

УДК 621.382

С. В. Пискунов

(Новосибирск)

### МНОГОСЛОЙНАЯ УНИВЕРСАЛЬНАЯ КЛЕТочНАЯ СТРУКТУРА И ЕЕ ЭЛЕКТРООПТИЧЕСКАЯ РЕАЛИЗАЦИЯ\*

Предлагается трехмерная универсальная клеточная структура, разработанная с использованием компьютерной системы моделирования распределенных вычислений (алгоритмов параллельных подстановок). Структура состоит из четырех слоев. Все слои структуры, кроме одного, фактически являются памятью с оптическими входами и выходами ячеек. Единственный отличный от памяти слой устроен достаточно просто: он состоит из итеративно повторяющейся схемы совпадений. Показано, что структура может обладать высокой технологичностью и быстродействием, позволяет компактно размещать имитируемое цифровое устройство.

**Введение.** Интерес к исследованию универсальных вычислительных структур, таких как однородные среды, клеточные структуры (например, клеточные автоматы), которые можно настраивать на реализацию заданного цифрового устройства, достаточно устойчив [1—4]. В этой области, наряду с теоретическими, имеются также практические результаты [5]. Притягательность универсальных структур объясняется их высокой технологичностью: тиражируется одно и то же изделие, находящееся самые разнообразные применения.

При построении универсальных структур приходится находить компромисс между сложностью отдельной клетки и их общим числом, необходимым для реализации того или иного устройства. Этот компромисс определяет не только сложность реализации устройства в универсальной структуре, но и его быстродействие. Поэтому чем больше различных функций при одной и той же технической сложности может выполнять клетка структуры, тем эта структура перспективней по сравнению со структурами, не обладающими такой возможностью.

В данной работе предлагается способ преобразования двумерной универсальной клеточной структуры в трехмерную, составленную из простых клеток. В этом плане работа продолжает исследования по расслоению двумерных структур. Но в отличие от преобразований, представленных в [6, 7], новое преобразование обеспечивает превращение двумерной структуры в трехмерную (четырёхслойную), состоящую из верхнего и нижнего слоев памяти и расположенных между ними операционного и регистрирующего слоев. Клетки памяти хранят записи микроопераций, выполняемых совместно клетками операционного и регистрирующего слоев. Такой прием позволяет построить операционный слой, клетки которого реализуют три элементарные операции:

- а) сравнение двух символов: одного, взятого из верхнего слоя памяти, другого — из собственной ячейки памяти клетки операционного слоя;
- б) копирование символа из нижнего слоя памяти в собственную ячейку памяти клетки операционного слоя, если были выполнены успешные срав-

\* Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (№ 93-01-01000).

нения по пункту «а» в регистрирующем слое для клетки и ее ближайших соседей;

в) сдвиг содержимого ячеек памяти клеток операционного слоя вдоль некоторой оси координат.

Выполнение в клетках структуры только этих элементарных операций предопределяет невысокую сложность клеток. Вместе с тем возможность записи в слои памяти разнообразных микроопераций как раз и обеспечивает настройку клетки операционного слоя и ее ближайших соседей на выполнение некоторого преобразования информации из широкого перечня возможных преобразований.

1. Предпосылки и основные понятия. В качестве исходной двумерной универсальной структуры выберем структуру, описанную в [4]. Основанием для нашего решения служат такие ее черты, как возможность непосредственно моделировать цифровую логику и компактность цифровой схемы, реализованной в структуре [4]. Отметим также, что набор преобразований информации, выполняемых клетками структуры: распространение, пересечение, ветвление сигналов, логические функции от одной, двух переменных, практически совпадает с соединительным и функциональным базисом однородной вычислительной среды [1].

Для описания функционирования клеточных структур используются алгоритмические средства, предложенные в [8, 9]. Сама структура изображается в виде одного или нескольких клеточных массивов в графическом представлении, преобразования состояний клетки и ее ближайших соседей записываются в виде команд параллельных подстановок также в графическом представлении [9]. На последующих рисунках клетка изображается квадратом с вписанным в него состоянием клетки. Изображение двумерного клеточного массива — это прямоугольник, составленный из изображений клеток. Начало координат в нем всегда задается в самой верхней левой клетке, ось  $i$  направлена вниз, ось  $j$  — вправо. Трехмерный ( $r$ -слойный) клеточный массив изображается в виде совокупности из  $r$  ступенчато расположенных плоских массивов. Самый нижний плоский массив является ближайшим к наблюдателю. Начало координат выбирается в его верхней левой клетке, ось  $k$  направлена от наблюдателя. Над изображением каждого массива записывается его имя. Команда подстановки может осуществлять преобразование информации сразу в нескольких клеточных массивах. Графические изображения конфигураций, образующих левую (правую) часть команды, помещают рядом и соединяют символом «\*». Изображение конфигурации совпадает с изображением клеточного массива, но только самой верхней левой клетке вместо имени массива ставится в соответствие тождественная именуемая функция, являющаяся вектором, образованным из имени клеточного массива, которому поставлена в соответствие конфигурация, и всех (или части) переменных  $i, j, k$ . Соединение конфигураций при помощи символа «\*» означает, что перемещение их изображений по изображениям соответствующих клеточных массивов осуществляется согласованно: путем подстановки во все именуемые функции всех конфигураций в левой и правой частях команды одного и того же набора значений переменных  $i, j, k$ .

На рис. 1, 2 дано описание двумерной универсальной клеточной структуры. Оно состоит из двух частей:

а) представление исходного состояния структуры: клеточные массивы  $sr$  и  $gr$  (см. рис. 1);

б) запись правил функционирования структуры: подстановки  $\Theta_1—\Theta_{16}$  (см. рис. 2).

Каждая клетка массива  $sr$  может находиться в одном из четырех состояний, которые будем называть (в скобках даны их символьные и графические обозначения):

