

ПЕРЕСТРАИВАЕМЫЕ АВТОМАТЫ И ОДНОРОДНЫЕ СТРУКТУРЫ ДЛЯ ПОСТРОЕНИЯ УПРАВЛЯЮЩИХ УСТРОЙСТВ

1.1. ВВОДНЫЕ ПОНЯТИЯ

В настоящее время существуют различные подходы к определению понятия конечного автомата, которые могут быть разбиты на группы *макроподхода* и *микроподхода*. При *макроподходе* интересуются внешним поведением устройства, тем, как оно осуществляет переработку входной информации в выходную информацию и последовательность состояний, отвлекаясь от внутреннего его состояния. На этом пути приходят к *понятию абстрактного конечного автомата*. Тем самым абстрактный конечный автомат может быть задан с помощью набора отображений, описывающих его «внешнее» функционирование. При *микроподходе* учитывается структура устройства, функционирование и связь между собой его частей. На этом пути приходят к *понятию структурного конечного автомата*, называемого также автоматной схемой, логической сетью либо многофункциональным логическим модулем (МЛМ). Структурный конечный автомат задается конечным множеством абстрактных автоматов, конечной схемой их соединения и указанием влияния частей схемы друг на друга. Понятия конечного абстрактного и конечного структурного автоматов можно считать составляющими понятия конечного автомата.

Обобщение понятия конечного автомата получается путем обобщения понятий конечных абстрактного и структурного автоматов.

Абстрактный автомат получается в результате выбора произвольных (необязательно конечных) множеств входных и выходных сигналов, множеств состояний, а также при расширении понятия зависимости состояния и выходного сигнала от входного сигнала и состояния. Структурный автомат получается в результате выбора произвольных множеств автоматов и схем их соединений.

1.2. СПОСОБЫ ЗАДАНИЯ ФУНКЦИЙ АЛГЕБРЫ ЛОГИКИ

Функция алгебры логики $f(x_1, \dots, x_n)$ полностью определяется заданием ее значений на всех наборах аргументов. Область определения в любой функции алгебры логики конечна, поскольку число аргументов и число значений каждого аргумента также конечны. Функции алгебры логики могут задаваться рядом способов [6]:

- 1) табличным, когда функция задается в виде таблицы истинности;
- 2) графическим, когда функция задается в виде n -мерного единичного куба;
- 3) координатным, когда функция задается в виде координатной карты состояний (карты Карно);
- 4) числовым, когда функция задается в виде десятичных или восьмеричных эквивалентов наборов тех аргументов, на которых она принимает значение единицы;
- 5) аналитическим, когда функция задается в виде аналитического выражения, получаемого путем совершения каких-либо логических операций с переменными алгебры логики, т.е. в виде формул, например:

$$f = x_1 \bar{x}_2 \vee x_3.$$

Здесь и далее символ \vee обозначает операцию «дизъюнкция», а операцию «конъюнкция», которая обозначается символом $\&$, будем заменять точкой или опускать какой-либо знак между аргументами.

1.3. КЛАССИФИКАЦИЯ АВТОМАТОВ

По количеству выполняемых преобразований информации все множество автоматов можно разделить на два класса: неперестраиваемые, или монофункциональные, автоматы; перестраиваемые, или многофункциональные, автоматы (M-автоматы).

Монофункциональный автомат имеет жесткую структуру и всегда реализует одно и то же преобразование $\{X\} \rightarrow \{Y\}$. Для такого автомата величина его функциональности $L = 1$.

К M-автоматам относятся все управляемые автоматы, реализующие путем их настройки некоторое множество $D_A = \{A_i\}, i = \overline{1, L}$, автоматных преобразований $\{X_i\} \rightarrow \{Y_i\}$, где $L > 1$.

Если функции переходов δ и выходов λ задаются на конечных множествах $D_\delta \subseteq \{X\} \times \{Z\}$ и $D_\lambda \subseteq \{X\} \times \{Z\}$ и М-автомат реализует любые автоматные преобразования (путем его настройки), определяемые парами функций $\delta^{(i)}$ и $\lambda^{(i)}$ и задаваемые на конечных множествах D_δ и D_λ , то такой автомат будем называть универсальным, или У-автоматом.

Выполнение в М-автомате требуемого автоматного преобразования $A_i \in D_A$ осуществляется соответствующей его настройкой. По принципу настройки М-автоматы делятся на три класса: с функциональной, структурной и программной настройкой.

