



Мультиклеточная архитектура. Состояние и перспективы



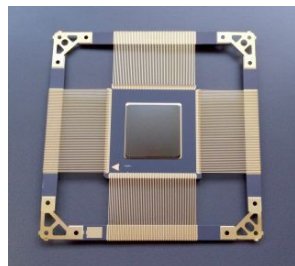
ОАО "Мультиклет"

разрабатывает процессоры и процессорные ядра на базе мультиклеточной архитектуры.



Мультиклеточная архитектура:

- ♦ параллельная, с естественной реализацией параллелизма;
- ♦ универсальная;
- ♦ не-фон-неймановская;
- ♦ не имеет зарубежных аналогов;
- ♦ патентно защищенная.



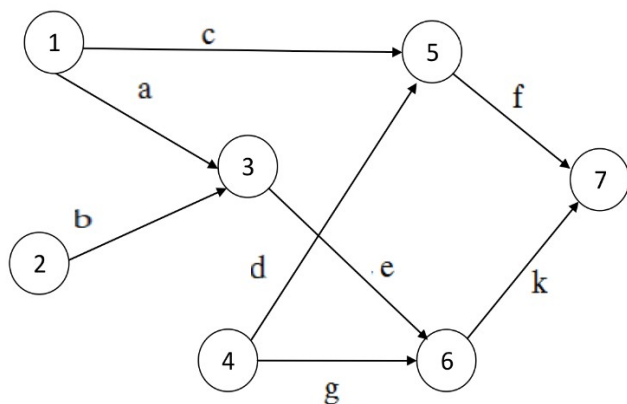
1. Копирование и доработка чужих технических решений — дорога аутсайдеров. В лучшем случае оно консервирует отставание, что подтверждается развитием компьютерной индустрии в нашей стране за последние пятьдесят лет.
2. Единственная возможность нашей страны выйти в число лидеров — это разработка своих, принципиально новых и патентно защищенных решений.

Общие положения

1. Любой алгоритм, выполняемый компьютером может быть представлен в виде ориентированного ациклического графа $G=(V,E)$, где (см. рис.):

$$V = \{1,2,3,4,5,6,7\},$$

$$E = \{a,b,c,d,e,f,g,k\}.$$



2. Любой граф может быть представлен **двумя** (!) списочными формами, либо списком смежности:

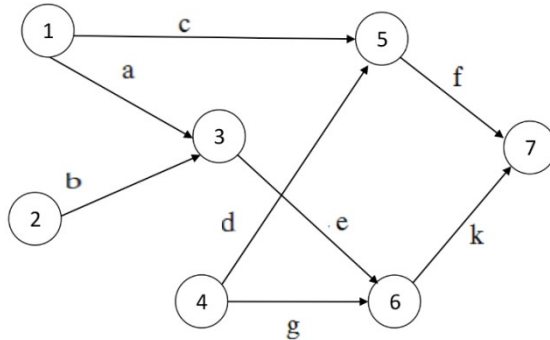
$$(5,3);2(3);3(6);4(5,6);5(7);6(7);7(\emptyset),$$

либо списком инцидентности:

$$1(a,c)^-;2(b)^-;3(a,b)^+(e)^-;4(d,g)^-;5(c,d)^+(f)^-;$$

$$(e,g)^+(k)^-;7(f,k)^+$$

Архитектуры на базе списка смежности



1(5,3);2(3);3(6);4(5,6);5(7);6(7);
7(∅)

Использование
обеспечивает:

списка смежности

естественную реализацию параллелизма (исполнение очередной операции инициируется исполнением предшествующих); минимальные затраты на передачу результатов, так как все связи заданы явно.

Необходимое условие – биективное отображение G на программно-аппаратную среду.

