

# Организация ЭВМ и систем

---

## Лекция № 7

- **VLIW архитектура**
- **Особенности IA – 64**
- **Особенности EPIC**

# VLIW архитектура

Предложенные в IA – 64 архитектурные идеи близки к концепции VLIW (Very Large Instruction World – сверх большое командное слово).

- VLIW -это набор команд, реализующий горизонтальный микрокод.
- Несколько (4-8) простых команд упаковываются компилятором в длинное слово. Такое слово соответствует набору функциональных устройств. VLIW архитектуру можно рассматривать, как статическую суперскалярную архитектуру. Имея ввиду, что распараллеливание кода производится на этапе компиляции, а не динамически во время исполнения. То есть, в машинном коде VLIW присутствует явный параллелизм.

# Особенности IA – 64

В IA – 64 2 нововведения по сравнению с RISC процессорами:

- Применение технологии явного параллелизма на уровне команд (EPIC – Explicitly Parallel Instruction Computing).
- Применение предикатных вычислений. **Предикаты** – способ обработки ветвлений (условных переходов).

# Особенности EPIC

- Большое количество регистров.
- Масштабируемость архитектуры до большого количества функциональных устройств. Это свойство представители фирмы Intel и HP называют «наследственно масштабируемый набор команд».
- Явный параллелизм в машинном коде. Поиск зависимостей между командами производит не процессор, а компилятор.
- Предикация. Команды из разных ветвей условного Ветвления снабжаются предикатными полями (полями условий) и запускаются параллельно.
- Загрузка по предположению. Данные из медленной основной памяти загружаются заранее.

# Описание IA - 64

## Регистры IA – 64:

- 128 64-разрядных регистров общего назначения.
- 128 80-разрядных регистров вещественной арифметики.
- 64 1-разрядных предикатных регистров.

## Формат команды IA – 64:

- Идентификатор команды.
- Три 7-разрядных поля операндов – 1 приемник и 2 источника.
- Особые поля для вещественной и целой арифметики.
- 6-разрядное предикатное поле ( $64=2^6$ ).

## Предикация

## Загрузка по предположению

# Архитектура E2K

(реализована в суперкомпьютере Эльбрус 3)

В E2K используются *команды переменной длины*.  
Общий формат команд E2K представлен на рисунке.

Заголовок	Слог 1	...	Слог №
-----------	--------	-----	--------

Команда E2K состоит из слогов длиной 32 разряда каждый. Число этих слогов может меняться от 2 до 16, причем данную архитектуру можно еще расширить – до 32 слогов.

Любая команда всегда включает 1 слог заголовка и еще от 1 до 15 слогов, указывающих на операции, которые могут выполняться параллельно. Слог заголовка содержит информацию о структуре команды и ее длине, что облегчает дешифрацию команды переменной длины. Применение заголовка позволяет не проводить предварительного декодирования команд перед их помещением в кэш команд.

В архитектуре E2K представлен *сверхбольшой файл регистров*. Все регистры E2K являются универсальными и могут содержать как целочисленные данные, так и числа с плавающей запятой. Всего имеется 256 регистров длиной по 64 разряда каждый.

В E2K есть два почти симметричных кластера, каждый из которых содержит по 256 регистров. Всего в этом процессоре имеется 30 регистровых портов: 20 портов чтения (по 10 портов на кластер) и 10 портов записи.

В Alpha 21264 применяется реализованное во многих суперскалярных процессорах динамическое переименование регистров. Этого механизма в E2K нет, так как в нем подобные задачи возлагаются на компилятор, однако в циклах с постоянным шагом применяется аналогичная схема циклической замены используемых регистров.

Еще одна особенность E2K – *регистраемое окно для процедуры*. Это решение является традиционным для машин серии «Эльбрус», однако особенно важным оно является для E2K, поскольку он содержит сверхбольшое количество регистров – 256.

Затраты на сохранение/восстановление регистров в данной ситуации становятся весьма значительными. Поэтому реализация в E2K аппаратного механизма переключения окон представляется актуальной. Окно регистров в E2K имеет переменную длину (до 192 регистров). Адресация регистров внутри контекста происходит относительно текущей базы, и при вызове другой процедуры достаточно сменить значение базы.

