory Access). При такой организации памяти страницы не имеют собственных фиксированных машин.

Память, которая привлекает строки по мере необходимости, называется *attraction memory*. Использование основной памяти в качестве большой кэш-памяти увеличивает частоту успешных обращений в кэш-память, а, следовательно, и производительность.

В результате появляются две новые проблемы:

1. как разместить строки кэш-памяти;
2. если строка удаляется из памяти, что произойдет, если это последняя копия.

Машина СОМА обещает существенно увеличить производительности системы, но в настоящее время известно очень мало машин СОМА (2 машины: KSR-1 и Data Diffusion Machine).

**Введение.** Основным параметром классификации параллельных компьютеров является наличие общей (SMP) или распределенной памяти (MPP). Нечто среднее между SMP и MPP представляют собой NUMA-архитектуры, где память физически распределена, но логически общедоступна. Кластерные системы являются более дешевым вариантом MPP. При поддержке команд обработки векторных данных говорят о векторно-конвейерных процессорах, которые, в свою очередь могут объединяться в PVP-системы с использованием общей или распределенной памяти. Все большую популярность приобретают идеи комбинирования различных архитектур в одной системе и построения неоднородных систем.

При организациях распределенных вычислений в глобальных сетях (Интернет) говорят о **мета-компьютерах**, которые, строго говоря, не представляют из себя параллельных архитектур.

Более подробно особенности всех перечисленных архитектур будут рассмотрены далее на этой странице, а также в описаниях конкретных компьютеров – представителей этих классов. Для каждого класса приводится следующая информация:

- краткое описание особенностей архитектуры,
- примеры конкретных компьютеров,
- перспективы масштабируемости,
- типичные особенности построения операционных систем,
- наиболее характерная модель программирования (хотя возможны и другие).

**Примечание.** На данной странице рассматриваются наиболее типичные классы архитектур современных параллельных компьютеров и супер-ЭВМ. (Не рассматриваются устаревшие и проектируемые архитектуры.)

## Массивно-параллельные системы (MPP)

Архитектура	<div>Система состоит из однородных <i>вычислительных узлов</i>, включающих:</div> <ul style="list-style-type: none"><li>• один или несколько центральных процессоров (обычно RISC),</li><li>• локальную память (прямой доступ к памяти других узлов невозможен),</li><li>• коммуникационный процессор или сетевой адаптер</li><li>• иногда – жесткие диски (как в SP) и/или другие устройства В/В</li></ul> <div>К системе могут быть добавлены специальные узлы ввода-вывода и управляющие узлы. Узлы связаны через некоторую коммуникационную среду (высокоскоростная сеть, коммутатор и т.п.)</div>
Примеры	IBM RS/6000 SP2, Intel PARAGON/ASCI Red, CRAY T3E, Hitachi SR8000, транспьютерные системы Parsytec.
Масштабируемость	Общее число процессоров в реальных системах достигает нескольких тысяч (ASCI Red, Blue Mountain).
Операционная система	<div>Существуют два основных варианта:</div> <div>1. Полноценная ОС работает только на управляющей машине (front-end), на каждом узле работает сильно урезанный вариант ОС, обеспечивающие только работу расположенной в нем ветви параллельного приложения. Пример: Cray T3E.</div> <div>2. На каждом узле работает полноценная UNIX-подобная ОС (вариант, близкий к <i>кластерному</i> подходу). Пример: IBM RS/6000 SP + ОС AIX, устанавливаемая отдельно на каждом узле.</div>
Модель программирования	Программирование в рамках модели <i>передачи сообщений</i> ( MPI, PVM, BSPlib)

Дополнительная информация:

- [Лекция об архитектуре массивно-параллельных компьютеров, на примере CRAY T3D](#) (Вл.В.Воеводин).
- [Параллельные компьютеры с распределенной памятью](#) <sup>PDF</sup> (статья в ComputerWorld, #22, 1999 г.)
- [An Overview of the Intel TFLOPS Supercomputer](#) <sup>PDF</sup>

