

# Многопроцессорные системы

Последовательные и параллельные  
архитектуры ЭВМ

# Многопроцессорная система

Опр. Многопроцессорная система (multiprocessor system) – вычислительная система, использующая для обработки данных более одного процессора.

Примеры: двухпроцессорные персональные компьютеры, мощные серверы со множеством процессоров, суперкомпьютеры.

# Преимущества

- Большие инженерные и исследовательские приложения, выполняемые на суперкомпьютерах, обеспечивают прирост производительности за счет параллельной обработки данных на нескольких процессорах
- Коммерческие и научные организации используют многопроцессорные системы для повышения производительности, предоставления задачам достаточных ресурсов и достижения высокой надежности

# ОС многопроцессорных ЭВМ должны дополнительно гарантировать, что:

- Все процессоры загружены работой
- Процессы равномерно распределены в системе
- Выполнение взаимосвязанных процессов синхронизировано
- Процессы работают с состоятельными копиями данных, хранящихся в общей памяти
- Обеспечивается взаимное исключение для выхода из тупиковых ситуаций

# Классификация многопроцессорных систем

- По структуре каналов обработки данных
- По схеме соединений процессоров
- По способу распределения ресурсов между процессорами
- По разделению функций операционных систем между процессорами
- По способу работы с памятью

# Классификация последовательных и параллельных архитектур ЭВМ

- Архитектуры компьютеров можно классифицировать по структуре каналов обработки данных, основываясь на понятии потоков данных и команд:
  - SISD
  - MISD
  - SIMD
  - MIMD

# Поток

Опр. Поток (stream) – битовая последовательность данных или инструкций, передаваемых процессору.

Замечание. Битовый поток (stream) не надо путать с потоком (thread) – логическим объектом, описывающим последовательность независимо выполняемых программных инструкций внутри процесса.

# SISD

Опр. Архитектура с одним потоком команд и одним потоком данных (SISD – single instruction stream, single data stream) – архитектура компьютеров, в которой один процессор последовательно выполняет инструкцию за инструкцией из потока команд над элементами данных из одного потока данных; к этой архитектуре относятся традиционные одноядерные однопроцессорные компьютеры.



# MISD

Опр. Архитектура с одним потоком данных и несколькими потоками команд (MISD – multiple instruction stream, single data stream) – архитектура компьютеров, содержащих несколько вычислительных элементов, выполняющих независимые параллельные потоки операций над одним потоком данных; к этой архитектуре относятся многоядерные процессоры.

# SIMD

Опр. Архитектура с одним потоком команд и несколькими потоками данных (SIMD – single instruction stream, multiple data stream) – архитектура компьютеров, состоящих из нескольких процессорных элементов, одновременно выполняющих одни и те же инструкции над различными элементами данных; к этой архитектуре относятся векторные и матричные процессоры.

# Векторный процессор

Опр. Векторный процессор (vector processor) – разновидность SIMD-компьютера, в котором используется один одноядерный процессор, одновременно выполняющий одну и ту же инструкцию над несколькими элементами данных. Высокая производительность достигается за счет использования длинного конвейера, состоящего из множества узлов процессора, и высокой тактовой частоты.

# Матричный процессор

Опр. Матричный процессор (array processors) – SIMD-система, состоящая из множества (до десятков тысяч) простых процессоров, каждый из которых выполняет такое же действие, как и остальные процессоры системы, но над своим элементом данных. Применяются, например, в научных расчетах для матричных преобразований.

# MIMD

Опр. Архитектура с множеством потоков команд и данных (MIMD – multiple instruction stream, multiple data stream) – архитектура компьютеров, состоящая из множества полноценных процессоров, каждый из которых выполняет свою последовательность инструкций над своим потоком данных; это полноценная параллельная архитектура настоящих многопроцессорных систем.

# Вопрос для самопроверки

- Верно ли, что несколько вычислительных потоков могут одновременно выполняться только в системах MISD и MIMD? (Да/Нет)

# Вопрос для самопроверки

- Верно ли, что несколько вычислительных потоков могут одновременно выполняться только в системах MISD и MIMD? (Да/Нет)
- Да. В системах SISD и SIMD несколько вычислительных потоков параллельно выполняться не могут. Однако, потоки выполняемые в MISD-системах, работают с одними и теми же данными, и поэтому не являются независимыми. Параллелизм в полной мере свойственен только MIMD<sup>5</sup>.

# Вопрос для самопроверки

- Присутствует ли параллелизм в SISD-системах? (Да/Нет)



# Вопрос для самопроверки

- Присутствует ли параллелизм в SISD-системах? (Да/Нет)
- Да. Для повышения производительности процессоров в SISD-системах используется конвейерная обработка данных и суперскалярная архитектура. Это позволяет одновременно выполнять инструкции нескольких команд.

# Вопрос для самопроверки

- Могут ли одновременно в одной системе использоваться MIMD и MISD архитектуры? (Да/Нет)

# Вопрос для самопроверки

- Могут ли одновременно в одной системе использоваться MIMD и MISD архитектуры? (Да/Нет)
- Да. Например, если вычислительная система состоит из множества многоядерных процессоров.

# Вопрос для самопроверки

- Архитектура ЭВМ Джона Фон Неймана является MIMD архитектурой?

# Вопрос для самопроверки

- Архитектура ЭВМ Джона Фон Неймана является MIMD архитектурой?
- Нет. Архитектура ЭВМ Джона Фон Неймана является последовательной SISD архитектурой с одним потоком данных и одним потоком команд.

# Многопроцессорные системы

## Схемы соединений процессоров

# Схема соединений

Опр. Схема соединений (interconnection scheme) – определяет способ соединения компонентов многопроцессорной системы, например, процессоров и модулей памяти. Схемы соединений состоят из узлов и связей.

# Узел

Опр. Узел (node) – компонент системы, например, процессор, модуль памяти или коммутатор, подключенный к сети. Во многих системах в узле могут содержаться один или несколько процессоров, связанный с ними кэш, модуль памяти и коммутатор. Иногда группу соединенных друг с другом узлов тоже называют узлом (суперузлом).



# Коммутатор

Опр. Коммутатор (switch) – узел, пересылающий сообщения между узлами-компонентами.

# СВЯЗЬ

Опр. Связь (link) – соединение между двумя узлами многопроцессорной системы, по которому между узлами передается информация.

