М. Г. КАРПОВСКИЙ

КОНЕЧНЫЕ АВТОМАТЫ С ИСПРАВЛЕНИЕМ И ОБНАРУЖЕНИЕМ ОШИБОК

§ 1. ПОСТАНОВКА ЗАДАЧИ

1.1. Одной из основных проблем, встающих при синтезе дискретных устройств, является проблема надежности. Эта проблема должна решаться уже на этапах абстрактного и структурного синтеза. При этом представляется весьма многообещающим использование кодов с обнаружением и исправлением ошибок для кодирования состояний дискретных устройств с памятью с целью повышения надежности. Подобная задача рассматривалась в ряде работ [1, 2, 3], но при этом способ кодирования состояний не зависел от функций, реализуемых устройством. В работе [4] был предложен способ построения такого рода устройств, учитывающий функции, реализуемые этими устройствами. Однако использование этого способа приводит к сложным задачам целочисленного линейного программирования, и уже при сравнительно небольшом числе элементов памяти этот способ оказывается практически непригодным.

Далее предлагаются другие, более простые и эффективные методы построения дискретных устройств с памятью с исправлением ошибки заданной кратности l, существенно зависящие от функций, реализуемых устройством, и вследствие этого оказывающиеся в ряде случаев более экономичными, чем методы, предложенные в [1—4].

1.2. Уточним теперь постановку задачи на языке теории конечных автоматов.

Под словом «автомат» всюду далее понимается детерминированный, полностью определенный инициальный автомат, все состояния которого закодированы двоичными наборами некоторой длины m [5]. Автомат и реализуемое им отображение, там, где это не приводит к двусмысленности, будем обозначать одинаковым образом.

Пусть задана последовательность натуральных чисел:

$$0 = t_0 < t_1 < t_2 < \dots \tag{1}$$

[illegible] (1) [illegible] последовательность ξ: $\xi = \xi(1)\xi(2)\dots$, [illegible] образом:

$$\xi(t) = \begin{cases} 1 & \text{при } t = t_k \ (k = 0, 1, \dots), \\ 0 & \text{в остальных случаях}. \end{cases} \tag{2}$$

Определение 1. Пусть задана последовательность ξ, определяемая (2).

а) *ξ-преобразованием тактности* назовем оператор, сопоставляющий каждой последовательности $x = x(1)\,x(2)\dots$ последовательность $\xi \times x$, определяемую следующим образом:

$$(\xi \times x)(k) = x(t_k).$$

б) *ξ^{-1}-преобразованием тактности* назовем оператор, сопоставляющий каждой последовательности $x = x(1)\,x(2)\dots$ последовательность $\xi^{-1} \times x$, определяемую следующим образом: $(\xi^{-1} \times x)(t) = x(k)$ при $t_{k-1} < t \leqslant t_k$.

Применение к произвольной временной последовательности x ξ-преобразования тактности приводит к тому, что x рассматривается только в моменты t_k, а применение ξ^{-1}-преобразования тактности приводит к тому, что в x каждый символ $x(t)$ повторяется $t_k - t_{k-1}$ раз при $t_{k-1} < t \leqslant t_k$ $(k = 1, 2, \dots)$.

Пусть далее заданы два алфавита — X и Y.

Рассмотрим автомат $A(x)$, определяемый системой

$$\begin{cases} a(t) = f(x(t),\ a(t-1)), \\ y(t) = g(a(t)), \end{cases} \tag{3}$$

где $x(t) \in X$; $a(t) \in \{0, 1\}^m$; $y(t) \in Y$. (Здесь $\{0, 1\}^m$ — множество двоичных наборов длины m.)

Определение 2. Пусть задан автомат $A(x)$, определяемый (3).

а) *Автоматом с ошибкой, соответствующим автомату* $A(x)$, назовем автомат $A(x, \gamma)$, входом которого является упорядоченная пара (x, γ), где $\gamma(t) \in \{0, 1\}^m$, определяемый системой

$$\begin{cases} a(t) = f(x(t),\ a(t-1) \oplus \gamma(t)), \\ y(t) = g(a(t)). \end{cases} \tag{4}$$

(Здесь $\oplus$ означает покомпонентную сумму по модулю 2.)

б) Входной сигнал $\gamma(t)$ назовем *ошибкой в автомате* $A(x, \gamma)$ *в момент времени* t.

Условие (4) эквивалентно тому, что ошибки происходят в элементах памяти автомата и надежность автомата не зависит от его комбинационной части. Такое предположение часто выполняется на практике, так как обычно элементы памяти подвержены влиянию помех в течение всего времени работы автомата, тогда как комбинационная часть подвержена влиянию помех только в моменты прихода входных сигналов.