Примеры реализации

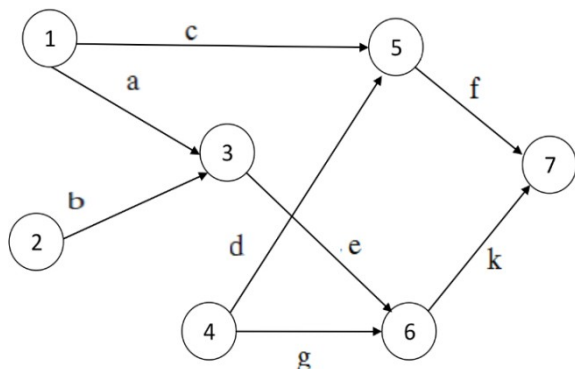
Аналоговые компьютеры

Функционально-ориентированные процессоры (например БПФ-процессоры)

Потоковые процессоры (проблемы, вызываемые сюръективным отображением не решены до сих пор)

Архитектуры на базе списка инцидентности

Использование списка инцидентности обеспечивает:



$1(a,c)^-; 2(b)^-; 3(a,b)^+(e)^-;$
 $4(d,g)^-; 5(c,d)^+(f)^-; 6(e,g)^+$
 $(k)^-;$
 $7(f,k)^+$

«естественную» реализацию параллелизма (исполнение операции инициируется готовностью требуемых ей результатов);

возможность использования сюръективного отображения G на программно-аппаратную среду и, как следствие, независимость описания G от среды.

Требуется организация хранения и общедоступность потока результатов до их использования, а также активизация исполнения операций потоком результатов.

Фон-неймановская(vN) и мультиклеточная архитектуры

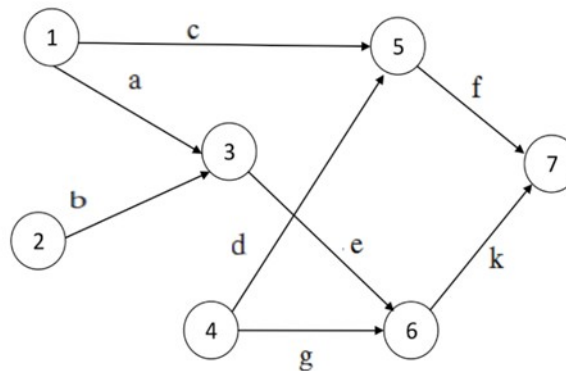


$3(a,b)+(e)^{-}$



add (7,5),(4)
(Rg4:=Rg7+Rg5)

vN -архитектура



$3(a,b)+(e)^{-}$



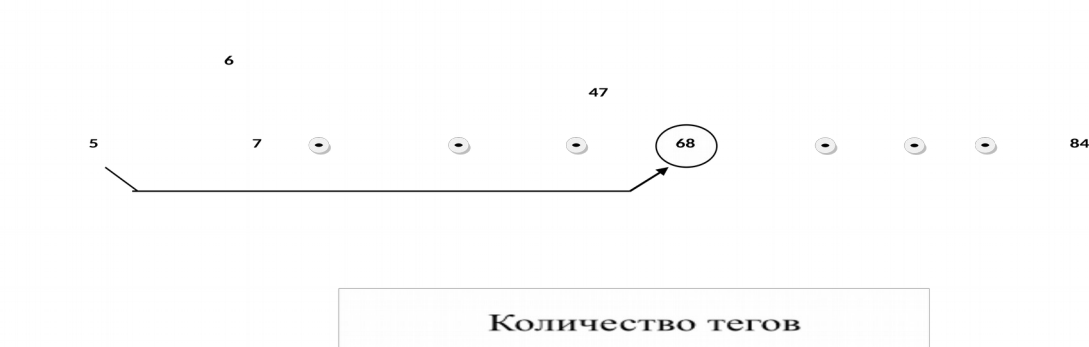
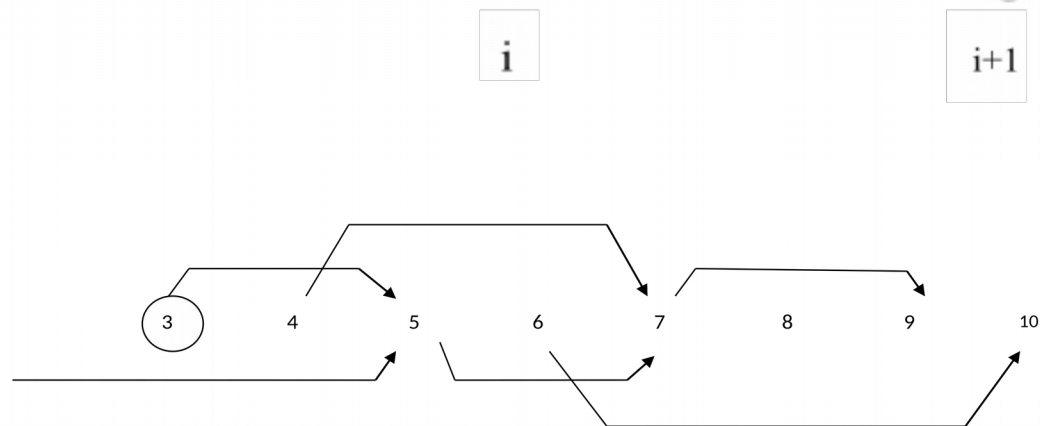
add ('1','2'),('3')



add (@2,@1)

