



ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ



## ЛЕКЦИЯ 5

# Универсальные микропроцессоры. Вычислительные системы класса SIMD, MIMD

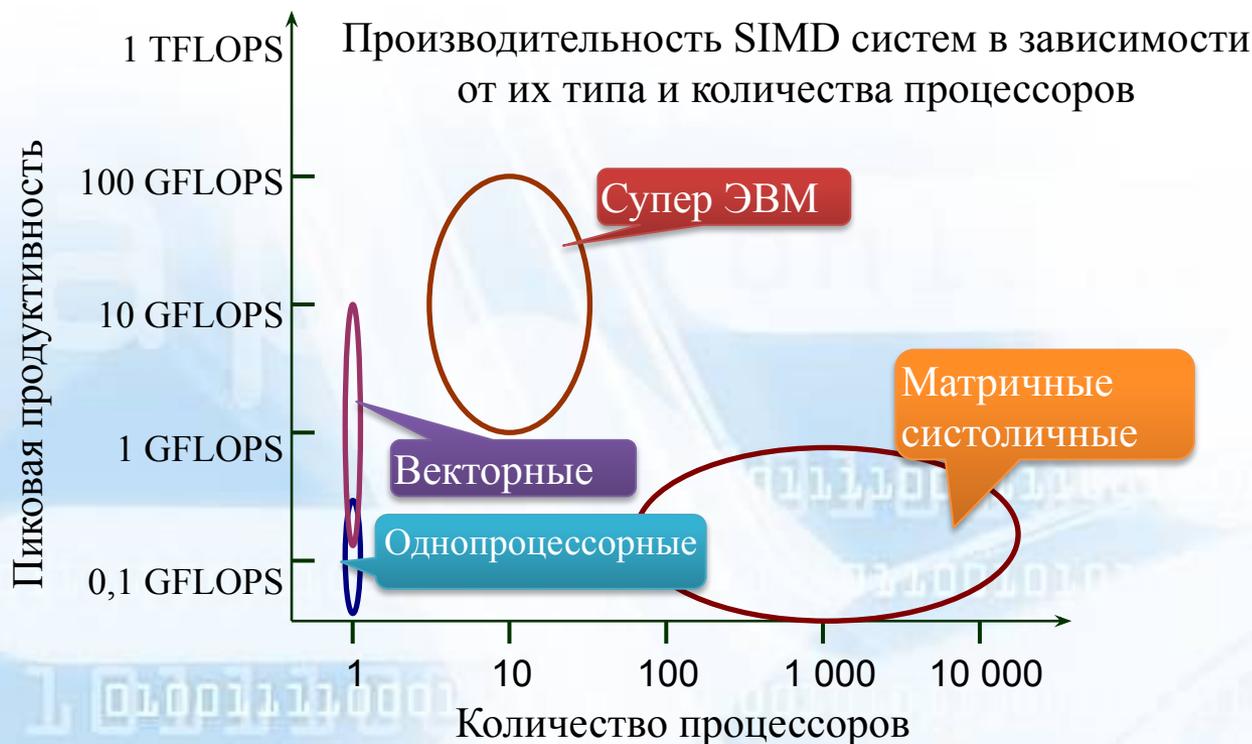
доцент кафедры информационных систем  
к.т.н., с.н.с. Евсеев Сергей Петрович

# ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ КЛАССА SIMD

К классу *SIMD* относят векторные, векторно-конвейерные, матричные, ассоциативные, систолические и системы с сверхдлинными командами.

Для них характерно:

- ❑ много элементов данных подвергаются параллельной, но однотипной обработке;
- ❑ используется одно устройство управления.



# ВЕКТОРНЫЕ И ВЕКТОРНО-КОНВЕЙЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

В средствах векторной обработки под *вектором* понимается одномерный массив однотипных данных (обычно в форме с плавающей запятой), регулярным образом расположенных в памяти ОС. Многомерные массивы рассматриваются как векторы.

## Размещение в памяти матрицы 4x5

$a_{11}$	$a_{12}$	$a_{13}$	$a_{14}$	$a_{15}$
$a_{21}$	$a_{22}$	$a_{23}$	$a_{24}$	$a_{25}$
$a_{31}$	$a_{32}$	$a_{33}$	$a_{34}$	$a_{35}$
$a_{41}$	$a_{42}$	$a_{43}$	$a_{44}$	$a_{45}$

## Размещение в памяти матрицы по строкам



## Размещение в памяти матрицы по столбцам



# ВЕКТОРНЫЕ И ВЕКТОРНО-КОНВЕЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

Обработка массивов сводится к однотипным операциям над парами элементов входных матриц, которые можно проводить параллельно во времени.

В универсальных ЭВМ, ориентированных на скалярные операции, обработка матриц осуществляется поэлементно и последовательно. Такая обработка занимает много времени и при большой размерности матриц универсальные компьютеры становятся неэффективными. Для решения таких задач используются специальные средства векторной обработки.

*Векторный процессор* – это процессор, в котором операндами некоторых операций могут выступать упорядоченные массивы данных - векторы.

*Векторный процессор* может быть реализован в двух вариантах:

- как дополнительный блок универсальной машины;
- как основа самостоятельной вычислительной системы.

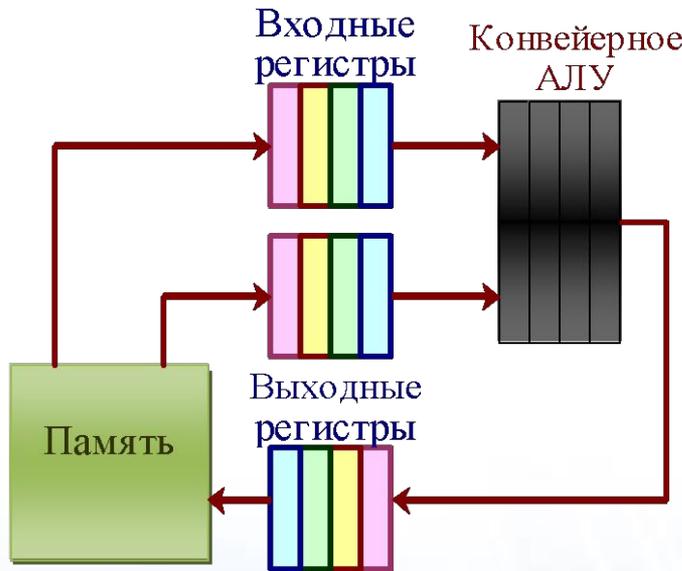
Средства векторной обработки по их архитектуре можно разделить на три группы:

- конвейерные арифметико-логические устройства;
- массив арифметико-логических устройств;
- массив процессорных элементов.