Обозначим теперь через [illegible] входом которого является упорядоченная пара [illegible] системой

$$\begin{cases} a(t) = f(x(t), \xi(t), [illegible] \\ y(t) = g(a(t)), \end{cases} \qquad (5)$$

где $x(t) \in X$; $\xi(t) \in \{0, 1\}$; $a(t) \in \{0, 1\}^m$; $y(t) \in Y$.

Обозначим, кроме того, через $\Gamma_m^{(l)}$ множество последовательностей γ таких, что в любой момент t двоичный вектор $\gamma(t)$ длины m имеет не более l единиц и в любой момент t_k $(k=1, 2, \ldots)$:

$$\gamma(t_k) = \underbrace{(00 \ldots 0)}_{m}. \qquad (6)$$

Определение 3. Пусть задан автомат $A(x, \xi)$, определяемый (5).

а) Автомат $A(x, \xi)$ будем называть *автоматом с исправлением l ошибок* и обозначать $A^{(l)}(x, \xi)$, если для соответствующего $A^{(l)}(x, \xi)$ автомата с ошибкой $A^{(l)}(x, \xi, \gamma)$ при любом фиксированном ξ (для которого $t_k - t_{k-1} > 1$, $k = 1, 2, \ldots$) для всех x и $\gamma \in \Gamma_m^{(l)}$ выполняется условие

$$\xi \times A^{(l)}(\xi^{-1} \times x, \ \xi, \ \gamma) = A^{(l)}(x, I), \qquad (7)$$

где $I(t) = 1$ для любого t.

б) Входной сигнал $\gamma(t)$ $(\gamma \in \Gamma_m^{(l)})$ будем называть *l-кратной ошибкой в автомате $A^{(l)}(x, \xi, \gamma)$ в момент t.*

Поясним смысл введенных определений.

Отметим сначала, что для любого дискретного устройства потенциального типа, на вход которого подается заданная последовательность x потенциальных сигналов, в качестве входной последовательности при этом можно рассматривать также $\xi^{-1} \times x$ для любого ξ. Действительно, обозначим моменты подачи сигналов на вход устройства $t_0 + 1$, $t_1 + 1, \ldots$ и каждый тактовый промежуток $[t_{k-1} + 1, t_k + 1)$ разобьем на $\eta_k + 1$ частей, где η_k — число нулей между $k-1$-й и $k+1$-й единицей в ξ. При этом последовательность x превращается в $\xi^{-1} \times x$. Пусть теперь задан автомат $A^{(l)}(x, \xi)$. Если, в соответствии с вышеизложенным, на вход автомата с ошибкой $A^{(l)}(x, \xi, \gamma)$ подается $(\xi^{-1} \times x, \xi, \gamma)$, то из условия (7) следует, что для любого $\gamma \in \Gamma_m^{(l)}$ выход $A^{(l)}(x, \xi, \gamma)$ в моменты $t_1, t_2, \ldots$ („после окончания переходного процесса“) совпадает с выходом $A^{(l)}(x, \xi)$, если на вход $A^{(l)}(x, \xi)$ подается x и $\xi = I$. Ограничение (6) для $A^{(l)}(x, \xi)$ означает, что существуют «защитные промежутки» $[t_k, t_k + 1)$, когда в $A^{(l)}(x, \xi, \gamma)$ не могут происходить ошибки. Это условие обычно выполняется на практике, так как при разбиении тактового промежутка $[t_{k-1} + 1, t_k + 1)$ на

[illegible] 1 [illegible] вероятность появления ошибок во время [illegible] стремится к нулю при [illegible] $\to \infty$. Отсюда следует, что [illegible] выборе ξ длительность „защитных промежутков“ может быть сделана сколь угодно малой, а условие (6) с приемлемой точностью будет выполняться.

Введем еще два определения для автоматов $A(x, \xi)$.

Определение 4. Автомат $A(x, \xi)$ будем называть *устойчивым*, если функция переходов его $f(x(t), \xi(t), a(t-1))$ удовлетворяет для любых x_i и a_j условию

$$f(x_i, 0, f(x_i, 1, a_j)) = f(x_i, 1, a_j). \qquad (8)$$

Для дискретных устройств с элементами памяти потенциального типа, для которых входной сигнал $x(t)$ может изменяться только в моменты $t = t_k + 1$, когда $\xi(t) = 1$, при отсутствии ошибок состояния могут изменяться также только в моменты $t = t_k + 1$ и, следовательно, условие (8) всегда выполняется.

Определение 5.

а) *Отображением, ξ-индуцированным автоматом* $A(x, \xi)$, будем называть отображение, $A(x, I)$.

