

НЕКОТОРЫЕ СПОСОБЫ ПОСТРОЕНИЯ ОДНОРОДНЫХ ВЫЧИСЛИТЕЛЬНЫХ МАШИН

Т.В. Кузовкина

(Москва)

Стремление повысить эффективность ВМ и расширить круг решаемых ими задач приводит к усложнению их структуры. При этом увеличиваются габариты и стоимость и снижается надежность ВМ. Кроме того, эффективность работы ВМ снижается из-за различной загруженности блоков в зависимости от типа решаемой задачи. С этой точки зрения, машины с перестраиваемой структурой будут более эффективны.

Наибольший интерес представляют итеративные вычислительные машины. Они состоят из большого числа одинаковых модулей, соединенных одинаковым образом между собой. Благодаря способности каждого модуля (или какой-либо группы) выполнять различные операции (запоминание, арифметические и логические), такие машины смогут более гибко приспосабливаться к решению различных задач. Возможность одновременно выполнять несколько программ приведет к повышению быстродействия. Высокая технологичность позволит увеличить емкость запоминаемой информации.

Сама идея использования распределенной памяти, арифметики и логики не является новой. Например, в первых автоматических ВМ Марс-1 и ЕНИАК функция внутреннего ЗУ еще не была специализирована и конструктивно отделена от других функций, выполняемых суммирующими устройствами. Такое разделение потребовало огромной затраты оборудования (18000 ламп) при сравнительно низкой емкости ($20 \cdot 10^4$ бит).

Отделение функции запоминания от вычислительных и управляющих позволило конструктивно упростить ЗУ настолько, что

машины с большим внутренним ЗУ стали технически и экономически осуществимыми.

С развитием технологии твердых схем количество оборудования не является ограничивающим фактором, более важна простота организации и соединений блоков ВМ. Поэтому метод распределенной памяти и логики сейчас привлекает внимание специалистов. Так, Хенни [1] исследовал вопросы анализа и синтеза одномерных, двумерных и многомерных сетей, состоящих из комбинационных и последовательностных элементов; Анжер [2] описал однородную ВМ, предназначенную для распознавания образов; Холланд [3,4,5] опубликовал целый ряд работ по однородным вычислительным машинам (ОВМ) и т.д. Все эти работы отличает большое разнообразие в выборе принципа организации и последовательности работы машины.

Например, Холланд [5] организовал работу ОВМ таким образом, чтобы поведение её было адаптивным. Его машина представляет двумерную решетку ячеек с одинаковой логической структурой. Каждая ячейка соединена с 4 соседями. Построение пути происходит при передаче активного состояния от соседа к соседу и является существенно последовательным процессом. Применение относительной адресации, сложность размещения данных делает реализацию машины Холланда затруднительной. Среди работ, продолжающих работы Холланда, следует отметить работы Комфорта [6] и Гонзалеса [7]. Комфорт использовал матрицу модулей только для хранения программы и данных, а для выполнения арифметических и логических операций служат специальные арифметические блоки (по одному на каждый столбец модулей в матрице). Благодаря этому ему удалось сократить количество оборудования в 5 раз и существенно упростить программирование.

Гонзалес, используя идею однородности, строит машину в виде слоистой структуры (3 платы), благодаря чему совмещаются во времени некоторые операции и связь между модулями появляется только в процессе определения пути и занимает меньше времени, чем в машине Холланда.

В данной работе предлагается однородная ВМ, где использован принцип слоистости однородной структуры, и связь между ячейками осуществляется в процессе выполнения программы. Благодаря упрощению процедуры построения пути, удалось сократить количество слоев до двух.

Итак, машина состоит из 2 слоев (рис.1), каждый из которых

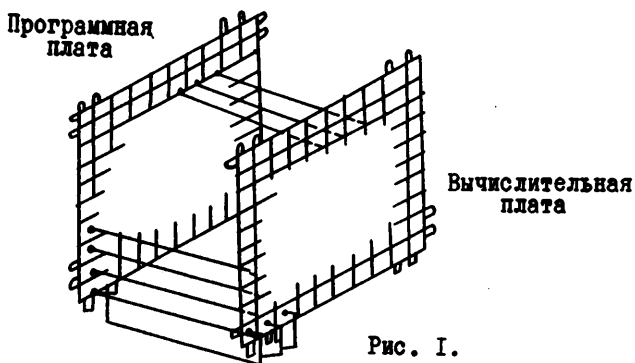


Рис. 1.