# Параметры схем соединений

- Степень узла
- Ширина сечения
- Диаметр сети
- Стоимость схемы соединений

# Степень узла

Опр. Степень узла (degree of node) – количество узлов, непосредственно соединенных с данным узлом.  
Разработчики пытаются свести к минимуму степень всех узлов, чтобы уменьшить стоимость и сложность коммуникационных интерфейсов узлов.

# Ширина сечения

Опр. Ширина сечения (bisection width) – минимальное количество связей, которые нужно перерезать, чтобы разделить сеть на две несвязанные части. Чем больше ширина сечения сети, тем больше ее отказоустойчивость.

# Диаметр сети

Опр. Диаметр сети (network diameter) - количество связей в кратчайшем пути между двумя самыми удаленными узлами. Чем меньше диаметр сети, тем выше ее производительность и меньше задержки при обмене информацией по каналам.

# Стоимость схемы соединений

Опр. Стоимость схемы соединений (cost of interconnection scheme) – общее количество соединений в схеме. Проектировщики стараются минимизировать стоимость схемы соединений.

# Типы схем соединений процессоров

- Общая шина
- Переключающая матрица
- Двумерная ячеистая сеть
- Гиперкуб
- Многоуровневая сеть

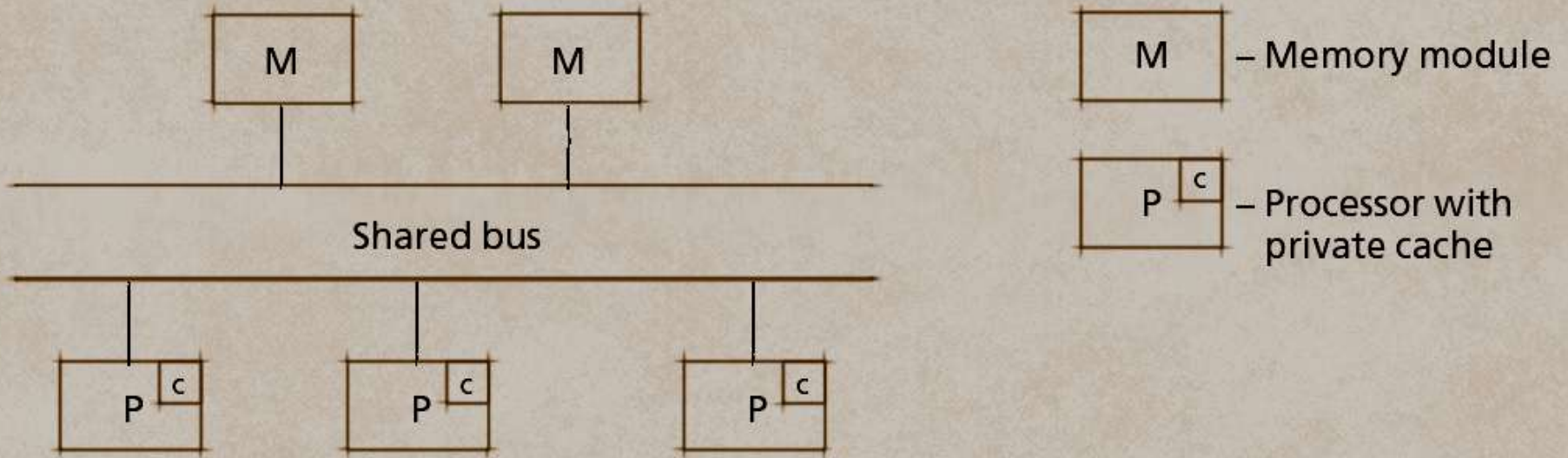


# Общая шина

Опр. Общая шина (shared bus) – схема соединений компонентов в многопроцессорных системах, в которой все компоненты, включая процессоры и модули памяти, соединены одной шиной. В каждый момент времени от одного узла ко всем остальным узлам может быть отправлена только одна единица данных.

Пр. Dual-processor Intel Pentium

# Многопроцессорная система с общей шиной



# Общая шина

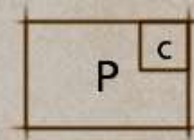
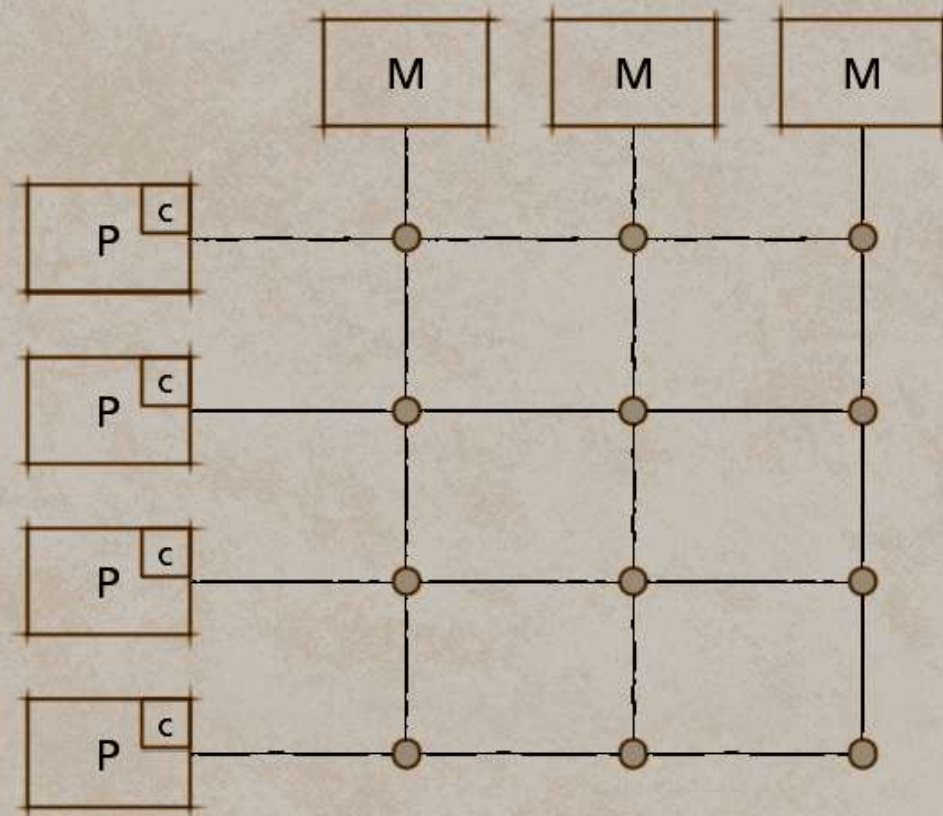
- Простой и дешевый способ связи между несколькими процессорами (на практике обычно не более 32)
- Применяется для небольших систем и для создания суперузлов в крупных системах
- Много производительных процессоров могут быстро загрузить общую шину
- Если к шине сразу обратится несколько процессоров возникнет конфликт, чтобы их сократить у каждого процессора есть свой кэш

# Переключающая матрица

Опр. Переключающая матрица (crossbar-switch matrix) – схема соединения процессоров, обеспечивающая отдельный путь от каждого узла-отправителя к каждому узлу-получателю. Все узлы одновременно могут отправить по одной единице данных, но в любой момент времени один узел может принимать лишь один поток данных.

