КОМПЛЕКСИРОВАНИЕ SIMD-СТРУКТУР ВНУТРИ КРИСТАЛЛА НА ПРИМЕРЕ БЛОЧНОГО УМНОЖЕНИЯ МАТРИЦ БОЛЬШОЙ РАЗМЕРНОСТИ

Затуливетер Ю.С., Фищенко Е.А.
Институт проблем управления им. В.А. Трапезникова РАН, г. Москва
zvt@ipu.rssi.ru, elena.fish@mail.ru

Аннотация. Отсутствие высокоэффективных многопроцессорных архитектур становится ключевой проблемой развития высокопроизводительной элементной базы. Рассматривается базовый вычислительный модуль (БВМ), который развивает масштабируемую архитектуру отечественного многопроцессорного компьютера ПС-2000, апробированную эффективными промышленными применениями на широких классах задач с массовым параллелизмом. Рассматривается возможность комплексирования БВМ внутри кристалла на примере задачи умножения матриц большой размерности.

Ключевые слова: блочное умножение матриц, большая размерность, комплексирование внутри кристалла, SIMD.

Введение

Многоядерные кристаллы на основе универсальных микропроцессоров не дают роста производительности пропорционального числу ядер, при том, что передовые технологии нанометрового диапазона обеспечивают возможности создания высокопроизводительных многопроцессорных вычислительных систем, с миллиардами транзисторов на одном кристалле, которые интегрируют в нем логические схемы и схемы памяти. Поэтому, в настоящее время сложилась ситуация, когда опережающие возможности СБИС- технологий оказываются недоиспользованными из-за недостатка высокопараллельных архитектур общего назначения, ориентированных на задачи с массовым параллелизмом, которые позволяют с высокой эффективностью использования своих ресурсов осуществлять вычисления с интенсивным обменами промежуточными данными как между процессорами, так и между процессорами и памятью.

Одним из главных инструментов наращивания производительности становятся однокристальные компьютерные архитектуры с массовым параллелизмом, и прежде всего SIMD- и VLIW-архитектуры.

Новейшие разработки однокристальных "ненеймановских" многопроцессорных архитектур с массовым параллелизмом (сотни и тысячи процессоров на кристалле) всё ещё находятся в начальной стадии индустриального становления и развития.

Наиболее известным примером промышленного производства таких архитектур служат новейшие графические процессорные устройства (Graphics Processing Unit – GPU) известных производителей видеокарт для ПК – nVIDIA и AMD (ATI). Однако оснований считать, что архитектуры, изначально ориентированные на использование в составе видеоплат ПК, или других узкопрофильных применений, сохранят высокую эффективность на других классах задачах с массовым параллелизмом, пока нет. Проблемы поиска и обоснования конкурентоспособных многопроцессорных архитектурных решений, отвечающих требованиям массовых применений в широких классах задач, в наступившем десятилетии становятся одними из наиболее приоритетных.

По сути, речь идёт о развитии новой ниши компьютерного рынка – высокопроизводительных однокристальных компьютеров общего назначения (General Purpose – GP) с массовым параллелизмом. Этот рынок охватит весь диапазон применений – от массовых устройств мобильной связи и встраиваемых систем управления, до суперкомпьютеров производительностью 1-1000 Пфлопс и более.

1. Базовый вычислительный модуль

В основу представленной в данной работе архитектуры однокристального MULTI-SIMD компьютера положены оригинальные научные и практические разработки Института проблем управления РАН и успешный опыт промышленного производства (1980-1988гг.) и применения (1980-1997гг.) отечественного многопроцессорного компьютера ПС-2000 [1].

На рис.1 представлена структура базового вычислительного модуля (БВМ), который содержит:

- набор из N процессорных элементов (ПЭ) (АЛУ + регистры);
- программируемые устройства активации, обеспечивающие возможность индивидуального отключения каждого ПЭ по вычисляемым предикатам;
- векторную (обмен между ПЭ) коммутацию;
- равномерно распределенную по ПЭ векторную память M с общей и локальной косвенной адресацией через регистры;
- общее устройство управления (ОУУ) с памятью общих команд, скалярным АЛУ и скалярной памятью H;
- широковещательную коммутацию от ОУУ к ПЭ;
- канал прямого доступа D в распределенную векторную память ПЭ;
- интерфейс X для обмена данными со скалярной памятью.

