

Лекция 4. Архитектуры современных вычислительных систем.

1. Влияние архитектуры МВС на эффективность решения задачи.

Цикл решения задачи: Физический объект – математическая модель – дискретная модель – алгоритм решения – программная реализация – вычисления на ЭВМ – анализ результатов. *На всех этапах* может понадобиться как *понятие параллелизма*, так и *необходимость его использования при вычислениях* для получения новых знаний об объекте. *При написании параллельной программы необходимо знание аппаратной среды*, т.е. архитектуры МВС. Иначе код может оказаться неэффективным или вообще не пригодным для расчетов.

2. Классификация архитектур ВС.

Основные параметры ВС: количество и архитектура процессоров, организация памяти, топология и скорость межпроцессорных коммуникаций, синхронность работы отдельных устройств, способ выполнения элементарных операций, технология программирования и др. Классификация ВС призвана эффективно решать задачу *отображения алгоритма решения конкретной задачи на архитектуру конкретной ВС*.

1) Классификация Флина (1966 г.) – в основе понятие потока команд и данных. 4 класса архитектур: **SISD (Single Instruction stream / Single Data stream):** УУ<--(ПК)-->ПР<--(ПД)-->ОЗУ (потоки команд и данных)

Пример: классические послед. ЭВМ – PDP-11, VAX 11/780, CDC 6600 (скалярная) и CDC 7600 (векторная).

SIMD (Single Instruction stream / Multiple Data stream): УУ<--(ПК)-->ПР<===(ПД)==>ОЗУ

Пример: ILLIAC IV (векторная), Cray-1 (векторно-конвейерная), все ВС которые имеют матрицы процессоров.

MISD (Multiple Instruction stream / Single Data stream): УУ<===(ПК)==>ПР<--(ПД)-->ОЗУ

Пример: данный класс пуст – таких машин не удалось пока создать

MIMD (Multiple Instruction stream / Multiple Data stream): УУ<===(ПК)==>ПР<===(ПД)==>ОЗУ

Пример: все мультипроцессорные системы: Cm*, C.mmp, Cray Y-MP, BBN Butterfly, Intel Paragon, Cray T3D

Все векторно-конвейерные машины тоже можно сюда отнести.

Эта классификация является основой современной, но имеет 2 недостатка: 1) не исчерпывает все виды ВС и 2) класс MIMD слишком широк.

2) Классификация Хокни – уточнение класса MIMD.

Класс MIMD подразделяется на 3 группы: **конвейерные, переключаемые, сети**. Переключаемые делятся на **ВС с общей и распределенной памятью**. Сети делятся на ВС, в которых процессоры связаны в виде **регулярных решеток, гиперкубов, иерархических структур, структур, изменяющих конфигурацию**.

Идея: множественный поток команд может обрабатываться *либо* одним конвейерным устройством, работающем в режиме разделения времени для отдельных потоков, *либо* каждый поток обрабатывается своим собственным устройством.

1-й класс MIMD (конвейерные однопроцессорные).

2-й класс MIMD (многопроцессорные) делится на 2: **переключаемые** (аппаратная связь каждый с каждым через переключатель) и **сети** (прямая аппаратная связь только с ближайшими соседями, удаленная связь с остальными через систему маршрутизации). Переключаемые разделяются на системы с **общей** и **распределенной** памятью, доступ к которой осуществляется через переключатель. Сетевые ВС все имеют распределенную память и подразделяются только в соответствии с топологией сети.

Примеры:

конвейерные - процессорные модули в Denelcor NEP, компьютеры Tera MTA;

переключаемые с общей памятью – BBN Butterfly, Cray C90;

переключаемые с распределенной памятью – PASM, PRINGLE, IBM SP2;

сети – 1CAP (звезда); **транспьютерные системы**, Parsytec PowerXplorer, Intel Paragon, Cray T3D (регулярные решетки разной размерности, подробнее о транспьютерах); NCube, Intel iPSC (гиперкубы); Cm*, CEDAR (кластеры – компьютеры на одном коммутаторе или свиче); Parsytec CC (сети компьютеров с иерархической структурой в виде пирамид, деревьев и т.д.); сети компьютеров, изменяющие конфигурацию (интеллектуальные коммутаторы и свичи).