б) *Автоматом, ξ-индуцированным автоматом* $A(x, \xi)$, будем называть подавтомат [6] $A(x, \xi)$ получаемый из $A(x, \xi)$ при $\xi = I$.

1.3. Рассмотрим теперь вопрос о необходимых и достаточных условиях существования устойчивых автоматов $A^{(l)}(x, \xi)$ с исправлением l ошибок и с заданным ξ-индуцированным отображением $A(x)$. Простейшим автоматом такого рода является автомат, получающийся $2l+1$-кратным дублированием [3] любого устойчивого автомата $A(x, \xi)$, для которого $A(x, I) = A(x)$. Однако это приводит к весьма существенному увеличению оборудования. Далее рассматриваются другие методы построения устойчивых автоматов $A^{(l)}(x, \xi)$, при которых для кодирования состояний автомата используются коды с исправлением ошибок. При этом, как будет показано далее, для широкого класса отображений $A(x)$ число элементов памяти увеличивается незначительно (или даже вообще не увеличивается). Это связано с тем, что предлагаемые методы построения автоматов $A^{(l)}(x, \xi)$, в отличие от [1—4], существенно зависят от $A(x)$. Вследствие этого предлагаемые методы могут привести к значительно более простым и надежным в общем случае устройствам, нежели известные методы [1—4].

Определение 6. Пусть задан автомат $A(x)$, определяемый (3). Состояние a_j автомата $A(x)$ будем называть *x_i-достижимым*, если существует достижимое состояние a_s такое, что

$$a_j = f(x_i, a_s).$$

Обозначим далее через $n_a^{(l)}$ — число состояний $A^{(l)}(x, \xi)$; через $]N[$ — ближайшее сверху к N целое число.

Теорема 1. *Для существования устойчивого автомата $A^{(l)}(x, \xi)$ с исправлением l ошибок с заданным ξ-индуцированным отображением $A(x)$, для которого $]\log_2 n_a^{(l)}[= m$, необходимо и достаточно, чтобы существовал автомат $A(x)$ (реализующий отображение $A(x)$), такой, что для любого x_i для любых двух x_i-достижимых неэквивалентных состояний a_p и a_q автомата $A(x)$ выполнялось бы условие*

$$\rho(a_p, a_q) \geqslant 2l + 1. \qquad (9)$$

(Здесь $\rho(a_p, a_q)$ — расстояние Хэмминга [7] между двоичными наборами длины m, которыми закодированы состояния a_p и a_q.)

Доказательство необходимости (которое здесь не приводится) состоит в следующем: показывается, что в качестве $A(x)$ может быть выбран автомат $A^{(l)}(x, l)$, ξ-индуцированный $A^{(l)}(x, \xi)$.

Приведем доказательство достаточности. Построим $A^{(l)}(x, \xi)$ с ξ-индуцированным отображением $A(x)$, для которого $]\log_2 n_a^{(l)}[= m$. В качестве множества состояний для $A^{(l)}(x, \xi)$ примем $\{0, 1\}^m$. Функцию переходов $f^{(l)}(x(t), \xi(t), a(t-1))$ для $A^{(l)}(x, \xi)$ определим следующим образом:

$$\text{для любых } x_i \text{ и } a_j \begin{cases} f^{(l)}(x_i, 1, a_j) = f(x_i, a_j), & (10) \\ f^{(l)}(x_i, 0, a_j) = f(x_i, a_s), & (11) \end{cases}$$

где a_s удовлетворяет условию

$$\rho(a_j, f(x_i, a_s)) \leqslant l. \qquad (12)$$

(Причем, если для некоторых (x_i, ξ_r, a_j) $f^{(l)}(x_i, \xi_r, a_j)$ не определена из (10) и (11), то $f^{(l)}(x_i, \xi_r, a_j)$ доопределяется произвольным образом.) Из условия (9) следует, что если существует несколько a_s, удовлетворяющих условию (12), то все $f(x_i, a_s)$ эквивалентны, и при этом $f^{(l)}(x_i, 0, a_j)$ принимается равной любому из этих эквивалентных состояний. Кроме того, для того, чтобы синтезируемый автомат $A^{(l)}(x, \xi)$ был устойчивым в соответствии с (11) и (12), положим: если существует a_s такое, что

$$a_j = f(x_i, a_s), \text{ то } f^{(l)}(x_i, 0, a_j) = a_j.$$

Функцию выходов $g^{(l)}(a(t))$ для $A^{(l)}(x, \xi)$ определим следующим образом: для любого $a_j \in \{0, 1\}^m$

$$g^{(l)}(a_j) = g(a_j). \qquad (13)$$

(Причем если $g^{(l)}(a_j)$ не определена из (13), то $g^{(l)}(a_j)$ доопределяется произвольным образом.) Нетрудно видеть, что построенный таким образом автомат является автоматом с исправлением l ошибок с ξ-индуцированным отображением $A(x)$, причем для построенного автомата $A^{(l)}(x, \xi)$ заданный автомат $A(x)$ является ξ-индуцированным.