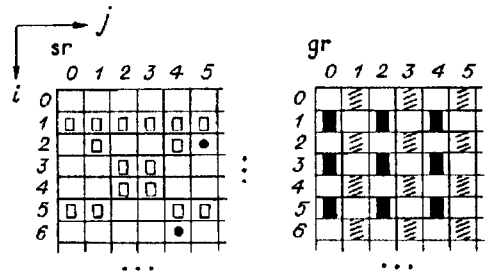


Рис. 1

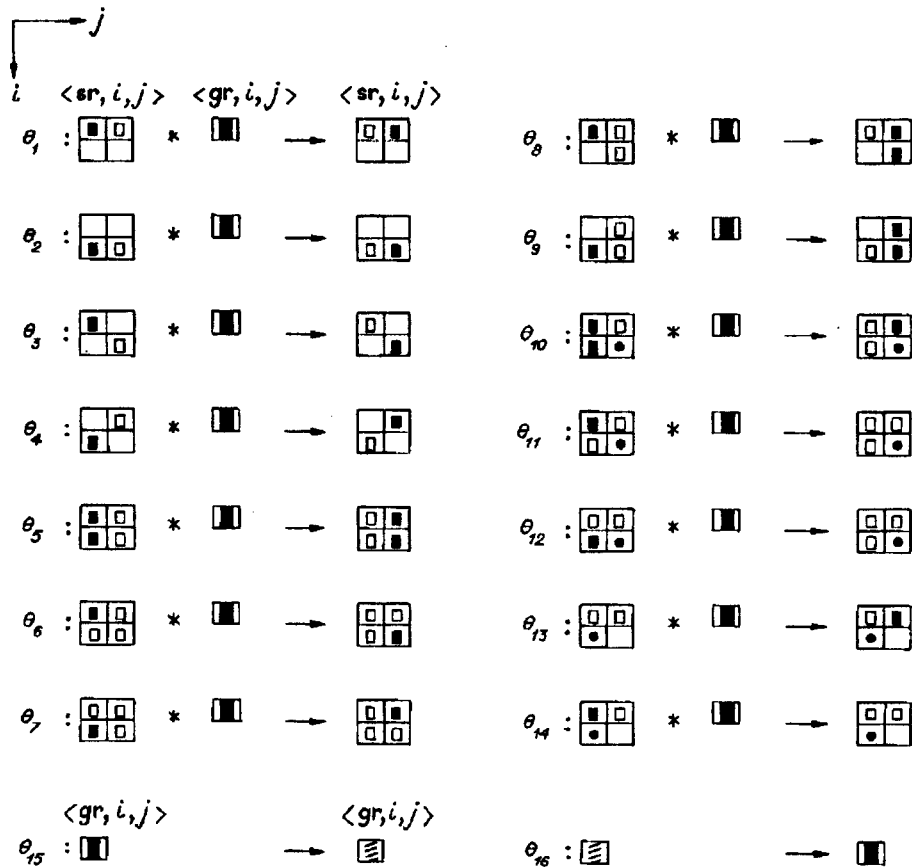


Рис. 2

«изоляция» (O,  $\square$ ), «канал» (C,  $\Pi$ ), «сигнал» (S,  $\blacksquare$ ), «вентиль» (G,  $\boxplus$ ). Символьными обозначениями состояний будем пользоваться в тексте, графическими — на рисунках. Отметим, что «каналом» также будем называть цепочку клеток в состоянии C, расположенных по вертикали, горизонтали или диагонали.

Исходное состояние массива *sg* универсальной структуры, реализующей заданное цифровое устройство, будем называть клеточным представлением этого устройства. При задании исходного состояния используются только символы O, C, G. Далее ограничимся в основном реализацией в структуре произвольных булевых функций.

Для представления состояний массива *gr*, наряду с символом O, вводятся два новых символа: «синий» (B,  $\blacksquare$ ) и «красный» (R,  $\boxplus$ ).

На рис. 2 полностью изображены только команды  $\theta_1$  и  $\theta_{15}$ . Изображения команд  $\theta_2$ — $\theta_{14}$  аналогичны изображению команды  $\theta_1$ , изображение команды  $\theta_{16}$  — изображению команды  $\theta_{15}$ . Чтобы не загромождать рисунок, эти изображения упрощены: в них опущены записи именующих функций. Команды  $\theta_1$ — $\theta_{14}$  определены на двух массивах *sg* и *gr*. Левая часть каждой из команд состоит из композиции двух графических изображений конфигураций, первая конфигурация далее будет называться базой команды. Команды  $\theta_{15}$ ,  $\theta_{16}$  определены на одном массиве *gr*. Объединение всех команд образует систему параллельных подстановок (обозначается СПП Ф). СПП Ф применяется к массивам *sg* и *gr* синхронно и итеративно: на одном такте отыскиваются вхождения всех левых частей всех команд в клеточные массивы, на другом — для тех команд, чьи левые части найдены, выполняется замена символов в найденных вхож-

дениях левых частей символами из соответствующих правых частей и т. д. Как происходят поиск и замена, покажем на примере одной команды, пусть это будет  $\Theta_1$ . Графическое изображение базы команды  $\Theta_1$  накладывается на клетки массива *sg* так, чтобы его верхняя левая клетка имела такие же координаты, как соответствующая ей клетка в состоянии В в массиве *gr* (первый такт). И везде, где графическое изображение базы совпадает с изображением, образованным клетками клеточного массива *sg*, образ базы в клеточном массиве *sg* заменяется образом правой части  $\Theta_1$  (второй такт).

Команды  $\Theta_{15}$ ,  $\Theta_{16}$  обеспечивают чередование в массиве *gr* двух разбиений, элементами каждого из которых являются квадратные блоки, составленные из четырех клеток и содержащие верхнюю левую клетку в состоянии В. Смена состояний этой клетки обеспечивает в клеточном массиве *sg* частичное наложение друг на друга образов баз команд, применимых на соседних итерациях, а значит, возможность перемещения и преобразования сигналов в структуре. Считая клетки с координатами  $\langle 1, 0 \rangle$ ,  $\langle 5, 0 \rangle$  массива *sg* входными для изображенного в нем представления цифрового устройства и задавая в них различные комбинации символов *S* и *S*, можно убедиться, что команды  $\Theta_1$ — $\Theta_4$  обеспечивают движение сигналов, команды  $\Theta_5$ — $\Theta_7$  — пересечение сигналов, команды  $\Theta_8$ ,  $\Theta_9$  — ветвление сигналов, команды  $\Theta_{10}$ — $\Theta_{12}$  реализуют логическую функцию И от двух входов, а команды  $\Theta_{13}$ ,  $\Theta_{14}$  — одноходовую логическую функцию НЕ.

Предлагаемое в следующем разделе преобразование двумерной структуры в трехмерную было разработано с использованием системы моделирования алгоритмов параллельных подстановок [9, 10]. Основные черты компьютерной модели клеточной структуры покажем на примере модели для структуры (см. рис. 1, 2). В первую очередь присвоим имена всем различным графическим изображениям конфигураций в командах подстановок. Например, базу команды  $\Theta_1$  назовем *srll*, правую часть команды  $\Theta_1$  — *srgl*, левую часть команды  $\Theta_{15}$  — *blue*, правую часть команды  $\Theta_{15}$  — *red* и т. д. Образы клеточных массивов и всех различных конфигураций отображаются на экране монитора. Преобразование клеточных массивов задается схемой программы, которая также отображается в специальном окне на экране. В этой схеме итеративное синхронное применение СПП Ф задается оператором *ex*. Оператор *in* с именем клеточного массива (или нескольких массивов, разделенных символом «\*») задает то клеточное пространство, где применяется команда подстановки. Сама команда подстановки задается двумя операторами: *at* с именем конфигурации (или перечнем имен конфигураций, разделенных символом «\*» и расположенных в порядке, согласованном с порядком клеточных массивов в операторе *in*) из левой части и *do*, устроенным аналогично, но с использованием правой части команды.