Пр. Sun UltraSPARC-III

# Многопроцессорная система с переключательной матрицей



– Processor with private cache



– Memory module

○ – Crossbar switch

# Переключающая матрица

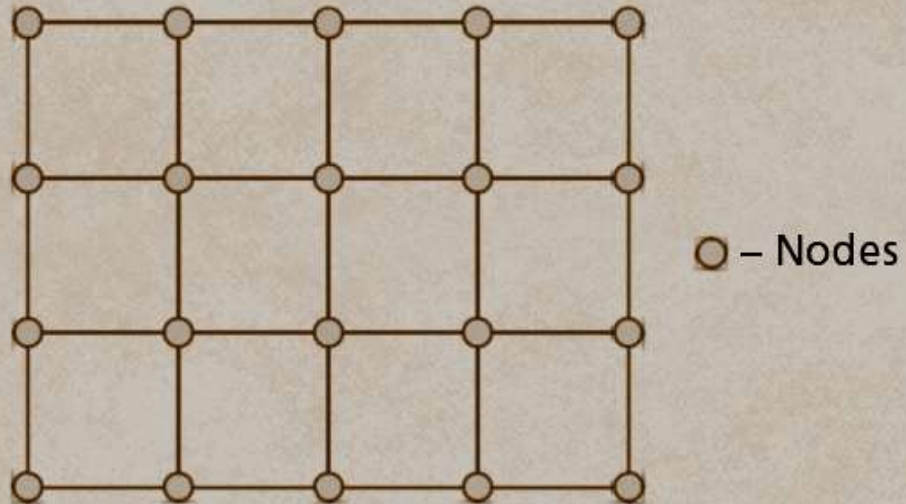
- Диаметр сети практически равен 1, значит система обладает максимальной производительностью
- Чтобы разделить надвое такую сеть нужно перерезать половину связей, значит ширина сети равна  $(n \times m)/2$ , где  $n$  – число процессоров, а  $m$  – число модулей памяти; поэтому система обладает высокой отказоустойчивостью
- Цена переключающих матриц велика и растет пропорционально  $n \times m$
- Аппаратное обеспечение дешевеет, и матрицы все чаще используются в больших системах 38

# Двумерная ячеистая сеть

Опр. Двумерная ячеистая сеть (2-D mesh network) – схема соединения процессоров, в которой узлы организованы в прямоугольник размерности  $n \times m$ . Каждый узел состоит из нескольких процессоров и модуля памяти и непосредственно соединен с соседними узлами слева, справа, снизу и сверху от него.

Пр. Intel Paragon

# Двумерная ячеистая сеть





# Двумерная ячеистая сеть

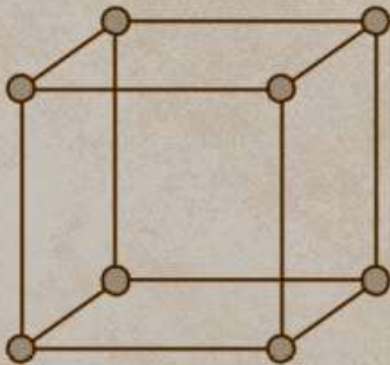
- Степень каждого узла равна от 2 до 4, это относительно дешевая архитектура
- Ширина сечения равна  $\min(m,n)$ , значит отказоустойчивость невысока
- Диаметр сети очень велик, значит и производительность невысока
- Эта архитектура используется при построении систем, в которых обмен данными преимущественно осуществляется между соседними узлами

# Гиперкуб

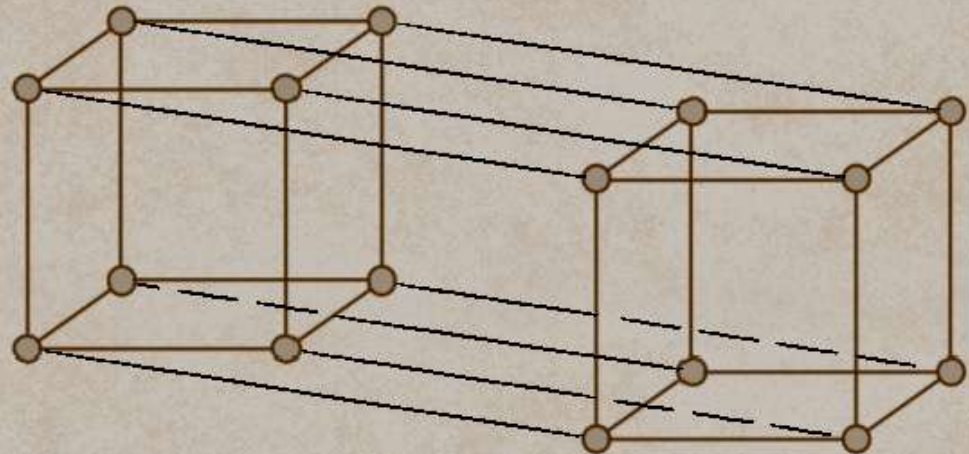
Опр. Гиперкуб (hypercube) – схема соединения процессоров, содержащая  $2^n$  узлов. Каждый узел в такой схеме связан с  $n$  соседними узлами. Например, двумерный гиперкуб – это двумерная ячеистая сеть размерности  $2 \times 2$ .

