

Классификация вычислительных систем

Одним из наиболее распространенных способов классификации ЭВМ является систематика Флинна (Flynn), в рамках которой основное внимание при анализе архитектуры вычислительных систем уделяется способам взаимодействия последовательностей (потоков) выполняемых команд и обрабатываемых данных. В результате такого подхода различают следующие основные типы систем

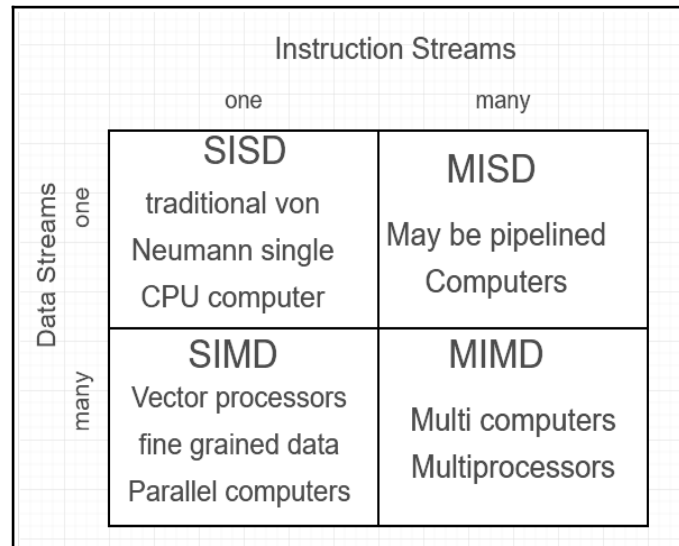


Рис. Таксономия (Классификация) Флинна

- **SISD** (Single Instruction, Single Data) – системы, в которых существует одиночный поток команд и одиночный поток данных; Это традиционный компьютер фон-Неймановской архитектуры с одним процессором, который выполняет последовательно одну инструкцию за другой, работая с одним потоком данных. В данном классе не используется параллелизм ни данных, ни инструкций, и, следовательно, SISD-машина не является параллельной. К этому классу также принято относить конвейерные, суперскалярные и VLIW-процессоры.
- **SIMD** (Single Instruction, Multiple Data) – системы с одиночным потоком команд и множественным потоком данных; Типичными представителями *SIMD* являются векторные процессоры, обычные современные процессоры, когда работают в режиме выполнения команд векторных расширений, а также особый подвид с большим количеством процессоров — матричные процессоры.
- **MISD** (Multiple Instruction, Single Data) – системы, в которых существует множественный поток команд и одиночный поток данных; примеров конкретных ЭВМ. К классу *MISD* ряд исследователей относит конвейерные ЭВМ, однако это не нашло окончательного признания. Также, возможно считать MISD системами, системы с горячим резервированием. Помимо этого, к архитектуре MISD некоторые относят систолические массивы процессоров.
- **MIMD** (Multiple Instruction, Multiple Data) – системы с множественным потоком команд и множественным потоком данных. Класс *MIMD* включает в себя многопроцессорные системы, где процессоры обрабатывают множественные потоки данных. Сюда принято относить традиционные мультипроцессорные машины, многоядерные и многопоточные процессоры, а также компьютерные кластеры.