Зам. Есть на самом деле еще один класс *гибридные ВС* – сочетающие несколько архитектур. Пример: Connection Machine 2 – на внешнем уровне гиперкуб, каждый узел которого есть кластер процессоров с полной связью.

3) Классификация Фенга (1972 г.) – классификация по двум признакам – число бит n в машинном слове, обрабатываемых параллельно при выполнении машинных инструкций (сейчас совпадает с длиной машинного слова), число слов m , одновременно обрабатываемых ВС: $C=(n,m)$ – n – число битовых слоев, m – ширина битового слоя. $P=n*m$ – максимальная степень параллелизма ВС.

В результате имеем 4 класса ВС:

- разрядно-последовательные и пословно-последовательные ($n=m=1$);
- разрядно-последовательные, пословно-параллельные ($n=1, m>1$);
- разрядно-параллельные, пословно-последовательные ($n>1, m=1$);
- разрядно- параллельные и пословно-параллельные ($n>1, m>1$).

Недостаток: нет разницы между процессорными матрицами, векторно-конвейерными и многопроцессорными системами (т.е. не уточняется способ вычисления m). Достоинство – введение единой метрики для сравнения ВС.

4) Классификация Хендлера – классификация на основе различия между тремя уровнями обработки данных: уровень **выполнения программы** (выборка и дешифрация команд УУ); уровень **выполнения команд** (исполнение команд АЛУ); уровень **битовой обработки** (обработка битов элементарными логическими схемами - ЭЛС).

Пусть k – число УУ, d – число АЛУ, w – число групп ЭЛС в каждом АЛУ. Тогда $t(C)=(k,d,w)$ – характеристика компьютера без учета наличия конвейера. Если учесть такую возможность, то после расширения трехуровневой модели получаем характеристику $t(C)=(k, k', d, d', w, w')$, где k -число процессоров, работающих параллельно, k' -глубина макроконвейера из процессоров, d -число АЛУ в каждом, параллельно работающем процессоре, d' -глубина конвейера в каждом АЛУ, w -число разрядов, обрабатываемых в АЛУ параллельно, w' -число ступеней в конвейере ЭЛС, обрабатывающих отдельные разряды. $P = k*k'*d*d'*w*w'$ – максимальная степень параллелизма.

Дополнительно Хендлер вводит **три операции** для описания более сложных (гибридных) архитектур и учета других устройств ВС, отличных от процессоров: «x» – конвейерная обработка данных; «+» – параллельная обработка данных, «v» – альтернативная обработка данных (одна и та же ВС может работать в нескольких режимах).

5) Классификация Шнайдера (1988) – классификация на основе разделения потоков команд и данных на адреса и их содержимое. Обозначения: $S = \{(a_1<t_1>) (a_2<t_2>)..., (b_1<u_1>) (b_2<u_2>)..., (c_1<v_1>) (c_2<v_2>)... \}$ – **поток ссылок**, конечное множество бесконечных последовательностей пар. Каждая пара содержит адрес и набор из n значений (неотрицательных чисел). **I** – **поток команд** - если значения рассматривать как команды, **D** – **поток данных**, если значения рассматривать как данные. Последовательность пар адрес-значение это история выполнения команд или перемещения данных. Число последовательностей в потоке равно числу инструкций (различных данных), которые компьютер может выполнять (обрабатывать) одновременно. $S_a = \langle a_1 b_1 c_1 \rangle \langle a_2 b_2 c_2 \rangle \dots$ - **последовательность адресов** потока S . $S_v = \langle t_1 u_1 v_1 \rangle \langle t_2 u_2 v_2 \rangle \dots$ - **последовательность значений** потока S . $w(S_x)=n$ – **ширина** последовательности S_x , равная длине элементов. $\Rightarrow w(S_a)=|S|, w(S_v)=n*|S|, |S|$ - мощность множества S .

Пара (I,D) – **вычислительный шаблон (ВШ)**. Все ВС можно классифицировать по ВШ.

ВС может исполнить ВШ (I,D) , если она в состоянии:

- 1) выдать $w(I_a)$ адресов команд для одновременной выборки из памяти;
- 2) декодировать и проинтерпретировать одновременно $w(I_v)$ команд;
- 3) выдать одновременно $w(D_a)$ адресов операндов;
- 4) выполнить одновременно $w(D_v)$ операций над различными данными.