*Кэш данных первого уровня* в E2K имеет емкость всего 8 Кбайт и продублирован в каждом из кластеров. Этот кэш является прямоадресуемым, использует алгоритм сквозной записи данных.

*Кэш данных второго уровня* в E2K имеет емкость 256 Кбайт при времени доступа в 8 тактов. Он является двухканальным частично-ассоциативным и имеет 4 банка, то есть обеспечивает 4-кратное расслоение кэш-памяти. В кэше данных второго уровня применяется алгоритм обратной записи. Он также является неблокирующим.

Кроме этого в E2K представлен *специализированный кэш предварительной выборки*, который разработчики назвали буфером предварительной подкачки. Он является частью устройства доступа к массивам и задействуется только при работе с массивами в циклах. Его емкость составляет всего 4 Кбайт, и он состоит из 2 банков с 2 портами в каждом из них. За один такт в буфер можно считать, следовательно, до 4 слов длиной 8 байт. Буфер организован как очередь FIFO и имеет до 64 зон предварительной выборки.

В E2K предусматриваются *два варианта подключения третьего уровня – кэш*: непосредственно к процессору E2K, что позволяет разгрузить «системную шину» – коммутатор, или через набор коммутаторных микросхем.

*Функциональные устройства (ФУ) E2K разнесены по двум кластерам.* Эти кластеры содержат по 3 одинаковых целочисленных конвейера – АЛУ (правда, один из кластеров имеет также ФУ деления – целочисленного и с плавающей запятой).

В каждом кластере представлены также адресные сумматоры, которые имеются для 2 из 3 путей («каналов») данных. В результате каждый кластер может одновременно выполнять до 2 операций загрузки регистров или 1 операцию записи в оперативную память. Возможен и смешанный случай: 2 загрузки плюс одна запись.

Кроме того, имеется 4 канала для данных с плавающей запятой, по 2 на кластер. В каждом канале может выполняться команда типа *MADD* – «умножить-и-сложить», что дает темп 8 результатов с плавающей запятой за такт.

Сам набор команд E2K «богаче», чем у традиционных RISC-процессоров: в нем представлены *четырёхадресные команды*, например, типа  $d=a+b+c$ . Такого нет и в IA-64. Что касается команд с плавающей запятой, то кроме полной поддержки IEEE754 в E2K реализована работа с 80-разрядным представлением Intel x86. При этом операнды хранятся в парах 64-разрядных регистров E2K. Правда, сложение/умножение таких чисел не полностью конвейеризовано. Кроме того, для приближения системы команд E2K к x86 в E2K реализованы также команды расширения MMX.

В E2K *целочисленный конвейер имеет длину 8 тактов* (собственно выполнение идет на седьмом такте, а обратная запись – на восьмом) против 7 тактов в Alpha 21264.

E2K обеспечивает очень высокий уровень одновременно выполняемых операций: в команде их кодируется до 23 (сюда кроме арифметико-логических операций входят также доступ в оперативную память, приращение индекса массива и т.п.). Эффективные показатели параллельной работы ФУ у E2K выше, чем у всех суперскалярных процессоров.

В архитектуре E2K, как и в IA-64, делается все, чтобы по возможности исключить обычные операции перехода. Для этого в E2K *имеется 32 одноразрядных регистра-предиката*, причем команда способна сформировать до 7 предикатов: 4 в операциях сравнения в АЛУ и еще 3 – в операциях логики.

Хотя в IA-64 предикатных регистров формально в 2 раза больше, чем в E2K, реально их практически столько же, так как в IA-64 хранятся пары – предикат и его отрицание. В IA-64 поля предикатов всегда представлены в команде, а в E2K – могут отсутствовать. Предикаты могут использоваться в канале АЛУ или в канале доступа к массивам; для указания на это используются условные слоги, содержащие маски предикатов и ФУ. Всего в этих слогах может кодироваться до 6 предикатов, указывающих на то, нужно ли выполнять соответствующие операции из «широкой» команды.

*Компилятор E2K порождает коды для обеих ветвей программы, возникающих при условном переходе, и, пользуясь большим числом ФУ и регистров, заставляет процессоры выполнять обе ветви программы. Та же процедура применяется и в IA-64. До тех пор, пока условие перехода остается неизвестным, обе ветви выполняются спекулятивно. Когда, наконец, условие найдено, выбираются нужные результаты. Признак спекулятивного выполнения взводится при этом в специальном бите в коде операции в соответствующем слоге. При возникновении ситуации исключения (exception) результат снабжается тегом недействительного значения.*

В файле предикатов E2K, как и в регистровом файле, используются окна.