Под *функциональной* настройкой М-автомата будем понимать такую, при которой код настройки Z_i не изменяется в течение всего времени выполнения преобразования A_i . При этом связи между функциональными элементами структурной схемы М-автомата могут оставаться прежними или изменяться в зависимости от реализуемого преобразования. В последнем случае изменение связей в структурной схеме М-автомата по существу эквивалентно преобразованию самой его структуры. Настройку автомата, в результате которой его структурная схема претерпевает соответствующие изменения (за счет изменения связей), будем называть *структурной* настройкой.

В *программно* настраиваемом автомате преобразования A_i выполняются за некоторое число n_{t_i} шагов, при этом код настройки на каждом шаге преобразования меняется.

Функциональная настройка является наиболее простым видом настройки и характерна для простейших М-автоматов (Мили, Мура, тривиальных и др.). Структурная настройка является более сложной и характерна для автоматов с более сложной структурной организацией, представляющей собой композицию некоторых функциональных элементов (автоматов) и элементов (автоматов) коммутации или связи. Код настройки такого М-автомата состоит из двух частей: кодов настроек функциональных автоматов и кодов настроек коммутационных автоматов. При этом сами функциональные и коммутационные автоматы могут быть функционально или структурно перестраиваемыми.

1.4. ПЕРЕСТРАИВАЕМЫЕ АВТОМАТЫ

Под перестраиваемым автоматом будем понимать модель широкого класса дискретных устройств, называемых иногда многофункциональными и универсальными логическими модулями, настраиваемыми логическими устройствами и т.п. [8].

Конечные детерминированные автоматы, называемые ниже просто автоматами, представляют собой объекты, имеющие конечное число внутренних состояний c_k ($k = 1, \dots, q$), конечное число входных сигналов x_i ($i = 1, \dots, n$) и конечное число выходных сигналов f_j ($j = 1, \dots, p$).

Автомат работает в дискретном времени t_v ($v = 1, 2, \dots$) и в каждый момент времени может находиться только в одном состоянии [8].

Пусть существует конечное число так называемых внутренних сигналов y_r ($r = 1, \dots, m$); сопоставим с каждым состоянием c_k набор значений сигналов y_r так, чтобы различным состояниям соответствовали различные наборы значений y_r .

Тогда в каждый момент времени значения внутренних и выходных сигналов можно задать как функции от входных и внутренних сигналов:

$$\begin{cases} y_1(v) = \alpha_1(x_1(v), \dots, x_n(v), y_1(v-1), \dots, y_m(v-1)), \\ \dots \\ y_m(v) = \alpha_m(x_1(v), \dots, x_n(v), y_1(v-1), \dots, y_m(v-1)); \end{cases} \quad (1.1)$$

$$\begin{cases} f_1(v) = \varphi_1(x_1(v), \dots, x_n(v), y_1(v-1), \dots, y_m(v-1)), \\ \dots \\ f_p(v) = \varphi_p(x_1(v), \dots, x_n(v), y_1(v-1), \dots, y_m(v-1)). \end{cases} \quad (1.2)$$

Описываемая модель автомата называется *структурным автоматом*, и, соответственно, системы (1.1) и (1.2) называются структурными функциями переходов и выходов.

Рассмотрим, например, комбинационный автомат, представленный на рис. 1.1. Обозначим сигналы, подаваемые на входы 1, 2, 3, 4, соответственно переменными x_1 , x_2 , z_1 , z_2 . При этом если сигнал на некотором входе равен 1, то значение соответствующей переменной равно 1, а если сигнал равен 0, то и значение соответствующей

переменной равно 0. Тогда функция $f(x_1, x_2, z_1, z_2)$, реализуемая на выходе автомата, имеет вид

$$f = \bar{x}_2 \bar{z}_1 z_2 \vee x_1 x_2 z_1 \vee \bar{x}_1 z_1 z_2.$$

Пусть в некоторый момент времени значения переменных z_1 и z_2 фиксированы и не изменяются. Нас интересует функция от переменных x_1 и x_2 , реализуемая на выходе автомата. Тогда автомат может реализовывать любую из четырех формул от двух переменных x_1 и x_2 :

при $z_1 = 0, z_2 = 0$

$$f = 0; \quad (1.3)$$

при $z_1 = 0, z_2 = 1$

$$f = \bar{x}_2; \quad (1.4)$$

при $z_1 = 1, z_2 = 0$

$$f = x_1 x_2; \quad (1.5)$$

при $z_1 = 1, z_2 = 1$

$$f = x_1 x_2 \vee \bar{x}_1. \quad (1.6)$$

Фиксацию сигналов на входах 3, 4 назовем *настройкой*, а сами входы 3, 4 и соответствующие им переменные z_1 и z_2 – *настроечными*. В интегральной технологии различают два этапа настройки: *жесткую* (схемная, технологическая, постоянная) настройку, выполняемую путем травления, выжигания при изготовлении интегральной схемы, и *мягкую* (программная, оперативная, переменная, гибкая), выполняемую многократно в процессе использования схемы. Здесь и в дальнейшем рассматривается настройка второго этапа, называемая просто *настройкой*. Входы 1, 2 назовем *информационными*. Выражения (1.3) – (1.6) задают множество реализуемых автоматом формул и алгоритм настройки на любую из них.