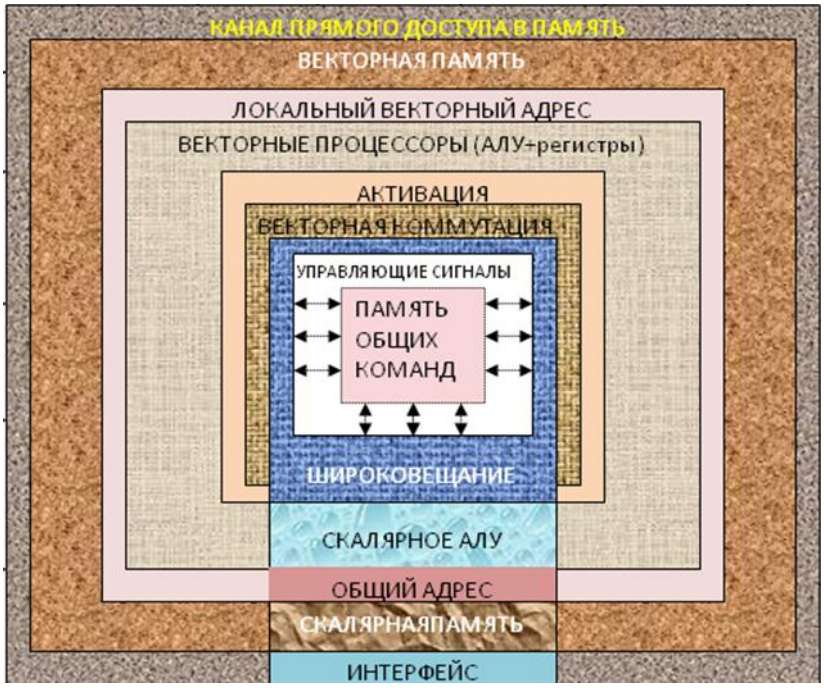

Рис. 1. Структура базового вычислительного модуля (БВМ)

Из памяти общих команд в ходе исполнения программ подается поток команд в виде многокомпонентных операций, которые на каждом такте задают множество одновременно выполняемых действий для всех параллельно работающих векторных и скалярныхустройств БВМ.

SIMD-архитектура БВМ характеризуется:

- высоким соотношением производительность/стоимость, которое обеспечивается высокой гибкостью и, одновременно, простотой структурно-аппаратных решений:
  - достаточно мощными полноразрядными ПЭ (АЛУ + регистры),
  - VLIW командами (простота дешифрации, конвейер команд организует оптимизирующий транслятор),
  - сегментируемым каналом векторной коммутации;

- масштабируемостью:
  - по числу ПЭ (8÷32 и более),
  - по объему памяти, обеспеченной косвенной (через регистры) адресацией как общей, так и локальной в ПЭ;
  - по пропускной способности канала ввода/вывода с прямым доступом в распределенную память;
  - по частоте (0,5–2 ГГц);
- комплексируемостью в параллельные и конвейерные структуры как внутри кристалла, так и между кристаллами.

2. Комплексировании БВМ внутри кристалла

На рис.2 приведена структура, которая позволяет программной реконфигурацией комплексировать БВМ внутри кристалла. БМВ в данном примере содержит 32 процессорных элементов (ПЭ) и общее устройство управления (ОУУ). БМВ в зависимости от решаемой задачи могут объединяться в параллельные и конвейерные структуры различной конфигурации.