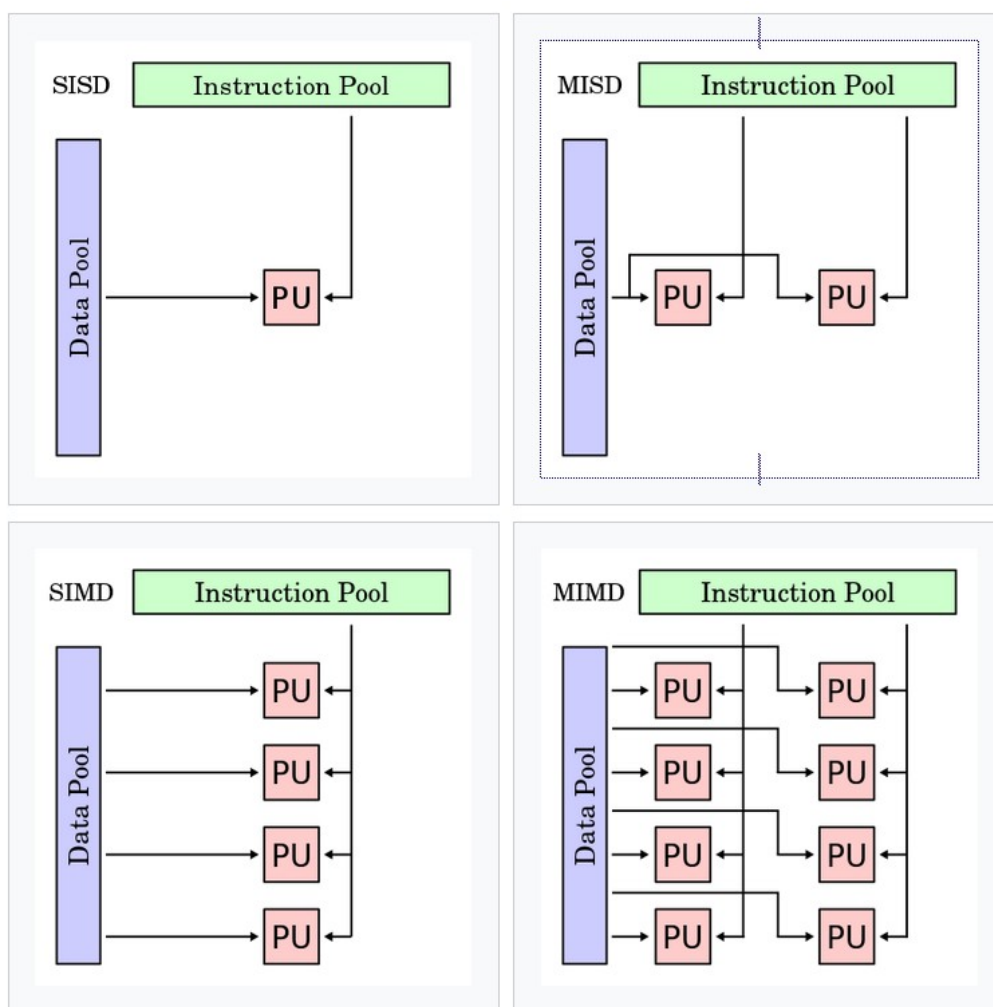


Рис. Архитектура вычислительных систем

Какие существующие системы относятся к SIMD?

Короткие SIMD инструкции (64 или 128 бит) стали появляться в процессорах общего назначения в 1990-х годах.

Модель *SIMD* впервые была реализована в процессорах Intel Pentium с поддержкой технологии MMX. Повышение производительности основывается на *векторизации* вычислений, для этого несколько скалярных операндов упаковываются в одном векторном регистре. Размер вектора, а следовательно и количество операций, выполняемых одной инструкцией, определяется соотношением размерностей регистра и операнда. Современные процессоры Intel с поддержкой технологии AVX512, используют 512-битные регистры (*ZMM0-ZMM31*). Таким образом, за одну векторную инструкцию может быть обработано, например, восемь 64-битных чисел двойной точности, либо шестнадцать 32-битных чисел одинарной точности.

Каждое обновление SIMD-расширений от Intel (MMX, SSE, AVX256, AVX512) несло не только количественные, но и качественные различия, включая организацию регистровых файлов и принципов кодирования инструкций, что приводило к необходимости редизайна компиляторов, приложений, фреймворков, системных процедур ОС. Несмотря на совместимость расширенной системы команд с базовым x86-набором, для получения преимуществ от увеличения разрядности векторных регистров требуется основательная модификация и ре-оптимизация ПО.

Сильные стороны SIMD-архитектуры очевидны при обработке больших массивов информации по одинаковому алгоритму. Минусом можно считать накладные расходы (затраты процессорных тактов) на упаковку скалярных данных в векторный регистр (*gathering*) и обратную распаковку (*scattering*), добавляемые в случае необходимости сопряжения скалярных и векторных фрагментов выполняемого кода.