Если эти условия выполнены, то ВС описывается как $I_{w(I_a),w(I_v)} D_{w(D_a),w(D_v)}$.

Классификация Флина в этой интерпретации:

SISD: $|I|=|D|=1, w(I_a)=w(D_a)=w(I_v)=w(D_v)=1 \Rightarrow I_{1,1} D_{1,1}$

SIMD: $|I|=1, w(I_a)=w(I_v)=1, |D|>=1, w(D_a)>1, w(D_v)>1$

Для более детальной классификации вводятся три значения, которые может принимать w : $w=s$ ($s=1$), $w=c$ (c – небольшая константа – жесткое ограничение аппаратуры – пример VLIW процессоры), $w=m$ (m – величина от 1 до большого конечного числа – компьютер по данному параметру масштабируем).

В итоге получаем следующие классы систем:

$I_{ss}D_{ss}$ – SISD системы; $I_{ss}D_{sc}$ – SISD системы, способные выполнять одну операцию над несколькими данными, расположенными по одному адресу, но с разным смещением (режим обработки двух полуслов);

$I_{ss}D_{sm}$ – SIMD системы без возможности получения уникального адреса данных в каждом процессорном элементе;

$I_{ss}D_{mm}$ – SIMD системы с возможностью изменения адресов операндов в каждом процессорном элементе;

$I_{sc}D_{cc}$ – системы, выбирающие и исполняющие одновременно несколько команд, для доступа к которым используется один адрес (VLIW-компьютеры); $I_{mm}D_{mm}$ – все MIMD системы.

Результат: гибкая систематизация SIMD, но MIMD системы опять в одном классе.

б) Классификация Скилликорна (1989) – специальное расширение классификации Флина, основанное на описании архитектуры с помощью составляющих ее 4-х компонент:

а) процессор команд (IP – Instruction Processor) – интерпретатор команд как отдельное ФУ (может отсутствовать);

б) процессор данных (DP – Data Processor) – АЛУ;

в) иерархия памяти (IM – Instruction Memory, DM – Data Memory) – ЗУ для команд и данных, пересылаемых между процессорами;

г) переключатель (SW – Switch) – абстрактное устройство, обеспечивающее связь между процессорами и памятью (может отсутствовать).

Функции процессора команд схожи с функциями УУ последовательных машин: на основе своего состояния и информации от DP процессор команд 1) определяет адрес следующей команды, 2) осуществляет доступ к IM для ее выборки; 3) получает и декодирует команду; 4) сообщает DP команду, которую надо выполнить; 5) определяет и посылает DP адреса операндов команды в DM; 6) получает от DP информацию о результате выполнения команды.

Функции процессора данных схожи с функциями АЛУ: процессор данных 1) получает от IP команду, которую надо выполнить; 2) получает от IP адреса ее операндов; 3) выбирает операнды из DM; 4) выполняет команду; 5) запоминает результат в DM; 6) возвращает в IP информацию о состоянии после выполнения команды.

Схема: $DM \leftarrow (\text{их адреса}) (\text{операнды}) \Rightarrow DP \leftarrow (\text{команды}) (\text{инфо о состоянии}) \Rightarrow IP \leftarrow (\text{команды}) (\text{их адреса}) \Rightarrow IM$. Она не содержит переключателей. Однако в других ВС они есть. Поэтому ввел 4 типа абстрактных переключателей:

1-1 – переключатель, связывающий пару ФУ;

n-n – переключатель, связывающий попарно устройства из двух множеств;

1-n – переключатель, связывающий одно устройство со всеми ФУ из некоторого множества;

nxn – переключатель, обеспечивающий связь каждый с каждым для двух множеств ФУ.

Примеры:

1) все матричные процессоры имеют переключатель типа **1-n** для связи единственного процессора команд с множеством процессоров данных;

2) в системах, где каждый процессор данных имеет локальную память, эта связь обозначается как **n-n**;

3) если каждый процессор команд может связаться с любым другим процессором, то эта связь обозначается как **nxn**.