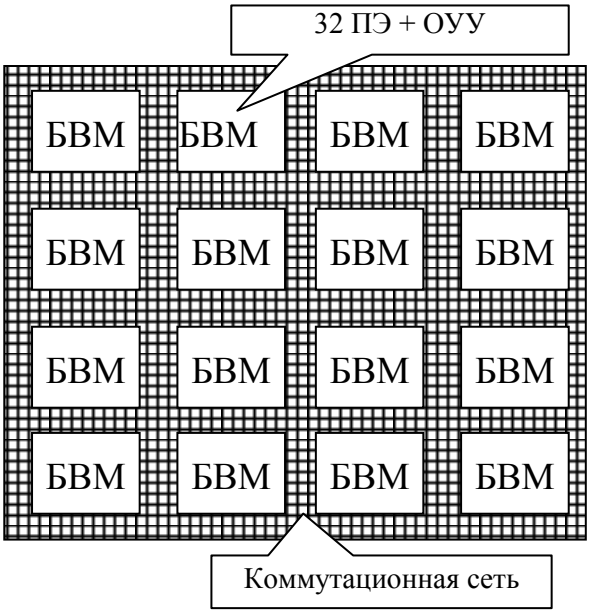

Рис.2. Комплексирование БВМ внутри кристалла

3. Блочное умножение матриц

Для иллюстрации организации вычислительного процесса на Z базовых вычислительных модулях (БВМ) ПС-2000М, каждый из которых содержит N ПЭ, рассмотрим пример распараллеливания алгоритма умножения матриц размером P×P, в котором Z=3, P=12×12, N=4.

Умножение будем выполнять блочным методом. Для этого разбиваем матрицы на клетки размером 4×4, см. рис. 3.

$$\begin{pmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \end{pmatrix} \times \begin{pmatrix} B_{1,1} & B_{1,2} & B_{1,3} \\ B_{2,1} & B_{2,2} & B_{2,3} \\ B_{3,1} & B_{3,2} & B_{3,3} \end{pmatrix} = \begin{pmatrix} C_{1,1} & C_{1,2} & C_{1,3} \\ C_{2,1} & C_{2,2} & C_{2,3} \\ C_{3,1} & C_{3,2} & C_{3,3} \end{pmatrix}$$

Рис. 3. Блочноее представление матриц для N=4, P=12×12.

Для решения задачи необходимо вычислить девять подматриц:

$$\begin{aligned} C_{1,1} &= A_{1,1} \times B_{1,1} + A_{1,2} \times B_{2,1} + A_{1,3} \times B_{3,1} \\ C_{1,2} &= A_{1,1} \times B_{1,2} + A_{1,2} \times B_{2,2} + A_{1,3} \times B_{3,2} \\ C_{1,3} &= A_{1,1} \times B_{1,3} + A_{1,2} \times B_{2,3} + A_{1,3} \times B_{3,3} \end{aligned} \quad (1)$$

$$\begin{aligned} C_{2,1} &= A_{2,1} \times B_{1,1} + A_{2,2} \times B_{2,1} + A_{2,3} \times B_{3,1} \\ C_{2,2} &= A_{2,1} \times B_{1,2} + A_{2,2} \times B_{2,2} + A_{2,3} \times B_{3,2} \\ C_{2,3} &= A_{2,1} \times B_{1,3} + A_{2,2} \times B_{2,3} + A_{2,3} \times B_{3,3} \end{aligned} \quad (2)$$

$$\begin{aligned} C_{3,1} &= A_{3,1} \times B_{1,1} + A_{3,2} \times B_{2,1} + A_{3,3} \times B_{3,1} \\ C_{3,2} &= A_{3,1} \times B_{1,2} + A_{3,2} \times B_{2,2} + A_{3,3} \times B_{3,2} \\ C_{3,3} &= A_{3,1} \times B_{1,3} + A_{3,2} \times B_{2,3} + A_{3,3} \times B_{3,3} \end{aligned} \quad (3)$$

В $БВМ_1$ будем вычислять выражение (1), последовательно подматрицы $C_{1,1}, C_{1,2}, C_{1,3}$; в $БВМ_2$ выражение (2), последовательно подматрицы $C_{2,1}, C_{2,2}, C_{2,3}$; в $БВМ_3$ выражение (3), последовательно подматрицы $C_{3,1}, C_{3,2}, C_{3,3}$. Для этого, см. рис.4, в память M_1 $БВМ_1$ по каналу D_1 последовательно в две выделенные области памяти в режиме «пинг-понга» вводятся подматрицы $A_{1,1}, A_{1,2}, A_{1,3}$, в память M_2 $БВМ_2$ по каналу D_2 подматрицы $A_{2,1}, A_{2,2}, A_{2,3}$, в память M_3 $БВМ_3$ по каналу D_3 подматрицы $A_{2,1}, A_{2,2}, A_{2,3}$. По каналу X последовательно в две выделенные области памяти в режиме «пинг-понга» вводятся подматрицы $B_{1,1}$, $B_{1,2}$, $B_{1,3}$, $B_{2,1}$, $B_{2,2}$, $B_{2,3}$, $B_{3,1}$, $B_{3,2}$, $B_{3,3}$, каждая из которых одновременно записывается в память H_1, H_2, H_3 трех БВМ.

