

УДК 681.142.353

К РЕАЛИЗАЦИИ МНОГОРЕГИСТРОВЫХ
ПЕРИОДИЧЕСКИ ОПРЕДЕЛЕННЫХ ПРЕОБРАЗОВАНИЙ
В ОДНОЙ АБСТРАКТНОЙ МОДЕЛИ ВЫЧИСЛИТЕЛЬНОЙ СРЕДЫ

В.И. Кекелия, Г.Е. Цейтлин

В настоящее время развивается направление, начатое работами В.М. Глушкова [1,2] по теории микропрограммных алгебр, связанное с формализацией проблем, возникающих при проектировании электронно-вычислительных машин (ЭВМ). Специфика этих задач приводит к изучению однородных структур (бесконечные регистры) [1] и различных преобразований на них. В частности, важным является класс многорегистровых периодически определенных преобразований^{*}, которые используются при определении абстрактной модели машины как композиции операционного и управляющего автоматов [1]. С другой стороны, интенсивно развиваются методы проектирования вычислительных устройств, связанные с применением вычислительных сред (ВС). В работах [3-5] подчеркиваются такие важные свойства ВС, как: 1) программная настройка, 2) принцип близкодействия, 3) однородность.

Перечисленные свойства ВС позволяют надеяться, что их можно успешно применить для решения задач алгоритмического этапа

^{*} В дальнейшем, для краткости, выражение "периодически определенное" будет опускаться.

проектирования ЭВМ [2]. Настоящая статья посвящена разработке методов реализации многорегистровых преобразований в одной абстрактной модели ВС, в терминах которых, в частности, удалось описать микропрограммы основных арифметических операций.

1. В работах [6,7] подробно изучаются алгебры преобразований на бесконечном регистре и ряд свойств, связанных с их структурой. Рассмотрим одно обобщение понятия преобразования на регистре.

Пусть $X^1, X^2, \dots, X^n$ - перечень бесконечных в обе стороны двоичных регистров. Рассмотрим последовательность целых чисел $K_1, K_2, \dots, K_n$ (коэффициентов) и базовую функцию:

$f(\tau_0^1, \tau_1^1, \dots, \tau_{K_1}^1, \tau_0^2, \tau_1^2, \dots, \tau_{K_2}^2, \dots, \tau_0^n, \tau_1^n, \dots, \tau_{K_n}^n); z_j > 0$, у которой пары аргументов $(\tau_0^1, \tau_{K_1}^1), (\tau_0^2, \tau_{K_2}^2), \dots, (\tau_0^n, \tau_{K_n}^n)$ не фиктивны.

Пусть, далее, $\alpha^j = \{\alpha_{-1}^j, \alpha_0^j, \alpha_1^j, \dots\}$ - состояние регистра X^j , где $j = 1, 2, \dots, n$. Новое состояние $\beta^m = \{\beta_{-1}^m, \beta_0^m, \beta_1^m, \dots\}$ регистра X^m является результатом преобразования $F_{K_1, K_2, \dots, K_n; f}^{z_1, z_2, \dots, z_n}$ над регистрами $X^1, X^2, \dots, X^n$, если

$$\beta_i = f(\alpha_{i+K_1}^1, \alpha_{i+K_1+1}^1, \dots, \alpha_{i+K_1+K_2}^1, \dots, \alpha_{i+K_1+K_n}^1, \alpha_{i+K_1+K_2}^2, \dots, \alpha_{i+K_1+K_n}^n, \alpha_{i+K_1+K_2}^n, \dots, \alpha_{i+K_1+K_n}^n) \quad (-\infty < i < \infty)$$

В частности, m может принадлежать множеству $\{1, 2, \dots, n\}$. К числу многорегистровых относятся следующие преобразования.

1.1. Сдвиг $\mathcal{L}_{K; f}(X^j) / \mathcal{L}_K^j$, где $f(\tau_0) = \tau_0$. Коэффициент K - число разрядов, на которое осуществляется сдвиг на регистре. В зависимости от того, является ли число K положительным или отрицательным, сдвиг называется правосторонним или левосторонним, соответственно.

1.2. Инверсия переменной $\overline{7}_{0; f}^j(X^j) / \overline{7}^j$, где $f(\tau_0) = \overline{\tau_0}$.

1.3. Установка регистра X^j в нулевое (единичное) состояние $0^j(1^j)$ задается базовой функцией $f=0$ (соответственно $f=1$).

1.4. Логическое сложение

$$\vee_{0,0; f}^m(X^j, X^j); / \vee^m$$

где $f(\tau_0^j, \tau_0^{j'}) = \tau_0^j \vee \tau_0^{j'}$. (Здесь и далее $j \neq j'$; m - номер регистра, в который записывается результат преобразования).

1.5. Логическое умножение

$$\&_{0,0; f}^m(X^j, X^{j'}); / \&^m$$

где $f(\tau_0^j, \tau_0^{j'}) = \tau_0^j \& \tau_0^{j'}$.

