

© 2012 г. ЛИСЕНКОВ М.А., аспирант,  
ВНУКОВ А.А., к.т.н., доцент  
(Московский Государственный Институт Электроники и Математики (МИЭМ  
НИУ ВШЭ), Москва)

## **ПОСТРОЕНИЕ ПАРАЛЛЕЛЬНОГО АЛГОРИТМА ОБРАБОТКИ ДАНЫХ ДЛЯ ЭФФЕКТИВНОГО РЕШЕНИЯ ЗАДАЧИ ПРЕОБРАЗОВАНИЯ КООРДИНАТ С ИСПОЛЬЗОВАНИЕМ ПЕРЕМНОЖЕНИЯ МАТРИЦ**

В статье рассматриваются вопросы параллельных вычислений с использованием высокопроизводительных систолических полей. Ставится задача, которая предполагает разработку эффективного параллельного алгоритма перемножения матриц. Предлагается вариант построения спроектированной вычислительной архитектуры с использованием множества процессорных элементов.

**PARALLEL DATA PROCESSING ALGORITHM FOR COORDINATE TRANSFORMATION TASKS AND MATRIX MULTIPLICATION /** M.A.Lisenkov, Moscow State Institute of Electronics and Mathematics NRU HSE (Technical University), B. Trekhsvjatitel'skiy, Moscow, 109028, Russia, E-mail: lisenkov.maxim@gmail.com; A.A.Vnukov, Moscow State Institute of Electronics and Mathematics NRU HSE (Technical University), B. Trekhsvjatitel'skiy, Moscow, 109028, Russia, E-mail: awnkw@mail.ru. In this article, we review the questions of parallel computing. In particular systolic architectures which increase performance in labor-intensive computing. And Systolic Multipliers which used to accelerate calculations in matrix multiplication.

### **1. Введение**

Разработка вычислительных моделей алгоритмов аппаратно-программных комплексов актуальна и сегодня. Реализация режима масштаба реального времени для многих технических систем на современном этапе невозможна без разработки специальных специализированных устройств, значительно ускоряющих процесс вычислений (спецвычислителей).

Большинство задач робототехники ряда дисциплин, среди которых механика и приводы роботов, информационные, электронные и вычислительные устройства роботов, методы искусственного интеллекта, управление сложными мехатронными системами, обладают высокой вычислительной сложностью и требуют соответствующих высокопроизводительных вычислительных ресурсов. Одним из решений этой проблемы является распараллеливание вычислений, как на аппаратном, так и на программном уровне. Для пространственных механизмов (например,

промышленный робот-манипулятор) эффективными методами определения функции положения являются векторный метод и метод преобразования координат.

Для реализации такого программного обеспечения требуются системы с высоким быстродействием, а возможности современных вычислительных систем часто ниже требующихся.

## 2. Основная задача

Программная система последовательно должна решать две динамические задачи для 6-ти осевого манипулятора. Прямая задача по заданному закону движения определяет обобщённые силы звеньев, обратная задача – по известным обобщённым силам – закон движения в заданных промежутках времени. Цель: ускорить выполнение алгоритма программы. Разработка эффективного параллельного алгоритма обработки данных, в частности для решения задачи перемножения матриц.

## 3. Вычислительная архитектура и процессорные элементы

Цифровые сигнальные процессоры (ЦСП) и транспьютеры могут быть использованы при практической реализации вычислительных архитектур (ВА) на СБИС (сверхбольшая интегральная схема). Минимальный структурный размер определяется технологическими нормами производства СБИС. Функциональное устройство может состоять из большого количества активных и пассивных элементов.

Рассмотрим такой класс вычислительной архитектуры, как систолическая ВА. Данная архитектура является специализированной и позволяют эффективно реализовать узкий класс алгоритмов. В процессоре систолической ВА записываются достаточно простые вычислительные модули (программы), которые в дальнейшем не изменяются. Данная ВА представляет собой совокупность процессорных элементов (ПЭ), которые имеют постоянные неизменяемые соединения между собой и, таким образом, обеспечивается вычислительная сеть.