Пр. Системы потоковой обработки видеоданных nCUBE используют гиперкубы до 13 измерений (8912 узлов)<sup>43</sup>

# 3 и 4-мерные гиперкубы



(a)



● - Nodes

(b)

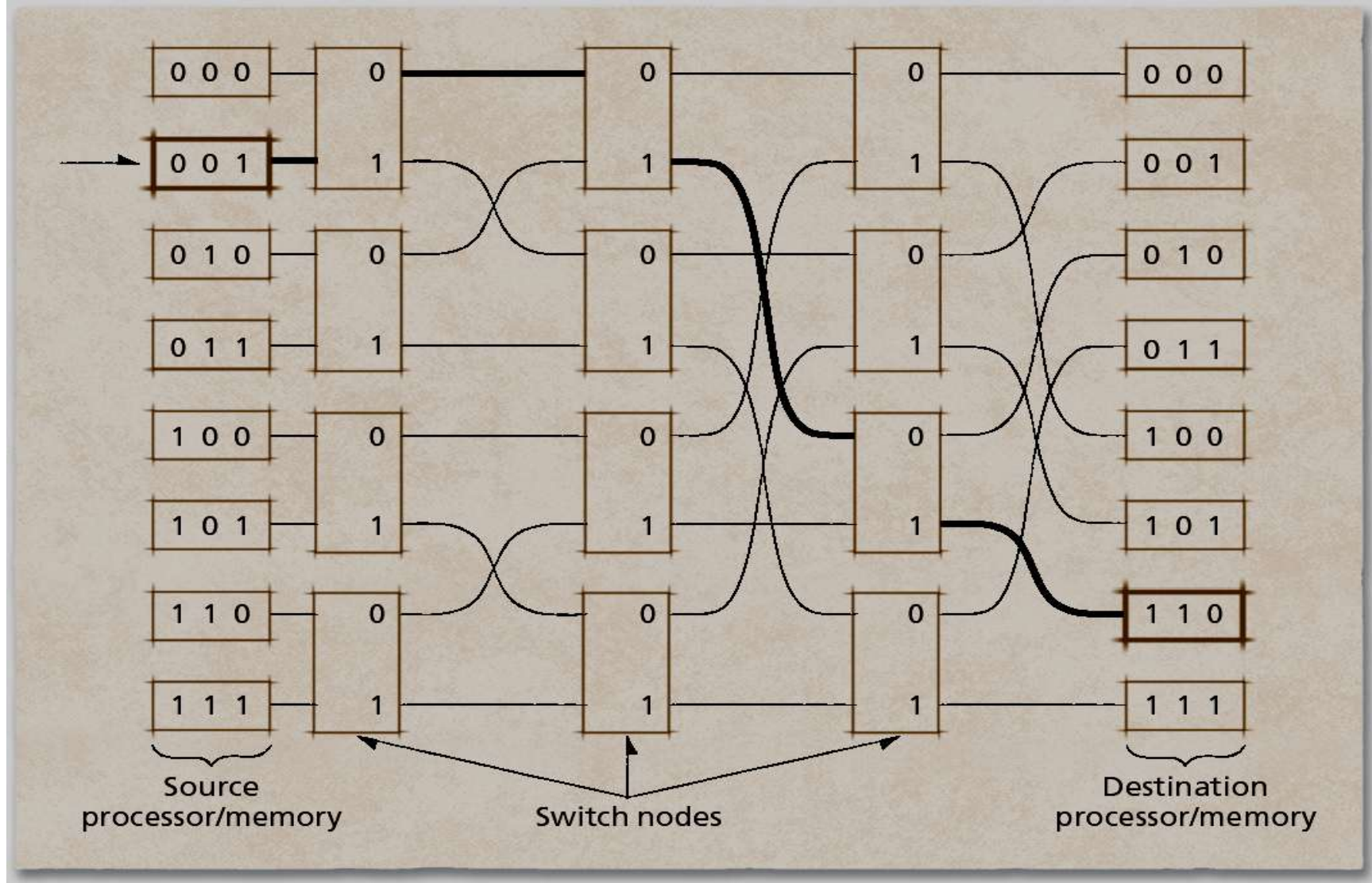
# Гиперкуб

- Более производительная и отказоустойчивая, но и более дорогая схема, чем двумерная ячеистая сеть
- Удобная схема для небольшого количества процессоров, более дешевая, чем переключаящая матрица

# Многоуровневая сеть

Опр. Многоуровневая сеть (multistage network) – схема соединения процессоров, в которой используются специальные узлы-коммутаторы для связи между отдельными процессорными узлами, снабженными локальной памятью.

Пр. Суперкомпьютеры IBM серии SP



**Путь между процессорами в многоуровневой сети.  
 Направление следующего шага определяется значением соответствующего разряда номера процессора-приемника, начиная с младшего (правого) разряда.**

# Многоуровневая сеть

- Попытка компромисса между ценой и производительностью
- Каждый процессор может связаться с каждым, не передавая данные между промежуточными процессорами
- Дешевле, чем переключаящая матрица но ее производительность и отказоустойчивость ниже

# Вопрос для самопроверки

- Верно ли что диаметр сети, основанный на переключающей матрице самый маленький? (Да/Нет)



# Вопрос для самопроверки

- Верно ли что диаметр сети, основанный на переключающей матрице самый маленький? (Да/Нет)
- Нет. Диаметр схемы с общей шиной тоже равен 1.

# Вопрос для самопроверки

- Верно ли, что ширина сечения схемы с общей шиной минимальна? (Да/Нет)

# Вопрос для самопроверки

- Верно ли, что ширина сечения схемы с общей шиной минимальна? (Да/Нет)
- Да. Достаточно перерезать только одну связь (шину), чтобы сеть перестала функционировать. Все остальные схемы гораздо более отказоустойчивы.

# Вопрос для самопроверки

- Верно ли, что стоимость переключающей матрицы самая высокая? (Да/Нет)

# Вопрос для самопроверки

- Верно ли, что стоимость переключающей матрицы самая высокая? (Да/Нет)
- Да. Только эта схема обеспечивает связь каждого узла с каждым по отдельному каналу связи. Поэтому количество связей, т.е. стоимость этой схемы, максимальна.

# Вопрос для самопроверки

- Верно ли, что четырехмерный гиперкуб – это два трехмерных гиперкуба? (Да/Нет)

# Вопрос для самопроверки

- Верно ли, что четырехмерный гиперкуб – это два трехмерных гиперкуба? (Да/Нет)
- Да. Четырехмерный гиперкуб – это два трехмерных гиперкуба, в которых связаны между собой соответствующие узлы. Это же справедливо и для гиперкубов других размерностей.