1.6. Сложение по $\text{mod}_2 - \Sigma_{0,0; f}^m(X^j, X^{j'}) / \Sigma^m$, где $f(\tau_0^j, \tau_0^{j'}) = \tau_0^j \oplus \tau_0^{j'} \vee \tau_0^j \& \tau_0^{j'}$.

Приведем микропрограммы операции сложения $\Sigma^{j,m}$ и умножения $\&^{j,m}$ для целых положительных чисел, записанные в терминах многорегистровых преобразований. Здесь предполагается, что операнды и результат операции расположены в соответствующих регистрах слева от элементов α_N , так что последние содержат их младшие разряды. В силу того, что число разрядов, занятых операндами, конечно, результат операции и все промежуточные вычисления также занимают конечные участки в регистрах, на которых реализуется данная микропрограмма. Таким образом, существует пара чисел N и N' таких, что элементы α_i при $i > N$ или $i < N'$ фиктивны для всякого регистра, участвующего в реализации микропрограммы.

Регулярное выражение микропрограммы сложения положительных чисел, записанное в терминах многорегистровых преобразований, имеет следующий вид:

$$\left\{ \Sigma_{0,0; f}^j(X^1, X^2) \right\} = \left\{ \Sigma_{0,0; f}^j(X^1, X^2) \&_{0,0; f}^2 \mathcal{L}_{-1}^2(X^2) \right\},$$

где Θ^j (здесь $j=2$) - логическое условие, истинное, когда содержимое регистра X^j не равно нулю, и ложное - в противном случае.

Предполагается, что операторы, заключенные в итерационные скобки, выполняются одновременно. При этом результат сложения будет находиться на регистре X^1 .

Регулярное выражение микропрограммы умножения положительных целых чисел, записанное в терминах многорегистровых преобразований, имеет вид:

$$0^1 [\text{ev} \{ \text{ev} (\&_{0,0; f}^{1,2} \vee \&_{0,0; f}^{2,1}) \mathcal{L}_{-1}^1 \mathcal{L}_{-1}^2 \}] 0^2 0^3,$$

где β_z^j - логическое условие, истинное, когда содержимое z -го разряда (здесь старшего) регистра $X^j/j=3/$ равно единице, и ложное - в противном случае, e - тождественное преобразование.

Аналогично с помощью сдвигов и инверсии можно представить операции деления и вычитания целых положительных чисел.

2. Для удобства реализации многорегистровых преобразований рассмотрим следующую абстрактную модель ВС, состоящую из двух двумерных параллельных плат, площадь каждой из которых не ограничена $|\infty \times \infty|$. Нижнюю плату A^0 назовем информационной, а верхнюю A^1 - вспомогательной. Для ориентации в ВС введем координатные оси: U и u . Элемент ВС, находящийся на пересечении координатных осей (начало координат), имеет координаты $0,0$. Указанные платы $A^z, z=\{0,1\}$, состоят из элементов $\alpha_{u,u}^z$, каждый из которых жестко соединен с пятью соседними элементами (рис. 1).

Предполагается, что элемент данной модели ВС извне может быть настроен на выполнение логических функций конъюнкции ($\&$), дизъюнкции $/V/$, инверсии $/7/$, а также соединительного элемента "крест без точки" (элемент Д) [8]. Это позволяет осуществить переработку, прием и передачу информации в требуемом направлении. Заметим, что выбор элемента ВС не влияет на общность дальнейших рассуждений. Необходимо лишь, чтобы базис элемента данной модели ВС был функционально и соединительно полным.

В дальнейшем при структурной интерпретации многорегистрового преобразования каждая плата рассматривается как решетка, в узлах которой расположены элементы данной модели, а направление передачи информации между элементами ВС указывается стрелками. При этом предполагается, что выполнена необходимая настройка элементов ВС.

Заметим, что необходимая информация для решаемых задач, а также программы работы системы в целом хранятся в запоминающем устройстве (ЗУ) ЭВМ, связанной с данной ВС (предполагается, что

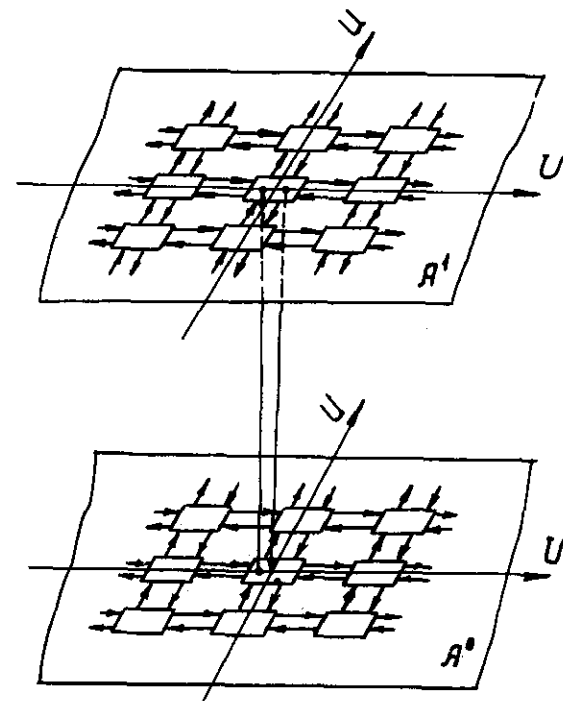


Рис. 1.