Обозначим ПЭ, входящий в систолическое поле.

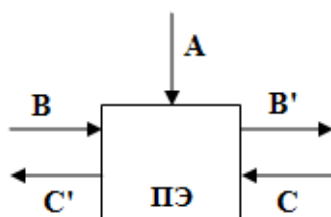


Рис. 1. ПЭ систолического поля для вычисления  $C=A * B$ , где  $A$  – матрица,  $B$  – вектор.

Для определения ПЭ в процессорной сети необходимо задать все входные и выходные потоки данных с указаниями направления. Для систолического поля умножение матрицы на вектор должно быть 3 входа и 2 выхода, в соответствии с количеством входов и выходов подбирается реальный процессорный элемент, который обладает таким же количеством входов и выходов и может быть использован в процессорном поле в качестве базового строительного элемента.

Для каждого ПЭ рассматриваемого поля необходимо задать процессорную функцию. Таким образом, процессорный элемент является определенным с точки зрения его

топологических характеристик, входных и выходных потоков данных (входов и выходов ПЭ) и процессорных функций, которые в данном случае являются одинаковыми для всех ПЭ в систолическом поле. Один поток команд – это команды, которые поступают разом на все ПЭ. Геометрия процессорной сети составляет глобальную модель процессорного поля. Локальная модель (F) процессорного поля определяется процессорной функцией для каждого ПЭ.

Таким образом, топологические особенности ПЭ, входные и выходные потоки данных для каждого ПЭ, процессорная функция – эта совокупность является заданием глобальной и локальной моделей процессорного поля. Модель системы управления определяется заданием временных интервалов для выполнения операций, фаз коммуникаций и вычислений.

#### 4. Систолическое вычислительное поле и интерфейс

В общем случае, обобщенная вычисляемая модель систолического поля описывается тройкой (G, F, T), где G - определяет геометрию процессорной сети, т.е. представляет глобальную модель. F - определяет процессорную функцию для G (глобальной модели). T - определяет время выполнения для F (локальная модель), т.е. представляет модель управления систолическим полем. Для модели управления задаются длительности фаз коммуникации, вычислений и, иногда, длительность собственно фазы управления, во время которой на процессорное поле подается один поток команд.

Для подключения систолических полей к вычислительной или технической системе необходима интерфейсная система. Интерфейсная система рассматривается главной вычислительной системой, как внешнее устройство. Главная вычислительная система и интерфейсная система могут функционировать одновременно. Такой интерфейс поддерживает работу одного или нескольких систолических полей и обеспечивает минимальные потери времени, не связанные с вычислительным процессом. Для организации правильного пространственно-временного потока данных на систолическое поле нужно иметь:

- 1) Специальным образом организованную память;
- 2) УУ такой памятью, выполняющее как операции ввода/вывода данных в/из систолического поля, так и синхронизацию работы процессоров сети по фазам коммуникации.

Рассмотрим использование ленточной магазинной памяти в качестве составного устройства специализированного интерфейса системы. Пример использования специальной ленточной магазинной памяти для выполнения матричной операции  $A*B=C$  (рис. 2).

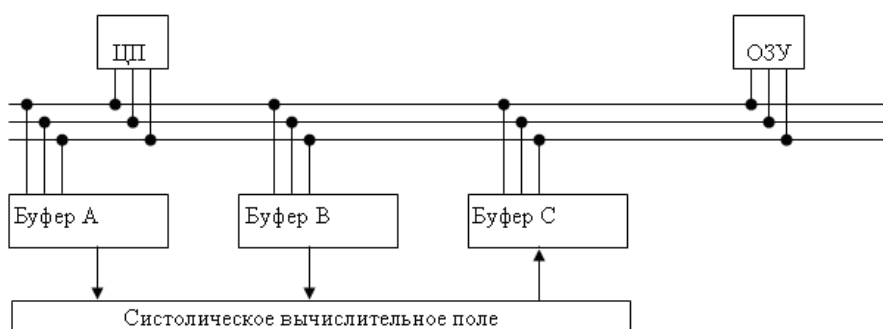


Рис. 2. Подключение вычислительного поля.