Таким образом, любой автомат путем фиксации значений сигналов на части его входов может реализовывать некоторое множество автоматных отображений наборов значений выходных сигналов, поступающих на остальные входы, в выходные сигналы.

Перестраиваемым называется автомат, для которого задано множество реализуемых им автоматных отображений и определен алгоритм настройки на реализацию каждого из этих автоматных отображений [8].

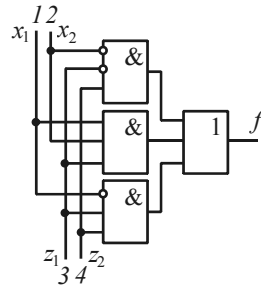


Рис. 1.1. Пример перестраиваемого автомата

Перестраиваемый автомат можно представить как множество автоматов с одними и теми же выходами, причем настройка определяет тот автомат, выходы которого считаются при этой настройке выходами всего перестраиваемого автомата. Поэтому перестраиваемый автомат ничего, кроме автоматных отображений, реализовать не может.

В случае, если перестраиваемый автомат при любых настройках формирует значение сигналов на его выходах, зависящих только от значений сигналов на его входах, он называется *комбинационным перестраиваемым автоматом*, или *многофункциональным логическим модулем*.

Если же для некоторых настроек значения сигналов на выходах автомата зависят от его состояния, то он называется *перестраиваемым автоматом с памятью*.

В основе построения автоматов с перестраиваемой структурой лежат три принципа.

1. *Параллельность*. Повышение быстродействия вычислительных средств за счет повышения тактовой частоты и параллельного (одновременного) выполнения большого числа операций, причем по мере повышения требований к быстродействию вклад от параллельного выполнения операций увеличивается [3].

2. *Перестраиваемость*. Надежность, гибкость и структурная универсальность (возможность создания для каждой задачи соответствующей структуры) управляющих вычислительных средств обеспечиваются схемно за счет программного изменения связей между элементами и автоматных функций самих элементов.

3. *Однородность*. Простота технологии изготовления вычислительных средств обеспечивается благодаря использованию одинаковых элементов и однотипных связей между ними.

1.5. ОДНОРОДНЫЕ СТРУКТУРЫ

Однородные структуры относятся к одному из важнейших видов управляющих систем. Они представляют собой дискретную математическую модель широкого класса реальных систем вместе с протекающими в них процессами.

Однородными структурами будем называть структуры [7], состоящие из функциональных ячеек и соединений между ними, которые отвечают следующим основным требованиям:

1) структура представляет собой геометрически правильную решетку, имеющую не менее двух осей симметрии, в узлах которой расположены функциональные ячейки (автоматы);

2) все функциональные ячейки структуры однотипны;

3) каждая функциональная ячейка соединена геометрически одинаковым образом с соседними, причем каждую из ячеек можно условно считать центром симметрии относительно связей ее с окружающими ячейками структуры;

4) функциональная ячейка структуры может быть настроена на выполнение в данный момент любой одной функции хотя бы одного полного базиса и функции канала передачи сигналов в заданном направлении, т.е. ячейка должна обладать функциональной и соединительной полнотой;

5) сигнал от любой ячейки A может быть передан к любой ячейке B структуры (хотя бы при помощи других ячеек);

6) изотропность структуры, вытекающая из предыдущих требований, позволяет реализовать любую заданную функцию на любом участке или области структуры.

Перечисленным признакам может отвечать большое количество разнообразных структур, отличающихся друг от друга геометрией решетки, функциональной схемой ячейки, количеством и характером связей, способом настройки и т.п.

Если в определенный момент времени настроить функциональные ячейки структуры на внутреннее состояние с учетом их информационных связей, то в однородной среде начнутся изменения состояний ячеек, определяемые законами их функционирования и связями между ними. Таким образом, развитие процесса в однородных структурах осуществляется во времени и пространстве с учетом причинно-следственных связей внутри них и, кроме того, в параллельном режиме.

Структуры, характеризующиеся не всеми указанными признаками, а лишь частью их, будем считать не полностью однородными.

Очевидно, что однородные структуры могут быть условно классифицированы по многим признакам, отражающим их особенность [7].