Схема программы для структуры (см. рис. 1, 2) начинается строками:

```
ex
in sr * gr
```

Затем следуют пары строк, поставленные в соответствие командам подстановок  $\Theta_1$ — $\Theta_{14}$ . Например, команде  $\Theta_1$  ставятся в соответствие строки:

```
at srll * blue
do srgl
```

Далее снова следует строка с оператором *in*:

```
in gr
```

Затем две пары строк, поставленные в соответствие командам  $\Theta_{15}$ ,  $\Theta_{16}$ . Первая пара имеет вид:

at blue  
do red

Вторая пара записывается аналогично, только blue и red меняются местами. Описание схемы завершено.

От алгоритмического описания клеточной структуры можно перейти к ее представлению в виде сети автоматов [8, 9]. Автоматы сети, поставленные в соответствие клеткам двумерного массива  $sg$ , имеющим такие же координаты, как клетки массива  $gr$ , состояния которых отличны от 0, будем считать активными. Это означает, что каждый из них выполняет список микроопераций, порожденных командами  $\Theta_1—\Theta_{14}$ . Микрооперация порождается командой  $\Theta_i$ ,  $i = 1, \dots, 14$ , при подстановке в именуемые функции команды конкретные значения переменных, пусть это будут  $i_0, j_0$ , и обозначается  $\Theta_i((i_0, j_0))$ . Команды  $\Theta_{15}, \Theta_{16}$  фактически имитируют две тактирующие последовательности, а массив  $gr$  показывает, к автоматам с какими координатами они подключены. Все остальные автоматы сети пассивные: это ячейки памяти, в которые может записываться и с которых может считываться информация. Чтобы построить связи между автоматами, нужно только заметить, что для выполнения микроопераций списка каждый активный автомат с координатами  $(i_0, j_0)$  должен получать информацию от своих ближайших соседей — автоматов с координатами  $(i_0, j_0 + 1)$ ,  $(i_0 + 1, j_0)$ ,  $(i_0 + 1, j_0 + 1)$ , т. е. иметь входы, соединенные с их информационными выходами, и записывать информацию в ячейки памяти своих ближайших соседей, т. е. иметь выходы, соединенные с их входами записи. Так как сложность активных автоматов, составляющих половину от общего числа автоматов в сети, определяется размером списка микроопераций, то любое расширение функциональных возможностей универсальной структуры, требующее увеличения числа команд подстановок, приводит к увеличению сложности каждого активного автомата, а значит, и сети в целом. Зададимся вопросом: можно ли избежать такого усложнения сети? Да, если заметить, что каждый активный автомат структуры, после того как его роль в клеточном представлении цифрового устройства определена, никогда не выполняет более трех микроопераций. Следовательно, в каждом активном автомате можно иметь память для хранения только трех микроопераций, если одновременно с заданием клеточного представления устройства произвести запись необходимых микроопераций в память автомата. Это фактически означает введение настройки в сеть. Переход к многослойной структуре, о котором речь пойдет в следующем разделе, позволяет вынести такую настраиваемую память в отдельные слои. В этом случае при ориентации на электрооптическую схемотехнику [11—13] обеспечивается существенное упрощение топологии слоев за счет превращения большей части межавтоматных и внутриавтоматных связей двуслойной структуры в межслойные оптические связи, а также дальнейшее уменьшение аппаратной сложности сети за счет возможности использования отдельных участков настраиваемой памяти не одним, а несколькими активными автоматами операционного слоя сети по очереди.

2. Процедура преобразования двумерной универсальной структуры в трехмерную универсальную структуру. Процедура состоит из двух больших частей: построения компьютерной модели универсальной структуры и представления структуры в виде сети автоматов.

2.1. Построение компьютерной модели трехмерной универсальной структуры. 2.1.1. Назначение слоев. Объединению клеточных массивов  $sg$  и  $gr$ , каждый из которых имеет размеры  $m \times n$ , ставится в соответствие пятислойный клеточный массив  $srd$  с номерами слоев 0, 1, 2, 3, 4. Все слои имеют одинаковые размеры не менее чем  $(2m + 1) \times (2n + 1)$ . Разделение двумерной универсальной структуры на слои в своей основе содержит ту же идею, что и построение однородной машины [8, 9] по сети автоматов, интерпретирующей алгоритм параллельных подстановок. Идея состоит в том, что память команд, повторяющаяся в активных автоматах сети, выносится в отдельный блок. Оставшаяся сеть называется памятью данных. Активные авто-

маты в ней просты: они выполняют сравнение состояния своей окрестности с левой частью каждой команды из блока команд, а при успешном сравнении изменение состояния окрестности в соответствии с правой частью команды. В расслоенной структуре блок команд превращается в распределенную память микроопераций, разделенную на два слоя: нулевой слой, в который записаны базы микроопераций, порожденных командами  $\Theta_1$ — $\Theta_{14}$ , и четвертый слой, в который записаны правые части микроопераций, порожденных теми же командами. Память данных также делится на два слоя: первый (операционный), содержащий клеточное представление цифрового устройства; второй (регистрирующий), фиксирующий совпадение символов, записанных в соответствующие клетки нулевого и первого слоев. В третьем слое задается разбиение. Этот слой вводится только в компьютерную модель, в сети автоматов он сливается со вторым слоем. В целом можно заметить, что расслоение сделано так, чтобы и слоев было немного, и при этом функции клетки были максимально просты.

2.1.2. Конструирование слоев. При задании состояний клеток слоев, наряду с символами O, C, S, G, B, R, будут использоваться дополнительные символы: «безразличное состояние» (#,  $\square$ ), «совпадение» ( $\rightarrow$ ,  $\boxtimes$ ), «окно» ( $\leftarrow$ ,  $\boxminus$ ).

Детальное конструирование слоев выполним для небольшого примера двумерной структуры, а затем покажем, как перейти к общему случаю.

На рис. 3 изображено устройство слоев массива *srd* трехмерной структуры, полученной из двумерной структуры, представленной массивами *sg* и *gr* размером  $3 \times 4$ . Массив *sg* содержит горизонтальный канал в первом ряду. Если клетку с координатами  $i = 1, j = 0$  массива *sg* перевести в состояние S, то движение сигнала по горизонтальному каналу описывают микрооперации  $\Theta_2((0, 0))$ ,  $\Theta_1((1, 1))$ ,  $\Theta_2((0, 2))$ , порожденные командами  $\Theta_1, \Theta_2$ . Для удобства читателя слои массива *srd* помечены номерами и расположены друг под другом, а не лесенкой, как на экране монитора.

Опишем получение всех пяти слоев.