В итоге классификация Скилликорна включает восемь параметров:

1) кол-во IP; 2) кол-во IM; 3) тип переключателя IP-IM; 4) кол-во DP; 5) кол-во DM; 6) тип переключателя DP-DM; 7) тип переключателя IP-DP; 8) тип переключателя DP-DP.

Примеры: Connection Machine 2 (**1, 1, 1-1, n, n, n-n, 1-n, nxn**); BBN Butterfly (**n, n, n-n, n, n, nxn, n-n, -**).

3. Особенности современных вычислительных систем. Гибридные архитектуры.

1) **векторно-конвейерные системы** - Cray-1(1976), Cray C90 (1990), NEC Earth Simulator (2002);

2) **системы с общей памятью** – SMP-системы, ccNUMA – HP Superdome (2001);

3) **системы с распределенной памятью** –

3а) **суперкомпьютеры с внешней шиной** Intel Paragon, IBM SP1/SP2, Cray T3D/T3E (1990), Parsytec CC (1995);

3б) **кластеры** – Beowulf (1994), Avalon (1998), Velocity+ (2000), MBC-1000M (2001), MBC-15K (2005), MBC-100K (2008), MBC-10П (2013);

3с) **сети МВС, метакомпьютеры** – отличаются неоднородностью.

4) **гибридные системы.**

Первые две группы бьют рекорды производительности по вычислительной мощности, приходящейся на один процессор, однако число процессоров жестко ограничено \Leftrightarrow не масштабируемы. Третья группа – наиболее распространены, не имеют ограничений по числу процессоров (масштабируемы), объему ОЗУ и внешней памяти. Четвёртая группа имеет самый высокий показатель - пиковая производительность на единицу потребляемой мощности.

4. Структура и особенности гибридных ВС.

Вычислительный кластер

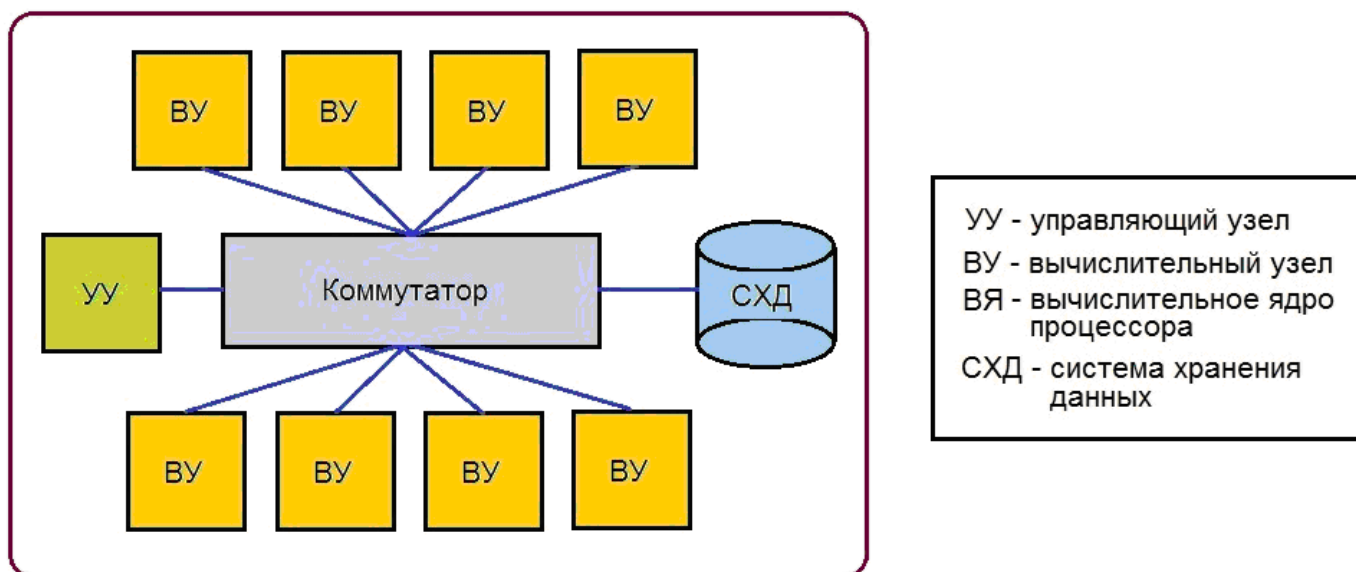


Рис. 1. Архитектура вычислительного кластера.