Практически все современные процессоры общего назначения (CPU) умеют работать в режиме SIMD.

В GPU (*graphics processing unit*) также используется технология похожая на SIMD, ее часто называют *SIMT*. Процедуры обработки информации объединяются в группы. В каждой такой группе, называемой *warp*, один поток инструкций управляет обработкой нескольких (в типовом случае 32) потоков данных. Иерархия вычислительных модулей GPU оптимизирована для ситуации, при которой все данные в пределах warp требуется обработать по общему алгоритму, это значит, что для управления группой из 32 потоков данных, можно использовать единственный поток инструкций, интерпретируемый одним управляющим блоком на группу из 32 арифметико-логических устройств.

Какие существующие системы относятся к MIMD?

Класс MIMD включает в себя большое множество разнородных систем. Как результат, многими исследователями предпринимались неоднократные попытки детализации систематики Флинна.

Так, например, для класса MIMD предложена практически общепризнанная структурная схема в которой дальнейшее разделение типов многопроцессорных систем основывается на используемых способах организации оперативной памяти в этих системах (см. рис. 1.1). Данный подход позволяет различать два важных типа многопроцессорных систем – multiprocessors (системы с общей разделяемой памятью или shared memory MIMD) и multicomputers (системы с распределенной памятью (раздельной памятью) или distributed memory MIMD).

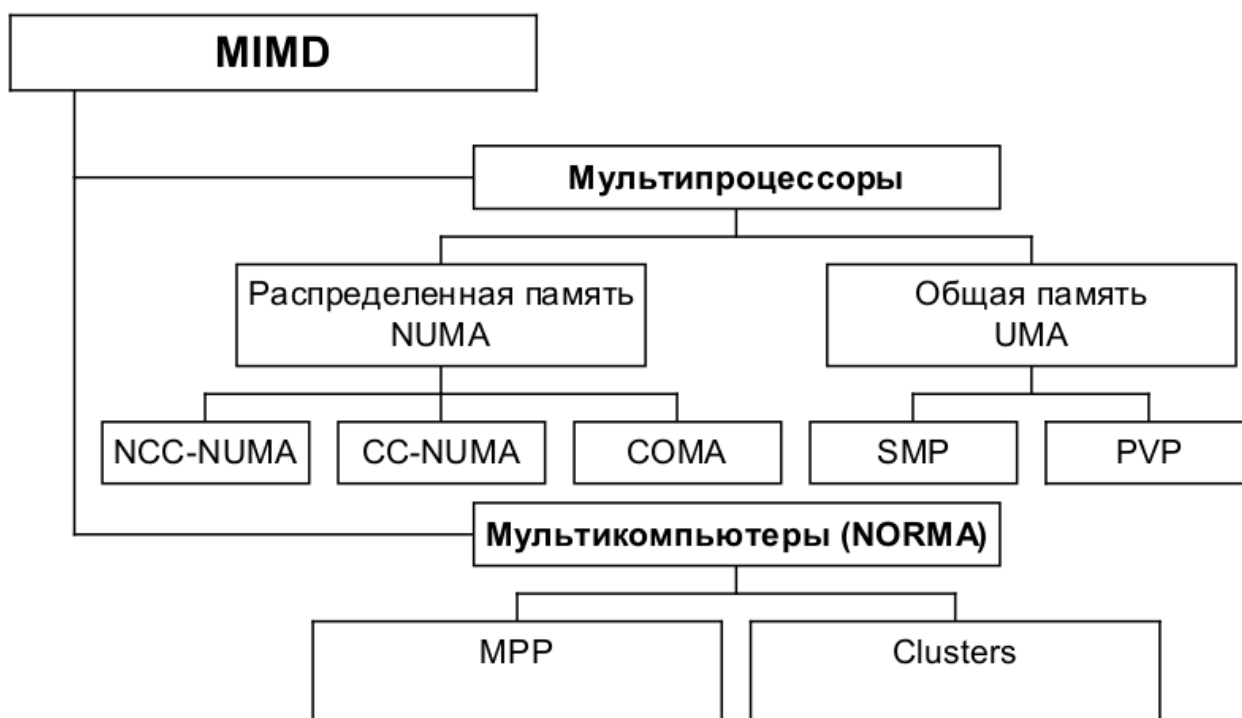


Рис. Подклассификация MIMD систем

SM-MIMD (shared memory MIMD)