## Симметричные мультипроцессорные системы (SMP)

Архитектура	Система состоит из нескольких однородных процессоров и массива общей памяти (обычно из нескольких независимых блоков). Все процессоры имеют доступ к любой точке памяти с одинаковой скоростью. Процессоры подключены к памяти либо с помощью общей шины (базовые 2–4 процессорные SMP-сервера), либо с помощью crossbar-коммутатора (HP 9000). Аппаратно поддерживается когерентность кэшей.
Примеры	HP 9000 V-class, N-class; SMP-сервера и рабочие станции на базе процессоров Intel (IBM, HP, Compaq, Dell, ALR, Unisys, DG, Fujitsu и др.).
Масштабируемость	Наличие общей памяти сильно упрощает взаимодействие процессоров между собой, однако накладывает сильные ограничения на их число – не более 32 в реальных системах. Для построения масштабируемых систем на базе SMP используются <i>кластерные</i> или NUMA-архитектуры.
Операционная система	Вся система работает под управлением единой ОС (обычно UNIX-подобной, но для Intel-платформ поддерживается Windows NT). ОС автоматически (в процессе работы) распределяет процессы/нити по процессорам (scheduling), но иногда возможна явная привязка.
Модель программирования	Программирование в модели <i>общей памяти</i> . (POSIX threads, OpenMP). Для SMP-систем существуют сравнительно эффективные средства автоматического распараллеливания.



Системы с неоднородным доступом к памяти (NUMA)

Архитектура	Система состоит из однородных базовых модулей (плат), состоящих из небольшого числа процессоров и блока памяти. Модули объединены с помощью высокоскоростного коммутатора. Поддерживается единое адресное пространство, аппаратно поддерживается доступ к удаленной памяти, т.е. к памяти других модулей. При этом доступ к локальной памяти в несколько раз быстрее, чем к удаленной. В случае, если аппаратно поддерживается когерентность кэшей во всей системе (обычно это так), говорят об архитектуре <b>cc-NUMA</b> (cache-coherent NUMA)
Примеры	HP HP 9000 V-class в SCA-конфигурациях, SGI Origin2000, Sun HPC 10000, IBM/Sequent NUMA-Q 2000, SNI RM600.
Масштабируемость	Масштабируемость NUMA-систем ограничивается объемом адресного пространства, возможностями аппаратуры поддержки когерентности кэшей и возможностями операционной системы по управлению большим числом процессоров. На настоящий момент, максимальное число процессоров в NUMA-системах составляет 256 (Origin2000).
Операционная система	Обычно вся система работает под управлением единой ОС, как в SMP. Но возможны также варианты динамического "подразделения" системы, когда отдельные "разделы" системы работают под управлением разных ОС (например, Windows NT и UNIX в NUMA-Q 2000).
Модель программирования	Аналогично SMP.

Параллельные векторные системы (PVP)

Архитектура	Основным признаком PVP-систем является наличие специальных векторно-конвейерных процессоров, в которых предусмотрены команды однотипной обработки векторов независимых данных, эффективно выполняющиеся на конвейерных функциональных устройствах. Как правило, несколько таких процессоров (1-16) работают одновременно над общей памятью (аналогично SMP) в рамках многопроцессорных конфигураций. Несколько таких узлов могут быть объединены с помощью коммутатора (аналогично MPP).
Примеры	NEC SX-4/SX-5, линия векторно-конвейерных компьютеров CRAY: от CRAY-1, CRAY J90/T90, CRAY SV1, CRAY X1, серия Fujitsu VPP.
Модель программирования	Эффективное программирование подразумевает векторизацию циклов (для достижения разумной производительности одного процессора) и их распараллеливание (для одновременной загрузки нескольких процессоров одним приложением).

Дополнительная информация:

- [Лекция](#) об архитектуре векторно-конвейерных супер-ЭВМ CRAY C90. (Вл.В.Воеводин)

Кластерные системы

Архитектура	Набор рабочих станций (или даже ПК) общего назначения, используется в качестве дешевого варианта <b>массивно-параллельного</b> компьютера. Для связи узлов используется одна из стандартных сетевых технологий (Fast/Gigabit Ethernet, Myrinet) на базе шинной архитектуры или коммутатора.  При объединении в кластер компьютеров разной мощности или разной архитектуры, говорят о <b>гетерогенных</b> (неоднородных) кластерах.  Узлы кластера могут одновременно использоваться в качестве пользовательских рабочих станций. В случае, когда это не нужно, узлы могут быть существенно облегчены и/или установлены в стойку.
Примеры	NT-кластер в NCSA, Beowulf-кластеры.
Операционная система	Используются стандартные для рабочих станций ОС, чаще всего, свободно распространяемые – Linux/FreeBSD, вместе со специальными средствами поддержки параллельного программирования и распределения нагрузки.
Модель программирования	Программирование, как правило, в рамках модели передачи сообщений (чаще всего – MPI). Дешевизна подобных систем оборачивается большими накладными расходами на взаимодействие параллельных процессов между собой, что сильно сужает потенциальный класс решаемых задач.