содержит $m \cdot n$ одинаковых ячеек (модулей). На одной плате (программной) записывается программа, исходные и промежуточные данные, на другой (вычислительной) выполняются все команды. Ячейки на программной плате можно сделать проще, чем на вычислительной: им не требуется накопитель. В остальном они одинаковы и содержат накопитель, регистр, два дешифратора адресов (по горизонтали и вертикали), дешифратор КОП, систему клапанов (рис.2). В плоскости каждой платы происходят не связанные между собой непрерывные горизонтальные и вертикальные шины (информационные). К ним по определенным командам от дешифраторов подключаются ячейки для выдачи и приема информации. Таким образом, связи у всех ячеек одинаковы и не зависят от их положения на плате. Каждый модуль программной платы связан с соответствующим модулем на вычислительной плате тоже непрерывными линиями. Такое соединение позволяет передавать данные с одной платы на другую одновременно из любого числа ячеек, что значительно повышает быстродействие решения задачи.

Используемая команда (Пр.КОП. $A_1 A_2 A_3$) относится к командам со свободным порядком. Для упрощения системы команд можно условиться, что операция выполняется в ячейках I -го операнда, а результат отсылается в соответствующую ячейку на ПП. Каждый адрес команды состоит из горизонтальной и вертикальной частей. Использование координатной адресации диктуется принятым способом установления связей между ячейками. Координатная адресация удобна, когда может возникнуть необходимость увеличения размера платы и когда в качестве операнда участвуют не-

сколько ячеек. Тогда указывается начало и конец этой совокупности ячеек. Это дает возможность производить одни и те же операции над целым массивом чисел или разместить одно слово в нескольких ячейках, благодаря чему отпадает необходимость вводить масштабные числа или нормализовать числа, что упрощает программирование.

Последовательность работы излагается для случая, когда каждый операнд занимает одну ячейку. В первый такт к горизонтальной информационной линии подключаются вертикальные дешифраторы, и из тех ячеек, где признаковый разряд в команде равен "1", на горизонтальную линию через клапаны K_{II} выдается код B_3 (вертикальный адрес наследника, т.е. ячейки, где записана следующая команда). Дешифратор той ячейки, чей адрес (B_i) совпадает с информацией B_3 , вырабатывает сигнал, который открывает клапан K_{III} , соединяющий горизонтальную и вертикальную информационные линии при подаче следующих тактов ($T_2 + T_n$), количество которых определяется разрядностью адресов ячеек ($n = 2$ для матрицы 3×3). Со следующим тактом (T_4) подключаются горизонтальные дешифраторы ($ГД$), и на информационную линию выдается I'_3 . При совпадении адресов $ГД$ посылает сигнал через схему совпадения на признаковый разряд регистра, устанавливая его в состояние "I", т.е. к работе готова ячейка, где записана следующая команда. Клапан K_{III} приводится в исходное состояние. Аналогично на этой же плате находится A_2 (угол пути образуется новый). Срабатывание $ГД$ в T_{10} означает, что найдена ячейка $П$ -го операнда и сигнал с $ДГ$ поступает на клапан K_I и вход накопителя подключается к вертикальной информационной линии. В регистре, где хранится число $П$ -го операнда, часть разрядов остаются свободными. Туда поступает информация из исходной ячейки. Информация ($КОП A_I$) пересылается в ячейку, где хранится число $П$ -го операнда. В таком виде содержимое ячейки передается в соответствующую ячейку $ВП$, где сигналом с триггера (рис.3) признаковый разряд устанавливается в состояние "I". Начинается 2-я фаза работы. После построения пути от ячейки $П$ -го операнда к ячейке I -го операнда пересылается $КОП$, а затем число I -го операнда и выполняется команда. Результат с $ВП$ передается в соответствующую ячейку $ПП$. 2-я фаза I -ой команды начинает выполняться одновременно с I -ой фазой следующей команды. Время, затрачиваемое на решение одной последовательной