Третий слой. Построение слоев массива *srd* начинается с третьего слоя, потому что он является опорным для остальных слоев структуры: в нем задается разбиение. Этот слой является образом массива *gr*: каждой клетке с координатами  $\langle i, j \rangle$  массива *gr* ставится в соответствие гнездо клеток с координатами  $\langle 3, 2i, 2j \rangle$ ,  $\langle 3, 2i + 1, 2j \rangle$ ,  $\langle 3, 2i, 2j + 1 \rangle$ ,  $\langle 3, 2i + 1, 2j + 1 \rangle$ . Состояние клетки с координатами  $\langle 3, 2i, 2j \rangle$  совпадает с состоянием клетки, породившей гнездо. Клетка с координатами  $\langle 3, 2i + 1, 2j \rangle$  устанавливается в состояние O, клетки с координатами  $\langle 3, 2i, 2j + 1 \rangle$ ,  $\langle 3, 2i + 1, 2j + 1 \rangle$  — в состояние #. Цель построенного отображения — превращение двух разбиений массива *gr* в два подразделения третьего слоя массива *srd*. Любой элемент первого подразделения выглядит так же, как блок, составленный из клеток с координатами  $\langle 3, 0, 0 \rangle$ ,  $\langle 3, 0, 1 \rangle$ ,  $\langle 3, 0, 2 \rangle$ ,  $\langle 3, 0, 3 \rangle$ ,  $\langle 3, 1, 0 \rangle$ ,  $\langle 3, 1, 1 \rangle$ ,  $\langle 3, 1, 2 \rangle$ ,  $\langle 3, 1, 3 \rangle$ . Элемент второго подразделения отличается тем, что у него верхняя левая клетка находится в состоянии R. Эту клетку и в том, и в другом подразделении будем называть опорной.

Все остальные слои делятся на блоки клеток, которые расположены точно над (под) элементом разбиения, образуя этажерку. При дальнейшем построении слоев будем соблюдать следующий принцип: пусть блок первого слоя содержит фраг-

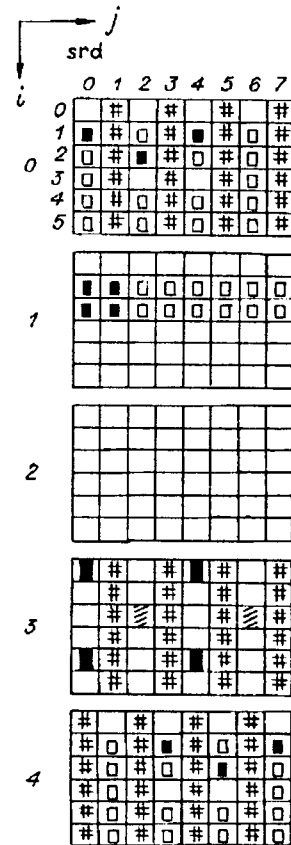


Рис. 3

мент клеточного представления цифрового устройства, тогда блок нулевого (четвертого) слоя содержит запись базы (правой части) той микрооперации, которая может быть применена к этому фрагменту.

Первый слой. Этот слой является образом массива  $sg$ : каждой клетке с координатами  $\langle i, j \rangle$  массива  $sg$  ставится в соответствие гнездо клеток с координатами  $\langle 1, 2i - 1, 2j \rangle, \langle 1, 2i, 2j \rangle, \langle 1, 2i - 1, 2j + 1 \rangle, \langle 1, 2i, 2j + 1 \rangle$ . Клетки с координатой  $2j$  называются основными, с координатой  $2j + 1$  — буферными. Состояния клеток гнезда дублируют состояние породившей их клетки массива  $sg$ . Легко заметить, что линии рисунка клеточного представления устройства получаются вдвое шире, чем в двумерном случае. Это сделано с той целью, чтобы для выбранного варианта разбиения, в котором подразделения не пересекаются, обеспечить возможность перемещения и преобразования сигналов в структуре за счет наложения образов баз микроопераций, применимых на соседних итерациях, на дубликаты клеток, задающих клеточное представление имитируемого устройства.

Нулевой (четвертый) слой. Установим вначале все клетки четных столбцов нулевого слоя массива  $sgd$  в состояние  $C$ , а нечетных — в состояние  $\#$ , в четвертом слое сделаем наоборот. Затем запишем базы (правые части) микроопераций в нулевой (четвертый) слой массива  $sgd$ . База микрооперации  $\Theta_2(\langle 0, 0 \rangle)$  записывается так: состояние  $O$  клетки с координатами  $\langle 0, 0 \rangle$  присваивается клетке с координатами  $\langle 0, 0, 0 \rangle$ , состояние  $O$  клетки с координатами  $\langle 0, 1 \rangle$  — клетке с координатами  $\langle 0, 0, 2 \rangle$ , состояние  $S$  клетки с координатами  $\langle 1, 0 \rangle$  — клетке с координатами  $\langle 0, 1, 0 \rangle$ , состояние  $C$  клетки с координатами  $\langle 1, 1 \rangle$  — клетке с координатами  $\langle 0, 1, 2 \rangle$  нулевого слоя. Аналогично база микрооперации  $\Theta_1(\langle 1, 1 \rangle)$  записывается в клетки с координатами  $\langle 0, 2, 2 \rangle, \langle 0, 2, 4 \rangle, \langle 0, 3, 2 \rangle, \langle 0, 3, 4 \rangle$ , а база микрооперации  $\Theta_2(\langle 0, 2 \rangle)$  — в клетки с координатами  $\langle 0, 0, 4 \rangle, \langle 0, 0, 6 \rangle, \langle 0, 1, 4 \rangle, \langle 0, 1, 6 \rangle$  нулевого слоя. При записи правой части микрооперации  $\Theta_2(\langle 0, 0 \rangle)$  в четвертый слой есть важное отличие, состоящее в сдвиге состояний клеток вправо: состояние  $O$  клетки с координатами  $\langle 0, 0 \rangle$  присваивается клетке с координатами  $\langle 4, 0, 1 \rangle$ , состояние  $O$  клетки с координатами  $\langle 0, 1 \rangle$  — клетке с координатами  $\langle 4, 0, 3 \rangle$ , состояние  $C$  клетки с координатами  $\langle 1, 0 \rangle$  — клетке с координатами  $\langle 4, 1, 1 \rangle$ , состояние  $S$  клетки с координатами  $\langle 1, 1 \rangle$  — клетке с координатами  $\langle 4, 1, 3 \rangle$ . Аналогично правая часть микрооперации  $\Theta_1(\langle 1, 1 \rangle)$  записывается в клетки с координатами  $\langle 4, 2, 3 \rangle, \langle 4, 2, 5 \rangle, \langle 4, 3, 3 \rangle, \langle 4, 3, 5 \rangle$ , а правая часть микрооперации  $\Theta_2(\langle 0, 2 \rangle)$  — в клетки с координатами  $\langle 4, 0, 5 \rangle, \langle 4, 0, 7 \rangle, \langle 4, 1, 5 \rangle, \langle 4, 1, 7 \rangle$  четвертого слоя.

Второй слой. Клетки второго слоя устанавливаются в состояние  $O$ .