# Многопроцессорные системы

Тесносвязанные и  
слабосвязанные системы



# Классификация по способу распределения ресурсов между процессорами

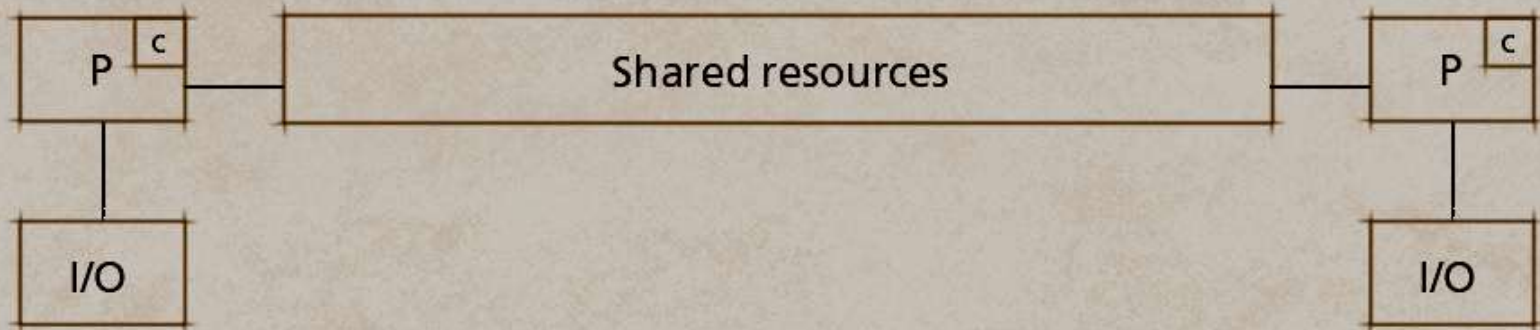
- Тесносвязанные системы
- Слабосвязанные системы

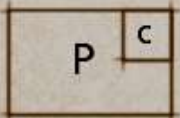
# Тесносвязанная система

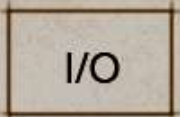
Опр. Тесносвязанная система (tightly coupled system) – система, в которой каждому процессору доступно большинство ее ресурсов. В таких системах часто используется общая шина и процессоры обмениваются данными через общую физическую память.

Пр. Dual-processor Intel Pentium

# Тесно связанная система



 – Processor with private cache

 – I/O subsystem

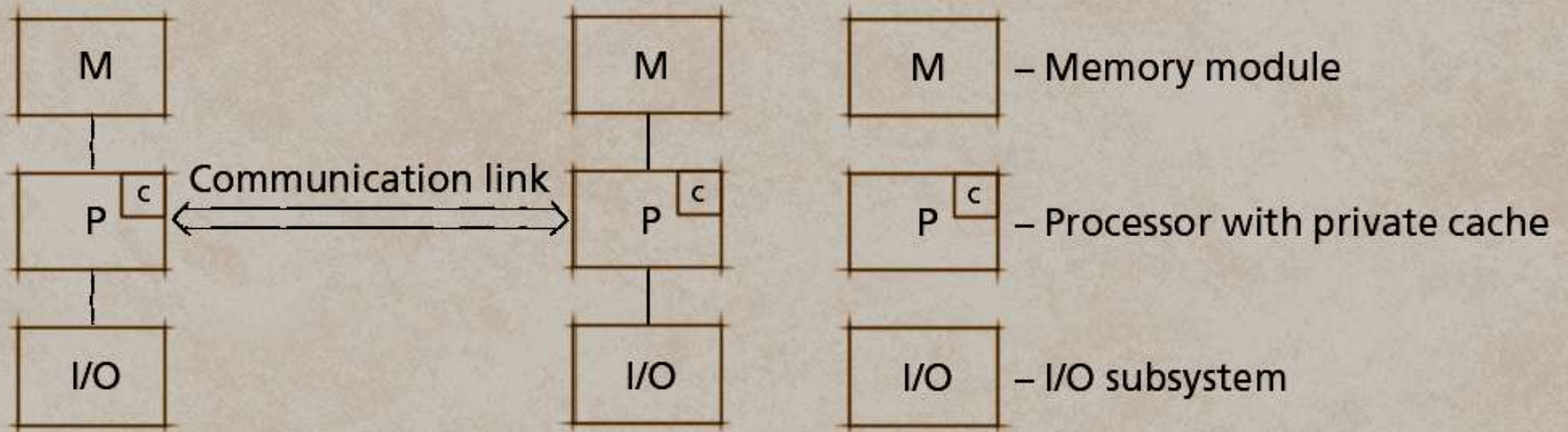
# Слабосвязанная система

Опр. Слабосвязанная система (loosely coupled system) – система, в которой каждый процессор не может напрямую обратиться к любому ресурсу.

Компоненты обычно соединяются специальными каналами связи, взаимодействие процессоров, у каждого из которых есть своя собственная память, осуществляется передачей сообщений.

Пр. Суперкомпьютер Earth Simulator

# Слабосвязанная система



# Сравнительные характеристики

- Слабосвязанные системы более расширяемые и отказоустойчивые, но менее производительны, чем тесносвязанные
- Создание системного программного обеспечения систем для тесносвязанных систем проще, чем для слабосвязанных

# Вопрос для самопроверки

- Тесносвязанные системы эффективны для создания ЭВМ с большим числом процессоров? (Да/Нет)

# Вопрос для самопроверки

- Тесносвязанные системы эффективны для создания ЭВМ с большим числом процессоров? (Да/Нет)
- Нет. Большие системы обычно создаются слабосвязанными, чтобы уменьшить вероятность конфликтов при доступе к разделяемым ресурсам и вероятность выхода системы из строя при отказе одного компонента.



# Вопрос для самопроверки

- Тесносвязанные системы редко используются на практике? (Да/Нет)

# Вопрос для самопроверки

- Тесносвязанные системы редко используются на практике? (Да/Нет)
- Нет. Маленькие системы обычно создаются тесносвязанными, поскольку в них невысока вероятность конфликтов при доступе к разделяемым ресурсам. Кроме того, многие крупные системы состоят их групп тесносвязанных компонентов, соединенных каналами связи в слабосвязанную систему.