Так, например, однородные структуры могут быть классифицированы по:

1) метрике пространства: одно-, двух-, и трехмерные;

- 2) временным параметрам изменения состояний структуры: синхронные, асинхронные и смешанные;
- 3) характеру выбора направления передачи сигналов: на структуры с индивидуальным выбором и структуры с коллективным выбором;
- 4) числу возможных направлений передачи сигналов и числу связей с окружающими ячейками: структуры с двумя, тремя, четырьмя и большим числом связей и направлений передачи сигналов;
- 5) способам настройки ячеек структуры: сигналами извне (непосредственный, координатный, пошаговый, адресный способ и др.) или сигналами от соседних ячеек;
- 6) способам подачи входных сигналов и отбора выходных сигналов: по краям или изнутри структуры;
- 7) функциональной способности ячейки структуры (набор функций, реализуемых ячейкой);
- 8) степени универсальности структур: универсальные и специализированные.

1.5.1. Синхронные и асинхронные структуры

Однородные структуры называются *синхронными*, если изменения логических уравнений сигналов ячеек синхронизованы общей тактовой частотой. В синхронных структурах можно предсказать состояние всякой ячейки в момент времени, соответствующий любому синхронизирующему импульсу, если известны входные переменные и управляющие сигналы, а также свойства и состояния структуры.

В *асинхронных* структурах, в которых работа ячеек не синхронизована общей тактовой частотой, предсказание состояний ячеек невозможно, поскольку каждая операция осуществляется со скоростью, определяемой параметрами этих ячеек. В таких структурах при завершении операции образуется сигнал, который управляет началом следующей операции.

Асинхронные структуры в общем случае позволяют обеспечить большее быстроедействие реализуемых в них устройств, чем синхронные, за счет более быстрого распространения сигнала, в то время как в синхронных структурах скорость передачи сигнала ограничена тактовой частотой.

В некоторых частных случаях может быть достигнуто большее быстроедействие синхронных структур, чем асинхронных.

Реализация какого-либо устройства в асинхронных структурах в общем случае требует меньшего числа ячеек, чем реализация того же устройства в синхронной структуре.

Наряду чисто с асинхронными и синхронными структурами существуют структуры смешанного типа.

Общеизвестно [7], что реализация комбинационных схем более эффективна в асинхронных структурах, поэтому наиболее перспективными оказываются структуры, в которых при решении комбинационных задач структура выступает как асинхронная, а при решении последовательных задач допускается использование тактовой частоты.

1.5.2. Структуры с индивидуальным и коллективным выбором направления передачи сигналов

Реализация в однородных структурах практически любых логических или последовательных схем требует, чтобы любая ячейка была способна выполнять набор функций хотя бы одного полного базиса, причем в каждый момент времени она может выполнять только одну функцию из полного набора (например, ячейка может быть способна выполнять функцию ИЛИ-НЕ или И-НЕ, или полный набор функций «запрет +1» и т.п., причем в последнем случае в любой момент ячейка выполняет только запрет или единичную функцию).

В связи с этим классификация структур с точки зрения функциональной способности их ячеек не является целесообразной.

Особым отличительным признаком с точки зрения поведения ячеек в структуре является их способность к образованию каналов передачи сигналов в заданном направлении из всех возможных направлений передачи сигналов в структуре.

Если при передаче сигнала через данную ячейку структуры в любом заданном направлении требуется настройка только этой ячейки, то такие структуры условно будем называть структурами с *индивидуальным выбором направления* передачи сигналов.

Если для выбора направления передачи сигнала через данную ячейку требуется настройка совокупности нескольких ячеек, включая и данную ячейку, будем называть такие структуры структурами с *коллективным выбором направления* передачи сигналов.

1.5.3. Направленность передачи сигнала и связи ячейки

Одним из важных конструктивных признаков однородной структуры является количество возможных направлений передачи сигнала. Будем называть структуру *N-направленной*, если от любой функциональной ячейки сигнал может быть передан в любом из N направлений. Практически целесообразными являются структуры, в которых $N \leq 8$ [7].

Количество возможных направлений передачи сигнала N , определяющее количество связей ячейки с окружающими, в значительной степени определяет и гибкость структуры с точки зрения реализации в ней различных функциональных схем. С увеличением числа направлений N гибкость структуры возрастает, однако вместе с тем усложняется конструкция структуры, так как при этом, естественно, возрастает число связей ячейки с окружающими ячейками.

В общем случае число связей ячейки в каждом из возможных направлений не ограничивается одной связью с соседней ячейкой, так как ячейка может быть симметрично связана не только с соседними ячейками.

1.5.4. Настройка структуры

В начальном состоянии однородная структура является недетерминированной относительно реализации в ней заданной функции. Для реализации заданной функции в структуру должна быть введена некоторая информация, однозначно определяющая реализацию требуемой функции.