Прежде чем переходить к дальнейшим пунктам, приведем дополнительные сведения о компьютерной модели (разд. 1), которые нам требуются. Первое дополнение касается введения операторов, представляющих параллельные функциональные подстановки [8, 9], т. е. подстановки, содержащие (наряду с конкретными символами некоторого алфавита) в записи левых частей переменные символы, правых частей — функциональные символы. Значения как тех, так и других определяются при выполнении подстановок. В компьютерном представлении таких подстановок оператор  $at$  заменяется оператором  $ab$ . В соответствующие клетки конфигурации, имя которой следует после оператора  $ab$ , вписываются имена термов (переменных и функциональных символов). После оператора  $do$  следует имя функции, в которой указано, каким из термов присваиваются новые значения. Каждая функция является записью одного оператора языка Си и показывается в специальном окне монитора. Второе дополнение касается правил выполнения операторов. Итерация может состоять из последовательности шагов. Шаг задается оператором  $ch$ , предшествующим оператору  $in$ . Изменения данных в клеточных массивах происходят после каждого шага. Третье дополнение касается использования трехмерных клеточных массивов и конфигураций. Для указания слоя массива, по клеткам которого перемещается ближайший к наблюдателю слой конфигурации, используется оператор  $op$  с номером слоя. Оператор  $op$  следует после оператора  $in$ . И в заключение два замечания технического характера. Строки в записи функций и схеме программы, начинающиеся с символа «;», являются

комментарием. Пара операторов at (ab), do, соответствующая команде подстановки, далее называется просто оператором с присвоенным ему номером.

### 2.1.3. Описание функций.

```
fsup ()
{
  x=y ? z=>;
}
```

Эта функция реализует первую элементарную операцию (см. введение).

```
frew ()
{
  p=0 && q==< ? x=z, y=z;
}
```

```
fcop ()
{
  z!=# ? x=y;
}
```

Эта пара функций реализует вторую элементарную операцию (см. введение).

2.1.4. Конструирование схемы программы. Схема программы, описывающей функционирование трехмерной структуры, представленной массивом *src* (см. рис. 3), и графическое представление конфигураций, используемых в схеме, изображены на рис. 4.

Дадим содержательный комментарий к схеме программы. Итерация, заданная оператором *ex*, состоит из следующих шагов.

Первый шаг. Оператор 1 схемы переводит в состояние  $\rightarrow$  те клетки во втором слое, которые расположены под клетками с совпадающими состояниями нулевого и первого слоев.

Второй шаг. Оператор 2 выделяет вхождения баз микроопераций в первый слой, отыскивая во втором слое для каждой опорной клетки в состоянии В и с координатами  $\langle 3, i, j \rangle$  образы, состоящие из четырех клеток, которые имеют состояние  $\rightarrow$  и координаты  $\langle 2, i, j \rangle$ ,  $\langle 2, i, j + 2 \rangle$ ,  $\langle 2, i + 1, j \rangle$ ,  $\langle 2, i + 1, j + 2 \rangle$ , и переводя клетки с координатами  $\langle 2, i, j + 1 \rangle$ ,  $\langle 2, i, j + 3 \rangle$ ,  $\langle 2, i + 1, j + 1 \rangle$ ,  $\langle 2, i + 1, j + 3 \rangle$  в состояние  $\leftarrow$ . Третий оператор выполняется одновременно со вторым и играет служебную роль. Он убирает «мусор», переводя все клетки второго слоя, находившиеся в состоянии  $\rightarrow$ , в состояние 0.

Третий шаг. Наличие клеток в состоянии  $\leftarrow$  для элемента разбиения с опорной клеткой в состоянии В и с координатами  $\langle 3, i, j \rangle$  служит приказом для переноса состояний клеток, хранящих правую часть микрооперации, в буферные клетки первого слоя: состояние клетки с координатами  $\langle 4, i, j + 1 \rangle$  присваивается клетке с координатами  $\langle 1, i - 1, j + 1 \rangle$ ,  $\langle 1, i, j + 1 \rangle$ , состояние клетки с координатами  $\langle 4, i, j + 3 \rangle$  — клетке с координатами  $\langle 1, i - 1, j + 3 \rangle$ ,  $\langle 1, i, j + 3 \rangle$ , состояние клетки с координатами  $\langle 4, i + 1, j + 1 \rangle$  — клетке с координатами  $\langle 1, i + 2, j + 1 \rangle$ ,  $\langle 1, i + 1, j + 1 \rangle$ , состояние клетки с координатами  $\langle 4, i + 1, j + 3 \rangle$  — клетке с координатами  $\langle 1, i + 2, j + 3 \rangle$ ,  $\langle 1, i + 1, j + 3 \rangle$ . Перенос выполняют операторы 4, 5 схемы. Оператор 6 служебный, он убирает «мусор».

Четвертый шаг. Оператор 7 в первом слое осуществляет запись состояний буферных клеток в основные клетки, операторы 8, 9 выполняют смену подразбиений.

2.1.5. Коррекция компьютерной модели трехмерной структуры. Выше фактически была описана модель, в которой над (под) отдельными фрагментами клеточного представления цифрового устройства размещались записи базы и правой части единственной микрооперации подстановки. Однако не менее часто требуется разместить две или три микрооперации (например, в случае фрагмента, изображающего разветвление, пересечение или вентиль).