# Многопроцессорные системы

Многопроцессорные операционные  
системы

# Схемы организации операционных систем для многопроцессорных ЭВМ

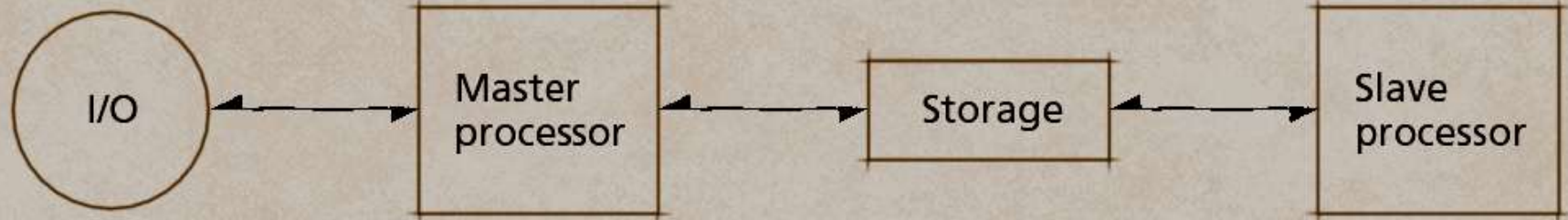
- Схема ведущий / ведомый
- Схема с отдельными ядрами
- Схема с симметричным обслуживанием всех процессоров

# Схема ведущий / ведомый

Опр. Многопроцессорная система схемы ведущий / ведомый (master / slave multiprocessor organization) – система, в которой на одном процессоре (ведущем, главном) выполняется операционная система, а остальные процессоры (ведомые, подчиненные) заняты выполнением только пользовательских задач.

Пр. Системы nCUBE

# Многопроцессорная система схемы ведущий / ведомый



**Storage – устройство хранения**

# Особенности схемы ведущий / ведомый

- Асимметричность аппаратуры – когда процессу, выполняющемуся на ведомом процессоре требуется услуга операционной системы, он генерирует прерывание и ожидает, пока ведущий процессор его обработает
- Вводом / выводом управляет только ведущий процессор, если выполняемые на ведомых процессорах задачи требуют интенсивного ввода / вывода они будут часто обращаться к ведущему процессору
- При отказе ведущего процессора система становится полностью неработоспособной

# Вопрос для самопроверки

- Системы схемы ведущий / ведомый хорошо масштабируются до больших размеров? (Да/Нет)



# Вопрос для самопроверки

- Системы схемы ведущий / ведомый хорошо масштабируются до больших размеров? (Да/Нет)
- Нет. Операционная система выполняется только на одном процессоре. Между пользовательскими процессами, выполняющимися на других процессорах, возникает конкуренция при обращении к функциям ввода / вывода.

# Вопрос для самопроверки

- Схема ведущий / ведомый хорошо подходит для интенсивных вычислений?  
(Да/Нет)

# Вопрос для самопроверки

- Схема ведущий / ведомый хорошо подходит для интенсивных вычислений? (Да/Нет)
- Да. Системы такой схемы лучше всего подходят для сред, в которых большая часть задач связана с интенсивными вычислениями. Эти задачи будут выполняться ведомыми процессорами без частых обращений к ведущему.

# Схема с разделенными ядрами

Опр. Многопроцессорная система с разделенными ядрами (separate kernels multiprocessor organization) – система, в которой на каждом процессоре выполняется отдельная операционная система, но процессы могут обращаться к некоторым глобальным данным (например, к списку всех работающих в системе процессов).

Пр. Отказоустойчивые системы Tandem <sup>76</sup>

# Особенности схемы с разделенными ядрами

- Процесс, запущенный на определенном процессоре, выполняется на нем до своего завершения
- Каждый процессор работает с ресурсами, доступными только ему, например, – файлами и устройствами ввода / вывода
- Полный отказ системы практически не возможен, система будет работать даже на одном последнем процессоре

# Вопрос для самопроверки

- Схема с разделенными ядрами более отказоустойчива, чем схема ведущий / ведомый? (Да/Нет)

# Вопрос для самопроверки

- Схема с разделенными ядрами более отказоустойчива, чем схема ведущий / ведомый? (Да/Нет)
- Да. Схема с разделенными ядрами является слабосвязанной. У каждого процессора есть свои ресурсы, и он не взаимодействует с другими процессорами при выполнении своих задач. Если любой из процессоров выходит из строя остальные продолжают работать. При отказе ведущего процессора схема ведущий / ведомый полностью отказывает.

# Симметричная схема

Опр. Многопроцессорная система симметричной схемы (symmetrical multiprocessor organization) – система, в которой операционная система может выполняться одновременно на всех процессорах. Операционная система распоряжается набором одинаковых процессоров, каждый из которых может обращаться к любому устройству ввода / вывода.

Пр. BBN Butterfly



# Особенности симметричной схемы

- Самые мощные, но сложные системы
- Необходимо гарантировать взаимное исключение при работе с общими структурами данных
- Высокая конкуренция процессов при доступе к системным ресурсам
- При отказе одного из процессоров производительность снижается плавно, так как любой процесс может выполняться на любом из процессоров

# Вопрос для самопроверки

- Удвоение количества процессоров в многопроцессорной системе симметричной схемы увеличит ее производительность в два раза? (Да/Нет)

# Вопрос для самопроверки

- Удвоение количества процессоров в многопроцессорной системе симметричной схемы увеличит ее производительность в два раза? (Да/Нет)
- Нет. Добавление в систему новых процессоров приводит к росту конкуренции при обращении к ресурсам и увеличивает накладные расходы операционной системы, поэтому часть производительности добавленных процессоров теряется.

# Вопрос для самопроверки

- Симметричная схема лучше масштабируется, чем схема ведущий / ведомый? (Да/Нет)

# Вопрос для самопроверки

- Симметричная схема лучше масштабируется, чем схема ведущий / ведомый? (Да/Нет)
- Да. Симметричная схема лучше масштабируется, чем схема ведущий / ведомый поскольку операционная система может выполняться на всех процессорах.

# Вопрос для самопроверки

- Симметричная схема обеспечивает лучшую кооперацию между процессорами, чем схема с разделенными ядрами? (Да/Нет)

# Вопрос для самопроверки

- Симметричная схема обеспечивает лучшую кооперацию между процессорами, чем схема с разделенными ядрами? (Да/Нет)
- Да. Симметричная схема позволяет обеспечить в полной мере взаимодействие процессов и эффективнее выполнять процессы в параллельном режиме.