Таким образом, настройка есть введение в структуру информации, которая обеспечивает перекоммутацию ячеек структуры для реализации схемы, реализующей заданную функцию.

Однородная среда называется *настраиваемой*, если функциональные ячейки среды могут быть настроены на реализацию той или иной функции только один раз, например путем выжигания отдельных соединений. Если однородная среда построена на основе многократно настраиваемых функциональных ячеек, она называется *перестраиваемой* [5]. В дальнейшем будем рассматривать только перестраиваемые однородные структуры.

Здесь предполагается, что программа настройки ячеек структуры, содержащая информацию о состояниях, в которые должны быть переведены каждая из настраиваемых ячеек, имеется вне данного участка структуры или всей структуры.

Существуют различные способы подачи управляющих сигналов на ячейки структуры.

Одним из наиболее простых является *координатный* способ выборки ячеек структуры, сопровождающийся передачей n -разрядного кода настройки, где в частном случае $n = 1$. В этом способе допускается произвольный порядок выборки настраиваемых ячеек. Выбор и настройка требуемой ячейки реализуются путем подачи определенной совокупности сигналов по соответствующим координатным шинам. В каждый момент времени в общем случае настраивается только одна из ячеек структуры, однако координатный способ допускает и одновременную выборку нескольких настраиваемых ячеек.

При *непосредственном* способе настройки ячеек структуры необходим индивидуальный внешний вывод от каждой ячейки, и применение его целесообразно при относительно небольшом числе ячеек.

Адресный способ настройки предполагает, что код настройки, сопровождаемый адресным кодом, посылается в общую систему связи, соединяющую все ячейки структуры. При этом каждая ячейка декодирует и анализирует адресный код и при совпадении его с собственным кодом записывает соответствующий код настройки.

Во многих однородных структурах отдельно используются управляющие и рабочие сигналы, которые передаются по независимым каналам.

Обычно управляющие сигналы служат для выборки требуемых ячеек (из всего множества ячеек структуры) и настройки их на выполнение заданных функций, а рабочие – являются входными аргументами реализуемых ячейками функций.

Однако в некоторых разновидностях однородных структур одни и те же сигналы являются как управляющими, так и рабочими, в зависимости от состояния ячейки.

Избранный способ настройки влияет на выбор конструктивных решений подачи рабочих сигналов (входных переменных) на структуру, а также на выбор способов отбора входных сигналов подачи и отбора по крайним ячейкам структуры или по любой ячейке структуры.

Наиболее целесообразным способом реализации однородных структур на микроэлектронных схемах является подача входных и отбор выходных сигналов по крайним ячейкам, так как не требует непосредственного доступа ко всем ячейкам структуры.

Иной способ требует осуществления внешнего вывода для каждой ячейки структуры, но при этом уменьшается количество ячеек, участвующих в реализации заданной функции, за счет сокращения длины соединительных каналов.

1.5.5. Функциональная способность ячейки

При реализации в однородной структуре любой логической или временной функции ячейка должна быть способна реализовать хотя бы один полный базис, а также выполнять функции канала связи между входными и выходными полюсами в заданном направлении, поэтому в общем случае ячейка структуры должна содержать в себе элементы для выполнения функций выбранного полного набора и элементы для образования канала связи. Выполняемая ячейкой в данный момент функция задается кодом настройки.

Тем не менее могут быть созданы структуры, в которых не имеет места разделение функциональной схемы ячейки; в таких структурах выполнение ячейкой функций базиса или функции канала связи осуществляется на одном и том же функциональном элементе в зависимости от сигналов настройки — извне или от окружающих ячеек.

1.5.6. Степень универсальности структур

Вместе с универсальными структурами, обеспечивающими решение задач классов почти с равной эффективностью, могут быть созданы структуры, в которых с наибольшей эффективностью реализуются функции некоторого класса R ; такие структуры можно считать специализированными в классе R .

Однородная среда называется *универсальной*, если в ней может быть реализован любой алгоритм функционирования устройства управления. Однородная среда называется *специализированной*, если в ней может быть реализован алгоритм из некоторого класса алгоритмов.

В частности, можно предположить, что в однородных структурах с адаптивными свойствами самоорганизующиеся и самонастраивающиеся

ся устройства могут быть реализованы более эффективно, чем детерминированные устройства с жесткой программой функционирования.

В то же время могут быть созданы структуры для реализации в них, например, только комбинационных схем или для выполнения последовательных функций определенного класса. Такие структуры также являются специализированными.

1.6. ЗАДАНИЕ АЛГОРИТМОВ ФУНКЦИОНИРОВАНИЯ И ПЕРЕРАБОТКИ ИНФОРМАЦИИ В АВТОМАТИЧЕСКИХ СИСТЕМАХ УПРАВЛЕНИЯ

Большинство устройств логического управления построено непосредственно по словесному описанию их работы без использования формализованных методов (автоматные таблицы, графы переходов и т.п.) и применяются лишь в некоторых наиболее сложных случаях, как правило, для построения и минимизации отдельных небольших подсистем.