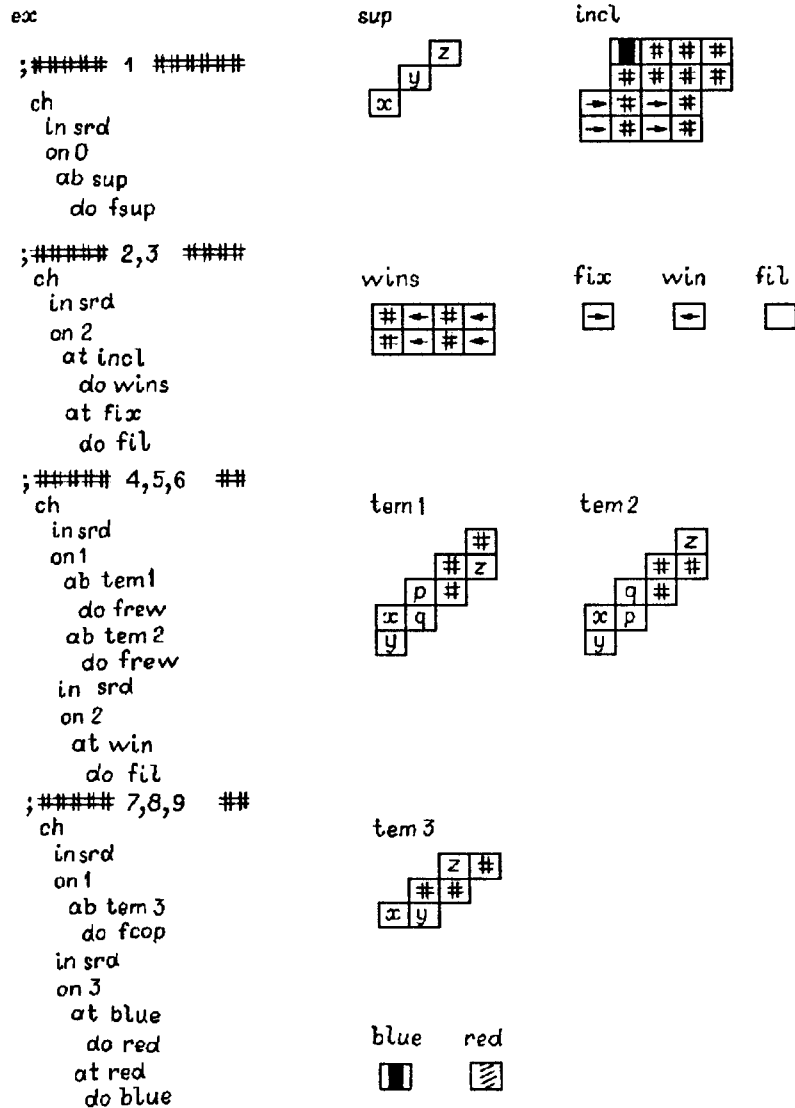


Рис. 4

Коррекция модели состоит в учете этого требования. Теперь записи микроопераций размещаются не только в той этажерке, где помещен фрагмент, с которым они связаны, но и в соседних по оси  $i$  этажерках, причем таких, которые относятся к тому же подразбиению, что и этажерка, содержащая фрагмент. Более того, часто не требуется, чтобы исходно запись какой-либо микрооперации помещалась в той же этажерке, что и фрагмент клеточного представления устройства; достаточно, если запись и фрагмент совместятся при сдвиге информации в первом слое вдоль оси  $i$ . Конечно, при этом могут возникать наложения записей друг на друга. Чтобы избежать этого, нужно проводить размещение микроопераций в нулевом и четвертом слоях одновременно с размещением клеточного представления устройства в первом слое, если необходимо, видоизменяя представление устройства еще в двумерном случае.

Коррекция затрагивает также описание функционирования трехмерной структуры, так как теперь необходимо обеспечить встречу фрагмента по очереди с записями трех микроопераций. Для этого следует ввести возможность сдвига всего содержимого первого слоя вдоль оси  $i$ . Мы не будем приводить

полностью формальное описание как в разд. 2.1.3 и 2.1.4, чтобы не загромождать изложение, поскольку, по сути, коррекция описания очень проста. Укажем только, что текст схемы (см. рис. 4) с изъятими операторами 7, 8, 9 становится ядром нового описания, которое задает функционирование структуры последовательностью вида

$$K \uparrow K \uparrow K \{789\} K \downarrow K \downarrow K \{789\}$$

Последовательность итеративно повторяется, в ней  $K$  — выполнение ядра,  $\uparrow, \downarrow$  — сдвиги содержимого первого слоя вдоль оси  $i$  вверх и вниз,  $\{789\}$  — параллельное выполнение операторов 7, 8, 9. Конечно, к описанию схемы добавляется описание новой функции, которая реализует третью элементарную операцию (см. введение), и графическое изображение двух конфигураций.

Следует также отметить, что, записав микрооперации в отдельные слои, мы получили возможность приписывать фрагментам имитируемого устройства микрооперации любых булевых функций от двух переменных. Можно также, строя некоторые каналы из символов  $S$ , а в качестве сигнала используя символ  $C$ , создавать делители тактовых импульсов, представлять обратную связь в имитируемом устройстве и т. д.

На рис. 5 дано клеточное представление 4-разрядного сумматора с пирамидой переносов [14] для двумерного случая. Изображение повернуто на  $90^\circ$  против часовой стрелки по сравнению с тем же изображением на экране монитора. В массиве  $g$  клетки в состоянии  $B$  имеют координаты  $(2i + 1, 2j)$ ,  $i, j = 0, 1, 2, \dots$ , клетки в состоянии  $R$  — координаты  $(2i, 2j + 1)$ ,  $i, j = 0, 1, 2, \dots$ . Клеточное представление сумматора построено с учетом последующего отображения в первый слой и размещения микроопераций в нулевом и четвертом слоях трехмерной структуры. По аналогии с командами подстано-

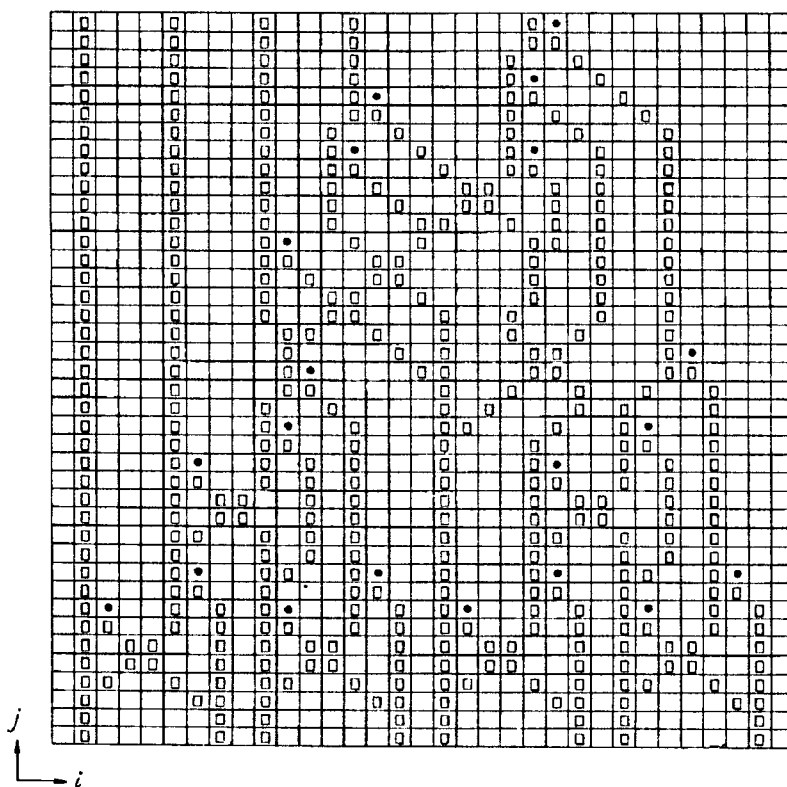


Рис. 5

вок, задающими операцию И, записываются команды подстановок, задающие функции ИЛИ и МОД2 (сложение по mod2). Затем производится распределение микроопераций, задающих эти логические функции. К вентилям с координатами  $\langle 6, 9 \rangle$ ,  $\langle 10, 17 \rangle$ ,  $\langle 13, 32 \rangle$ ,  $\langle 14, 9 \rangle$ ,  $\langle 21, 32 \rangle$ ,  $\langle 21, 36 \rangle$ ,  $\langle 22, 9 \rangle$  привязываются микрооперации для И, к вентилям с координатами  $\langle 2, 7 \rangle$ ,  $\langle 10, 7 \rangle$ ,  $\langle 10, 27 \rangle$ ,  $\langle 14, 35 \rangle$ ,  $\langle 18, 7 \rangle$ ,  $\langle 22, 15 \rangle$ ,  $\langle 26, 7 \rangle$  — микрооперации для МОД2, к вентилям с координатами  $\langle 11, 20 \rangle$ ,  $\langle 22, 39 \rangle$ ,  $\langle 28, 21 \rangle$  — микрооперации для ИЛИ. Клетки с координатами  $\langle 25, 0 \rangle$ ,  $\langle 17, 0 \rangle$ ,  $\langle 9, 0 \rangle$ ,  $\langle 1, 0 \rangle$  служат входными для разрядов первого слагаемого, клетки с координатами  $\langle 31, 0 \rangle$ ,  $\langle 23, 0 \rangle$ ,  $\langle 15, 0 \rangle$ ,  $\langle 7, 0 \rangle$  — для разрядов второго слагаемого. Установка входной клетки в состояние S соответствует наличию единицы в разряде слагаемого. Сумма получается в выходных клетках с координатами  $\langle 21, 39 \rangle$ ,  $\langle 13, 39 \rangle$ ,  $\langle 9, 39 \rangle$ ,  $\langle 5, 39 \rangle$ ,  $\langle 1, 39 \rangle$ . Отметим высокую производительность устройства: каждую следующую пару слагаемых можно записывать во входные клетки на том шаге, когда происходит выполнение оператора 7.