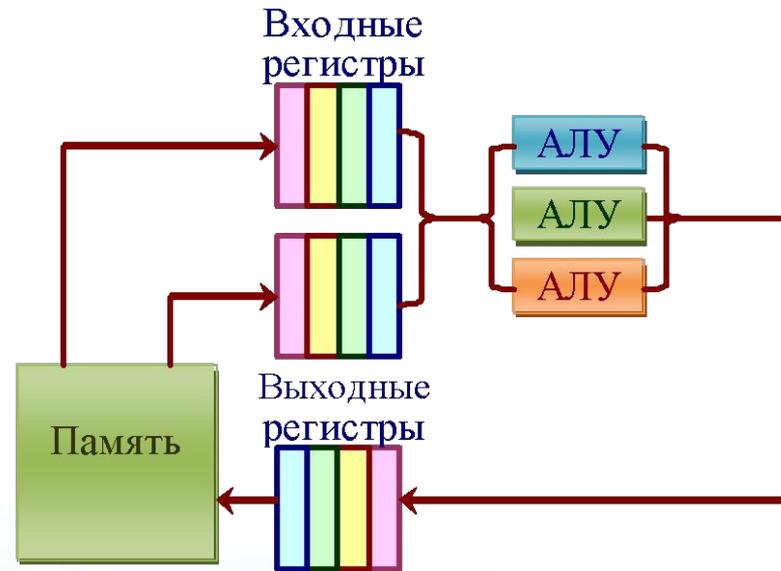
Первые две группы устройств используются при построении векторных процессоров, а третий - представляет многопроцессорную систему, которая носит название *матричной вычислительной системы*.

# ВЕКТОРНЫЕ И ВЕКТОРНО-КОНВЕЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

Вычисления с конвейерным АЛУ



Вычисления с несколькими АЛУ



В варианте с конвейерным АЛУ обработка элементов векторов осуществляется в виде операций над числами с плавающей запятой в виде конвейера. Операции над числами с плавающей запятой достаточно сложны, но поддаются разделению на отдельные шаги.

Так операцию сложения двух чисел можно разделить на четыре этапа.

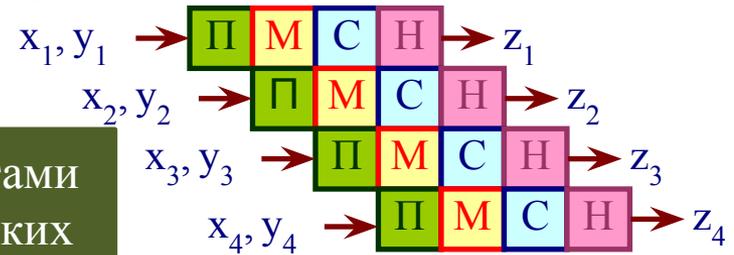
Каждый этап может быть реализован с помощью отдельной ступени конвейера.

## Структура арифметического конвейера для чисел с плавающей запятой



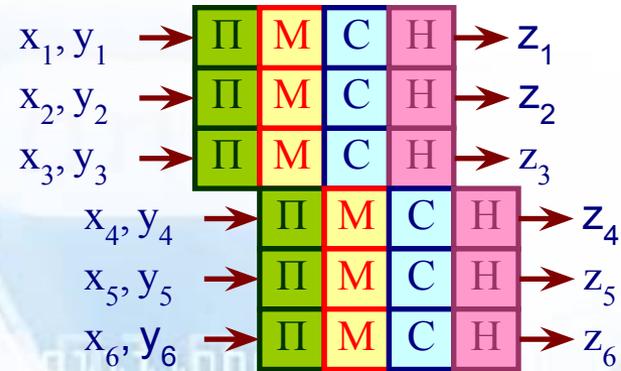
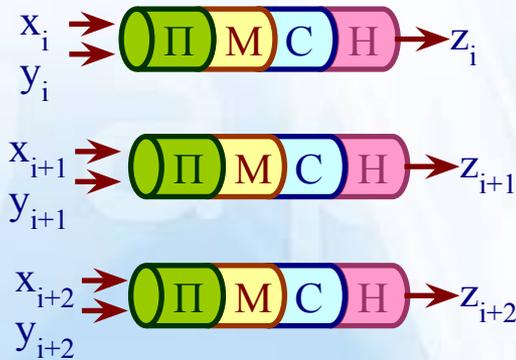
# ВЕКТОРНЫЕ И ВЕКТОРНО-КОНВЕЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

## Обработка векторов в конвейерном АЛУ



Одновременные операции над элементами векторов можно проводить с помощью нескольких АЛУ, работающих параллельно. Каждое АЛУ отвечает за обработку одной пары элементов массивов.

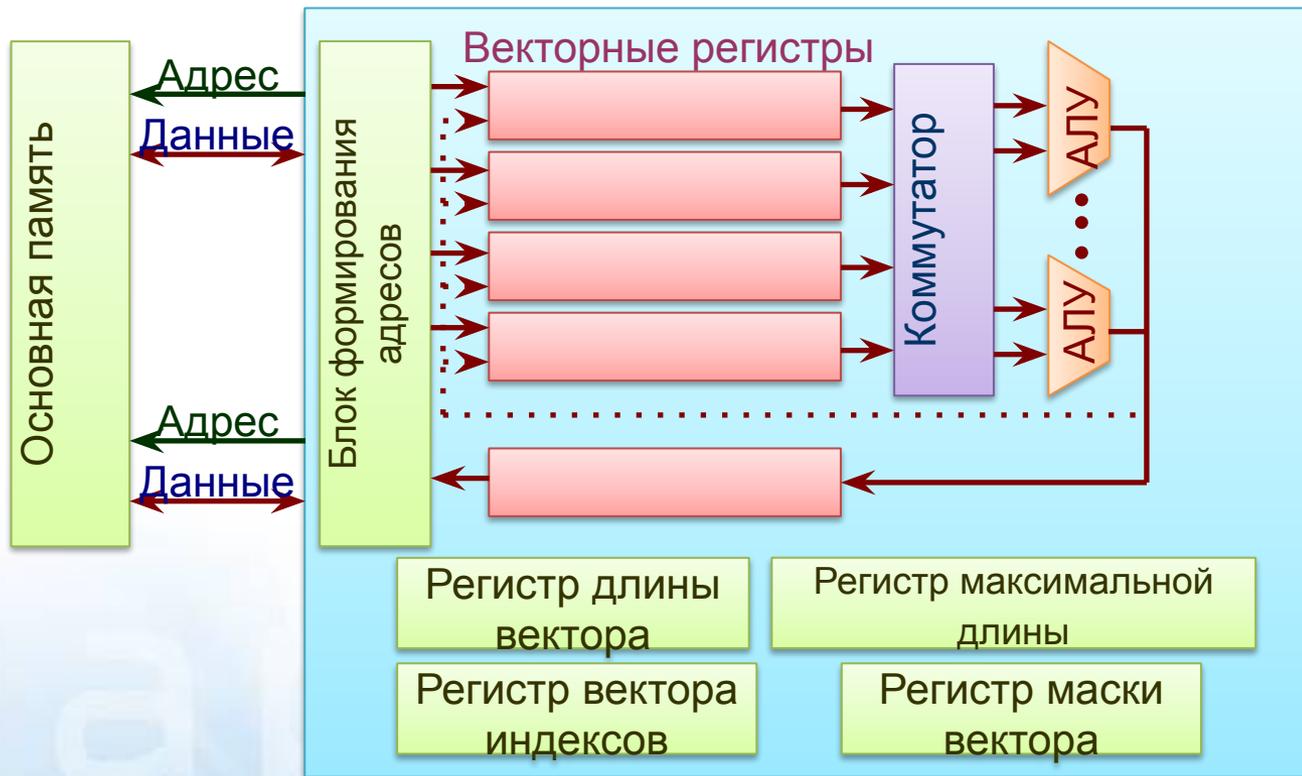
## Параллельная обработка векторов несколькими конвейерными АЛУ



При параллельно используемых конвейерных АЛУ используется еще один уровень конвейеризации. Вычислительные системы, в которых используется эта идея, называются векторно-конвейерными.