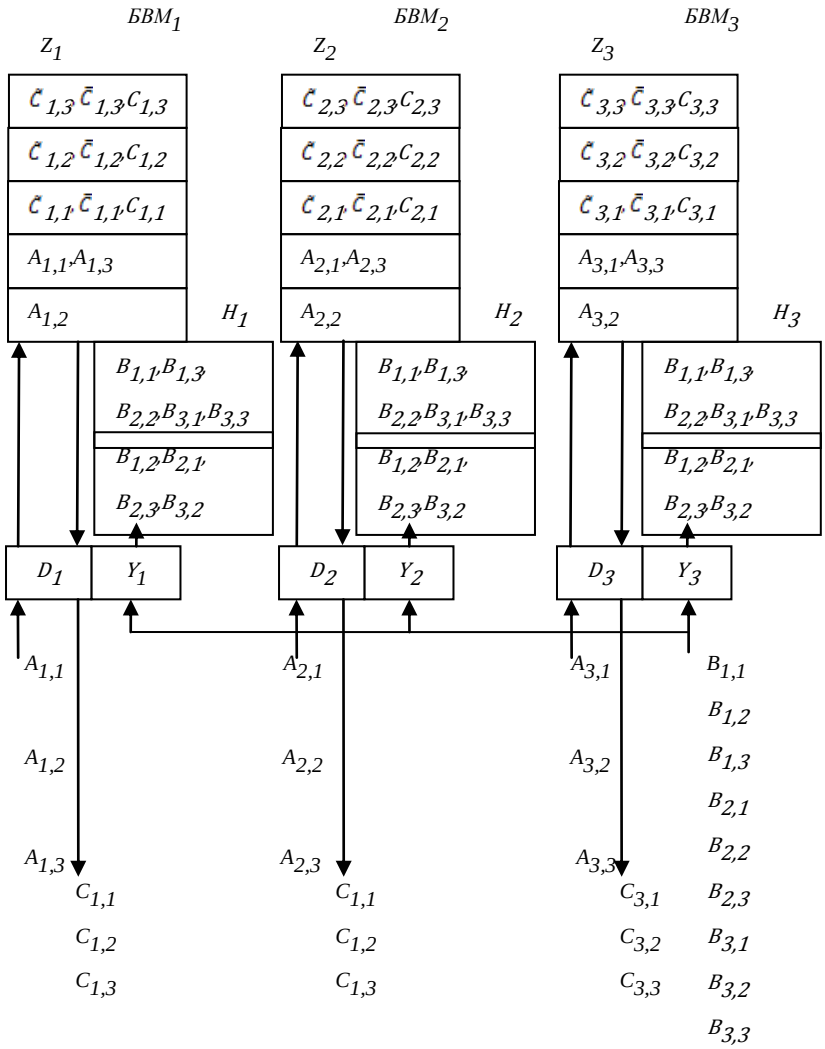

Рис. 4. Распределение памяти для блочного умножения матриц на трех БВМ

Задача решается за 11 шагов, которые приведены в табл.1.

Таблица 1. Вычислительный процесс блочного умножения матриц

№		$БВМ_1$	$БВМ_2$	$БВМ_3$
1	Ввод	$D_1 = A_{1,1}, X_1 = B_{1,1}$	$D_2 = A_{2,1}, X_2 = B_{1,1}$	$D_3 = A_{3,1}, X_2 = B_{1,1}$
2	Счет	$\tilde{C}_{1,1} = A_{1,1} \times B_{1,1}$	$\tilde{C}_{2,1} = A_{2,1} \times B_{1,1}$	$\tilde{C}_{3,1} = A_{3,1} \times B_{1,1}$
	Ввод	$X_1 = B_{1,2}$	$X_2 = B_{1,2}$	$X_3 = B_{1,2}$
3	Счет	$\tilde{C}_{1,2} = A_{1,1} \times B_{1,2}$	$\tilde{C}_{2,2} = A_{2,1} \times B_{1,2}$	$\tilde{C}_{3,2} = A_{3,1} \times B_{1,2}$
	Ввод	$X_1 = B_{1,3}$	$X_2 = B_{1,3}$	$X_3 = B_{1,3}$
4	Счет	$\tilde{C}_{1,3} = A_{1,1} \times B_{1,3}$	$\tilde{C}_{2,3} = A_{2,1} \times B_{1,3}$	$\tilde{C}_{3,3} = A_{3,1} \times B_{1,3}$
	Ввод	$X_1 = B_{2,1}, D_1 = A_{1,2}$	$X_2 = B_{2,1}, D_2 = A_{2,2}$	$X_3 = B_{2,1}, D_3 = A_{3,2}$
5	Счет	$\bar{C}_{1,1} = \tilde{C}_{1,1} + A_{1,2} \times B_{2,1}$	$\bar{C}_{2,1} = \tilde{C}_{2,1} + A_{2,2} \times B_{2,1}$	$\bar{C}_{3,1} = \tilde{C}_{3,1} + A_{3,2} \times B_{2,1}$
	Ввод	$X_1 = B_{2,2}$	$X_2 = B_{2,2}$	$X_3 = B_{2,2}$