ЭВМ обладает достаточным объемом памяти).

3. Перейдем к реализации многорегистровых преобразований в описанной абстрактной модели ВС.

Пусть базовая функция преобразования $F_{\kappa_1, \kappa_2, \dots, \kappa_n} f^{\kappa_1, \kappa_2, \dots, \kappa_n}$ задана в дизъюнктивной нормальной форме (д.н.ф.):

$$f(\tau_0^1, \tau_1^1, \dots, \tau_{k-1}^1, \dots, \tau_0^n, \tau_1^n, \dots, \tau_{k-1}^n) = \alpha_1 V \dots V \alpha_s, \quad (1)$$

где α_k - элементарное произведение ($k=1, \dots, s$).

По обе стороны от оси u вычислительную среду условно разобьем на равные слои S_z , каждый из которых состоит из двух

полосе $B_i^z, z \in \{0, 1\} (-\infty < z < \infty)$, причем левая грань полосы B_i^z проходит по оси $u(v=0)$. Ширина каждой такой полосы задается формулой:

$$b = n + s + 1.$$

Известно, что каждое преобразование $F_{k_1, k_2, \dots, k_n; f^1, f^2, \dots, f^n}(F)$ может быть представлено в следующем виде:

$$F_{k_1, k_2, \dots, k_n; f^1, f^2, \dots, f^n} = L_{k_1}^1 X L_{k_2}^2 X \dots X L_{k_n}^n F_{0, 0, \dots, 0; f^1, f^2, \dots, f^n}$$

где X - композиция, $L_{k_j}^j$ - оператор сдвига на регистре $X^j (j=1, \dots, n)$.

Реализация оператора сдвига $L_{k_j}^j$ сводится к реализации оператора $L_{k_j}^j$ при $k_j > 0$ или оператора $L_{k_j}^j$, если $k_j < 0$. Для реализации операторов $L_{k_j}^j, L_{k_j}^j$ в ВС реализуются два регистра X^j и X_{-j}^j . При этом в ЭВМ, далее называемой устройством управления УУ, вырабатываются следующие управляющие сигналы: $y_n(y_n)$ и y^{j-1-j} . Процедуру осуществления сдвига можно описать следующим образом: синхронно с управляющим сигналом $y_n(y_n)$ содержимое регистра X^j пересылается на регистр X_{-j}^j со сдвигом на один разряд вправо (влево), а синхронно с сигналом y^{j-1-j} содержимое регистра X_{-j}^j пересылается на регистр X^j . При повторении описанной процедуры k раз регистр X^j будет переведен в новое состояние, соответствующее результату преобразования $L_{k_j}^j$.

В дальнейшем регистр $X^j (j=1, \dots, n)$ располагается на плате A^0 по прямой, уравнение которой имеет вид $u^j = \text{const}$. Причем, полосе B_i^0 ставится в соответствие элемент x_i регистра X^i . Известно, что каждый элемент регистра - триггер может быть построен соединением двух или любого другого четного числа инверторов (схемы НЕ) с обратной связью. При этом реализацию элемента x_i регистра X^i в данной модели ВС можно описать следующим образом: элементы ВС $a_{v, u}^0, u^j$

$$a_{v, u}^0 \text{ и } a_{v, u}^0 \text{ (} v=1, \dots, v-u^j-1 \text{)}$$

настраиваются соответственно на выполнение логической функции отрицания и соединительного элемента Д, где $v^j = b \cdot i + j - 1$ и $v = v^j + s$ - номер элемента ВС по оси v , а u^j - номер элемента ВС по оси u . Условимся снимать выходной сигнал элемента x_i регистра X^i , соответствующего i -й полосе, с выходных полюсов элемента $a_{v, u}^0$ ВС. Заметим, что регистр X_{-j}^j реализуется аналогично X^j .