Замечание. Условие (9) теоремы 1 (а также излагаемые в § 2 достаточные условия существования $A^{(l)}(x, \xi)$) остается достаточным, если в определении $A^{(l)}(x, \xi)$ условие (7) заменить более сильным условием.

$$\xi \times A_j^{(l)}(\xi^{-1} \times x, \xi, \gamma) = A_j^{(l)}(x, l) \quad (j = 1, 2, \ldots, n_a^{(l)}).$$

(Здесь $A_j^{(l)}$ — отображение, реализуемое автоматом $A^{(l)}$, если начальным состоянием $A^{(l)}$ выбрано a_j.)

Отметим еще одно важное свойство автомата $A^{(l)}(x, \xi)$, построенного при доказательстве теоремы 1, сохраняющееся также и для всех автоматов $A^{(l)}(x, \xi)$, которые будут строиться в § 2. В моменты времени t, когда $\xi(t) = 0$, в автомате $A^{(l)}(x, \xi)$, как видно из (11) и (12), отсутствуют „критические гонки“ [5]. Таким образом, для решения проблемы „гонок“ в автомате $A^{(l)}(x, \xi)$ достаточно добиться устранения „критических гонок“ только в моменты t, когда $\xi(t) = 1$, т. е. в моменты, когда функция переходов $A^{(l)}(x, \xi)$ совпадает с функцией переходов ξ-индуцированного автоматом $A^{(l)}(x, \xi)$ автомата $A^{(l)}(x, l)$ (см. (10)), реализующего заданное отображение.

В заключение этого раздела, отметим, что из теоремы 1 следует, что для любого автоматного отображения $A(x)$ всегда существует устойчивый автомат $A^{(l)}(x, \xi)$ с ξ-индуцированным отображением $A(x)$. Действительно, закодируем состояния автомата $A(x)$ кодом с исправлением l ошибок. При этом выполняется (9), и, следовательно, существует устойчивый автомат $A^{(l)}(x, \xi)$ с ξ-индуцированным отображением $A(x)$.

В § 2 рассматриваются другие, в общем случае более экономичные, методы построения устойчивых автоматов $A^{(l)}(x, \xi)$ с заданным ξ-индуцированным отображением $A(x)$, причем в дальнейшем для краткости слово „устойчивый“ обычно опускается, $A^{(l)}(x, \xi)$ обозначается $A^{(l)}(x)$ и ξ-индуцированные автомат и отображение называются просто индуцированными автоматом и отображением.

§ 2. НЕОБХОДИМЫЕ И ДОСТАТОЧНЫЕ УСЛОВИЯ СУЩЕСТВОВАНИЯ АВТОМАТОВ, ИСПРАВЛЯЮЩИХ l ОШИБОК И ИМЕЮЩИХ ЗАДАННОЕ ИНДУЦИРОВАННОЕ ОТОБРАЖЕНИЕ

2.1. Будем всюду далее считать заданным автоматное отображение $A(x)$. Дадим теперь необходимые и достаточные условия существования автомата $A^{(l)}(x)$ с индуцированным отображением $A(x)$, который может быть реализован с помощью заданного числа m элементов памяти.

Введем следующие обозначения: n_a — число состояний минимального автомата $A_{\min}(x)$, реализующего отображение $A(x)$;

$n_a(x_i)$ — число x_i — достижимых состояний $A_{\min}(x)$; n_x — число символов входного алфавита X; $n_x(a_j)$ — число символов входного алфавита, которыми на графе [5] $A_{\min}(x)$ помечены стрелки, подходящие к состоянию a_j; $A(m, d)$ — максимальное число слов m-разрядного двоичного кода с минимальным расстоянием Хэмминга между словами, равным d; $B(m, d)$ — максимальное число слов m-разрядного группового [7] двоичного кода с минимальным расстоянием Хэмминга между словами, равным d.

