

# Многопроцессорность

Материал из Википедии — свободной энциклопедии

**Многопроцессорность** (**Мультипроцессорность**, **Многопроцессорная обработка**, англ. *Multiprocessing*) — использование пары или большего количества физических процессоров в одной компьютерной системе.

Устройство называется **многопроцессорным**, если в его составе используется два или более физических процессора.

Операционная система или программа называется многопроцессорной, если способна распределять задачи между процессорами (см. многопоточность).

Существует много вариантов данного понятия, и определение многопроцессорности может меняться в зависимости от контекста, главным образом в зависимости от того, как определены процессоры (много ядер в одном кристалле, множество чипов в одном корпусе, множество корпусов в одном системном модуле и так далее).

Многопроцессорностью иногда называют выполнение множественных параллельных программных процессов в системе в противоположность выполнению одного процесса в любой момент времени. Однако термины многозадачность или мультипрограммирование являются более подходящими для описания этого понятия, которое осуществлено главным образом в программном обеспечении, тогда как многопроцессорная обработка является более соответствующей, чтобы описать использование множественных аппаратных процессоров. Система не может быть и многопроцессорной и мультипрограммированной, только одной из двух, или ни той и ни другой.

## Содержание

### Типы

- Процессорная симметричность
- Потоки команд и данных
- Соединения процессоров

### Программные реализации

- Многопроцессорная обработка с SISD
- Многопроцессорная обработка SIMD
- Многопроцессорная обработка MISD
- Многопроцессорная обработка MIMD

### Ссылки

## Типы

### Процессорная симметричность

В многопроцессорной системе все центральные процессоры могут быть равными, или некоторые могут быть зарезервированы для особых целей. Комбинация конструктивных соображений программного обеспечения аппаратной и операционной системы определяет симметрию (или отсутствие её) в данной системе. Например, аппаратные или программные соображения могут потребовать, чтобы только один центральный процессор отвечал на все аппаратные прерывания, тогда как вся другая работа в системе может быть распределена одинаково среди процессоров; или выполнение кода привилегированного режима может быть ограничено только одним процессором (или определенным процессором, или только один процессор за один раз), тогда как код непривилегированного режима может быть выполнен на любой комбинации процессоров. Часто многопроцессорные системы проще проектировать, если введены такие ограничения, но они имеют тенденцию быть менее эффективными чем системы, в которых используются все центральные процессоры.

Системы, которые обрабатывают все центральные процессоры одинаково, называют системами с симметричной многопроцессорной обработкой (SMP). В системах, где все центральные процессоры не равны, системные ресурсы могут быть разделены многими способами, в том числе асимметричной многопроцессорной обработкой (ASMP), многопроцессорной обработкой с неоднородным доступом к памяти (NUMA) и кластеризованной многопроцессорной обработкой (qq.v.).

## Потоки команд и данных

В многопроцессорной обработке процессоры могут использоваться для выполнения одной последовательности команд во множественных контекстах (единственная машинная команда, множественные данные или SIMD, часто используемый в векторной обработке), множественные последовательности команд в единственном контексте («множественный поток команд, одиночный поток данных» или Архитектура MISD, используемая для избыточности в отказоустойчивых системах и иногда применяемая, чтобы описать конвейерные процессоры или гиперпоточность), или множественные последовательности команд во множественных контекстах («множественный поток команд, множественные поток данных» или MIMD).

## Соединения процессоров

*Многопроцессорные системы с сильной связью (англ. Tightly-coupled multiprocessor systems)* содержат несколько процессоров, которые подключены на шинном уровне. Эти процессоры могут иметь доступ к центральной разделяемой памяти (SMP или UMA), или могут участвовать в иерархии памяти и с локальной и с разделяемой памятью (NUMA). IBM p690 Regatta является примером мощной системы SMP. Процессоры Intel Xeon доминировали над многопроцессорным рынком для деловых PC и были единственной x86-опцией до выпуска линейки процессоров AMD Opteron в 2004 году. Обе линейки процессоров имели свой собственный встроенный кэш, но по-разному обеспечивают доступ к разделяемой памяти: процессоры Xeon через общий канал, а процессоры Opteron через независимые магистрали к системной оперативной памяти.

Многопроцессорные (многоядерные) чипы, включает больше одного процессора, помещённого в однокристалльную схему, и могут считаться самой предельной формой многопроцессорной обработки с сильной связью. Мэйнфреймовые системы со множеством процессоров — часто являются системами с сильной связью.

*Многопроцессорные системы с гибкой связью (англ. Loosely-coupled multiprocessor systems)*, часто называемые кластерами, основаны на множественных автономных одиночных или двойных компьютерах, связанных через высокоскоростную систему связи (например, Gigabit Ethernet).

Кластер Беовульфа под управлением Linux — пример гибко связанной системы.

Системы с сильной связью работают лучше и физически они меньше, чем гибко связанные системы, но исторически потребовали больших начальных инвестиций и могут быстро амортизовываться; узлы в гибко связанной системе — обычно недорогие компьютеры и могут быть использованы как независимые машины после удаления из кластера.