В эту группу попадают многопроцессорные машины с общей памятью, многоядерные процессоры с общей памятью. Классический и самый распространенный пример — мультипроцессоры — многопроцессорные SMP-сервера. В таких машинах память каждому процессору видна как общее адресное пространство, и процессоры обмениваются друг с другом данными по общей адресной шине через общие переменные (*shared variables*). Для каждого процессора доступ к любому участку памяти является одинаковым (UMA—«однородный доступ к памяти»).

Достоинства: относительно легко программировать, поддержка SMP существует уже давно во всех ведущих операционных системах.

Недостатком данных машин является их невысокая масштабируемость: чем больше процессоров в системе, тем выше становится нагрузка на общую шину. В коммерческих вариантах таких систем максимальное число процессоров не превышает 64.

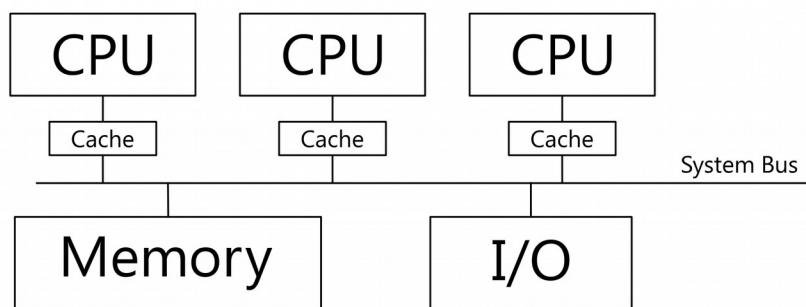


Рис. Пример архитектуры SMP сервера (SM-MIMD)

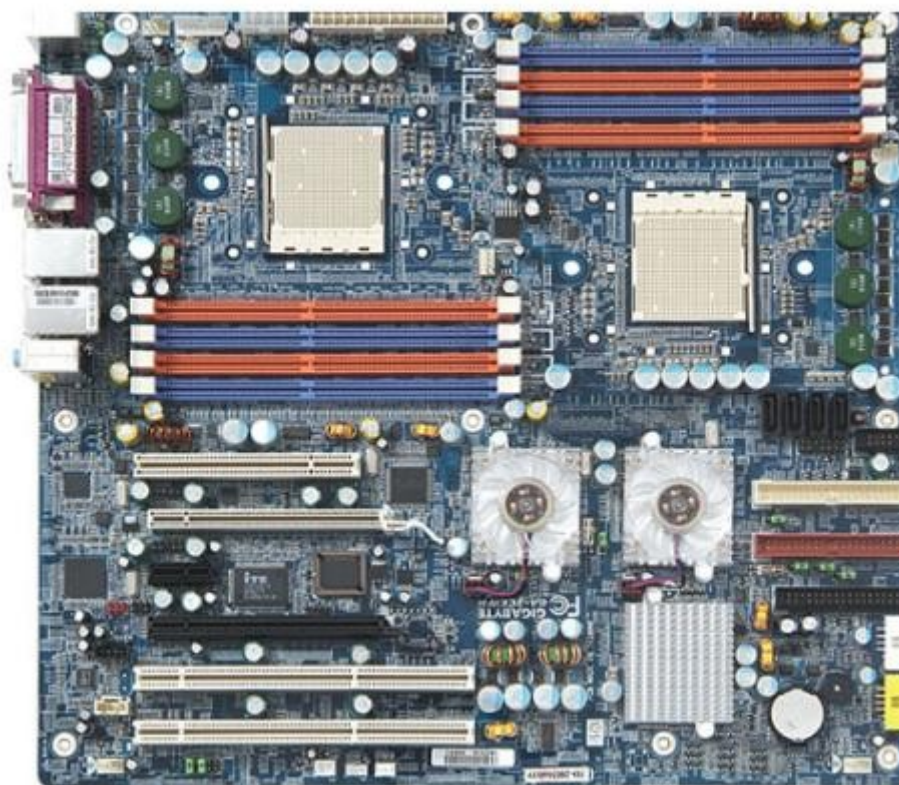


Рис. SMP матерплата.