Мультиклеточная
архитектура

Мультиклеточная архитектура. Выборка команд и присвоение тегов



Мультиклеточная архитектура. Формирование программы



Jump next

Rd a

Rd b

Rd c

Rd d

+ 4,1

- 4,3

wr 2,e

wr 2,f

Команды в программе объединены в параграфы - линейные и информационно-замкнутые последовательности команд.

Каждый параграф имеет один вход и не менее одного выхода. При наличии нескольких выходов в процессе выполнения должен выполняться только один.

Состояние процессора меняется, как правило, по завершению параграфа.

Мультиклеточная архитектура. Сравнение программ



Load 4,a

Load 3,b

Load 2,c

Load 1,d

+ 4,1

(4):= a + d

- 3,2

(3):= b - c

Wr 4,e

Wr 3,f

Jmp next

Jmp next

Rd a

Rd b

Rd c

a + d

Rd d

b - c

+ 4,1

- 4,3

wr 2,e

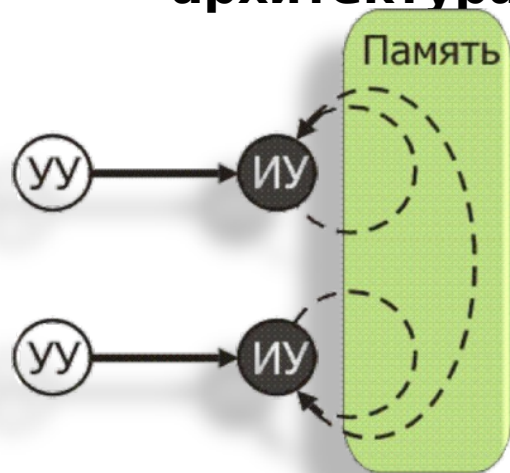
wr 2,f

- параграф

vN - процессор

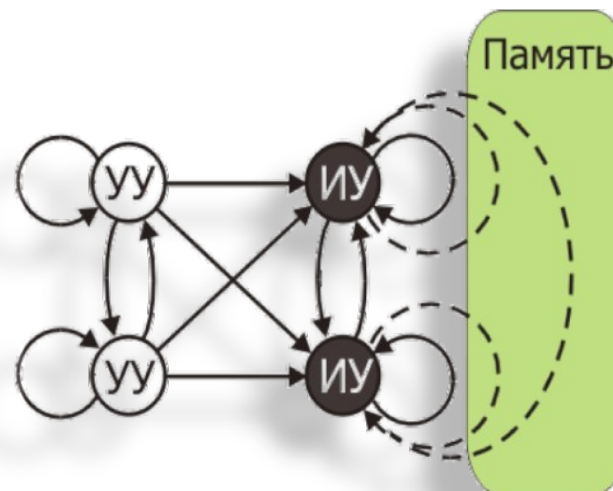
Сравнение архитектур

Многоядерная фон-неймановская архитектура



- ♦ Оперирует командами
- ♦ Команды упорядочены
- ♦ Информационный обмен между командами опосредован - через память («бутылочное горлышко»)