Еще некоторые особенности архитектуры E2K:

- *тегирование данных*, поддерживаемое во всей линейке процессоров ЭВМ «Эльбрус»;
- *сегментно-страничная организация памяти*;
- *поддержка мультипрограммирования в стиле x86*.

В сочетании с разработанными средствами двоичной компиляции и специальными аппаратными средствами ее поддержки это позволяет выполнять x86-коды на E2K.

Поддерживается также двоичная компиляция для SPARC-архитектуры.

По материалам: Отечественные микропроцессоры: Elbrus E2K  
Михаил Кузьминский/17.05.1999. [Открытые системы. #05-06/1999](#)

Характеристика	IA – 64 (VLIW)	E2K
1. Формат команд	Длина команд фиксирована. 3 команды в связке длиной 128 бит. В формат команды вводятся маски, которые указывают на зависимость между командами.	Команды переменной длины, состоит из слогов 32 разряда каждый (м.б. 2-16 слога, можно расширить до 32 слогов). Во всех командах <u>1 слог заголовка</u> : <ul style="list-style-type: none"> <li>▪содержит информацию о структуре команды и ее длине ;</li> <li>▪Позволяет не проводить предварительного декодирования команд перед их перемещением в I-кэш.</li> </ul>
2. Регистры	128 – целочисленных регистров и 128 регистров с плавающей точкой.	Сверхбольшой РФ, все регистры <u>универсальны</u> , т.е. могут содержать как целочисленные данные, так и с плавающей точкой. 256 64-разрядных регистров.
3. Система команд		Четырехадресные команды типа $d=a+b+c$ , поддержка 80-разрядного представления формата с плавающей точкой Intel x86. Реализованы <u>MMX команды</u> .
4. Обработка команд	64 регистра предиката. Хранятся пары – предикат и его отрицание. Поля предикатов всегда представлены в команде.	32 одноразрядных регистра – предиката. Команда способна сформулировать до 7 предикатов, но иногда предикаты могут отсутствовать.
	Компилятор порождает коды для обеих ветвей программы и заставляет процессоры выполнять обе ветви программы (спекулятивно).	
		Тегирование данных (теговая память). сегментно-страничная организация и мульти-программирование в стиле x86.

# Сравнительные характеристики E2K и Alpha 21264

		<b>E2K</b>	<b>Alpha 21264</b>
<b>I – кэш L1</b>	Емкость, Кб	64	64
	Тип	4-канальный частично ассоц.	2-канальный частично ассоц.
	Время доступа	2 такта	3 такта
<b>D – кэш L1</b>	Емкость, Кб	8+8	64
	Тип	Прямонадресуемый	2-канальный частично ассоц
	Время доступа	2 такта	3 такта
<b>D – кэш L2</b>	Емкость, Кб	256	Кэш второго уровня является внешним
	Тип	2-канальный частично ассоц.	
	Время доступа	8 тактов	
<b>I – TLB</b>	Емкость	64 строки	128 строк
<b>D – TLB</b>	Емкость	16 строк (ассоц.) +256 строк 4-канальн. частично ассоц.	128 строк (ассоц.)
<b>Регистры</b>	Число	По 256 на кластер	80x2 целоч., 72 веществ.
	Число портов	20 чтения, 10 записи	8 чтения, 6 записи (целоч.)

	<b>E2K</b>	<b>Alpha 21264</b>
<b>Тактовая частота, МГц</b>	1200	500-667(*)
<b>Число транзисторов</b>	28 млн.	15,2 млн.
<b>Площадь, кв.мм</b>	126	302
<b>Рас рассеяние тепла, Вт</b>	35	60
<b>Технология, мкм</b>	0,18	0,35(*)
<b>SPECint95</b>	135	30
<b>SPECfp95</b>	350	60

# Вопросы для самоконтроля

1. Основные особенности VLIW архитектуры.
2. Что нового появилось в архитектуре IA – 64 по сравнению с RISC процессорами.
3. Архитектурные особенности EPIC.
4. Что представляют собой команды переменной длины? Где они используются?
5. Перечислите особенности архитектуры E2K.