# ВЕКТОРНЫЕ И ВЕКТОРНО-КОНВЕЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

## Структурная схема векторного процессора



Обработка всех  $n$  компонентов векторов-операндов задается одной векторной командой.

АЛУ векторного процессора может реализовываться в виде единого конвейерного устройства, которое способно выполнять все операции над числами с плавающей запятой.

Более распространена схема в которой АЛУ состоит из нескольких блоков, каждый из которых выполняет одну операцию: сложения, деления, умножения. Каждый из этих блоков строится как конвейер. Кроме того, в состав процессора включаются блоки выполнения скалярных операций.

# ВЕКТОРНЫЕ И ВЕКТОРНО-КОНВЕЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

*Векторные регистры* используются для хранения операндов-векторов. Это совокупность скалярных регистров, которые соединены в очередь типа FIFO («First In, First Out» (англ.)). Принцип «первым пришёл — первым ушёл») и могут сохранять 50-100 чисел с плавающей запятой. Система команд векторного процессора поддерживает работу с векторными регистрами и обязательно включает в себя команды:

- загрузка векторного регистра содержанием последовательных ячеек памяти;
- выполнение операций над всеми элементами векторов, расположенных в векторных регистрах;
- хранение содержания векторного регистра в последовательности ячеек памяти, указанные адресу первой ячейки это последовательности.

*Регистр длины вектора* – определяет количество обрабатываемых элементов вектора, то есть количество повторений одной операции над элементами одной пары векторов.

*Регистр максимальной длины вектора* – определяет максимальное количество элементов вектора, которое можно одновременно обработать аппаратурой процессора. Регистр используется при разделении сильно длинных векторов на сегменты длины, соответствующей максимальному числу элементов, которые можно обработать в один прием.

*Регистр маски вектора* – указывает, какие элементы вектора результата следует записывать в векторный регистр результата. Каждому элементу вектора в регистре отведен один бит. Этот регистр используется в тех случаях, когда все элементы вектора принимают участие в векторной операции.

*Регистр вектора индекса* – имеет такой же формат, как регистр маски, и предназначен для выполнения операции упаковки/распаковки векторов. Они используются при обработке нескольких ненулевых элементов вектора. Операция упаковки формирует вектор, содержащий только ненулевые элементы входного вектора, операция распаковки проводит обратное преобразование.

# ВЕКТОРНЫЕ И ВЕКТОРНО-КОНВЕЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

## ИСПОЛЬЗОВАНИЕ ВЕКТОРНЫХ КОМАНД ПРИВОДИТ К ПОВЫШЕНИЮ ПРОИЗВОДИТЕЛЬНОСТИ БЛАГОДАРЯ ДВУМ КАЧЕСТВАМ:

- ❖ Вместо многократной выборки одной и той же скалярной команды проводится однократное чтение векторной команды, что позволяет сократить затраты времени в устройстве управления и уменьшить требования к пропускной способности памяти.
- ❖ Векторная команда обеспечивает процессор упорядоченными данными. При выполнении такой команды процессор знает, что ему нужно считать из основной памяти  $n$  пар операндов, расположенных в памяти регулярным способом. Когда используется память с чередованием адресов, эти пары могут быть получены со скоростью одной пары за один цикл процессора и направлены для обработки в конвейерный операционный блок. При отсутствии чередования памяти эффективность обработки векторов значительно снижается.

Приведенная модель векторного процессора используется в большинстве современных векторных вычислительных системах, например, семейства CREY - максимальная длина вектора - 64. Она использует операционные команды "Регистр-регистр", то есть вектора перед выполнением операции должны быть записаны во внутренние векторные регистры процессора, по окончании операции результат из векторного регистра должен быть перегружен в основную память. Такая система является наиболее эффективной при коротких векторах, когда их длина не больше длины векторных регистров.

Время загрузки векторов во внутренние регистры процессора называется процедурой запуска операции.

При длинных векторах такая система снижает эффективность, так как операция над всем вектором должна выполняться в несколько этапов. На каждом этапе нужно загружать и выгружать векторные регистры частями полных векторов. Это значительно увеличивает время запуска операции.

# ВЕКТОРНЫЕ И ВЕКТОРНО-КОНВЕЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

*Длинные операнды* более эффективно обрабатываются в системе типа "память-память", когда элементы считываются последовательно из памяти и непосредственно поступают в операционные блоки процессора и нет необходимости разбивать эти операнды на сегменты фиксированной длины. При этом отпадает необходимость иметь в процессоре векторные регистры.

Время выполнения операции в этом случае определяется временем полной загрузки конвейера из основной памяти:

$$T = s + \alpha \times N$$

где  $s$  – время запуска;

$\alpha$  – константа, зависящая от команды (обычно  $\frac{1}{2}$ , 1, или 2);

$N$  – длина вектора.

Время запуска операции в таком векторном процессоре значительно больше, чем в системе "регистр-регистр". Поэтому при коротких векторах такая система значительно ниже по эффективности.

К классу "память-память" относятся вычислительные системы Star-100, Cyber-200, ETA-10.

В настоящее время системы "память-память" практически не используются из-за большого времени запуска

# МАТРИЧНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

*Матричные вычислительные системы предназначены для обработки больших массивов данных. В основе матричных систем лежит матричный процессор, состоящий из регулярного массива процессорных элементов (ПЭ).*

## Обобщенная модель матричной SIMD системы



**Массив процессоров** – выполняет параллельную обработку элементов данных. Для хранения множественного набора данных в состав массива процессоров должно входить множество модулей памяти. Также в составе массива процессоров нужно реализовать сеть взаимосвязей как между процессорами, так и между процессорами и модулями памяти.

Дополнительную гибкость системе обеспечивает механизм маскировки, что позволяет включать в участие в операциях определенное количество процессоров. Маскировка осуществляется как на этапе компиляции, так и на этапе выполнения. Процессоры исключаются из обработки путем установления в нуль соответствующего разряда регистра маски массива процессоров.

# МАТРИЧНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

**Контроллер массива процессоров** – выполняет последовательный код, реализующий команды условного и безусловного перехода, транслирует в массив процессоров команды, данные и сигналы управления. Команды обрабатываются процессорами в режиме жесткой синхронизации. Сигналы управления используются для синхронизации команд и пересылок, а также для управления процессом вычислений и определения.