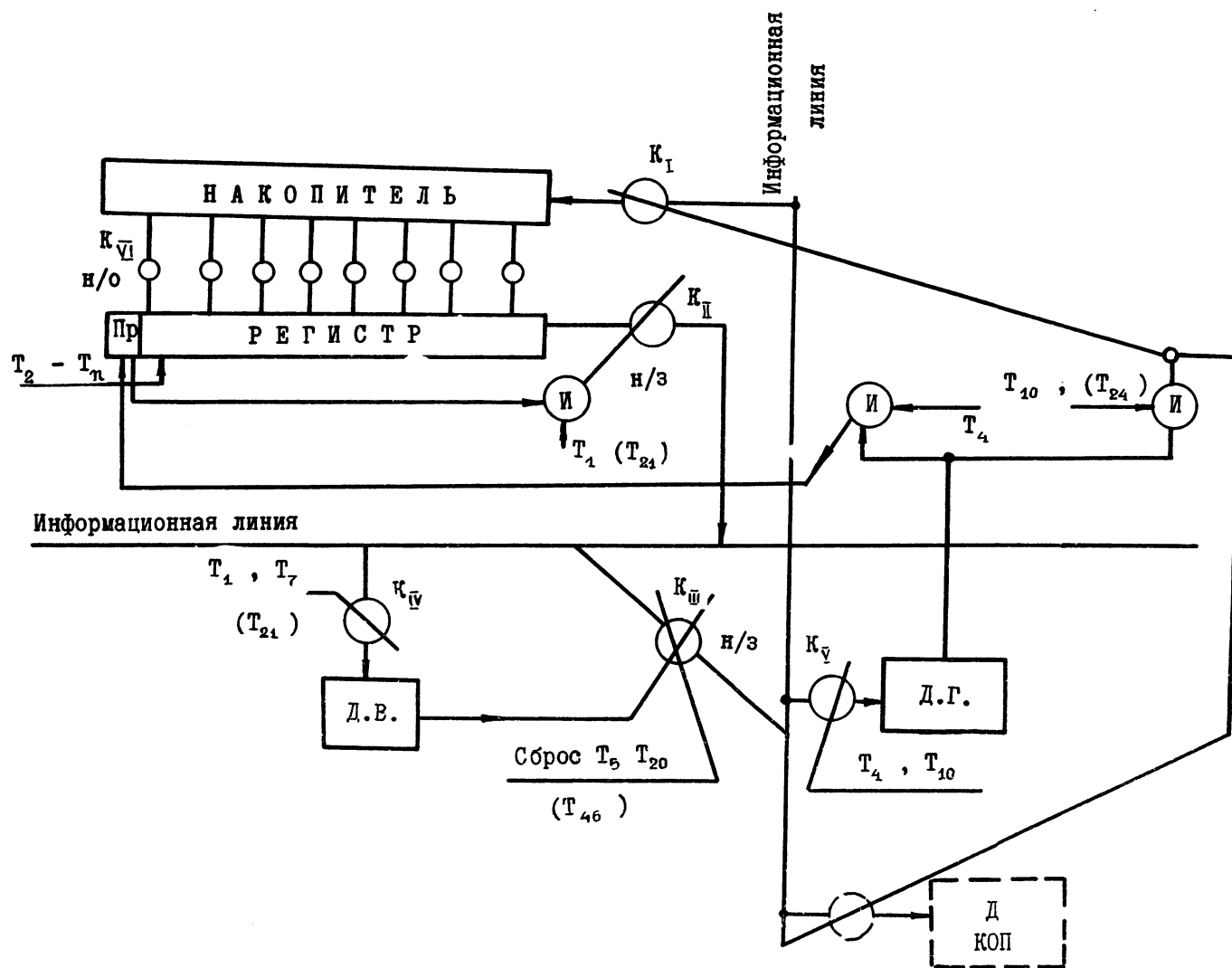


Рис. 2.

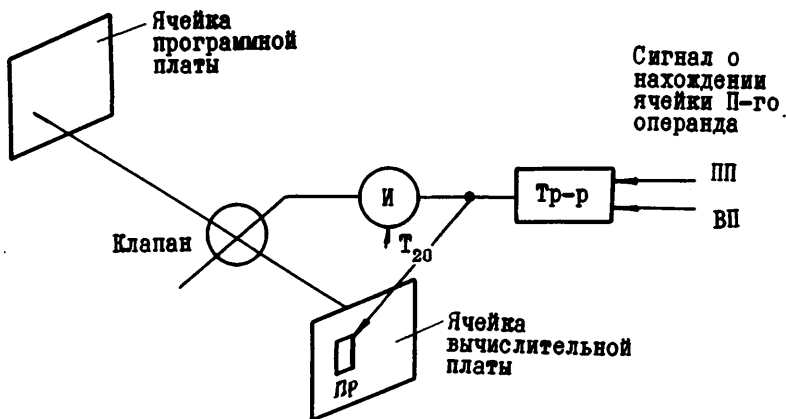


Рис. 3.