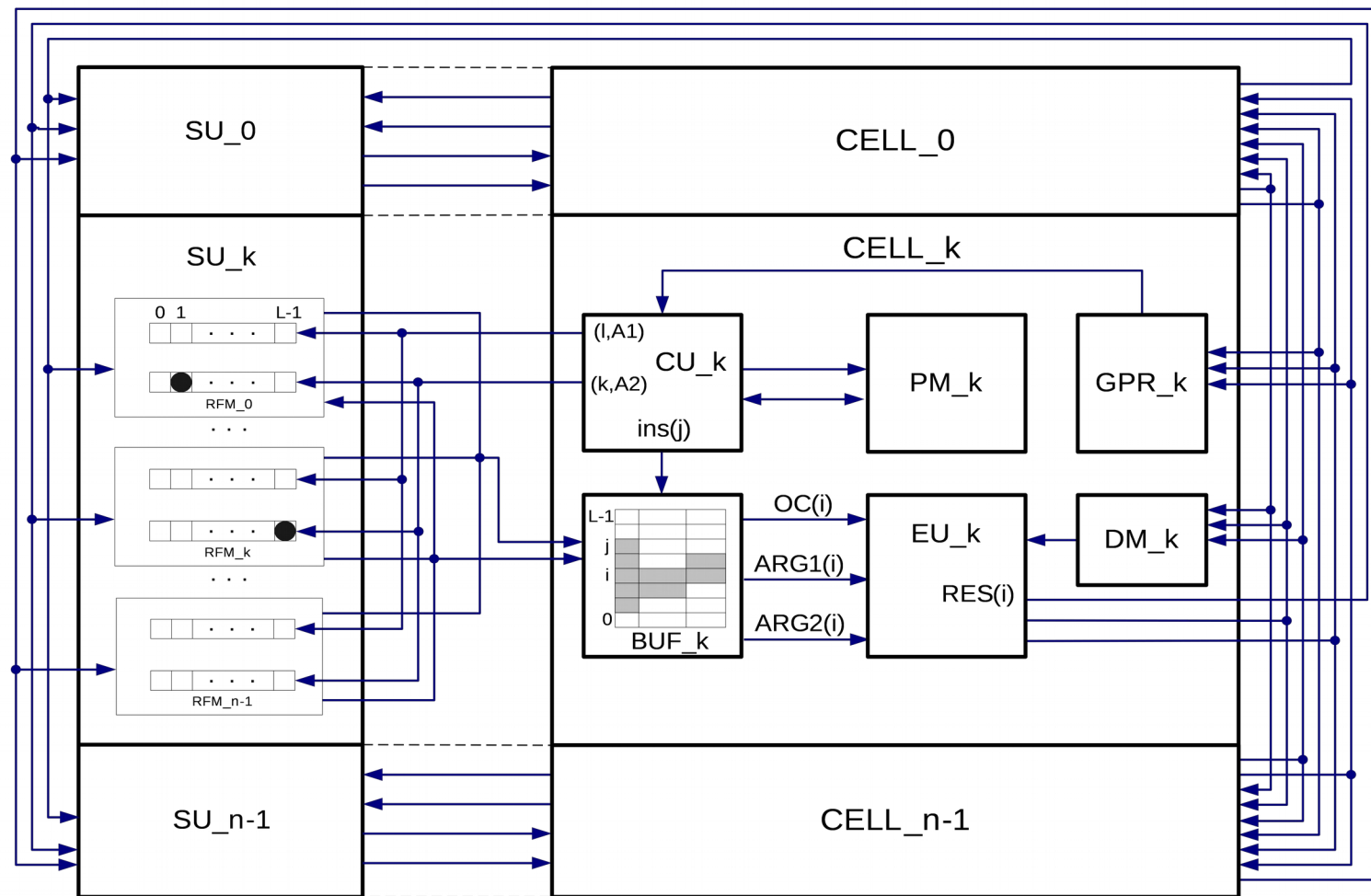
Мультиклеточная архитектура



- ♦ Оперирует параграфами - замкнутыми группами информационно связанных команд
- ♦ Команды в параграфе частично упорядочены
- ♦ Информационный обмен между командами - прямой, память в этот обмен не вовлекается

Мультиклеточная архитектура.