Сложность описания при использовании упомянутых формализованных подходов определяется числом входов и состояний синтезируемого автомата и почти не зависит от сложности алгоритма и соответственно от сложности предстоящей схемной реализации. Так, размер таблицы состояний автомата, имеющего n и содержащего s триггеров, будет $2^n \cdot 2^s$, причем он одинаков как для самого простого, так и для самого сложного устройства.

Идеальным представляется случай, когда сложность описания алгоритма пропорциональна сложности его предстоящей реализации. Отметим, что логические формулы обладают именно таким свойством: число букв в них линейно связано с числом элементов, требующихся для реализации этих формул.

Высокая стоимость разработки топологии интегральных схем диктует необходимость применения повторяющихся модульных структур, т.е. небольшого числа разных типов простых элементов. Задача состоит в определении набора элементарных ячеек (МЛМ), реализуемых наиболее эффективно и позволяющих оптимально использовать потенциальные возможности технологии изготовления новых устройств.

На основании проведенных исследований [1] установлено, что настраиваемые логические модули для построения логических устройств, исходя из специфики булевых формул, описывающих алгоритмы их

функционирования, должны быть способны реализовывать путем настройки произвольные функции n переменных, а лишь только те из них, для которых булевы формулы неповторны или обладают малой повторностью переменных, что обеспечит построение модулей с малой элементной сложностью и малым числом внешних выходов.

С конструированием матричных однородных структур связан целый ряд важнейших проблем, в частности проблемы соединений и их модульной организации.

Изотропные среды из многофункциональных логических модулей можно организовывать таким образом, чтобы обеспечить выполнение различных видов обработки информации.

Возможности и ограничения при организации соединений в изотропной среде существенным образом сказываются на быстродействии и оперативной гибкости данного устройства.

Разные алгоритмы требуют разной степени связи. Чтобы эффективно использовать специфические свойства способов организации соединений, математические операции, необходимые для обработки информации и выполнения вычислений, должны быть представлены в форме соответствующих алгоритмов.

При выполнении точечных операций каждая точка одномерной или двумерной матрицы должна обрабатываться совершенно независимо. Операции, при которых одномерный входной сигнал представляет собой временную последовательность, относятся к операциям «без запоминания». Все точки входной матрицы могут обрабатываться одинаковым образом или же для каждой из них имеется свой независимый набор правил обработки. В обоих случаях очевидно, что после ввода матрицы входных данных в матрицу изотропной среды каждый элемент может выполнять свою собственную задачу обработки независимо от остальных элементов среды. Поэтому необходимая для таких операций связь минимальна и проявляется только при вводе данных в изотропную среду или выводе из нее. Следовательно, такие операции могут выполняться параллельно.

Многие алгоритмы обработки информации могут быть выражены через матричные операции. Одной из основных операций матричной алгебры является умножение двух матриц (частным случаем такой операции можно считать умножение вектора на матрицу). Математически такое умножение описывается следующим образом:

$$C_{ij} = \sum_{k=1}^N A_{ik} B_{kj}, \quad i, j = 1, \dots, N, \quad (1.7)$$

где для удобства через A , B и C обозначим квадратные матрицы $N \times N$. Результирующую матрицу можно иначе определить как сумму N матриц внешнего произведения, полученных умножением вектор-столбцов матрицы A на соответствующие вектор-строки матрицы B :

$$C = \sum_{k=0}^N C^{(k)}, \quad C_{ij}^{(k)} = A_{ik} B_{kj}, \quad i, j = 1, \dots, N, \quad (1.8)$$

где вторая строка – это внешнее произведение k -го столбца матрицы A на k -ю строку матрицы B . Из вышеприведенных уравнений очевидно, что при данной операции высока степень связанности элементов входных матриц и результирующей матрицы. Так, все элементы данной строки матрицы A и данного столбца матрицы B дадут вклад в один элемент результирующей матрицы C и наоборот, один элемент матрицы A (или B) дает вклад во все элементы соответствующей строки (или столбца) матрицы C .

Регулярный характер связи, задаваемый уравнениями (1.7) и (1.8), означает, что операции можно выполнять также посредством рекуррентных и локально связанных алгоритмов, реализуемых с помощью изотропных сред из многофункциональных логических модулей [10].