Расположение регистров в ВС при реализации преобразования F следующее: пусть по прямой $u^{j-1} = \text{const}$ и $u^{j+1} = \text{const}$ соответственно реализованы регистры X^{j-1} и X^{j+1} . Тогда по прямой $u^j = u^{j-1} - 1 = \text{const}$ реализуется регистр X_{-j}^j , а по прямой $u^j = u^{j+1} - d = \text{const} - X_{-j}^j$, где d - расстояние между регистрами X_{-j}^j и X_{-j}^j , необходимое для настройки элементов ВС на реализацию оператора сдвига ($L_{k_j}^j$ или $L_{k_j}^j$). В случае $k_j = 0$ реализация регистра X_{-j}^j не требуется и, следовательно, $u_{-j}^j = u^j$. Заметим, что если результат преобразования F определен на регистре X^j и при этом коэффициент $k_j = 0$, то для выполнения условия устойчивости функционирования проектируемого автомата [9] необходимо по прямой $u_{-j}^j = u^j + d_j = \text{const}$ реализовать регистр X_{-j}^j . Элементы ВС, расположенные между этими регистрами (X_{-j}^j и X_{-j}^j), настраиваются на реализацию оператора пересылки содержимого регистра X_{-j}^j на регистр X_{-j}^j синхронно с управляющим сигналом y^{j-1-j} (d_j - расстояние между регистрами X_{-j}^j и X_{-j}^j , необходимое для настройки элементов ВС на реализацию оператора пересылки между этими регистрами).

Построим схему, реализующую преобразование $F_{a_{0,0}, 0; f^1, f^2, \dots, f^n}$. Основной принцип построения состоит в следующем (подразумевается, что описанная ниже идея была применена для настройки элементов ВС, расположенных в $i+1$ -м слое): ниже прямой $u^j = \text{const}$ выделяется параллелепипед, грани которого являются прямоугольниками R_{-j}^j и R_{-j}^j с вершинами a_1, a_2, a_3, a_4 и a_1', a_2', a_3', a_4' (см. рис. 2). Ширина каждого прямоугольника равна s - числу элементарных произведений в д.н.ф. базовой функции, а длина - $R = n(z_{\max} + 1)$, где R - число ар-

гументов данной функции; $z_{max} = \max(z_1, \dots, z_n)$

Далее, в прямоугольнике R_i^1 осуществляется трассировка каналов, обеспечивающих поступление сигналов, соответствующих значениям аргументов функции $f(\tau_0^1, \tau_1^1, \dots, \tau_{n-1}^1, \tau_n^1, \dots, \tau_{z_n}^1)$. Для этого необходимо отождествлять выходные и входные полюсы следующих элементов ВС:

а) На плате A^1 прокладывается путь π_0^j ($j=1, \dots, n$) между элементами $a_{v, u}^0$ и $a_{v_{a_2}, u_{a_2+j-1}}^1$ (где $u_{a_2} = u_{a_2}^1 = v_i^j + S - 1$, $u_{a_2} = u_{a_2}^1 = u_i^j + 3$).

$$\pi_0^j = a_{v, u}^0, a_{v, u^j + E_1}^1, a_{v - E_2, u^j}^1,$$

где

$$E_1 = 0, \dots, u_{a_2} - u_i^j + j - 1,$$

$$E_2 = 1, \dots, v_i - v_{a_2}.$$

На плате A^0 и A^1 (см. рис. 2) элементы ВС, соответствующие началу и концу пути π_0^j , соответственно обозначены через e_1, e_2, e_3 и a_1', e_1', e_2', e_3' , а трассируемые каналы указаны контурными линиями.

б) Выходные полюсы элементов ВС, расположенные на грани прямоугольника R_{i+1}^1 , отождествляются соответственно с входными полюсами элементов ВС, расположенных на грани прямоугольника R_i^1 . Требуемое отождествление полюсов осуществляется следующим образом: переменной ν присваивается последовательно значение из множества $\{1, \dots, z_{max}\}$. Если ν - нечетное число, тогда прокладывается путь π_ν^j , иначе - путь $\tilde{\pi}_\nu^j$ между элементами ВС

$$a_{v_{i+1}}^1, u_{a_2} + (\nu - 1)n + j - 1 \text{ и } a_{v_{a_2}}^1, u_{a_2 + \nu n + j - 1}.$$

Заметим, что для некоторого фиксированного значения ν переменная j последовательно принимает значения из множества $\{1, \dots, n\}$

$$\pi_\nu^j = a_{v_{i+1}}^1, u_{a_2} + (\nu - 1)n + j - 1, a_{v_{i+1}}^1, u_{a_2 + (\nu - 1)n + j - 1},$$

$$a_{v_{i+1} - E_1}^0, u_{a_2} + (\nu - 1)n + j - 1, a_{v_{i+1} - n + j + 2 - E_3}^0, u_{a_2 + \nu n + j - 1},$$

$$a_{v_{i+1} - n + j + 2 - E_3}^0, u_{a_2 + \nu n + j - 1}, a_{v_{a_2}}^1, u_{a_2 + \nu n + j - 1},$$

$$a_{v_{a_2}}^1, u_{a_2 + \nu n + j - 1}$$

$$\tilde{\pi}_\nu^j = a_{v_{i+1}}^1, u_{a_2} + (\nu - 1)n + j - 1, a_{v_{i+1} - E_1}^1, u_{a_2} + (\nu - 1)n + j - 1,$$

$$a_{v_{i+1} - n + j - 2}^1, u_{a_2} + (\nu - 1)n + j - 1 - E_2,$$

$$a_{v_{i+1} - n + j - 2 - E_3}^1, u_{a_2} + \nu n + j - 1,$$

где

$$E_1 = 1, \dots, n - j + 2,$$

$$E_2 = 1, \dots, n,$$

$$E_3 = 1, \dots, v_{i+1} - n + j - u_{a_2} - 2.$$

Нетрудно заметить, что в каналах π_ν^j и $\tilde{\pi}_\nu^j$ ($j=1, \dots, n$; $\nu=0, \dots, z_{max}$) проходят сигналы, соответствующие значениям переменных базовой функции τ_ν^j .