2.2. *Представление трехмерной универсальной структуры в виде сети автоматов.* Как указано в разд. 1, алгоритмическое описание задает действия, выполняемые автоматами сети, а список конфигураций (см. рис. 4) — связи между ними.

2.2.1. Описание структуры сети. Сеть получается четырехслойной, так как слой, задающий разбиение, можно убрать, совместив его со слоем, фиксирующим совпадения. Части слоев сети изображены на рис. 6. Блоки, на которые делятся слои, изображены штриховыми линиями на рис. 6, а. Блоки одной и той же этажерки занимают одинаковые позиции на рис. 6, а — с. Незаштрихованный квадрат обозначает элемент сети, который назовем

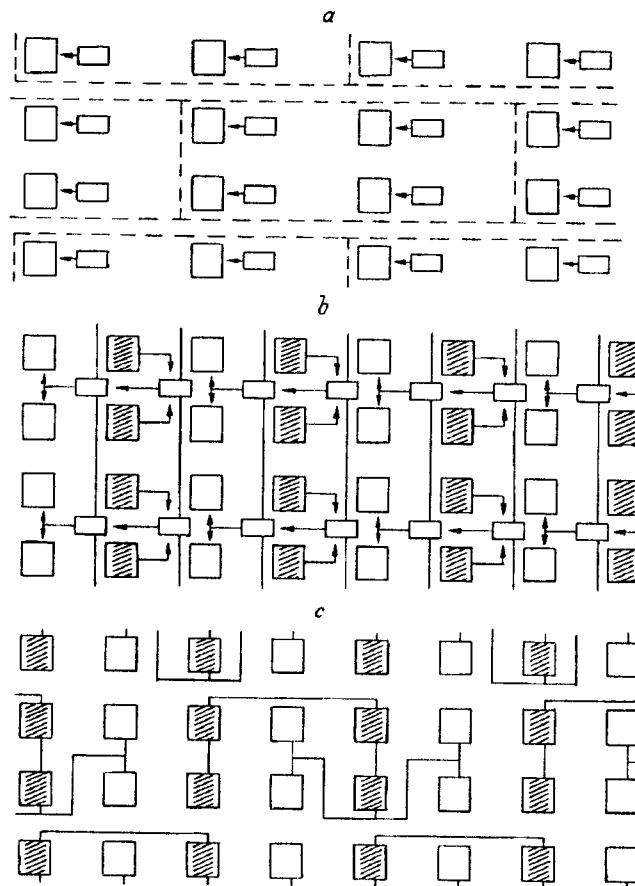


Рис. 6

фильтром, заштрихованный квадрат назовем приемником. Меньший по величине прямоугольник — это изображение четырехзначной ячейки памяти. На рис. 6, *a* приведено изображение части нулевого слоя: памяти, хранящей коды баз микроопераций подстановок. Слой составлен из столбцов, содержащих пары (фильтр, ячейка памяти) и разделенных пустыми промежутками, соответствующими столбцам, заполненным символами  $\#$  в модели. Изображение части третьего слоя: памяти, хранящей правые части микроопераций, — отличается от верхнего фрагмента только тем, что столбцы и пустые промежутки переставлены местами, и поэтому на рис. 6 не приведено. Рис. 6, *b* — это изображение части первого слоя, ячейки памяти которого хранят клеточное представление цифрового устройства. В этом слое используются ячейки памяти двух типов: основные и буферные. Выход основной ячейки памяти соединен со входами двух фильтров. Вход записи буферной ячейки памяти соединен с выходами двух приемников. В двух соседних столбцах выход буферной ячейки памяти соединен со входом записи основной ячейки памяти. Вертикальные шины обеспечивают сдвиг информации вверх и вниз вдоль оси  $i$ . Рис. 6, *c* — это изображение части второго слоя: слоя совпадений. В этом слое для каждого элемента разбиения четыре приемника образуют схему совпадений, у которой нагрузкой служат четыре параллельно соединенных фильтра. Тактирующие шины, задающие разбиение, подсоединяются к точкам соединения приемника и нагрузки (на рис. 6, *c* не показаны).

2.2.2. Описание функционирования сети. Введем понятие смешанного сигнала. Он содержит три составляющих  $\omega_1, \omega_2, \omega_3$ , поставленных в соответствие состояниям  $C, S, G$ , и поступает на каждый фильтр нулевого слоя перпендикулярно его площади сверху, на каждый фильтр третьего слоя перпендикулярно его площади снизу. Установим следующие требования к элементам, составляющим сеть. Фильтры нулевого, первого и третьего слоев настраиваются ячейками памяти: например, если ячейка имеет состояние  $C$ , то связанный с ней фильтр пропускает только составляющую  $\omega_1$  смешанного сигнала и т. д. В отличие от этих фильтров фильтры второго слоя широкополосные: когда они открыты, они пропускают все составляющие смешанного сигнала. Перейдем теперь к приемникам. Приемник первого слоя устанавливает буферную ячейку памяти в то состояние, которое соответствует составляющей смешанного сигнала, поступившей на него. Если никакая составляющая смешанного сигнала на приемник не поступает, он не оказывает влияния на состояние буферной ячейки памяти. Иначе ведет себя приемник второго слоя: при подаче на него любой составляющей смешанного сигнала он открывается («заключивает» свой участок схемы совпадений). Если никакая составляющая смешанного сигнала на приемник не поступает, он остается закрытым. Если все четыре приемника открыты, открываются фильтры, иначе они остаются закрытыми и не пропускают никаких составляющих смешанного сигнала. Чередование подразбиений задается во втором слое с помощью двух тактирующих последовательностей, одна из которых поступает на схемы совпадений, поставленные в соответствие первому подразбиению, другая — второму. Тактирующая последовательность обеспечивает замыкание или разрыв участка схемы совпадений там, где приемник соединяется с фильтрами.