Системы с сильной связью имеют тенденцию быть намного больше энергоэффективными, чем кластеры. Значительная энергоэффективность достигается за счёт того, что компоненты в таких системах заранее проектируются для работы в конкретной системе, в то время как гибко связанные системы зачастую используют компоненты спроектированные для работы в более широком классе систем.

## Программные реализации

---

### Многопроцессорная обработка с SISD

В компьютере с одиночным потоком команд и одиночным потоком данных один процессор последовательно обрабатывает команды; каждая машинная команда обрабатывает один элемент данных. Пример — фон-неймановская архитектура.

### Многопроцессорная обработка SIMD

В компьютере с одиночным потоком команд и множественным потоком данных один процессор обрабатывает поток команд, каждая из которых может выполнить параллельные вычисления на множестве данных.

Многопроцессорная обработка SIMD хорошо подходит для параллельной или векторной обработки, в которой большой набор данных может быть разделён на части, которые обрабатываются идентичными, но независимыми операциями. Одиночный поток команд направляет операцию модулей мультипрограммирования для выполнения однотипных манипуляций одновременно на потенциально большом количестве данных.

Для определённых типов вычислительных приложений этот тип архитектуры может дать значительное увеличение производительности с точки зрения затраченного времени. Однако, недостаток этой архитектуры состоит в том, что большая часть системы начинает простаивать при выполнении программ или системных задач, которые не могут быть разделены на модули (подзадачи), которые могут быть обработаны параллельно.

Кроме того, программы должны быть тщательно и специально написаны, чтобы иметь возможность максимально задействовать возможности архитектуры. Часто применяются специальные оптимизирующие компиляторы, спроектированные чтобы создать код специально для использования в этой среде. Некоторые компиляторы в этой категории обеспечивают специальные конструкции или расширения, чтобы позволить программистам непосредственно определять операции, которые будут выполнены параллельно (например, операторы DO FOR ALL в Фортране, используемого на ILLIAC IV, который был многопроцессорным суперкомпьютером с SIMD-архитектурой).

Многопроцессорная обработка SIMD находит широкое применение в некоторых областях, таких как компьютерное моделирование, но малополезна в универсальном настольном компьютере и бизнес-задачах.

## Многопроцессорная обработка MISD

Многопроцессорная обработка с множественным потоком команд и одиночным потоком данных предлагает главным образом преимущество избыточности, так как модули мультипрограммирования выполняют одни задачи на одних данных, уменьшая возможности неправильных результатов, если один из модулей выходит из строя. Архитектура MISD позволяет сравнивать результаты вычислений в целях обнаружения отказов. Кроме избыточности и отказоустойчивости у этого типа многопроцессорной обработки немного преимуществ. К тому же он весьма дорог. Он не увеличивает производительность.

## Многопроцессорная обработка MIMD

Архитектура многопроцессорной обработки MIMD является подходящей для большого числа разнообразных задач, в которых реализовано полностью независимое и параллельное выполнение команд, касающихся различных наборов данных. По этой причине и потому что это просто осуществить, MIMD преобладает в многопроцессорной обработке.


Обработка разделена на несколько потоков, каждый с собственным аппаратным состоянием процессора, в рамках единственного определённого программным обеспечением процесса или в пределах множественных процессов. Поскольку система имеет несколько потоков, ожидающих выполнения (системные или пользовательские потоки), эта архитектура эффективно использует аппаратные ресурсы.

В MIMD могут возникнуть проблемы взаимной блокировки и состязания за обладание ресурсами, так как потоки, пытаясь получить доступ к ресурсам, могут столкнуться непредсказуемым способом. MIMD требует специального кодирования в операционной системе компьютера, но не требует изменений в прикладных программах, кроме случаев когда программы сами используют множественные потоки (MIMD прозрачен для однопоточных программ под управлением большинства операционных систем, если программы сами не отказываются от управления со стороны ОС). И системное и пользовательское программное обеспечение, возможно, должны использовать программные конструкции, такие как семафоры, чтобы препятствовать тому, чтобы один поток вмешался в другой, в случае если они содержат ссылку на одни и те же данные. Такое действие увеличивает сложность кода, снижает производительность и значительно увеличивают количество необходимого тестирования, хотя обычно не настолько чтобы свести на нет преимущества многопроцессорной обработки.

Подобные конфликты могут возникнуть на аппаратном уровне между процессорами, и должен обычно решаться аппаратными средствами, или с комбинацией программного обеспечения и оборудования.

## Ссылки

---

- Современные операционные системы (<https://books.google.ru/books?id=jv6AckoevagC&pg=PA610&dq=%D0%9C%D1%83%D0%BB%D1%8C%D1%82%D0%B8%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80>) (пер. с англ.), Эндрю С. Таненбаум. Глава 8.1 Мультипроцессоры (англ. (<http://www.cs.vu.nl/~ast/books/mos2/sample-8.pdf>) 
  - multiprocessing (<http://www.britannica.com/EBchecked/topic/397249/multiprocessing>) - Encyclopædia Britannica, Inc.
-

---

**Эта страница в последний раз была отредактирована 11 мая 2020 в 03:31.**

Текст доступен по лицензии Creative Commons Attribution-ShareAlike; в отдельных случаях могут действовать дополнительные условия.

Wikipedia® — зарегистрированный товарный знак некоммерческой организации Wikimedia Foundation, Inc.