На рис. 2 изображена структурная интерпретация многорегистрового преобразования, основные параметры которого соответственно равны: $n=3$, $z_{max}=2$, $S=3$. Для упрощения рисунка предполагается, что коэффициенты $K_1=K_2=K_3=0$. На данном рисунке в качестве примера контурными линиями указаны пути π_1^2 и $\tilde{\pi}_2^2$, начало и конец которых обозначены соответственно через e_1 и e_2 , e_3 и e_4 .

Настройка элементов ВС, расположенных в прямоугольниках R_i^1 и R_i^0 , происходит по базовой функции, значения которой определяются элементом $a_{v_{i+1}}^1, u_{a_2} - 1$ ВС и синхронно с управляющим сигналом, например, y_i , засылаются вначале на регистр X_i^1 , а затем пересылаются на регистр X_i^0 . Здесь предполагается, что преобразование f определено на регистре X_i^1 , $j \in 1, \dots, n$. В случае если $j \notin 1, \dots, n$, то по прямой $u^m = u^m - 1 = const$ реализуется регистр X^m (расположение элемента памяти x_i регистра X^m совпадает с элементом x_i регистра X^1), в котором синхронно с управляющим сигналом y_i засылается значение базовой функции. При этом необходимо предварительно регистр X^m установить в нулевое состояние.

На рис. 2 показано, что значение базовой функции засылает-

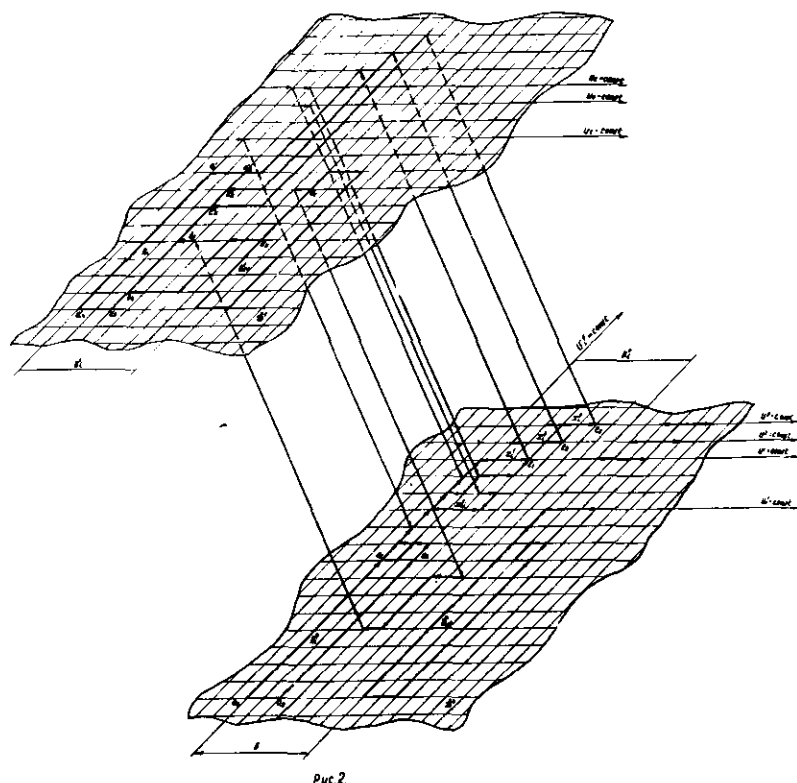


Рис. 2. Реализация преобразования

$F: 0, 0, \dots, 0, f^2, z_0, \dots, z_n$ в среде

ся в регистр X_1' после установки его в нулевое состояние сигналом y_0 , затем синхронно с сигналом y_2 пересылается в регистр X_1' . Каналы, по которым проходят управляющие сигналы, y_0, y_1 и y_2 , имеют координаты: $u_0 = u_1' - 1$, $u_1 = u_1' + 1$ и $u_2 = u_1' - 1$.

Таким образом, схема, реализующая преобразование F , построена.

Описанный метод реализации отражает основное свойство многорегистровых преобразований — их периодичность. Кроме того, введение вспомогательной платы упрощает общую конструкцию и уменьшает число обращений к УУ, обеспечивая связь между различными участками платы A^0 . Подтверждением сказанному служат приведенные ниже схемы микропрограмм основных операций машины, реализованные в данной модели ВС.