Теорема 2. *Необходимым условием существования* $A^{(l)}(x)$, *для которого* $]\log_2 n_a^{(l)}[= m$, *является*

$$\max_i n_a(x_i) \leqslant A(m, 2l+1). \tag{14}$$

Обозначим:

$$k_0 =]\log_2 \max_i n_a(x_i)[, \tag{15}$$

$$m_0 = k_0 +]\log_2 n_x[. \tag{16}$$

Следствие 1. Необходимым условием существования $A^{(l)}(x)$, для которого $]\log_2 n_a^{(l)}[= m$, является

$$k_0 \leqslant m - \log_2 \sum_{i=0}^{l} C_m^i. \tag{17}$$

2.2. Рассмотрим теперь достаточные условия существования $A^{(l)}(x)$ с числом элементов памяти, не меньшим m_0. Для этого будем строить автомат $A^{(l)}(x)$, индуцированным для которого будет некоторый автомат $A_0(x)$, состояния которого будем кодировать элементами смежных классов по групповому коду с исправлением l ошибок так, чтобы выполнялось (9).

Теорема 3. *Достаточными условиями существования* $A^l(x)$, *для которого* $]\log_2 n_a^{(l)}[= m$, *являются*

$$\begin{cases} m_0 \leqslant m, \\ \max_i n_a(x_i) \leqslant B(m, 2l+1). \end{cases} \tag{18}$$

Доказательство. Построим автомат $A_0(x)$, реализующий $A(x)$, для которого $n_{x_0}(a_j) = 1$ для любого a_j (здесь $n_{x_0}(a_j)$ обозначен параметр $A_0(x)$, аналогичный параметру $n_x(a_j)$ для $A_{\min}(x)$).

Граф $A_0(x)$ строится „расщеплением" [8] каждого состояния a_j автомата $A_{\min}(x)$ на $n_x(a_j)$ эквивалентных состояний так, что к каждому из этих $n_x(a_j)$ состояний будут подходить стрелки, помеченные одним и тем же входным символом.

Из (18) следует, что существует групповой (m, k_0) код с исправлением l ошибок. Так как из (16) $n_x \leqslant 2^{m_0 - k_0}$ и $m \geqslant m_0$, то можно сопоставить каждому входному символу смежный класс по этому коду так, чтобы различным входным символам соответствовали различные смежные классы. Далее для каждого символа x_i закодируем все x_i-достижимые состояния $A_0(x)$ элементами соответствующего смежного класса. При этом для $A_0(x)$ выполняется условие (9), и, следовательно, существует $A_0(x)$, для которого $]\log_2 n_a^{(l)}[= m$.

Следствие 1. Достаточными условиями существования $A^{(l)}(x)$, для которого $]\log_2 n_a^{(l)}[= m$, являются

$$\begin{cases} m_0 \leqslant m, \\ k_0 < m - \log 2 \sum\limits_{i=0}^{2l-1} C_{m-1}^i. \end{cases} \tag{19}$$

Следствие 2. Необходимым и достаточным условием существования автомата $A^{(1)}(x)$, исправляющего одиночную ошибку, для которого $]\log_2 n_a^{(1)}[= m$, где $m \geqslant m_0$, является

$$m + 1 \leqslant 2^{m-k_0}. \tag{20}$$

Отметим, что для наиболее часто встречающегося случая $3 < m < 8$ условие (20) принимает вид

$$m \geqslant k_0 + 3.$$

2.3. Доказательство теоремы 3 дает общий метод синтеза автоматов $A^{(l)}(x)$, для которых $]\log_2 n_a^{(l)}[\geqslant m_0$. Однако эти автоматы при больших n_x или k_0 могут оказаться весьма неэкономичными. Как будет показано далее, в ряде случаев существуют автоматы $A^{(l)}(x)$, для которых $]\log_2 n_a^{(l)}[< m_0$.

Введем обозначения: M — множество состояний автомата $A(x)$; M_i — множество x_i-достижимых состояний $A(x)$.

Определение 7. а) *Блочным покрытием* для автомата $A(x)$ будем называть покрытие множества M (т. е. множество подмножеств M, объединение которых равно M) такое, что каждый элемент этого покрытия есть объединение некоторых множеств M_i и каждое множество $M_i (i = 1, 2, \ldots, n_x)$ есть подмножество одного и только одного элемента покрытия.

б) Элементы блочного покрытия будем называть *блоками*. Для блочного покрытия λ введем обозначения: n_λ — число блоков; $n_a(\lambda_i)$ — число состояний блока λ_i; $n_\lambda(a_j)$ — число блоков, к которым принадлежит состояние a_j;

$$k_\lambda =]\log_2 \max_i n_a(\lambda_i)[; \tag{21}$$

$$m_\lambda = k_\lambda +]\log_2 n_\lambda[. \tag{22}$$

Каждому блочному покрытию λ сопоставим разбиение π_λ алфавита X (т. е. множество непересекающихся подмножеств множества X, объединение которых равно X) следующим образом: если M_i и M_r принадлежат одному и тому же блоку λ, то x_i и x_r принадлежат одному и тому же элементу разбиения π_λ.