**Шина широковещательной рассылки** передает команды, данные и сигналы управления от контроллера к массиву процессоров.

**Шина результата** передает к контроллеру от процессоров результаты вычислений для выполнения команд перехода.

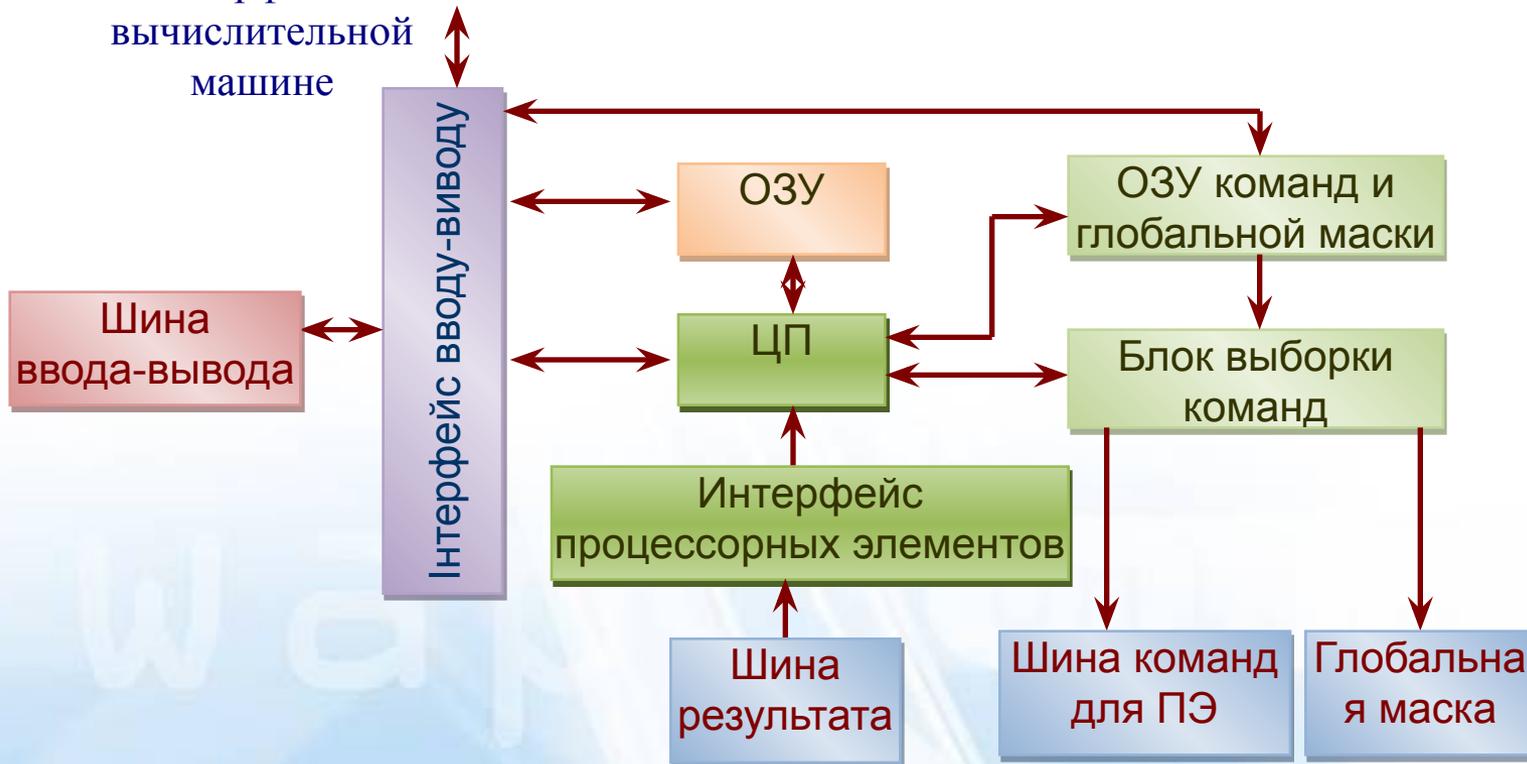
**Интерфейсная вычислительная машина** – это универсальный компьютер, который выполняет задачи загрузки программ и данных в контроллер массива процессоров. Этот компьютер соединяет матричную систему с внешним миром с помощью одного из сетевых интерфейсов, и используется для создания и отладки программ. Чаще всего этот компьютер работает под управлением операционной системы UNIX.

**Устройства ввода-вывода** – используются для ввода (вывода) данных в (из) массива процессоров.

# МАТРИЧНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

## Пример устройства контроллера массива процессоров системы PASM

К интерфейсной  
вычислительной  
машине



**Этап загрузки.** Программа интерфейсного компьютера через интерфейс ввода-вывода загружается в ОЗУ. Команды процессорных элементов и глобальная маска, сформированные на этапе компиляции, также через интерфейс ввода-вывода записываются в оперативное запоминающее устройство команд и глобальной маски.

**Этап выполнения.** Контроллер считывает или с ОЗУ скалярную команду, или с ОЗУ команд и маски множественной команду.

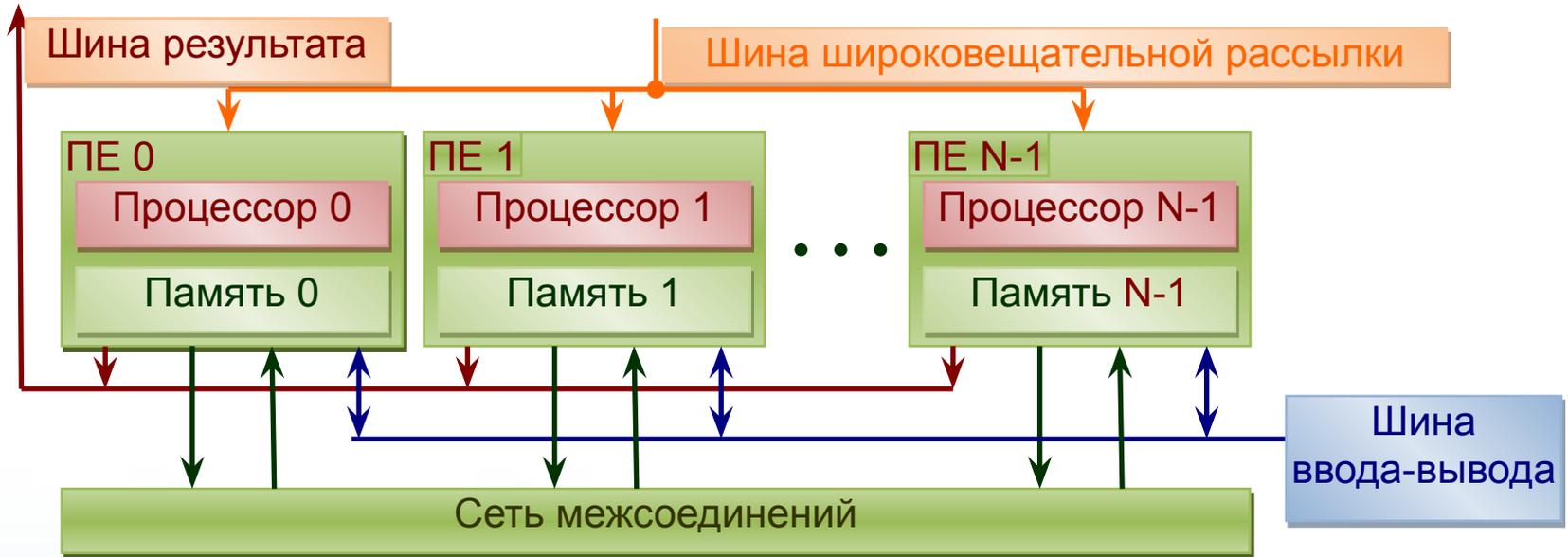
# МАТРИЧНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

*Скалярные команды* осуществляют операции над скалярными данными, хранящимися в ОЗУ на центральном процессоре, входящем в состав контроллера.