Две важные операции обработки информации, требующие простых связей между всеми элементами матрицы входных данных, – это фурье-преобразование и сортировка. Иными словами, при этих операциях на каждый элемент выходной матрицы влияют все элементы входной матрицы и каждый элемент входной матрицы влияет на все элементы выходной матрицы. Вычисление при фурье-преобразовании сводится к умножению и сложению комплексных чисел, тогда как сортировка является операцией сравнения.

Операция сортировки числовой последовательности включает в себя элементарные операции сравнения двух чисел и расположение их на выходе в убывающем порядке. Одним из эффективных алгоритмов сортировки является сортирующий алгоритм Батчера [9]. Основной принцип, положенный в основу этого алгоритма, состоит в применении метода «разделяй и властвуй», т.е. крупная задача делится на несколько задач меньшего масштаба, и комбинация их решений на выходе позволяет получить решение общей задачи.

В настоящее время можно определить три направления, по которым идет развитие элементной базы большой степени интеграции для построения разного рода управляющих устройств [11].

В рамках *первого направления* ведется разработка микропроцессоров и микропроцессорных наборов на базе одного кристалла. При этом вопрос соответствия структурных и функциональных особенностей микропроцессоров структуре ЭВМ по существу не был учтен. Это вызвало необходимость использования мультипроцессорной организации цифровых систем на основе микропроцессоров и разработки соответствующих принципов обработки информации и управления.

Второе направление предполагает достижение высокой эффективности элементной базы большой степени интеграции благодаря максимальному соответствию структуры интегральных схем структуре проектируемых управляющих устройств. Элементами реализации этого направления являются матричные БИС, программируемые логические матрицы и перепрограммируемые запоминающие устройства.

Третье направление базируется на автоматном принципе обработки информации, это определяет структурное сближение модулей обработки информации, управления и электронной части модулей внешних устройств.

Таким образом, предполагается модульное построение систем управления на основе принципов многофункциональности и регулярности. Отличительной особенностью последнего направления от предыдущих является строгая формализация процесса его синтеза, который базируется на основе теории многофункциональных автоматов.

1.7. АППАРАТНАЯ И ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМОВ ФУНКЦИОНИРОВАНИЯ УПРАВЛЯЮЩИХ УСТРОЙСТВ

Существует два основных принципа реализации алгоритма функционирования устройств управления (УУ) – *аппаратурная (схемная) реализация* в том или ином элементном базисе и *программная реализация* [2]. При аппаратурной реализации, когда по заданному алгоритму функционирования строится структурная схема УУ в каком-либо базисе элементов, говорят, что алгоритмическое описание переводится в *структурное описание*, представляющее собой структурную модель заданного алгоритма функционирования УУ.

Переход от алгоритмического описания к структурному в базе однородной среды для программируемого УУ означает, что по заданному алгоритму функционирования УУ осуществляется настройка многофункциональных элементов однородной среды.

При настройке многофункциональных элементов на реализацию одной из функций могут быть использованы различные способы, основными из которых являются *технологический*, когда настройка осуществляется удалением (например, выжиганием) и созданием (например, напылением пленок) отдельных соединений, и *функциональный*, когда настройка осуществляется подачей на какие-либо входы определенных сигналов.

Дискретные устройства с итеративной структурой его функциональной схемы будем называть однородной средой, если в качестве элементов функциональной схемы устройства использованы МЛМ.

Такое дискретное устройство путем определенной настройки его МЛМ, очевидно, может быть настроено на реализацию алгоритмов функционирования любого УУ.

Альтернативой структурному моделированию алгоритма функционирования УУ в виде аппаратурной реализации является программное моделирование. При программном моделировании вместо структуры УУ в том или ином элементном базисе алгоритм функционирования представляется в виде последовательности так называемых команд, каждая из которых определяет операцию над множеством исходных или промежуточных данных (операндов) и номер следующей команды.

Упорядоченная совокупность команд, однозначно описывающая заданный алгоритм функционирования, называется *программой*, а представление алгоритма функционирования в виде программы – *программным моделированием алгоритма*.

В отличие от аппаратурной реализации, при структурном моделировании алгоритма функционирования программа, реализующая тот же алгоритм функционирования при программном моделировании, является пассивной, т.е. представляет собой некоторую запись алгоритма функционирования. Для чтения этой записи и выполнения команд при программном моделировании можно использовать *микропроцессор*.

Таким образом, при структурном моделировании аппаратурно реализуется весь заданный алгоритм функционирования УУ. При программном моделировании микропроцессор, представляющий собой

универсальное устройство, настраиваемое на аппаратную реализацию одновременно только одной операции, обеспечивает выполнение всего алгоритма функционирования путем *последовательного* выполнения четко разграниченных отдельных операций (*актов алгоритма*), определяемых программой, хранящейся в запоминающем устройстве.