6	Счет	$\tilde{C}_{1,2} = \tilde{C}_{1,2} + A_{1,2} \times B_{2,2}$	$\tilde{C}_{2,2} = \tilde{C}_{2,2} + A_{2,2} \times B_{2,2}$	$\tilde{C}_{3,2} = \tilde{C}_{3,2} + A_{3,2} \times B_{2,2}$
	Ввод	$X_1 = B_{2,3}$	$X_2 = B_{2,3}$	$X_3 = B_{2,3}$
7	Счет	$\tilde{C}_{1,3} = \tilde{C}_{1,3} + A_{1,2} \times B_{2,3}$	$\tilde{C}_{2,3} = \tilde{C}_{2,3} + A_{2,2} \times B_{2,3}$	$\tilde{C}_{3,3} = \tilde{C}_{3,3} + A_{3,2} \times B_{2,3}$
	Ввод	$X_1 = B_{3,1}$, $D_1 = A_{1,3}$	$X_2 = B_{3,1}$, $D_2 = A_{2,3}$	$X_3 = B_{3,1}$, $D_3 = A_{3,3}$
8	Счет	$C_{1,1} = \tilde{C}_{1,1} + A_{1,3} \times B_{3,1}$	$C_{2,1} = \tilde{C}_{2,1} + A_{2,3} \times B_{3,1}$	$C_{3,1} = \tilde{C}_{3,1} + A_{3,3} \times B_{3,1}$
	Ввод	$X_1 = B_{3,2}$	$X_2 = B_{3,2}$	$X_3 = B_{3,2}$
9	Счет	$C_{1,2} = \tilde{C}_{1,2} + A_{1,3} \times B_{3,2}$	$C_{2,2} = \tilde{C}_{2,2} + A_{2,3} \times B_{3,2}$	$C_{3,2} = \tilde{C}_{3,2} + A_{3,3} \times B_{3,2}$
	Ввод	$X_1 = B_{3,3}$	$X_2 = B_{3,3}$	$X_3 = B_{3,3}$
	Вывод	$D_1 = C_{1,1}$	$D_2 = C_{2,1}$	$D_3 = C_{3,1}$
10	Счет	$C_{1,3} = \tilde{C}_{1,3} + A_{1,3} \times B_{3,3}$	$C_{2,3} = \tilde{C}_{2,3} + A_{2,3} \times B_{3,3}$	$C_{3,3} = \tilde{C}_{3,3} + A_{3,3} \times B_{3,3}$
	Вывод	$D_1 = C_{1,2}$	$D_2 = C_{2,2}$	$D_3 = C_{3,2}$
11	Вывод	$D_1 = C_{1,2}$	$D_2 = C_{2,2}$	$D_3 = C_{3,2}$

Время решения задачи ускоряется пропорционально числу БВМ

$$T=P/Z\times t+2\alpha, \text{ где}$$

T – время умножения двух матриц размером P×P в Z БВМ;
t – время умножения двух подматриц размером N×N в одном БВМ;
α- время ввода/вывода подматрицы размером N×N.

Литература

1. *Затуливетер Ю.С., Фищенко Е.А., Артамонов С.Е., Козлов В.А.* Элементы стратегии и архитектурные предпосылки опережения в области однокристальных многопроцессорных компьютеров с массовым параллелизмом // Информационные технологии. 2014. №2. Приложение. С. 1-32.