4. Перейдем к реализации микропрограмм основных арифметических операций с фиксированной запятой, записанных в терминах многорегистровых преобразований. Вначале построим схему сложения по ранее описанной микропрограмме (см. рис. 1). В рассматриваемой микропрограмме содержатся следующие три преобразования: E^1 , E^2 , Z^2 , для реализации которых, как и ранее, ВС разбивается на равные слои S_i шириной $b = 3$. Далее выделяем слой S' шириной, равной двум. Пересечением слоев $S_i / -\infty < i < \infty$ со слоем S' образуются параллелепипеды, гранями которых являются прямоугольники R_i^1 и R_i^2 , соответственно.

Пусть $u_0, u_1, u_2, u_3, v_0, v_1, v_2, v_3$ — координаты вершин прямоугольника. На плате A^0 в столбце с координатой $u_1' = u_2 - b, v_1' = v_1 - d$ реализуется регистр X_1' / X_1' , а в столбце с координатой $u_1'' = u_0 + 4, v_1'' = v_1 + d$ — регистр X_2' / X_2' . Выходной сигнал, который соответствует единичному состоянию триггера $x_i / N \geq i \leq N$ регистра $X^j / j = 1, 2$, снимается с выходных полюсов элемента ВС, с координатами v_1^j, u_1^j , где $u_1^j = b \cdot i = u_0$. Затем в прямоугольник R_1^1 через плату A^1 трассируется канал минимальной длины, по которому проходит сигнал, соответствующий значению переменной z_0' / z_0' базовой функции, то есть отождествляются выходные и входные полюсы элементов ВС с координатами соответственно $v_1^1, u_1^1, v_2^2, u_2^2, v_3^3, u_3^3, v_4^4, u_4^4$.

В прямоугольнике P_i^1 элементы ВС с координатами $v_{c_1}, u_{c_1}; v_{c_1}, u_{c_1}+1; v_{c_1}, u_{c_1}+2; \dots; v_{c_1}, u_{c_1}+N$ настраиваются на реализацию логических функций $v, 7, 7/7, 7, v/$. Каждый элемент ВС, расположенный в прямоугольнике P_i^1 , настраивается на прием информации с платы A^1 .

Для реализации базовых функций преобразований $\mathcal{E}^1$ и $\mathcal{E}^2$ прямоугольник P_i^1 разбивается на два - верхний и нижний. В первом прямоугольнике, ширина которого равна двум, строится базовая функция преобразования $\mathcal{E}^1$, а во втором - базовая функция преобразования $\mathcal{E}^2$, значения которых синхронно с управляющими сигналами y^1 и y^2 записываются соответственно на регистры X^1 и X^2 . Элементы ВС, расположенные между регистрами X^1 и X^2 , настраиваются на реализацию оператора пересылки (содержимое регистра X^1 пересылается на регистр X^2), а элементы ВС между регистрами X^2 и X^2 - на реализацию оператора сдвига $\mathcal{L}_2^2$.

На каждом такте реализации преобразований $\mathcal{E}^1$ и $\mathcal{E}^2$ в УУ выдается сигнал (логическое условие ω^2), нулевое значение которого означает, что регистр X^2 находится в нулевом состоянии (конец микропрограммы сложения). Канал, по которому проходит сигнал, задающий состояние регистра X^2 , трассируется в строке v^2-1 на плате A^1 . Схема, реализующая сложение по его микропрограмме (см. п.1), построена.

Предположим теперь, что числа C_1 и C_2 , участвующие в операции, представлены с фиксированной запятой и хранятся в ЗУ машины в прямом коде.

Пусть состояния α_i, α_{i+1} элементов x_i, x_{i+1} регистров X^1 и X^2 соответствуют значению знаковых разрядов чисел C_1 и C_2 , а состояния $\alpha_{i+1}, \alpha_{i+2}, \dots, \alpha_{i+N}$ элементов $x_{i+1}, x_{i+2}, \dots, x_{i+N}$ - значению мантиссы, где N - константа, указывающая ее разрядность.

Для реализации той или иной микропрограммы арифметических операций необходимо предварительно осуществить ряд преобразований и соответствующую настройку элементов ВС. Например, перед непосредственным сложением отрицательные коды данных чисел преобразуются в модифицированный дополнительный код. Пусть в столбце $v^3 = v^1 d, v^2 v^2 d$ реализован регистр X^3/X^4 для промежуточного хранения слагаемого C_1/C_2 . Элементы ВС между

регистрами X^1 и X^3/X^4 и X^2 и X^4 настраиваются на реализацию оператора пересылки, причем содержимое регистра X^1/X^2 пересылается на X^3/X^4 , и наоборот.