Будем теперь строить автомат $A^{(l)}(x)$, индуцированный для которого будет уже не автомат $A(x)$, который строился при доказательстве теоремы 3, а в общем случае некоторый другой автомат $A_\lambda(x)$, число состояний и функция переходов которого зависят от выбора блочного покрытия λ. При этом элементами одного и того же смежного класса по групповому коду с исправлением l ошибок будут кодироваться все состояния $A_\lambda(x)$, к которым на графе подходят стрелки, помеченные входными символами, принадлежащие одному и тому же элементу разбиения π_λ.

Таким образом, в данном случае уже не каждому входному символу (как это имело место при доказательстве теоремы 3), а каждому элементу π_λ сопоставляется смежный класс по групповому коду с исправлением l ошибок.

Теорема 4. *Если для некоторого автомата $A(x)$ существует блочное покрытие λ такое, что выполняются условия*

$$\begin{cases} m_\lambda \leqslant m, \\ \max\limits_i n_a(\lambda_i) \leqslant B(m, 2l+1), \end{cases} \tag{23}$$

то существует автомат $A^{(l)}(x)$, для которого $]\log_2 n_a^{(l)}[= m$.

Доказательство. Построим сначала автомат $A_\lambda(x)$, эквивалентный автомату $A(x)$, следующим образом. Произведем „расщепление" состояний $A(x)$, как это делалось при доказательстве теоремы 3 для $A_{min}(x)$, но только каждое состояние a_j будем расщеплять на $n_\lambda(a_j)$ эквивалентных состояний. Причем, если на графе $A(x)$ к состоянию a_j подходят стрелки, помеченные x_i и x_r, и, кроме того, x_i и x_r принадлежат одному и тому же элементу π_λ, то стрелки, помеченные x_i и x_r, подходят к одному и тому же состоянию $A_\lambda(x)$, полученному „расщеплением" a_j. Из (23) с учетом (21) следует, что существует групповой (m, k_λ) код с исправлением l ошибок. Так как из (22) $n_\lambda \leqslant 2^{m_\lambda - k_\lambda}$ и $m \geqslant m_\lambda$, то каждому блоку λ, а следовательно, и каждому элементу π_λ, можно сопоставить смежный класс по этому коду так, чтобы различным блокам соответствовали различные смежные классы. Закодируем состояния $A_\lambda(x)$ так, чтобы состояния, к которым на графе подходят стрелки, помеченные входными символами из одного элемента π_λ, кодировались элементами соответствующего этому элементу смежного класса. При этом для $A_\lambda(x)$ выполняется условие (9), и, следовательно, существует $A^{(l)}(x)$, для которого $]\log_2 n_a^{(l)}[= m$.

Доказанная теорема 4 является обобщением теоремы 3, так

как теорема 3 следует из теоремы 4, если выбрать в качестве $A(x)$ автомат $A_{min}(x)$ а в качестве λ блочное покрытие $\lambda_0 = \{M_1, M_2, \ldots, M_{n_x}\}$.

Отметим, что так как в общем случае m_λ может оказаться меньшим m_0, то теорема 4 дает возможность построения более экономичных автоматов с исправлением l ошибок, нежели теорема 3.

Для произвольного блочного покрытия λ произвольного автомата $A(x)$ выполняется неравенство:

$$\max_i n_a(x_i) \leqslant \max_i n_a(\lambda_i).$$

Следовательно, с учетом (15), для того чтобы выполнялось достаточное условие (23) существования $A^{(l)}(x)$, для которого $]\log_2 n_a^{(l)}[\leqslant m_0$, необходимо, чтобы существовало $C \geqslant 0$ такое, что

$$B(m_0, 2l+1) = 2^{k_0+c}. \tag{24}$$

Кроме того, отметим, что для нахождения минимального среди автоматов $A^{(l)}(x)$, которые могут быть построены на основании теоремы 4, достаточно рассмотреть только блочные покрытия для $A_{min}(x)$.

Обозначим:

$$\Delta_\lambda =]\log_2 \sum_{i=1}^{n_\lambda} n_a(\lambda_i)[- k_\lambda. \tag{25}$$

Следствие 1. Пусть для $A_{min}(x)$ удовлетворяется условие (24). Тогда, если для $A_{min}(x)$ существует блочное покрытие λ такое, что

$$\Delta_\lambda \geqslant m_0 - k_0 - c, \tag{26}$$

то для любого m, не меньшего m_λ, существует $A^{(l)}(x)$, для которого $]\log_2 n_a^{(l)}[= m$.