Как уже было сказано выше, память, которая программисту видна как одно общее адресное пространство, может быть физически распределена по узлам системы. Такой подкласс машин называется **DSM-MIMD** (*distributed shared memory MIMD*). В этом подклассе машин у каждого процессора имеется своя локальная память, а к другим участкам памяти процессор обращается через высокоскоростное соединение. Так как доступ к разным участкам общей памяти является неодинаковым (к своему локальному участку быстрее, к другим — медленнее), то такие системы носят название NUMA (от *Non-Uniform Memory Access*). Так как память физически распределена, возникает трудность с тем, чтобы каждый процессор видел в памяти изменения, сделанные другими процессорами. Придумано несколько способов решения этой проблемы: через согласование кэша — ccNUMA, без согласования кэша — pccNUMA.

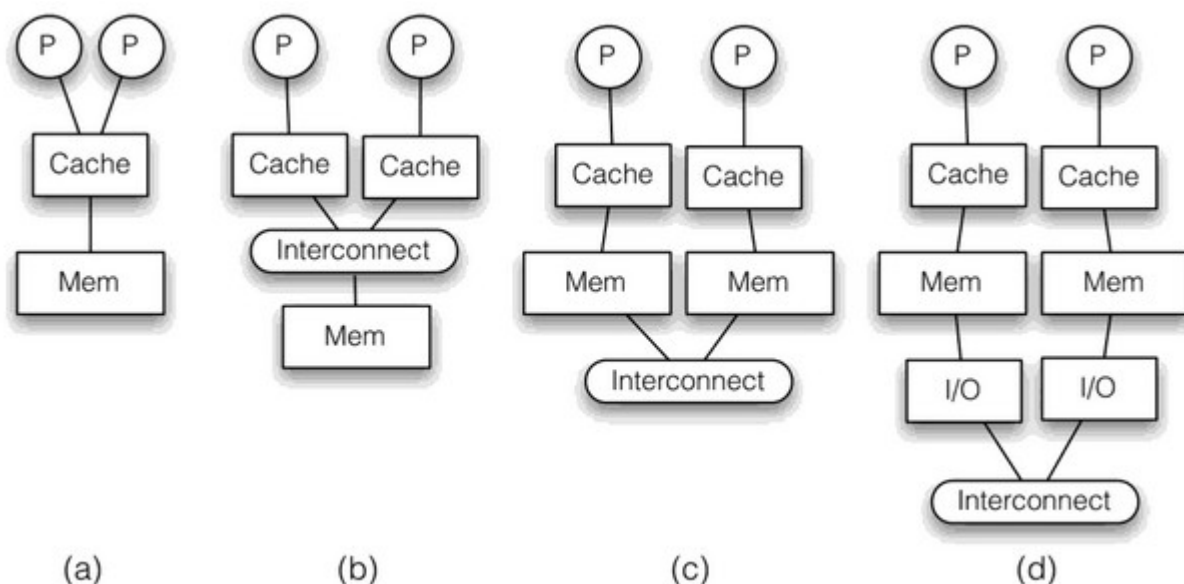


Рис. Подклассы MIMD. а) Shared cache, б) UMA— однородный доступ к памяти, с) (NUMA) *Non-Uniform Memory Access* — неравномерный доступ к памяти, д) распределенная память