программы, будет

$$T = T_{\Phi_1 \text{ 1-ой ком}} + \sum_{k=1}^n T_{\Phi_2 k}$$

Однородная ВМ позволяет выполнять несколько программ одновременно, но 1) ячейки не должны лежать на одной горизонтали, 2) вторые операнды разных программ должны ложиться по диагоналям платы, 3) первые операнды должны лежать на разных вертикалях.

Для иллюстрации приведена программа сложения чисел:

$$x = a_1 + a_2 + a_3 + a_4 + a_5.$$

Программа на плате займет $3 \times 3 = 9$ ячеек.

1к	a_3	a_1
3к	4к	2к
a_4	a_2	a_5

№ ком	Пр.	КОП	Γ_1	B_1	Γ_2	B_2	Γ_3	B_3
1к	I	+	0I	II	II	IO	IO	II
2к	0	+	0I	II	0I	IO	IO	0I
3к	0	+	0I	II	II	0I	IO	IO
4к	0	+	0I	II	II	II	00	00

Если программы записывать на программной плате, а исходные и промежуточные данные располагать на вычислительной, то увеличивается емкость машины, но несколько ухудшаются возможности для одновременной работы нескольких программ. Операнды (I и Π) находятся одновременно на разных платах. Для этого необходимо обеспечить передачу информации A_I с программной платы на информационную линию вычислительной платы, а также передачу состояния клапана K'_{II} (рис.4), который обеспечивает "угол пути" на программной плате, на клапан соответствующей ячейки

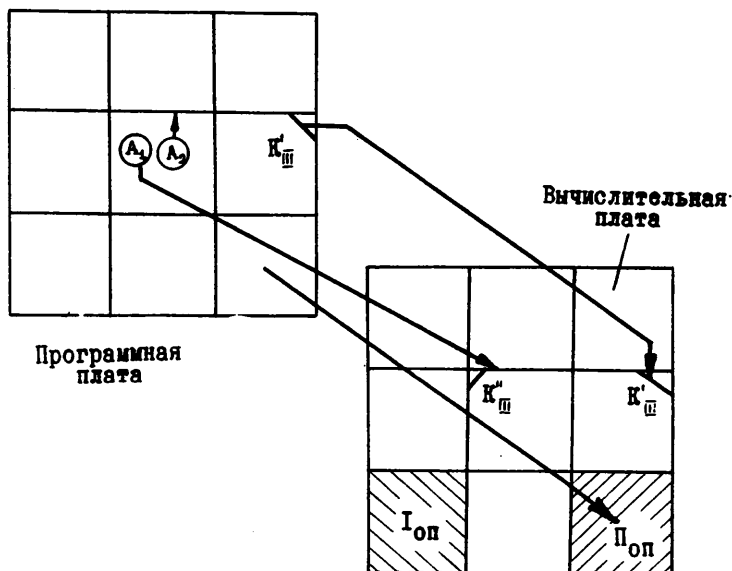


Рис. 4.

на вычислительной плате. Таким образом создается путь между первым и вторым операндами.

В заключение отметим, что вычислительные машины, построенные с использованием указанных выше принципов, будут обладать повышенным быстродействием, благодаря 1) возможности параллельной передачи чисел, 2) возможности распараллеливания задач и одновременного выполнения отдельных программ, 3) совмещению отдельных фаз работы.

Применение команд типа 2+1, которые при обычной организации машины требуют оптимального программирования, не приводит

и усложнению программирования, так как время обращения одинаково и меньше, чем в обычных ЭУ. Благодаря обработке данных массивом можно получить более эффективные алгоритмы и уменьшить время обработки, благодаря чему повышается точность. Все это способствует повышению надежности выполнения вычислительных операций.

Л и т е р а т у р а

1. F.C. Hennie. Iterative Arrays of Logical Circuits. G.Wiley & Sons, Inc., New York, 1961.
2. S.H. Unger. A Computer Oriented towards spatial problems. - Proc. IRE, 1958, vol. 46, Oct.
3. G.H. Holland. A universal computer capable of executing an arbitrary number of programs simultaneously. - Proc. EGCC, 1959, Dec., p. 1-3.
4. G.H. Holland. Iterative circuits computers. - Proc. of WGCC, 1960, New York, WGCC, San Francisco. Calif., May, p. 3-5.
5. G.H. Holland. Outline for a logical theory of adaptive systems. - Journal of ACM, 1962, vol. 9, N 3.
6. W.T. Comfort. Modified Holland machine. - AFIPS Conf., Proc., 1963, vol. 24.
7. R. Gonzalez. Multilayer Iterative Circuit Computers. - IRE Trans., 1963, vol. EC-12, N 6.