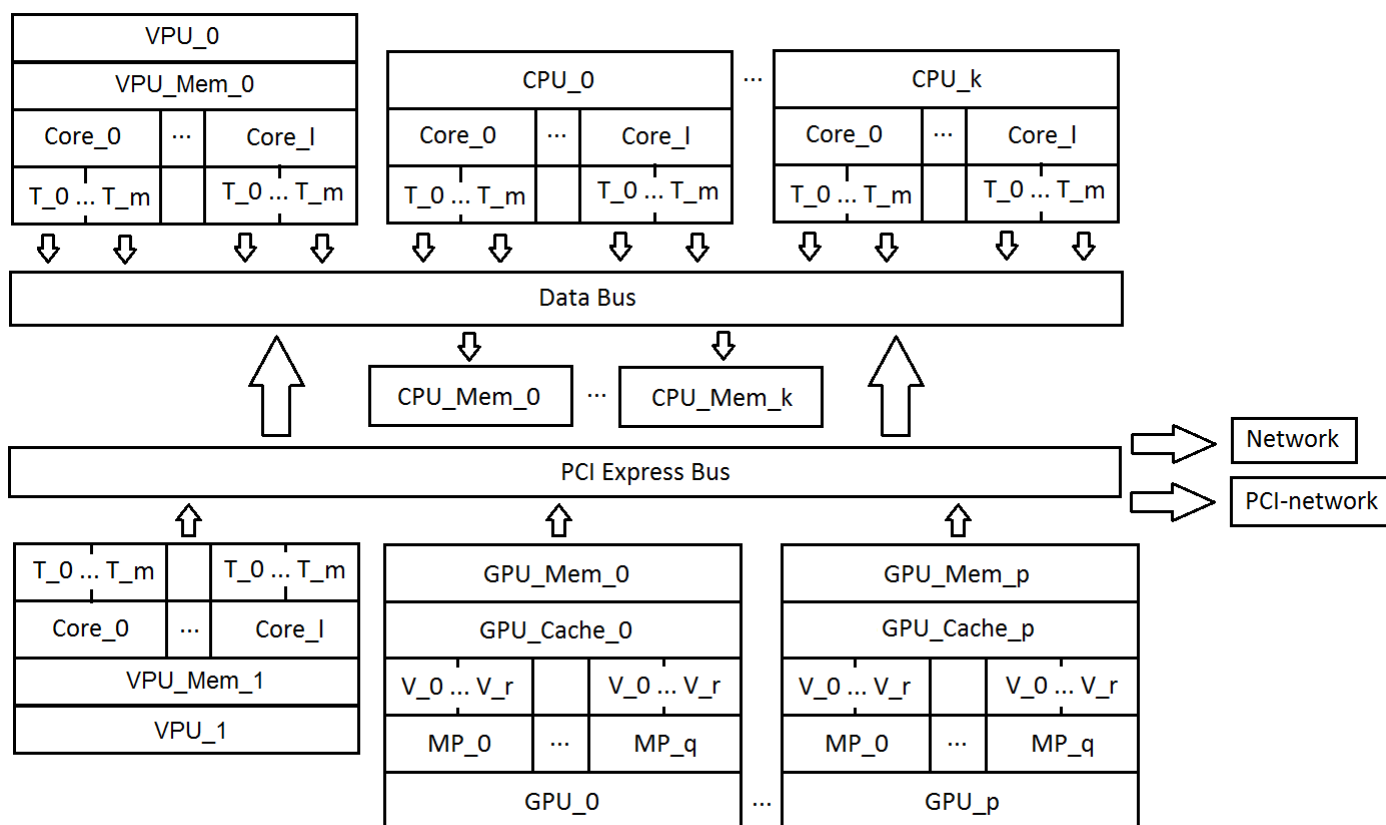


Рис. 2. Архитектура узла гибридного вычислителя.

Новые виды вычислителей: $APU = CPU + GPU$, $IPU = CPU + GPU + Mem$

APU (сокр. от **Accelerated or Agregated Processing Unit**, ускоренное или агрегированное обрабатывающее устройство) – гибридный центральный процессор, который объединяет центральный процессор с графическим процессором в одном кристалле.