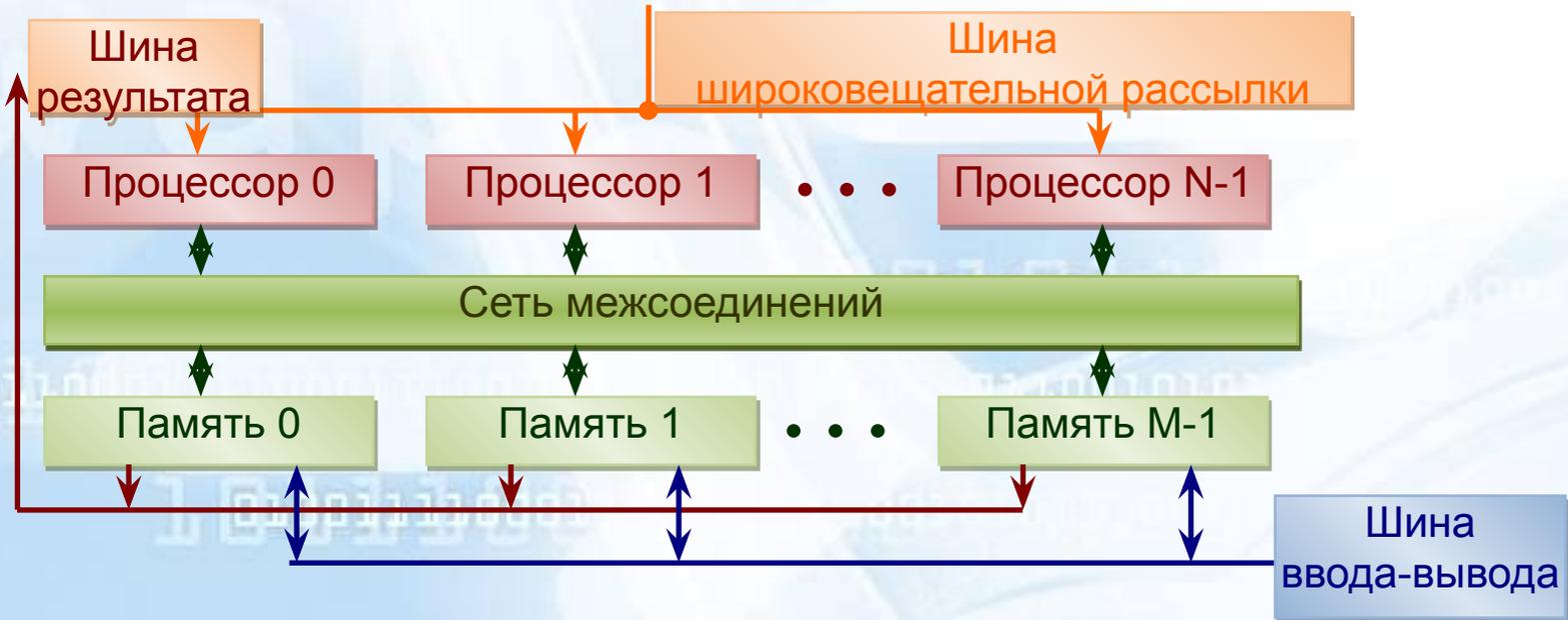
*Команды, оперирующие параллельными операндами*, хранящимися в каждом процессорном элементе, превращаются в блоке выборки команд в последовательность более простых конструкций - наноконанды. Наноконанды совместно с маской пересылаются через шину команд ПЭ на выполнение в массив процессоров. Например, команда сложения двух 32-разрядных слов системы MPP превращается в 32 наноконанды однокорядного сложения, которые каждым ПЭ обрабатываются последовательно.

Дальнейший порядок вычислений зависит от результатов и флагов предыдущих операций. Для обеспечения такого режима в матричных системах статусная информация, хранящаяся в процессорных элементах, собирается в одно слово и передается в контроллер массива процессоров для выработки решения о ветвлении в программе. Такая информация передается в контроллер по однокоравленной шине результата. Каждому процессу в нем отведена одна линия.

## Массив процессоров типа "процессорный элемент - процессорный элемент"



## Массив процессоров типа "процессор - память"



# МАТРИЧНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

В большинстве матричных SIMD-систем процессорными элементами выступают простые малоразрядные RISC-процессоры с локальной памятью ограниченной емкости.

Номер ПЭ – номер процессорного элемента, который является его адресом;

F – регистр флага разрешения маскировки. Его состояние определяет сигналы управления с контроллера массива ПЭ, или результаты операций в самом ПЭ, или и то и другое вместе.

## Структурная схема процессорного элемента



Данные могут поступать из своей локальной памяти и регистров, а также по шине широковещательной рассылки из контроллера массива. Кроме того, каждый ПЭ может получать данные от других ПЭ и направлять им свои результаты. Для этого используется сетевой интерфейс

Существенной характеристикой матричной системы способ синхронизации работы ПЭ. Так как все команды ПЭ получают и выполняют одновременно, то их работа жестко синхронизирована. Это особенно важно при выполнении операций обмена данными между ПЭ.