Регулярное выражение микропрограммы преобразования $\mathcal{P}$ отрицательного кода чисел C_1 и C_2 имеет вид:

$$\mathcal{P}(C_1): \omega^2 \left[X^{3+1} \right] \omega^1 \left[\mathcal{E}^1 \right] \omega^2 \left[X^{1+3} v e \right] \omega^1,$$

$$\mathcal{P}(C_2): \omega^1 \left[X^{4+2} \right] \omega^2 \left[\mathcal{E}^1 \right] \omega^2 \left[X^{1+2} X^{2+4} v e \right] \omega^2.$$

Процедура обмена информацией между регистрами X^1 и X^2 описывается следующим регулярным выражением:

$$X^2 \rightarrow X^1, / X^{2+1} / : \omega^1 \mathcal{E}^1 \omega^2, \\ X^1 \rightarrow X^2, / X^{1+2} / : \omega^2 \mathcal{E}^2 \omega^1.$$

Регулярное выражение микропрограммы сложения (вычитания) двух алгебраических чисел имеет вид:

$$\mathcal{P}(C_1) \mathcal{P}(C_2) X^{3+1} X^{4+2} \omega^{1,2}, \\ \mathcal{P}(C_1) \mathcal{P}'(C_2) X^{3+1} X^{4+2} \omega^{1,2}.$$

При этом подразумевается, что слагаемое C_1/C_2 на первом такте реализации микропрограммы было присвоено регистру X^3/X^4 . Регулярное выражение преобразования $\mathcal{P}'$ вычитаемого C_2 при реализации микропрограммы вычитания имеет вид:

$$\omega^1 \left[X^{4+2} \right] \omega^2 \left[v X^{4+2} \right] \omega^2 \left[\mathcal{P}(C_2) X^{1+2} X^{2+4} \right] \omega^1,$$

где ω^1 - логическое условие, истинное, когда знаковые разряды регистра X^1 установлены в единичное состояние, и ложное - в противном случае.

Отметим, что преобразования в рассматриваемых регулярных выражениях, записанные в одном столбце, выполняются параллельно.

При реализации описанных микропрограмм в УУ вырабатываются следующие управляющие сигналы: $y_{\text{ог}}^1, y_{\text{ог}}^2, y_{\text{ог}}^3, y_{\text{ог}}^4, y_{\text{ог}}^5$. Каналы, по которым проходят данные управляющие сигналы, трассируются в столбцах, координаты которых соответственно равны: v^1+2, v^1-2, v^1-3 (на плате A^1), v^2+2 (на плате A^2). При этом предполагается, что единичное значение сигнала $y_{\text{ог}}^1/y_{\text{ог}}^2$ устанавливает в нулевое состояние знаковый разряд

регистра X^j . Аналогично сигнал y_{i+1}^j / y_{i+1}^j устанавливает в единичное состояние $i+N$ -й разряд регистра X^j . Заметим, что установка регистра X^j в некоторое состояние α сводится к установке триггера x_i регистра X^j в состояние $\alpha_i / \alpha_i = 0, 1$.

После окончания операции сложения (вычитания) в УУ выдается сигнал $1/\omega^1$ о знаке результата и сигнал $1/\varphi^1$ о переполнении. Для определения значения сигнала φ^1 реализуется функция $f(\alpha_i, \alpha_{i+1}) = \alpha_i \bar{\alpha}_{i+1} \vee \bar{\alpha}_i \alpha_{i+1}$ на плате A^1 . Канал, по которому проходит сигнал φ^1 , трассируется в столбце с координатой v^1+2 .

При необходимости округления результата выполняется микропрограмма

$$0^2 [1^2 \vee [1^2]_{i+N+1}] C^{1,2}.$$

При этом в УУ вырабатывается управляющий сигнал y_{i+1}^2 / y_{i+1}^2 . Канал, по которому проходит сигнал y_{i+1}^2 / y_{i+1}^2 , трассируется в столбце с координатой v^2+4 на плате A^1 .

В случае, если требуется записать в ЗУ машины результат операции сложения (вычитания), проверяются знаковые разряды регистра X^1 . При значении $\omega^1=1$ содержимое регистра X^1 преобразуется микропрограммой $\Pi(C_1)$ в прямой код. Преобразованное число пересылается в ЗУ машины.

Составим микропрограмму умножения и построим реализующую ее схему, основным узлом которой является ранее построенная схема сложения (вычитания). При реализации умножения ВС перестраивается (достраивается) следующим образом: на плате A^0 в столбце v^3 реализуется регистр X^3 . Элементы ВС между регистрами X^1/X^1 и X^1/X^3 настраиваются на реализацию оператора сдвига Z_{+1} . На регистре X^3 сдвигается только дробная часть числа. Ранее продолженные связи (при сложении/вычитании) между мантийсскими частями регистров X^1 и X^3 разрываются. На первом такте работы микропрограммы (после того как регистры, участвующие в операции, установлены в нулевые состояния) значение множимого присваивается регистру X^2 , а множителя - регистру X^3 (на плате A^1). Если множимое (множитель) равно нулю, вырабатывается сигнал θ^2/θ_n^2 об окончании микропрограммы умножения. Канал, по которому проходит

сигнал, задающий состояние дробной части регистра X^3 (логическое условие θ_n^3), трассируется на плате A^1 в столбце с координатой v^3-1 . Если начальные состояния регистров X^1 и X^3 не равны нулю, содержимое мантийссы регистра X^3 сдвигается вправо и его $i+1$ -й разряд устанавливается в единичное состояние. Канал, по которому проходит управляющий сигнал y_{i+1}^3 , вырабатываемый в УУ, трассируется на плате A^1 в столбце с координатой v^3+2 . Единица, записанная в $i+1$ -й разряд регистра X^3 , является контрольной, задающей количество тактов $-N$ при умножении чисел C_1 и C_2 .