# Многопроцессорные системы

Архитектуры доступа к памяти



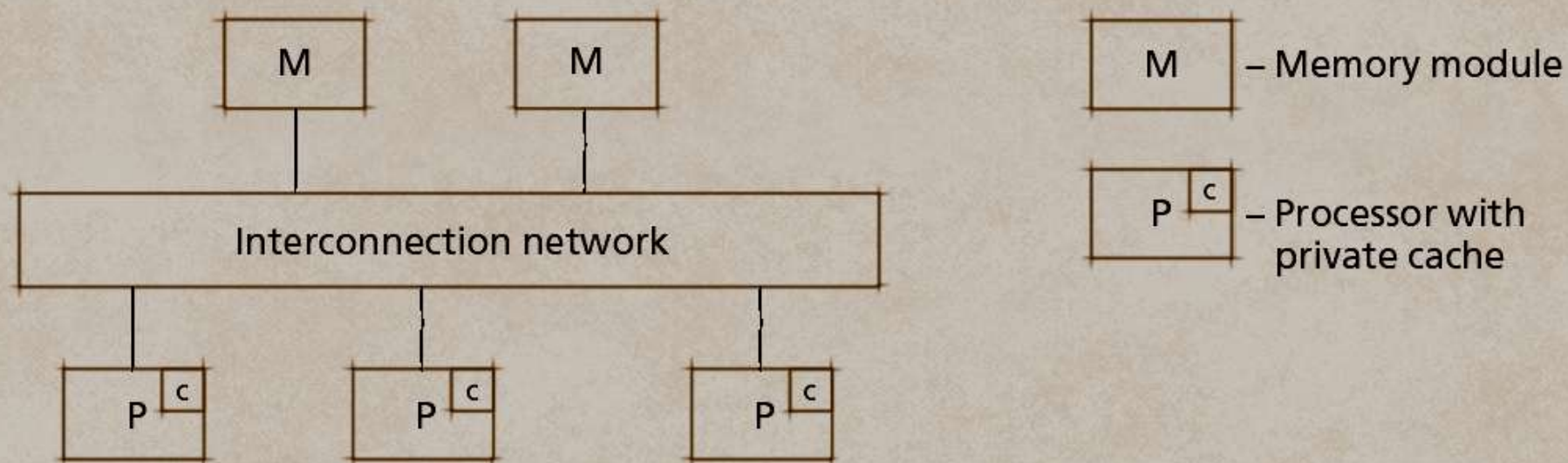
# Архитектуры доступа к памяти в многопроцессорных системах

- Архитектура однородного доступа к памяти (UMA)
- Архитектура неоднородного доступа к памяти (NUMA)
- Архитектура без доступа к удаленной памяти (NORMA)

# UMA

Опр. Однородный доступ к памяти (uniform memory access, UMA) – архитектура многопроцессорных систем, позволяющая всем процессорам обращаться к общей памяти; в общем случае время доступа к памяти в такой системе постоянное, независимо от того, какой процессор запрашивает данные, за исключением случаев, когда запрашиваемые данные находятся в кэше процессора.

# Многопроцессорная система архитектуры UMA



# Многопроцессорная система архитектуры UMA

- Также имеет название симметричной многопроцессорной системы, поскольку в ней любой процесс может выполняться на любом процессоре и все ресурсы (включая память, устройства ввода / вывода и процессы) доступны для всех процессоров
- Обычно используется в небольших системах (обычно – от двух до восьми процессоров)
- В качестве сети соединений используется общая шина или переключаящая матрица

# Вопрос для самопроверки

- Архитектура UMA не очень хорошо масштабируется? (Да/Нет)

# Вопрос для самопроверки

- Архитектура UMA не очень хорошо масштабируется? (Да/Нет)
- Да. В архитектуре UMA легко наступает насыщение шины, если одновременно большое число процессоров обращается к памяти, а переключательные матрицы весьма дороги даже при использовании в системах весьма скромного размера.

# Вопрос для самопроверки

- Ячеистые сети и гиперкубы подходят для соединений процессоров в системах архитектуры UMA? (Да/Нет)

# Вопрос для самопроверки

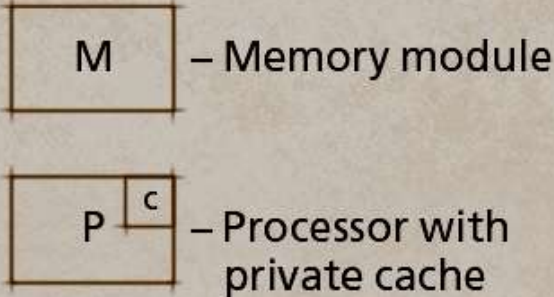
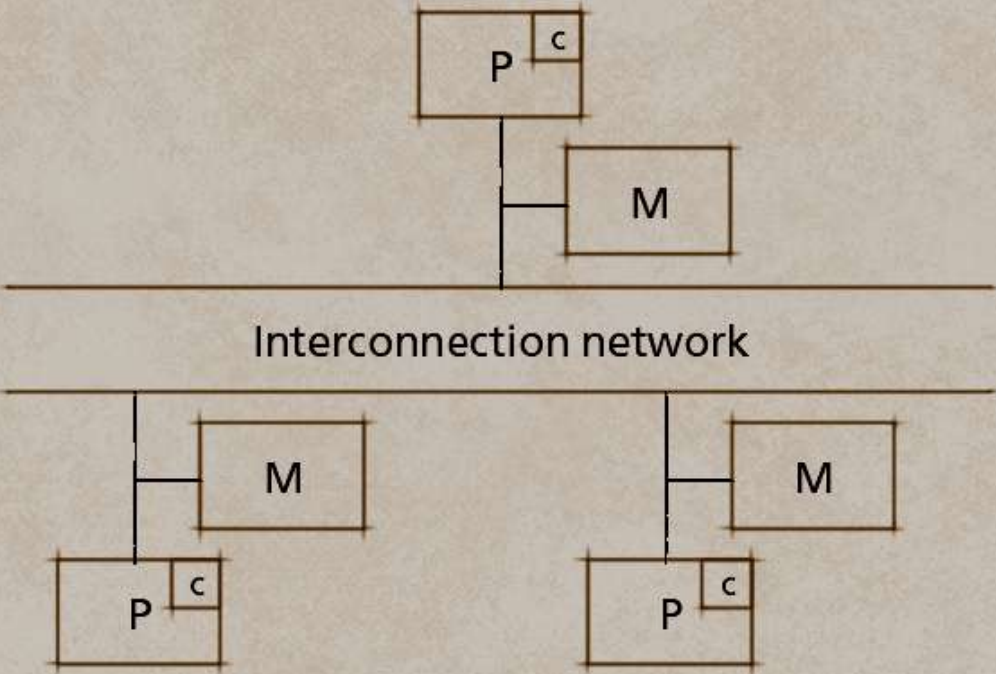
- Ячеистые сети и гиперкубы подходят для соединений процессоров в системах архитектуры UMA? (Да/Нет)
- Нет. В ячеистых сетях и гиперкубах процессоры и память размещаются в каждом узле, и обращения к локальной памяти осуществляются быстрее, чем к удаленной. Поэтому доступ к памяти в этих схемах соединений не будет однородным.