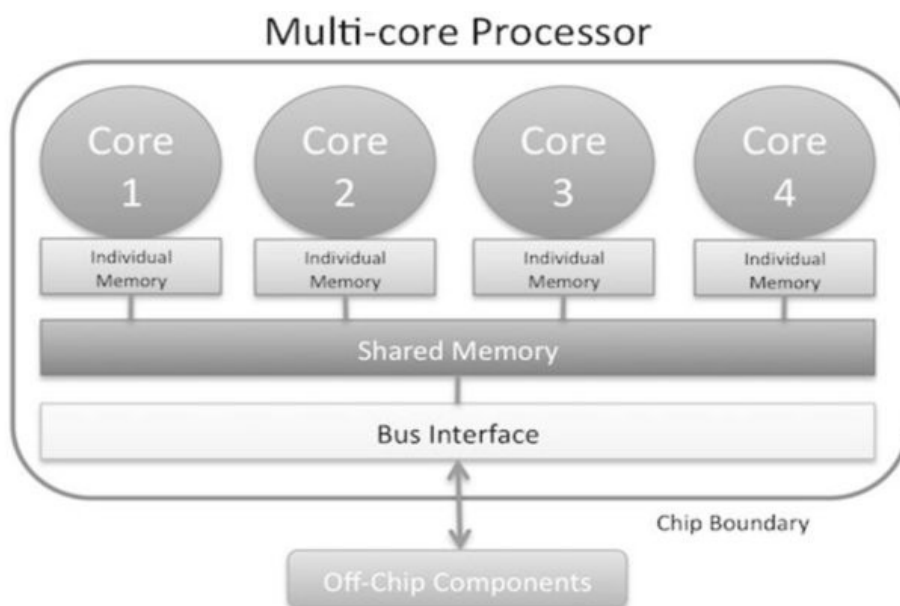


Рис. Архитектура многоядерного процессора. (MIMD)

DM-MIMD (distributed memory MIMD)

В этот под-класс попадают многопроцессорные MIMD-машины с распределенной памятью.

У каждого процессора имеется своя собственная локальная память, которая не видна другим процессорам. Каждый процессор в такой системе выполняет свою задачу со своим набором данных в своей локальной памяти. Если процессору нужны данные из памяти другого процессора, данный процессор обменивается с другим процессором сообщениями, то есть в таких системах используется модель программирования *Message Passing*, с помощью *Parallel Virtual Machine* (PVM) или какой-нибудь реализации *Message Passing Interface* (MPI).

Главное преимущество DM-MIMD маши — их высокая масштабируемость, позволяющая создавать массово-параллельные системы из нескольких сотен тысяч процессоров

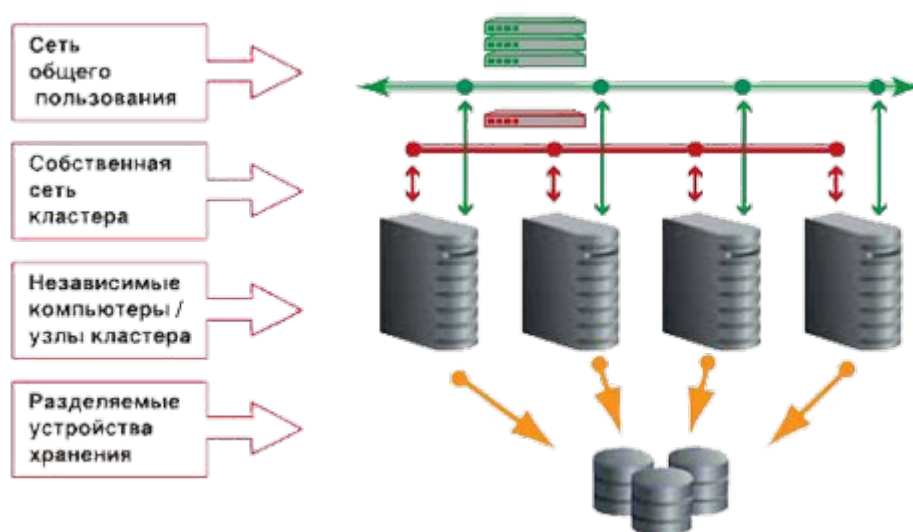


Рис. Архитектура вычислительного кластера (DM-MIMD).