# МУЛЬТИПРОЦЕССОРНЫЕ СИСТЕМЫ КЛАССА MIMD

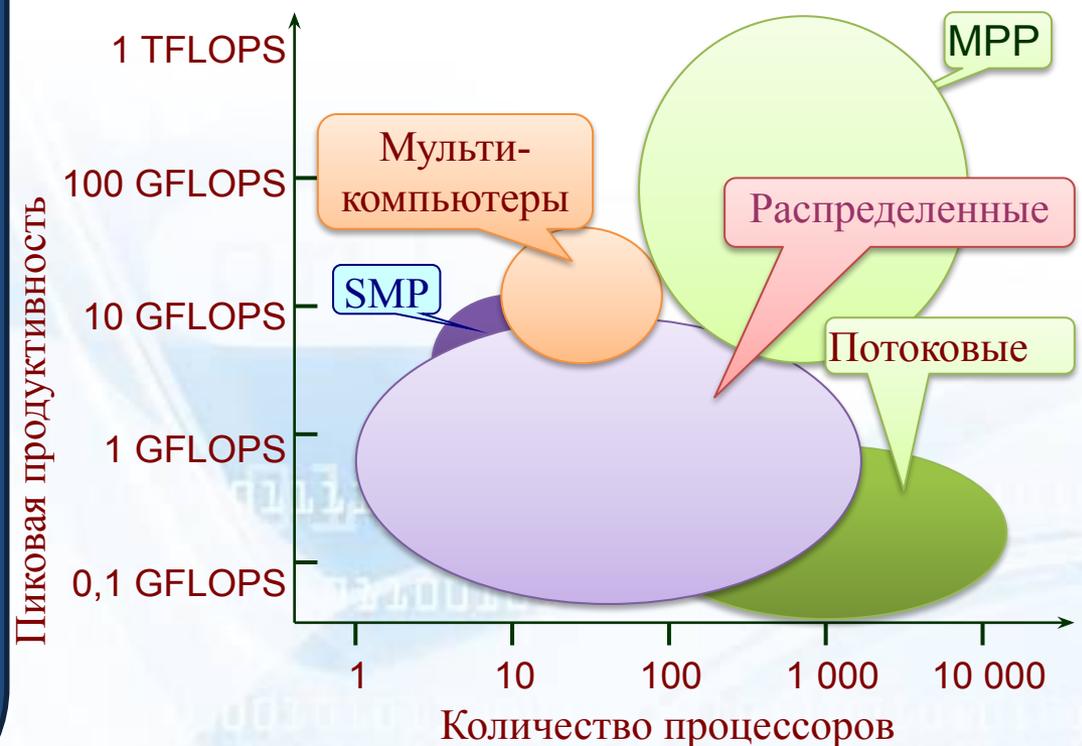
MIMD-системы имеют большую гибкость и могут работать как высокопроизводительные Системы одного пользователя так и многопрограммные системы, выполняющие параллельно множество задач параллельно.

В MIMD-системах каждый процессорный элемент (ПЭ) выполняет свою программу достаточно независимо друг от друга.

В зависимости от способа взаимодействия ПЭ между собой различают:

- Вычислительные системы с общей памятью - это сильно связанные системы, к которым относятся:
  - симметричные мультимикропроцессоры SMP (Symmetric Multiprocessor);
  - системы с неоднородным доступом к памяти NUMA (Non-Uniform Memory Access).
- Вычислительные системы с распределенной памятью - это слабо связанные системы, к которым относятся:
  - системы с массовым параллелизмом MPP (Massively Parallel Processing);
  - кластерные вычислительные системы.

## Производительность MIMD систем в зависимости от типа и количества процессоров



# СИММЕТРИЧНЫЕ МУЛЬТИПРОЦЕССОРНЫЕ СИСТЕМЫ - SMP

## SMP СИСТЕМА ОПРЕДЕЛЯЕТСЯ СЛЕДУЮЩИМИ СВОЙСТВАМИ:

- есть два или более процессора примерно равной производительности;
- процессоры совместно используют оперативную память и работают в едином виртуальном и физическом адресном пространстве.
- все процессоры связаны между собой по общей шине или по другой схеме так, что время доступа к памяти любого из них одинаково;
- все процессоры имеют доступ к устройствам ввода-вывода или через одни и те же каналы, либо через различные каналы, обеспечивающие доступ к одному и тому же внешнему устройству.
- все процессоры способны выполнять одни и те же функции - этим объясняется "симметричность" системы;
- все процессоры могут обслуживать внешние прерывания;
- вычислительная система управляется интегрированной операционной системой, которая организует и координирует взаимодействие между процессорами и программами на уровне задач, задач, файлов и элементов данных, что обеспечивает высокую связность между процессорами.

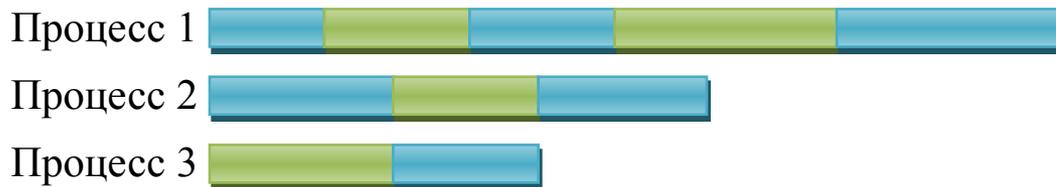
Хотя система симметрична, программное обеспечение вносит в нее некоторый перекос. На время загрузки системы один из процессоров получает статус ведущего (master), после загрузки системы все процессоры становятся равнозначными.

# СИММЕТРИЧНЫЕ МУЛЬТИПРОЦЕССОРНЫЕ СИСТЕМЫ - SMP

## Мультипрограммирование



## Мультипроцессорная обработка



По сравнению с однопроцессорными схемами SMP система имеет преимущества по показателям:

**Продуктивность** – значительный выигрыш получается тогда, когда задача поддается разделению на несколько параллельных частей.

**Готовность** – отказ одного процессора не ведет к потере работоспособности системы, а только снижает общую производительность.

**Расширяемость** – производительность системы может быть повышена включением в нее дополнительных процессоров.

**Масштабируемость** – путем варьирования количества процессоров можно создавать системы различной производительности и стоимости.

Все эти преимущества возможны только тогда, когда многопроцессорность поддерживает ОС, т.е. в ней есть средства поддержки параллелизма.

# СИММЕТРИЧНЫЕ МУЛЬТИПРОЦЕССОРНЫЕ СИСТЕМЫ - SMP

## Архитектура симметричной мультимикропроцессорной системы



В системе используется от 2 до 32 идентичных микропроцессоров.

Каждый процессор имеет локальную кэш-память, как правило, состоящую из кэша первого и второго уровня (L1 и L2). Когерентность кэш-памятей всех процессоров обеспечивается аппаратными средствами.

В некоторых системах проблема когерентности решается совместным использованием единой кэш-памяти. Однако, этот прием технически и экономически оправдан только при количестве процессоров не более 4. Использование общей кэш-памяти приводит к повышению стоимости и снижению производительности этого вида памяти

Коммуникационная система обеспечивает равный доступ всех процессоров к модулям основной памяти и устройствам ввода-вывода.

Процессоры общаются между собой через область общих данных основной памяти.

Основная память обычно строится по модульному принципу и организована так, что позволяет одновременное обращение к различным ее модулям.

В некоторых системах каждый процессор обладает также личной локальной основной памятью и каналами ввода-вывода.

# СИММЕТРИЧНЫЕ МУЛЬТИПРОЦЕССОРНЫЕ СИСТЕМЫ - SMP

С точки зрения взаимодействия процессоров с общими ресурсами - памятью и системой ввода-вывода - различают следующие архитектуры SMP систем:

- с общей шиной;
- с коммутатором типа кросс-бар;
- с многопортовой памятью
- с централизованным устройством управления.

## Организация SMP-системы с общей шиной



Физический интерфейс, логика адресации, арбитража и распределения времени в общей шине остаются такими же, как в однопроцессорной машине.

Общая шина позволяет легко расширять систему и отказ одного элемента системы не приводит к отказу всей системы.

Основной недостаток - скорость системы ограничена временем цикла шины. Это приводит к необходимости использования для каждого процессора локальной кэш-памяти большого размера. Возникает проблема когерентности, в свою очередь ограничивает количество процессоров.

# КЛАСТЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

**Кластер** – группа взаимно соединенных вычислительных систем (узлов), работающих совместно и составляющих единый вычислительный ресурс.

В качестве кластера может выступать как однопроцессорный компьютер, так и вычислительная система типа SMP или MPP. Важно лишь то, что каждый узел может функционировать самостоятельно и отдельно от кластера.

В плане архитектуры суть кластерных вычислений сводится к объединению нескольких узлов высокоскоростной сетью.

Классифицируют:

- Гомогенные кластеры – в качестве узлов используются одинаковые компьютеры;
  - Гетерогенные кластеры – в качестве узлов используются разнотипные компьютеры.
- По своей архитектуре кластерная вычислительная система является системой с слабыми связями.

## **Преимущества кластеризации:**

*Абсолютная масштабируемость* – возможность создания больших кластеров, по производительности будут превышать даже самые производительные вычислительные системы, так как кластер способен содержать десятки узлов, каждый из которых может быть мультипроцессорной системой.

*Наращиваемая масштабируемость* – возможность расширять возможности системы постепенно, порциями, путем подключения к кластеру новых узлов.

*Высокий коэффициент готовности* – отказ одного узла кластера не приводит к потере работоспособности всей системы в целом.

*Высокое соотношение цена / производительность* – можно создать кластер любой производительности путем соединения стандартных компьютеров. При этом его стоимость будет ниже, чем стоимость одиночной вычислительной системы с такой же мощностью.

# КЛАСТЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

На уровне аппаратного обеспечения кластер – совокупность независимых компьютеров или вычислительных систем, соединенных в обычных сетях. В соединении используются стандартные средства прямых связей между ПК, например, Ethernet, или более сложные высокоскоростные сети с пропускной способностью в сотни Мбайт/с.

Узлы кластера контролируют друг друга и обмениваются специфической, характерной для кластеров информацией. Контроль работоспособности осуществляется специальными сигналами (так называемыми heartbeat – сердцебиениями). Эти сигналы передаются узлами друг другу, чтобы подтвердить свое нормальное функционирование.

Обязательная часть кластера - специализированное программное обеспечение, обеспечивающее бесперебойную работу при отказе одного или нескольких узлов. ПО осуществляет перераспределение вычислительной нагрузки при отказе узла, а также обновления вычислений при сбое в узле. Кроме того, при наличии в кластере совместно используемых дисков кластерное ПО поддерживает единую файловую систему.

# КЛАСТЕРНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

## Конфигурация кластеров без совместно используемых дисков

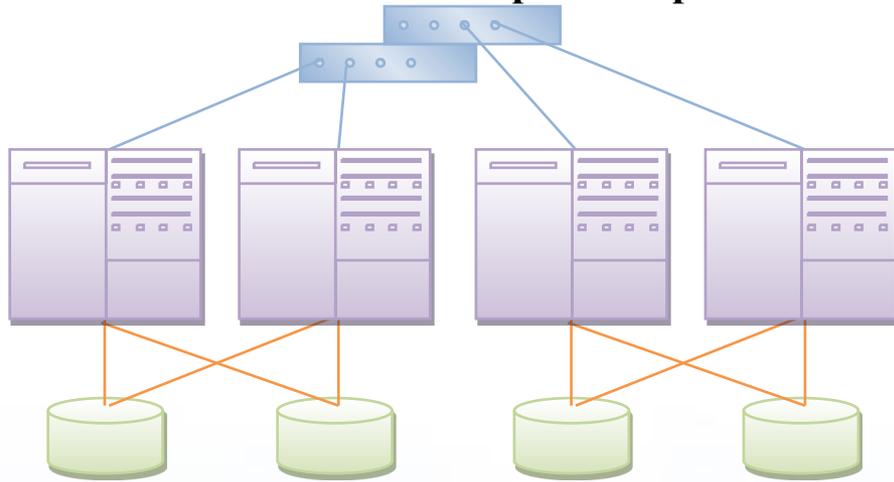


## Конфигурация кластеров с совместно используемыми дисками



# ТОПОЛОГИЯ КЛАСТЕРНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

## Топология кластерных пар



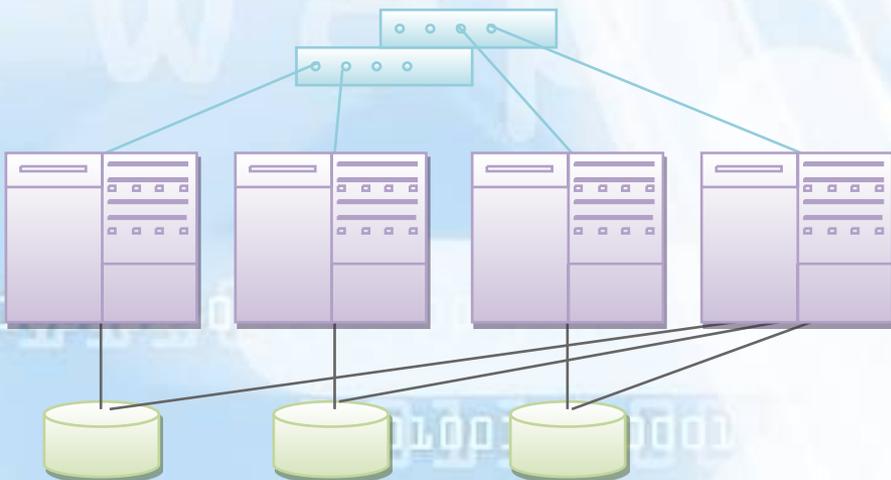
Используется при построении кластеров с двумя или четырьмя узлами.

Узлы группируются попарно. Дисковые массивы подключаются к обоим узлам пары, причем каждый узел имеет доступ ко всем дисковым массивам своей пары. Один из узлов является резервным для другого.

Четырехузловая кластерная пара является расширением двухузловой топологии. Обе кластерные пары с точки зрения администрирования и настройки рассматриваются как единое целое.

Эта топология используется для организации кластеров с высокой степенью готовности, но отказоустойчивость реализуется только в пределах пары.

## Топология N+1



Каждый дисковый массив подключается только к двум узлам. Дисковые массивы организованы по схеме RAID 1. Один сервер имеет соединение со всеми дисковыми массивами и выступает в качестве резервного для всех остальных узлов.

Используется для кластеров высокой готовности. Топология не позволяет реализовать глобальную отказоустойчивость системы.