Для эффективного обмена информацией в систолической структуре ПЭ может быть использована буферная ленточная память магазинного типа. Память состоит из входного буфера с исходными данными и выходного буфера с полученными результатами их обработки.

## 5. Задача перемножения матриц

Рассмотрим простую операцию:  $C = A * B$ .

С помощью систолического процессорного поля может быть вычислена следующая задача вида:

$$(1) \quad A_{(4 \times 4)} \cdot B_{(4 \times 1)} = C_{(4 \times 1)} \Rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \times \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}$$

$A$  - ленточная пятидиагональная матрица (имеет 5 ненулевых диагоналей подряд);  $B$  - вектор исходных данных;  $C$  - вектор полученных результатов.

Рассмотрим буфер элементов матрицы  $A$ . Если емкость ленточного буфера памяти достаточна, то загружаем всю матрицу в буферную память. Если же недостаточно, то в процессе обработки данных в систолическом поле, интерфейс микропроцессора осуществляет подкачку данных в буферной памяти. В буферной памяти каждый такт - это фактически сдвиг данных на одну ячейку в направлении сверху вниз во время фазы коммуникации.

Проанализируем организацию ленточной магазинной памяти для вектора  $B$  (исходных данных) и вектора  $C$  - результатов. Управляющий интерфейс микропроцессора переписывает данные из основной памяти в буфере элементов. Во время фазы коммуникации (внешней передачи информации) данные поступают из ленточной памяти в систолическое поле или одновременно выводятся из него в ленточную память.

Во время фазы вычислений в ленточной памяти возможно следующее:

- 1) содержимое ячеек памяти не меняется;
- 2) происходит обмен с ОЗУ более высокого уровня (при этом производится дополнение данными или считывание полученных результатов).

Синхронизация входных и выходных потоков по времени показывается относительно их моментов времени ввода и вывода в/из систолического поля. Элементы входных потоков вводятся один раз в процессорную сеть. При этом элементы вектора  $B$  используется в процессорной сети многократно. Выходной поток результатов движется в противоположном направлении по отношению к потокам элементов вектора  $B$ .

## 6. Разработка систолического вычислителя

В данном пункте рассматривается задача разработки специализированного вычислителя для умножения двух ленточных матриц, используемого при преобразовании координат. Цель работы - ускорить выполнение алгоритма программы.

Зададим операцию  $C = A * B$ , где матрицы  $(4 \times 4)$  имеют вид:

$$(2) \quad \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ 0 & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ 0 & b_{32} & b_{33} & b_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Построим для каждой матрицы матрицу, которая учитывает участие каждого элемента в вычислениях. Матрицы А и В – это пятидиагональная ленточная матрица; С - шестидиагональная ленточная матрица.

В качестве поддержки построения систолического поля сформулируем матрицу участия в вычислениях для каждого сомножителя  $a_{ij}, b_{ij}$ , а также матрицу с количеством умножений для каждого элемента результата  $c_{ij}$ . Элемент  $a_{11}$  вводится один раз в процессорное поле, используется в четырех вычислениях и прекращает свое существование. Элемент  $b_{11}$  вводится один раз в процессорное поле, используется в двух вычислениях и прекращает свое существование. Элемент  $c_{11}$  вводится один раз, используется два раза в вычислениях (операциях умножения) и один раз выводится. Вспомогательные матрицы создаются программно.

Выбираем локальную модель, задаем входные и выходные потоки и их направления:

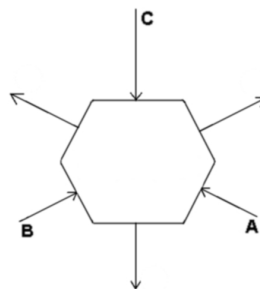


Рис. 3. Локальная модель ПЭ и направление потоков данных.