Легко понять, что при структурном моделировании изменение алгоритма функционирования требует изменения структурной схемы УУ, тогда как при программном моделировании необходимо изменить лишь хранящуюся в запоминающем устройстве программу.

Особенности реализации алгоритмов функционирования УУ в однородной среде. Если каждый элемент однородной среды (ОС) соединен непосредственно только со своими ближайшими соседями, то речь идет об ОС с близкодействием. Достоинством таких ОС является возможность локализации и исключения из числа используемых элементов неисправных путем соответствующей настройки соседних с ними элементов ОС. Однако наличие связей только между соседними элементами ОС с близкодействием приводит к тому, что для организации более протяженных связей приходится нерационально использовать большое число элементов среды. При этом возникает значительная избыточность. Кроме того, элементы таких ОС, выполняющие большое число различных операций, оказываются достаточно сложными, и требуется большой объем информации для их настройки.

Если ОС представляет собой прямоугольную итеративную решетку из H горизонтальных и V вертикальных шин, то будем называть ее матричной ОС. Разработка различных типов ОС привела к необходимости создания методов синтеза управляющих устройств в таких средах. Существующие методы синтеза можно разделить на две принципиально отличные группы:

- 1) моделирование в ОС алгоритма функционирования УУ;
- 2) размещение в ОС функциональной схемы, реализующей алгоритм функционирования УУ.

При использовании первого из указанных подходов осуществляется настройка ОС на реализацию исходного описания автомата.

Методы синтеза УУ в ОС, относящиеся ко второй группе, состоят в том, что по исходному описанию УУ строится функциональная схема в заранее выбранном логическом базисе и размещается в используемой для синтеза ОС. При этом говорят также о «натягивании» заданной

функциональной схемы, или логической, сети на выбранную ОС или о программировании ОС для реализации в ней заданной схемы [3].

Необходимо отметить, что реализация синтезируемого устройства ОС не заканчивается процессом размещения в ней логической схемы. В соответствии с выбранным размещением необходимо получить упорядоченный массив кодов настройки, который затем передается в настройочную память элементов ОС.

Таким образом, перенастройка программируемого УУ на реализацию другого алгоритма функционирования при программной его реализации состоит в замене программы в запоминающем устройстве, а при аппаратной реализации с помощью ОС – в перенастройке МЛМ.

1.8. ВЫВОДЫ

Основные результаты, полученные в данной главе, сводятся к следующему.

1. Даны общие понятия синтеза структурных конечных автоматов.
2. Приведены способы задания функции алгебры логики и классификация автоматов.
3. Описаны основные принципы построения перестраиваемых автоматов.
4. Рассмотрены основные понятия однородных структур.
5. Раскрыты способы задания алгоритмов функционирования и их аппаратная и программная реализация, а также особенности их реализации в однородных структурах.

ЛИТЕРАТУРА

1. *Артюхов В.Л., Копейкин Г.А., Шалыто А.А.* Настраиваемые модули для управляющих логических устройств. – Л.: Энергоатомиздат, 1981. – 168 с.
2. *Будинский Я.* Логические цепи в цифровой технике. – М.: Связь, 1977. – 392 с.
3. *Евреинов Э.В.* Однородные вычислительные системы, структуры и среды. – М.: Радио и связь, 1981. – 208 с.
4. *Ершова Э.Б. и др.* Основы дискретной автоматики в электросвязи. – М.: Связь, 1980. – 232 с.
5. *Лазарев В.Г., Пийль Е.И., Турута Е.Н.* Построение программируемых управляющих устройств. – М.: Энергоатомиздат, 1984. – 192 с.

6. *Поспелов Д.А.* Логические методы анализа и синтеза схем. – М.: Энергия, 1964. – 320 с.
7. *Прангшивили И.В., Абрамова Н.А., Бабичева Е.В., Игнатущенко В.В.* Микроэлектроника и однородные структуры для построения логических и вычислительных устройств. – М.: Наука, 1967. – 228 с.
8. *Пупырев Е.И.* Перестраиваемые автоматы и микропроцессорные системы. – М.: Наука, 1984. – 192 с.
9. *Фет Я.И.* Массовая обработка информации в специализированных однородных процессорах. – Новосибирск: Наука, 1976. – 200 с.
10. *Шальто А.А.* Логическое управление. Методы аппаратной и программной реализации алгоритмов. – СПб.: Наука, 2000. – 780 с.
11. *Шидловский С.В.* Задание алгоритмов управления и переработки информации в проблемно-ориентированных комплексах // Научная сессия ТУСУР: Материалы докл. межрегиональной науч.-техн. конф. 14–16 мая 2002 г. Томск, Россия. – Томск: ТУСУР, 2002. – Ч. 2. – С. 321–333.