Замечание. Изложенный при доказательстве теоремы 4 метод построения автомата $A^{(l)}(x)$, использующий для кодирования состояний групповые коды с исправлением l ошибок, может быть обобщен и на негрупповые коды. Достаточные условия существования $A^{(l)}(x)$, для которого $]\log_2 n_a^{(l)}[= m$, при этом принимают вид

$$\begin{cases} \max\limits_i n_a(\lambda_i) \leqslant A(m, 2l+1), \\ n_\lambda \leqslant \sum\limits_{i=0}^{l} C_m^i \end{cases}$$

(где λ есть некоторое блочное покрытие $A(x)$).

Обозначим L множество всех блочных покрытий λ для $A_{min}(x)$, для которых выполняются (24) и (26). Пусть далее $\lambda_{min} \in L$ и

Для любого $\lambda \in L$ $m_\lambda = m$ ... Тогда число состояний минимального автомата $A^{(l)}(x)$ не превышает $2^{\dots}$.

Пример. Построим сумматор последовательного действия с исправлением одиночной ошибки. Граф сумматора последовательного действия, если его рассматривать как автомат Мура, представлен на рис. 1 [10]. Здесь x_0, x_1, x_2 — входные сигналы 00, 01 или 10, 11 соответственно; y_0 и y_1 — выходные сигналы 0 и 1 соответственно.

Автомат, изображенный на рис. 1, является минимальным. Для этого автомата

$$M_0 = \{0, 1\};\; M_1 = \{1, 2\};\; M_2 = \{2, 3\}.$$

Рассмотрим для этого автомата блочное покрытие: $\lambda = \lambda_0 =$

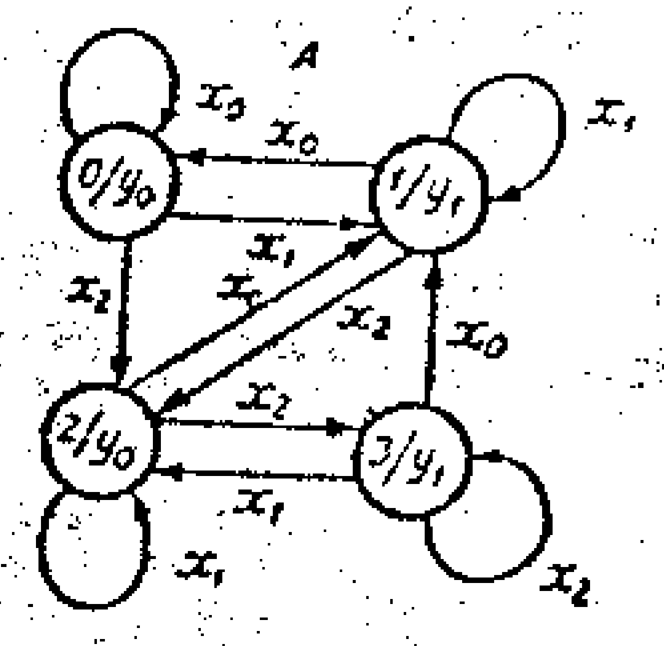

Рис. 1.