# NUMA

Опр. Неоднородный доступ к памяти (non-uniform memory access, NUMA) – архитектура многопроцессорных систем, в которой каждый узел состоит из одного или нескольких процессоров с кэшем и модуля памяти. Доступ к памяти, находящейся в том же узле, выполняется намного быстрее, чем к памяти, расположенной в других узлах.

# Многопроцессорная система архитектуры NUMA



# Многопроцессорная система архитектуры NUMA

- Более масштабируема, чем архитектура UMA, так как благодаря наличию локальной памяти процессы реже обращаются к шине
- Обычно системы, основанные на архитектуре NUMA, состоят из большего числа процессоров, чем системы архитектуры UMA
- В качестве сети соединений может быть использована любая из рассмотренных ранее схем соединений процессоров

# Вопрос для самопроверки

- Системы NUMA эффективнее, чем системы UMA при небольшом количестве процессоров? (Да/Нет)

# Вопрос для самопроверки

- Системы NUMA эффективнее, чем системы UMA при небольшом количестве процессоров? (Да/Нет)
- Нет. Поскольку системы UMA в отличие от систем NUMA обеспечивают одинаково быстрый доступ ко всей памяти для всех процессоров.

# Вопрос для самопроверки

- Возрастет ли производительность системы NUMA, если операционная система обеспечит размещение процесса и памяти, с которой этот процесс работает в одном узле? (Да/Нет)

# Вопрос для самопроверки

- Возрастет ли производительность системы NUMA, если операционная система обеспечит размещение процесса и памяти, с которой этот процесс работает в одном узле? (Да/Нет)
- Да. Если данные, используемые процессом, находятся в удаленной памяти, то производительность системы снижается.

# NORMA

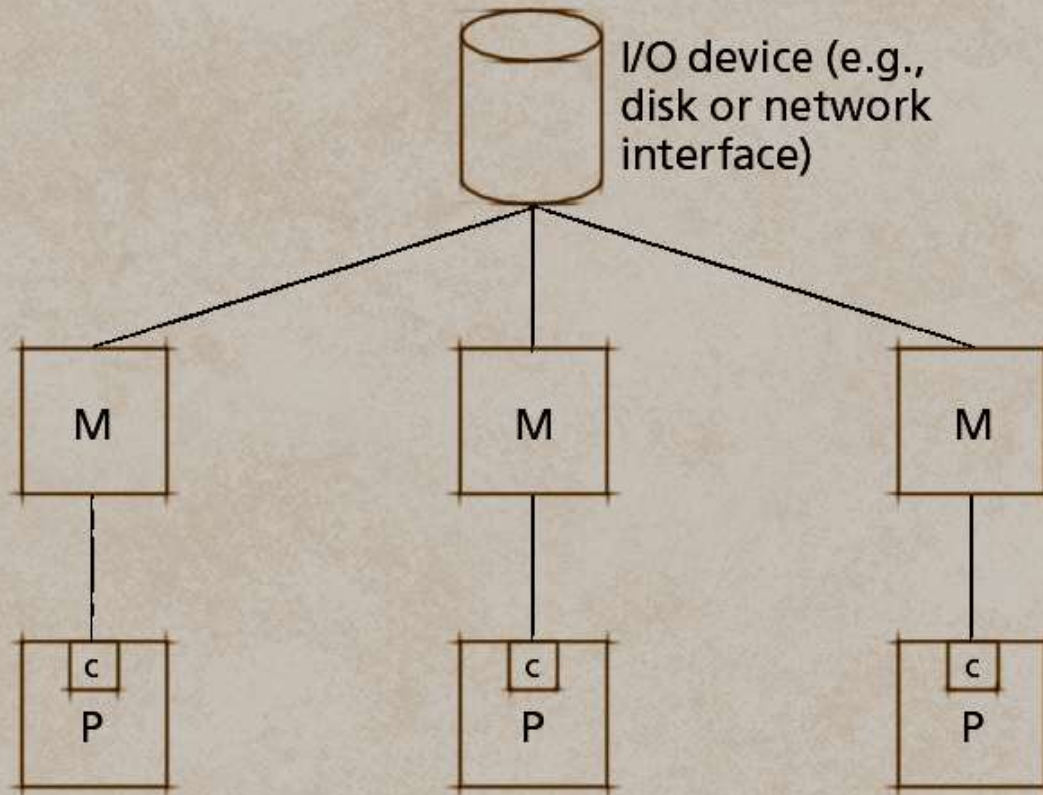
Опр. Архитектура без доступа к удаленной памяти (no remote memory access, NORMA) – архитектура многопроцессорных систем, в которой нет общей памяти, есть только локальная память компьютеров-узлов, в таких системах для обмена данными используется сетевой интерфейс, либо общая виртуальная память.



# Общая виртуальная память

Опр. Общая виртуальная память (shared virtual memory, SVM) – расширение концепции виртуальной памяти, используемое в многопроцессорных системах; SVM создает иллюзию присутствия в системе общей физической памяти.

# Многопроцессорная система архитектуры NORMA



# Многопроцессорная система архитектуры NORMA

- Часто использование SVM не эффективно, так как процессорам приходится передавать целые страницы данных по медленным каналам связи
- Узлы в системах NORMA, не использующих SVM, для обмена данными применяют протоколы передачи сообщений
- Системы NORMA представляют собой распределенные системы, управляемые одной операционной системой
- Поисковый портал Google, службы которого выполняются с помощью 15000 серверов начального уровня – пример системы NORMA

# Вопрос для самопроверки

- Архитектура NORMA плохо подходит для создания кластеров рабочих станций?  
(Да/Нет)

# Вопрос для самопроверки

- Архитектура NORMA плохо подходит для создания кластеров рабочих станций? (Да/Нет)
- Нет. Пользователи в кластере рабочих станций обычно не обращаются к памяти, используемой другими пользователями. Обычно они используют общие ресурсы, например, файловую систему и процессоры. Архитектура NORMA обеспечивает доступ к этим ресурсам. 109

# Вопрос для самопроверки

- Есть ли среды, где архитектура NORMA будет бесполезна? (Да/Нет)

# Вопрос для самопроверки

- Есть ли среды, где архитектура NORMA будет бесполезна? (Да/Нет)
- Да. Системы NORMA бесполезны в средах, в которых нужен доступ к общей памяти, особенно в системах с одним пользователем или с небольшим количеством процессоров. В таких средах лучше использовать архитектуры UMA или NUMA.