*Шаг 1.* Задали направления для трех потоков данных  $a_{11}, b_{11}, c_{11}$ , отметили точками начало вычислений, ПЭ, в которых они используются.

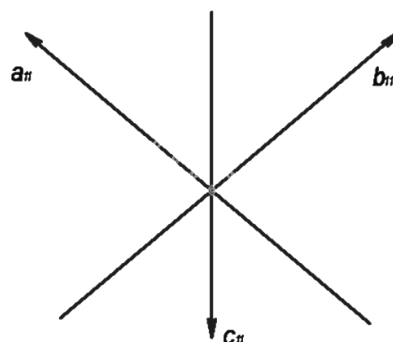


Рис. 4. Шаг 1.

*Шаг 2.* Через построенные точки на прямой  $a_{11}$  и  $c_{11}$  проводим прямые параллельные  $b_{11}$ . По этим прямым будет перемещаться поток данных с элементами ленточной матрицы  $B$ .

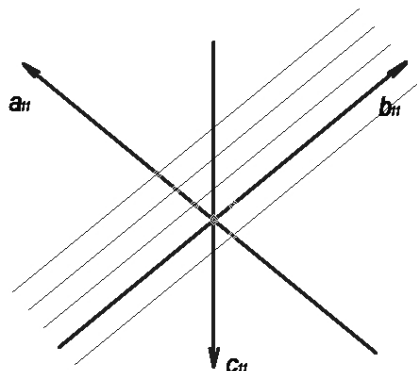


Рис. 5. Шаг 2.

*Шаг 3.* Проводится еще одна прямая через точку прямой  $b_{11}$  параллельно  $a_{11}$ .

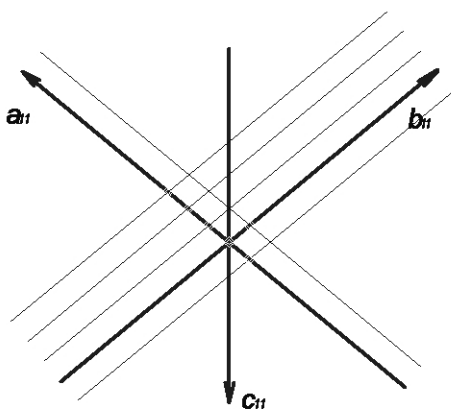


Рис. 6. Шаг 3.

*Шаг 4.* Теперь надо разработать структуры двух входных и одного выходного потока данных.

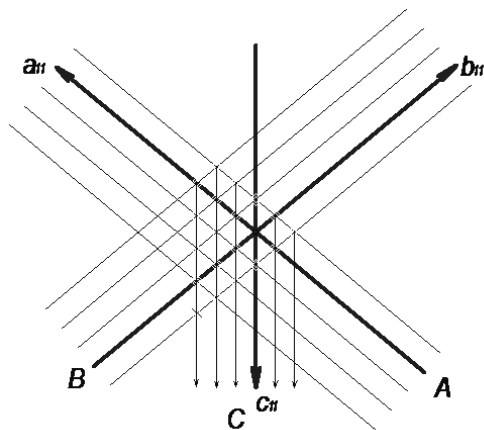


Рис. 7. Шаг 4.

Пусть в выделенной точке (рис. 8) начинаются вычисления.

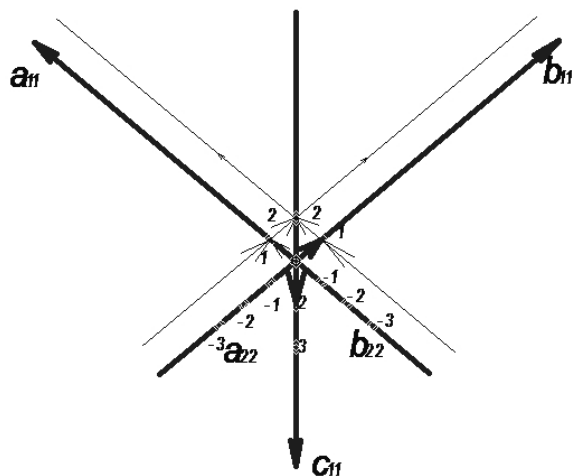


Рис. 8. Диаграмма следования элементов матриц в потоках данных.