Структура процессора



Конкурентные преимущества мультিকлеточной архитектуры

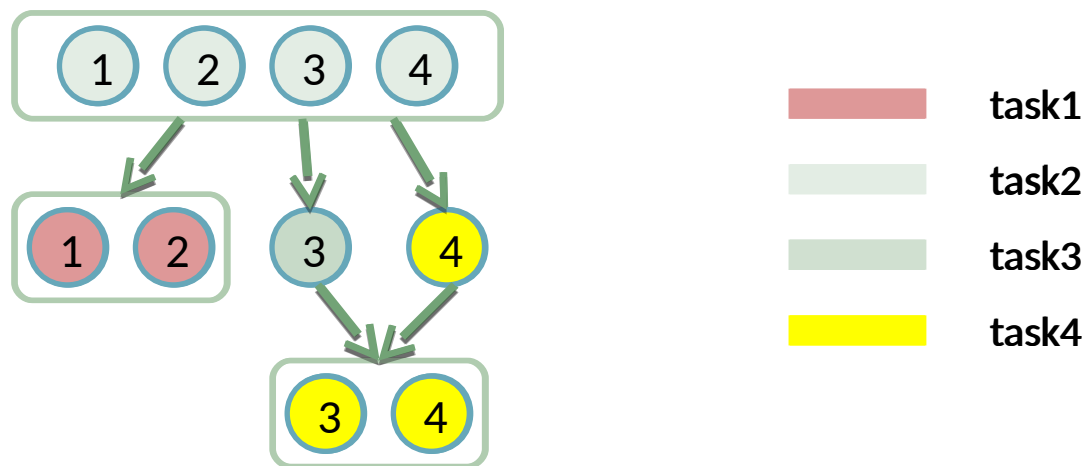


- ❖ Естественная реализация параллелизма (без решения задачи распараллеливания)
- ❖ Использование традиционных императивных языков высокого уровня для программирования и типовых инструментальных средств для разработки (LLVM)
- ❖ Уменьшение площади кристалла
- ❖ Эффективная реализация любого класса задач (коммутационная среда не накладывает ограничений на межклеточный обмен данными)
- ❖ Выполнение программы без перекомпиляции на любом количестве клеток
- ❖ Непрерывное выполнение программы при деградации аппаратной среды (отказах клеток)
- ❖ Динамическое перераспределение вычислительных ресурсов

Динамическая реконфигурация



Процессоры с мультиклеточной архитектурой – единственный тип процессоров, в которых возможна динамическая реконфигурация процессора в ходе решения потока задач



Производительность на CoreMark



	Multiclet P1-lcc compiler	Multiclet R1-lcc compiler	Multiclet R1-llvm compiler	1985B E91T Ми-ландр ARM Cortex-M3	STM32F4x Stmicroelectronics ARM Cortex-M4	RX62N Renesas	WIPS proActive RTL FPGA prototype	Intel Core i7-2760QM CPU@2.40GHz
Такт. част., МГц	80	100	100	80	168	100	31	2400
Общ. показ. Core Mark	24.49	24.95	56.45	117.6	501.85	311.54	137.10	85151.68
Core Mark/ МГц	0.31	0.25	0.56	1.47	2.98	3.12	4.42	35.48

Производительность на тесте porcnt



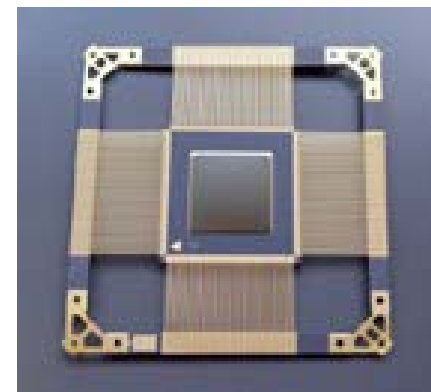
Сводная таблица результатов (количество тактов на один цикл расчёта 32-х бит)

	Multiclet R1 (C)	Multiclet R1 (ASM)	Pentium Dual Core 5700 3.0GHz	Intel Core i7- 4700HQ@2 400
BitHacks	59.4	5.0	9.5	4.7

Производительность

№/№	Параметры	Мультиклет R1	Аналоги		
			Процессор 1967BH034 компания "Миландр"	Семейство C66x компания TI	Процессор ADSP-TS201S компания Analog Devices
1	Количество операций выполняемых за один такт	4	6	8	24
2	Время выполнения CFFT-256 в тактах	2136	1812	1782	928
3	Вычислительн ый ресурс, затраченный на выполнение задачи	8544	10872	14256	22272

- ♦ Для DSP процессоров и процессоров общего назначения — оптимизация схемотехнических решений и, соответственно, уменьшение площади клеток при одновременном повышении их производительности
- ♦ Разработка решений обеспечивающих «живучесть» процессоров в системах с повышенной надежностью, обеспечивающих непрерывную работу системы при отказах клеток
- ♦ Увеличение количества клеток на кристалле и создание многоуровневых мультиклеточных систем для супервычислений



**СПАСИБО ЗА
ВНИМАНИЕ!**