Если задать функционирование сети итеративно повторяющейся последовательностью срабатываний

$$\frac{W_b \uparrow W_b \uparrow W_b R}{T_1} \quad \frac{W_b \downarrow W_b \downarrow W_b R}{T_2},$$

где  $W_b$  — запись в буферную ячейку памяти первого слоя состояния ячейки памяти третьего слоя;  $\uparrow, \downarrow$  — сдвиги содержимого ячеек памяти первого слоя вверх и вниз;  $R$  — перезапись в первом слое состояния буферной ячейки памяти в основную ячейку памяти;  $T_1$  — такт первой тактирующей последовательности;  $T_2$  — такт второй тактирующей последовательности, то можно убедиться, что функционирование сети в основном соответствует функционированию компьютерной модели. Некоторые отличия объясняются тем, что в

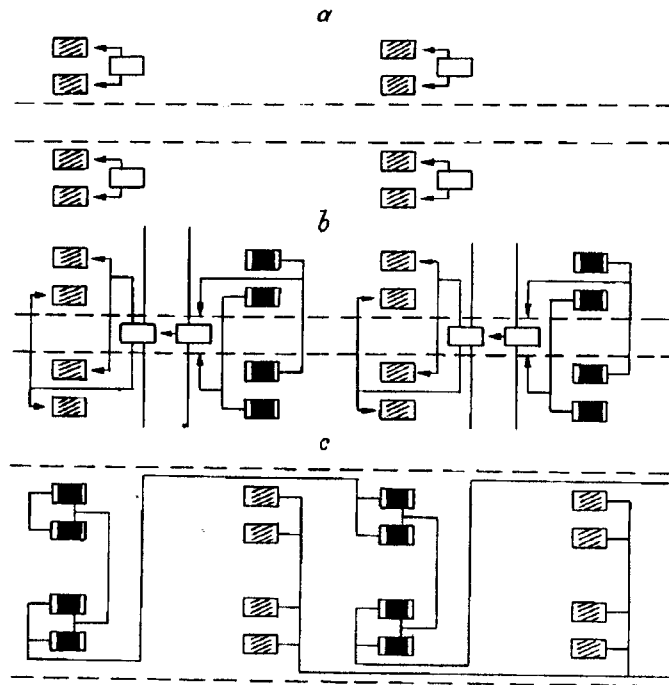


Рис. 7

сети фильтры и приемники собственной памяти не имеют (в отличие от соответствующих им клеток в модели), и поэтому отпадает необходимость в выполнении операторов, названных в модели служебными.

2.2.3. Переход от сети с многозначными ячейками памяти к сети с двоичными ячейками памяти. В разд. 2.2.1 сеть описана в достаточно общем виде. Конкретизируем представление сети, используя схемотехнику, предложенную в [11—13]. Четырехзначная ячейка памяти заменяется двумя потенциальными триггерами с двумя выходами (если нужно, то и с двумя входами), фильтр — модулятором, приемник — фотоприемником. Смешанный сигнал заменяется поляризованным светом, падающим на плоскости нулевого и третьего слоев сверху и снизу. На рис. 7 заштрихованные прямоугольники — это модуляторы, незаштрихованные — триггеры, зачерненные — фотоприемники. Границы блоков, порожденных подразделениями, показаны штриховыми линиями. На рис. 7, а для нулевого (аналогично для третьего) слоя сверху и снизу от границ блоков показаны пары схем. Каждая горизонтально расположенная пара заменяет четырехзначную ячейку памяти с фильтром на выходе. Двоичный фрагмент, заменяющий фрагмент из первого слоя, содержащий основную и буферную многозначные ячейки памяти с выходными модуляторами и входными приемниками, изображен на рис. 7, б. И наконец, на рис. 7, с показан фрагмент, заменяющий половину схемы совпадений из второго слоя. Используя изображенные на рис. 7 фрагменты, легко построить все слои структуры с двоичными ячейками памяти. Заметим, что модуляторы и фотоприемники первого и второго слоев из одной и той же этажерки образуют потенциальную электрооптическую ячейку [11]. Нами в качестве ячейки памяти в нулевом и третьем слоях был выбран триггер. Поэтому, чтобы иметь возможность перезаписи микроопераций электрическим путем, следует организовать из этих ячеек одномерный регистр сдвига. Очевидно, это не единственный вариант построения памяти, которая, в сущности, является матрицей модуляторов. Нам представляется, что память может получиться особенно простой, если в структуре реализуется единственный вариант клеточного представ-

ления цифрового устройства, т. е. нулевой и третий слои — это постоянная память. В этом случае память, например, может быть просто непрозрачной перфорированной пленкой.

**Заключение.** Многослойная универсальная структура, предложенная в статье, обладает следующими качествами:

— высокая технологичность: действительно, все слои структуры, кроме слоя регистрации, являются памятью с оптическими входами и выходами ячеек, требования к слоям памяти достаточно стандартны: запись (перезапись) информации и ее хранение, а для операционного слоя, кроме того, возможность сдвига хранимой информации; в свою очередь, и регистрирующий слой устроен весьма просто: он состоит из итеративно повторяющейся схемы совпадений;

— быстродействие, являющееся следствием глубокой конвейерности обработки данных в структуре: на входы реализованного в структуре цифрового устройства новый комплект данных можно подавать буквально после каждой перезаписи информации из буферных ячеек памяти в основные, даже если исходное устройство свойством конвейерной обработки данных не обладало;

— хорошая компактность и легкость размещения цифровой схемы в структуре, вытекающие из возможности использования при построении схемы большого набора преобразований состояний клеток (среди них, например, все преобразования, реализующие булевы функции от двух переменных).

#### СПИСОК ЛИТЕРАТУРЫ

1. Евреинов Э. В., Прангишвили И. В. Цифровые автоматы с настраиваемой структурой (однородные среды). М.: Энергия, 1974.
2. Codd E. Cellular Automata. New York, London: Academic Press, 1968.
3. Theory and Application of Cellular Automata /Ed. by S. Wolfram // World Scientific. Singapore, 1986.
4. Тоффоли Т., Марголюс Н. Машины клеточных автоматов. М.: Мир, 1991.
5. Bertin P., Roncin D., Vuillemin J. Introduction to programmable active memories // Systolic Array Processors /Ed. by J. McCanny, J. McWhirter, E. Swartzlander. N.Y: Prentice Hall, 1989.
6. Пискунов С. В. Построение многослойных электрооптических структур // Автометрия. 1992. № 4.
7. Пискунов С. В. Построение предельно расслоенных электрооптических клеточных устройств // Автометрия. 1993. № 3.
8. Методы параллельного микропрограммирования /Под ред. О. Л. Бандман. Новосибирск: Наука, 1981.
9. Achasova S., Bandman O., Markova V., Piskunov S. Parallel substitution algorithm. Theory and application // World Scientific. Singapore, 1994.
10. Погудин Ю. М. ALT — графическая система параллельного микропрограммирования // Параллельные алгоритмы и структуры. Новосибирск: ВЦ СО АН СССР, 1991.
11. Косцов Э. Г., Мишин А. И. Особенности построения оптических ЦВМ // Микроэлектроника. 1977. 6, вып. 2.
12. Егоров В. М., Косцов Э. Г. Перспективы создания оптических цифровых высокопроизводительных вычислительных устройств // Автометрия. 1985. № 1.
13. Egorov V. M., Kostsov E. G. Integrated optical digital computers // Appl. Opt. 1990. 29, N 8.
14. Карцев М. А., Брик В. А. Вычислительные системы и синхронная арифметика. М.: Радио и связь, 1981.

*Поступила в редакцию 9 февраля 1995 г.*