Регулярное выражение микропрограммы умножения имеет вид:

$$0^1 X^{4+2} \left[e \vee X^{3+1} \bar{e}^1 X^{1+3} 0^1 \right. \\ \left. \begin{matrix} \theta^2 \theta_n^3 & Z_{+1}^3 [1^3]_{i+1} & X^{2+4} \\ & 10^2 [& \end{matrix} \right] \\ \left\{ e \vee (X^{4+2} C^{1,2} \vee e) Z_{+1}^1 Z_{+1}^3 \right\} X^{3+1} \left. \right] 0^3 0^4. \\ \theta_n^3 \beta_{i+N+1}^3$$

Канал, по которому проходит сигнал, задающий состояние дополнительного $i+N+1$ -го разряда регистра X^3 , трассируется на плате A^1 в столбце с координатой v^3+3 . Результат умножения снимается с регистра X^1 .

В ЭВМ находят применение два метода деления: деление с восстановлением и без восстановления остатка. Регулярное выражение микропрограммы деления $(C_1 : C_2)$ без восстановления остатка имеет следующий вид: $(X^1 = C_1, X^2 = C_2)$.

$$0^1 0^3 0^4 0^5 \left[e \vee \begin{matrix} X^{2+4} \bar{e}^1 & 10^2 [\Pi(C_2) X^{3+4}] 0^1 [\\ X^{3+1} X^{4+5} & X^{1+3} [1^3]_{i+N+1} & X^{3+1} 0^1 \end{matrix} \right. \\ \left. \theta_n^3 \theta^2 \right] \\ \left\{ \beta_{i+1}^3 \varphi^1 ([1^3]_i C^{1,5} \vee X^{4+2} C^{1,2} [1^3]_{i+N+1}) Z_{+1}^1 Z_{+1}^3 \right\}$$

$$\left(\begin{matrix} 0^1 & X^{1+2} & \Sigma^1 \\ 0^2 & X^{3+1} & Z^3 \end{matrix} X^{1+3} \right) \left] 0^1 0^2 0^4 0^5, \right.$$

где ψ^j - логическое условие, истинное, когда знаковые разряды регистра X^j установлены в единичное состояние, и ложное - в противном случае. Для определения значения сигнала ψ^1 реализуется функция $f(\alpha_{i-1}, \alpha_i) = \alpha_{i-1} \cdot \alpha_i$ на плате A^1 .

Схему, реализующую микропрограмму деления, можно построить по описанному регулярному выражению. Заметим, что результат деления снимается с регистра X^3 .

В заключение заметим, что за счет введения вспомогательной платы (A^1) число обращений к ЭВМ, обслуживающей вычислительную среду, достаточно мало при выполнении одной микропрограммы. Это позволяет надеяться, что можно будет повысить быстродействие проектируемой системы, снимаемой в терминах многорегистровых преобразований.

Л и т е р а т у р а

1. ГЛУШКОВ В.М. Теория автоматов и вопросы проектирования структур цифровых машин, - "Кибернетика", Киев, 1965, № 1.
2. ГЛУШКОВ В.М. Теория автоматов и формальные преобразования микропрограмм, - "Кибернетика", Киев, 1965, № 5.
3. ЕВРЕИНОВ Э.В. Теоретические основы построения универсальных вычислительных сред, - "Вычислительные системы", Новосибирск, "Наука" СО, 1965, вып. 16.
4. ЕВРЕИНОВ Э.В., КОСАРЕВ Ю.Г. Однородные универсальные вычислительные системы высокой производительности. Новосибирск, "Наука", 1966.
5. ПРАНГИШВИЛИ И.В., АБРАМОВА Н.В. и др. Микроэлектроника и однородные структуры для построения логических и вычислительных устройств. М., "Наука", 1967.
6. БОНДАРЧУК В.Г., ЦЕЙТЛИН Г.Е. Об алгебрах периодически-определенных преобразований бесконечного регистра. - "Кибернетика", Киев, 1969, № 1.
7. ЦЕЙТЛИН Г.Е. Вопросы функциональной полноты для одной модификации алгебры логики, - "Кибернетика", Киев, 1969, № 4.
8. МИШИН А.И. Об одном варианте комбинированной вычислительной системы, - "Вычислительные системы", Новосибирск, "Наука" СО, 1967, вып. 26.
9. ГЛУШКОВ В.М. Синтез цифровых автоматов, Физматгиз, М., 1962.

Поступила в редакцию
3.УШ.1971 г.