Произведем замену узловых точек на последнем рисунке шестиугольными ПЭ. Спроектированное систолическое процессорное поле выполняет матричную операцию:  $C=A*B$  с наложенными ограничениями на размерность и внутреннюю структуру матриц операндов (рис. 9).

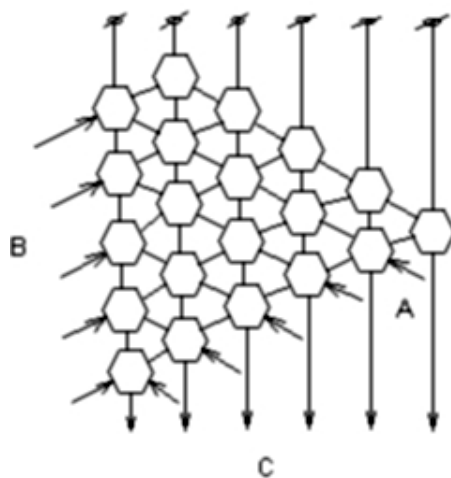


Рис. 9. Процессорное поле.

Правильность построения процессорного поля и можно проверить по вспомогательным матрицам. Время окончания работы систолического поля (время получения последнего элемента матрицы  $C$ ) составляет  $t=15$  тактов коммуникации и вычислений. Время получения элемента  $c_{ij}$  составляет:  $t(c_{ij})=(5+i)+2(j-1)$ .

Ускорение работы последовательного алгоритма осуществляется с помощью создания специального высокопроизводительного вычислительного блока. На основе разработанной структуры можно собрать заказную СБИС, в которой будет реализовано процессорное поле с 20 ПЭ с постоянными соединениями, как показано на рисунке 9.

При каждом перемножение матриц будем использовать разработанное систолическое поле, которое будет являться спецвычислителем, обеспечивающим параллельную обработку данных вместо последовательного алгоритма выполнения программы. Таким образом, модернизируем последовательный алгоритм умножения матриц.

## 7. Заключение

Стоит отметить, что разработанная модель СБИС (соответственно и алгоритм параллельной обработки данных) является универсальной и может использоваться не только в качестве блока робота-манипулятора, также может быть встроена в различные технические устройства для ускорения вычислений. Элемент перемножения матриц встречается не только в сфере робототехники, в частности в 3D-графике и видеообработке. Разработанное программное и аппаратное обеспечение является крайне полезным для обработки больших массивов данных, позволяет существенно увеличить быстродействие.

## СПИСОК ЛИТЕРАТУРЫ

1. *Гергель В.П., Стронгин Р.Г.* Принципы построения параллельных вычислительных систем // Учебное пособие. — Нижний Новгород, 2003.
2. *Фролов К. В.* Механика промышленных роботов. Том 1. Кинематика и динамика. Под редакцией К. В. Фролова, Е. И. Воробьева. — М.: “Высшая школа”, 1989.
3. *А.А.Внуков, М.В.Семиков, Д.А.Тренин* Разработка параллельного алгоритма для системы динамического управления манипуляционным роботом // Вестник Российского университета дружбы народов. Научный журнал. Серия «Инженерные исследования» (информационные технологии, управление и экономика), Издательство РУДН. — М.: 2008, №5, С. 107-122.
4. *Шахинпур М.* Курс робототехники. Под редакцией С. Л. Зинкевича. — М.: “Мир”, 1990.
5. *Попов Е. П.* Управление роботами-манипуляторами. — Техника, кибернетика, 1974.
6. *Шпаковский Г. И.* Организация параллельных ЭВМ и суперскалярных процессоров. — Минск, 1996, С. 219-256.