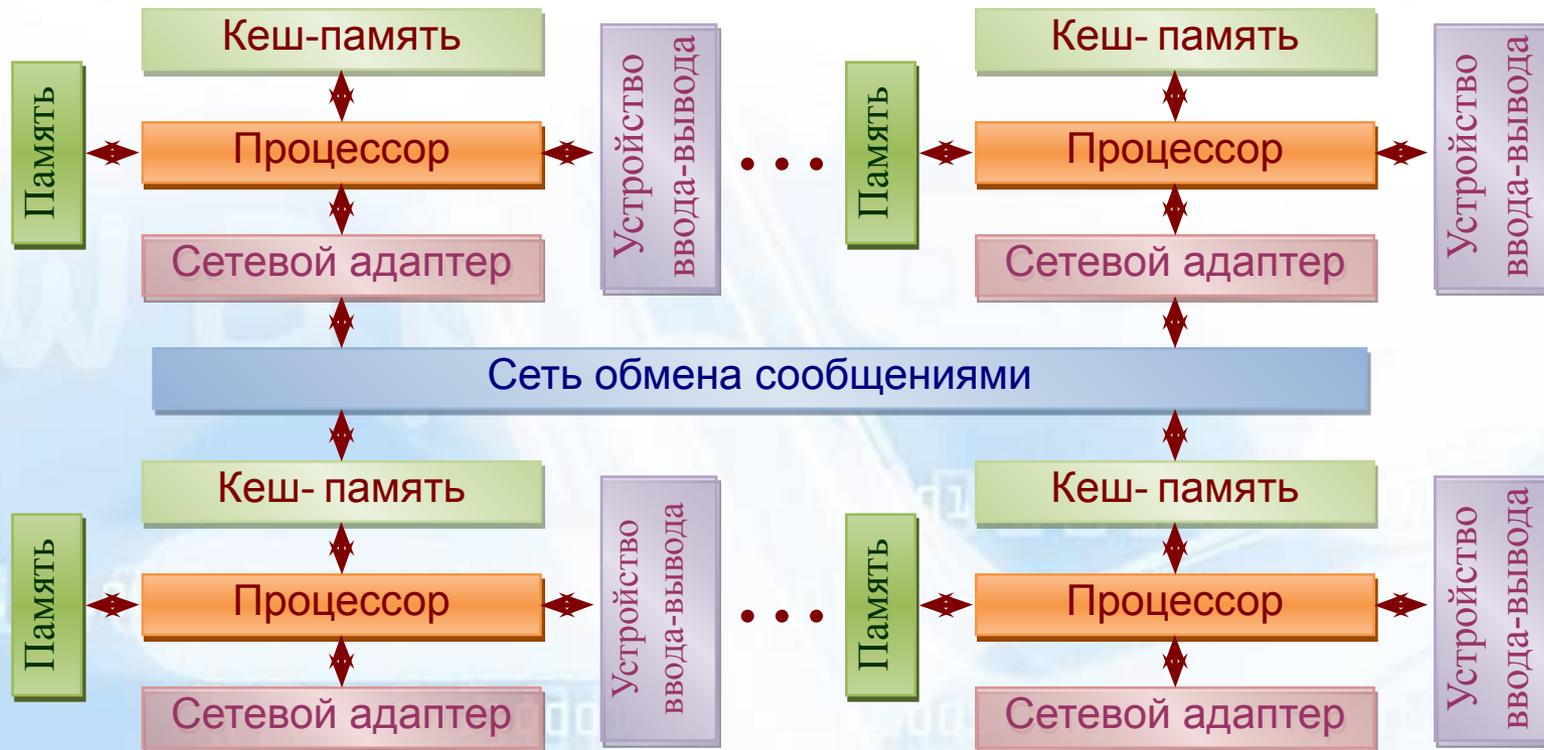
# СИСТЕМА С МАССОВОЙ ПАРАЛЛЕЛЬНОЙ ОБРАБОТКОЙ - MPP

MPP система - это система с числом процессоров  $n \geq 256$

Главные признаки класса MPP систем:

- стандартные микропроцессоры;
- физически распределенная память;
- сеть соединений с высокой пропускной способностью и малыми задержками;
- хорошая масштабируемость (до 1 000 процессоров);
- асинхронная MIMD система с пересылкой сообщений
- программа представляет собой множество процессов, имеющих отдельные адресные пространства.

## Архитектура вычислительной системы с массовой параллельной обработкой



# СИСТЕМА С МАССОВОЙ ПАРАЛЛЕЛЬНОЙ ОБРАБОТКОЙ - MPP

MPP система – это MIMD система с распределенной памятью, являющаяся дальнейшим распространением кластерных систем.

Основная цель создания MPP систем - построение вычислительных систем гигантской производительности при высокой степени масштабируемости, когда в зависимости от требований и экономических возможностей заказчика можно создавать вычислительные системы широкого диапазона по производительности и стоимости.

Характерная черта MPP систем - наличие единого устройства управления (процессора), распределяющего задачи между множеством подчиненных ему устройств. Эти устройства чаще всего являются одинаковыми, т.е. взаимозаменяемыми процессорами с памятью и устройствами ввода-вывода.

## СХЕМА ВЗАИМОДЕЙСТВИЯ УСТРОЙСТВ СИСТЕМЫ:

- центральное устройство управления формирует очередь заданий, каждому из которых назначается некоторый уровень приоритета;
- по мере освобождения подчиненных устройств им передаются задачи из очереди;
- подчиненные устройства оповещают центральный процессор о ходе выполнения задания, например, о завершении или о необходимости дополнительных ресурсов;
- у центрального устройства есть средства для контроля работы подчиненных процессоров, в том числе и для выявления нештатных ситуаций, прерывания хода выполнения задания в случае выявления более приоритетного задания.

Можно считать, что на центральном процессоре выполняется ядро операционной системы, а на подчиненных процессорах - выполняются приложения.

Подчиненность между процессорами может быть реализована как программно, так и аппаратно.

# ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ С НЕОДНОРОДНЫМ ДОСТУПОМ К ПАМЯТИ NUMA

В симметричных мультипроцессорных системах SMP есть предел числа процессоров, их составляющих. Это определяется тем, что для обеспечения когерентности локальной кэш-памяти процессора с основной памятью системы с ростом числа процессоров резко растет трафик общей шины. Поэтому количество процессоров ограничено числом 16.

Для снятия ограничений на количество процессоров используют кластерные системы. В них каждый узел имеет свою локальную оперативную память. Соседние процессоры не видят ячейки памяти соседних процессоров. Когерентность памяти в кластерных системах поддерживается на программном уровне. Это также значительно снижает общую производительность.

Согласно технологии NUMA, каждый процессор в системе обладает личной памятью, но с точки зрения процессоров имеется память, глобально адресуемая. Для этого каждая ячейка любой локальной памяти имеет уникальный системный адрес.

Когда процессор инициирует доступ к памяти и нужная ячейка отсутствует в его локальной кэш-памяти первого и второго уровней памяти, то процессор организует операцию выборки. Когда нужная ячейка есть в локальной основной памяти, выборка осуществляется с использованием локальной шины. Когда нужная ячейка хранится в удаленной основной памяти, то автоматически формируется запрос, посылаемый по сети соединений на нужную локальную шину и уже по ней до подключенного к данной локальной шине кэша или основной памяти. Все эти действия выполняются автоматически и прозрачно для процессора и его кэш-памяти.

Для адресации глобальной памяти в каждом узле есть справочник, где хранится информация о местоположении в системе каждой составляющей глобальной памяти, а также о состоянии своей кэш-памяти.

# ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ С НЕОДНОРОДНЫМ ДОСТУПОМ К ПАМЯТИ NUMA

## Архитектура вычислительной системы с массовой параллельной обработкой