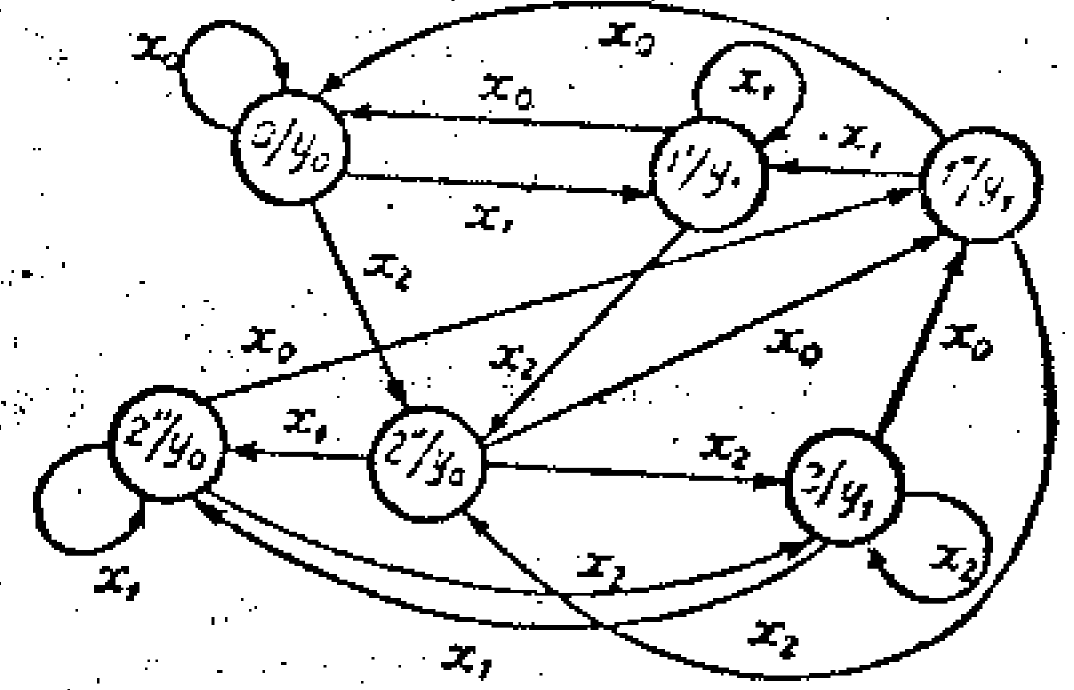

Рис. 2.

$= \{M_0, M_1, M_2\}$. Тогда $k_\lambda = k_0 = 1$, $m_\lambda = m_0 = 3$. Следовательно, по теореме 4, существует автомат $A^{(l)}(x)$, для которого $]\log_2 n_a^{(l)}[= 3$.

Граф автомата $A_\lambda(x) = A_0(x)$ представлен на рис. 2. Здесь состояния 1′,1″ и 2′,2″ получены „расщеплением" состояний 1 и 2 графа рис. 1 соответственно.

Тогда для исправления одиночной ошибки состояния $A_0(x)$ кодируются следующим образом:

0 — 000	2′ — 110
1′ — 001	2″ — 010
1″ — 111	3 — 101

Таким образом, сумматор последовательного действия с исправлением одиночной ошибки может быть реализован описанным методом на трех элементах памяти. Отметим, что при реализации этого же устройства методами, описанными в [1] и [3], потребовалось бы соответственно 5 и 6 элементов памяти.

§ 3. ПОСТРОЕНИЕ ФУНКЦИИ ПЕРЕХОДОВ ДЛЯ АВТОМАТОВ С ИСПРАВЛЕНИЕМ ОШИБОК

3.1. Функцию переходов $f^{(l)}(x(t), \xi(t), a(t-1))$ для $A^{(l)}(x)$ будем строить методом, описанным при доказательстве достаточного условия теоремы 1.

Из (10) видно, что $f^{(l)}(x(t), 1, a(t-1))$ определяется только функцией переходов автомата $A_\lambda(x)$, который является индуцированным автоматом для $A^{(l)}(x)$.

Рассмотрим теперь вопрос о реализации $f^{(l)}(x(t), 0, a(t-1))$.

При построении $A^{(l)}(x)$ методом, описанным при доказательстве теоремы 4, каждому входному символу x_i однозначным образом сопоставляется смежный класс с представителем $P(x_i)$ следующим образом: элементами смежного класса с представителем $P(x_i)$ кодируются все x_i-достижимые состояния $A_\lambda(x)$.

Пусть x_s удовлетворяет условию $P(x_s) = \left(\dfrac{00\ldots0}{m}\right)$.

Обозначим для любого a_j: $f^{(l)}(x_s, 0, a_j) = \varphi(a_j)$.

Теорема 5. *Для любых x_i и a_j справедливо следующее равенство:*

$$f^{(l)}(x_i, 0, a_j) = P(x_i) \oplus \varphi(a_j \oplus P(x_i)). \qquad (27)$$

Из теоремы 5 следует, что для реализации $f^{(l)}(x(t), 0, a(t-1))$ достаточно построить φ, которая является функцией декодирования используемого кода и не зависит от вида отображения $A(x)$.

Будем теперь строить схему, реализующую $f^{(l)}(x(t), \xi(t), a(t-1))$ в виде двух отдельных блоков, реализующих $f^{(l)}(x(t), 1, a(t-1))$ и $f^{(l)}(x(t), 0, a(t-1))$ и объединенных через схемы «ИЛИ». Блок, реализующий $f^{(l)}(x(t), 1, a(t-1))$, будем называть блоком, реализующим отображение $A(x)$ и сложность [9] его будем обозначать L_1, а блок, реализующий $f^{(l)}(x(t), 0, a(t-1))$, будем называть блоком исправления ошибок и сложность его будем обозначать L_0.

Тогда можно показать, что для любого из описанных в предыдущем разделе методов при фиксированном l, $n_x \to \infty$ и $n_a \to \infty$ справедливо следующее соотношение:

$$L_0 = O(L_1), \qquad (28)$$

т. е. при достаточно больших n_x и n_a сложность устройства, реализующего функцию переходов $A^{(l)}(x)$, не зависит от сложности блока исправления ошибок и определяется только сложностью блока, реализующего отображение $A(x)$.