

7. ПРЕОБРАЗОВАНИЕ КОДОВ С ПОМОЩЬЮ НЕЛИНЕЙНЫХ ДВОИЧНЫХ ДИНАМИЧЕСКИХ СИСТЕМ (КОНЕЧНЫХ АВТОМАТОВ)

Рассматриваются проблемы, связанные с использованием нелинейных двоичных динамических систем (НДДС), именуемых также конечными автоматами (КА), в составе устройств преобразования двоичных кодов. В качестве первоочередной ставится задача разработки методологии и алгоритмического обеспечения формирования аналитических представлений нелинейных модельных представлений ДДС. В связи с тем, что «нелинейность» в общесистемной постановке суть разновидность статической «памяти», то следует ожидать при использовании НДДС в составе устройств дискретной автоматики для решения задач преобразования кодов заметного сокращения размерности кода состояния ДДС, что влечет за собой уменьшения объема памяти, определяемого числом используемых триггеров. При этом разработчик устройств преобразования двоичных кодов (УПДК) средствами НДДС встречается с заметным их преимуществом перед ЛДДС, состоящим в возможности использования всего банка триггеров существующей триггерной логики как линейной, так и нелинейной, что в итоге существенно расширяет класс схемотехнических реализаций ДДС.

Поставленная задача решается двумя способами. Первый способ основан на использовании канонического алгоритма синтеза конечных автоматов, второй – на использовании граф – схем алгоритмов (ГСА) описания функционирования устройств дискретной автоматики с последующим погружением ГСА в автоматную среду.

7.1. Канонический алгоритм синтеза конечных автоматов (НДДС). Автоматные логики Мура и Мили

В настоящем параграфе в развитие положений параграфа 6.1, в котором в классе моделей «вход–состояние–выход» (ВСВ) (6.20) построены линейные представления правил (функций) λ перехода и δ выхода в форме (6.23) и (6.24), ставится задача конструирования их нелинейных аналогов. Для целей построения нелинейных модельных представлений правил λ и δ при описании ДДС используются возможности автоматной логики, которая опирается на процедуру канонического автоматного синтеза ДДС. Процедура канонического автоматного синтеза нелинейного модельного представления ДДС в форме НДДС реализуется в виде следующего алгоритма.

Алгоритм 7.1

конструирования модельного ВСВ представления НДДС на основе
канонического автоматного синтеза

1. Сформулировать постановку задачи преобразования кодов, решаемой конструируемой нелинейной ДДС (конечным автоматом).

2. Формализовать задачу преобразования кода в виде абстрактного автомата (АА), задаваемого в виде пятиэлементного макровектора

$$AA: \{ Z, S, W, \lambda, \delta \}, \quad (7.1)$$

где Z – алфавит высокого уровня (с возможным использованием вербальных описаний в виде глаголов в форме инфинитивов типа «сформировать» код...) входов абстрактного автомата мощности $[Z]=r_z$, S – алфавит высокого уровня его состояний мощности $[S]=n_s$, W – алфавит высокого уровня выходов АА мощности $[W]=m_w$, λ – правило (функция) перехода АА

$$s(k+1) = \lambda[s(k), z(k)], \quad (7.2)$$

где δ – правило (функция) выхода, задаваемое функциональными соотношениями соответственно

$$W(k) = \delta[s(k)] \quad (7.3)$$

в логике абстрактного автомата Мура, в соответствии с которой выходной сигнал однозначно определяется состоянием АА, и

$$W(k) = \delta[s(k), z(k)] \quad (7.4)$$

в логике абстрактного автомата Мили, в соответствии с которой выходной сигнал формируется на переходе из состояния в состояние. В (7.2) – (7.4) $s(k), s(k+1)$ – соответственно исходное состояние и состояние перехода АА, k – дискретное время, выраженное в числе тактов длительностью Δt . При этом основным математическим средством описания правил (функций) λ, δ на первом этапе конструирования являются графы переходов и выходов, на втором – таблицы переходов и выходов.

3. Осуществить переход от абстрактного автомата (7.1) к конечному автомату (КА)

$$KA: \{ U, X, Y, \lambda, \delta \} \quad (7.5)$$

над простым полем Галуа $GF(p)$ при $p=2$, путем кодирования элементов алфавитов высокого уровня АА (7.1) кодами, составленными из элементов поля $GF(p)$. В выражении (7.5) $U = K\{Z\}$, $X = K\{S\}$, $Y = K\{W\}$, где $K\{(*)\}$ – множество кодов (вектор-строк) элементов алфавита $(*)$ размерности $dim K\{(*)\}$.

Размерности кодов конечного автомата (7.5) и мощности алфавитов абстрактного автомата (7.1) связаны соотношениями:

$$\left. \begin{aligned} \dim U = r &= \operatorname{argmin}\{p^r \geq r_z\}_{p=2} = \operatorname{argmin}\{2^r \geq r_z\} \\ \dim X = n &= \operatorname{argmin}\{p^n \geq n_s\}_{p=2} = \operatorname{argmin}\{2^n \geq n_s\} \\ \dim W = m &= \operatorname{argmin}\{p^m \geq m_w\}_{p=2} = \operatorname{argmin}\{2^m \geq m_w\} \end{aligned} \right\}. \quad (7.6)$$

Коды алфавитов входа и выхода могут строиться в рамках требований (7.6) достаточно произвольно. Коды элементов алфавита состояния с тем, чтобы избежать начальной предустановки должны использовать нулевую комбинацию, а также учитывать специфику графа переходов АА. Так, если в графе переходов АА явно обнаруживается некоторая его цикличность, то из соображений простоты технической реализации НДДС коды ее состояний, соседние по графу, должны быть максимально приближены к соседним, то есть должны характеризоваться минимальным кодовым расстоянием.

Представить правила λ , δ (7.2) – (7.4) КА после процедуры кодирования соответствующих алфавитов АА, соответственно в виде

$$\lambda: x(k+1) = \lambda[x(k), u(k)], \quad (7.7)$$

и

$$\delta: y(k) = \delta[x(k)] \quad (7.8)$$

при использовании автоматной логики Мура, и

$$\delta: y(k) = \delta[x(k), u(k)] \quad (7.9)$$

при использовании автоматной логики Мили, где $x(k)$, $x(k+1)$ – соответственно коды исходного состояния и состояния перехода. Представить правило перехода λ (7.7) и выхода δ (7.7) и δ (7.8) сформированных КА в виде таблиц переходов и выходов КА.

4. Рассматривая таблицы переходов и выходов КА, сформированных в п.3 алгоритма как таблицы истинности функций $\lambda: x(k+1) = \lambda[x(k), u(k)]$, и $\delta: y(k) = \delta[x(k)]$ и $\delta: y(k) = \delta[x(k), u(k)]$ относительно логических переменных $x(k)$, $u(k)$ составить аналитические представления этих функций в булевом базисе.

5. Выбрать тип автоматной логики (Мура или Мили) функционирования конечного автомата на основе анализа требований, предъявляемых к НДДС по быстродействию и информационной надежности, и таблиц переходов и выходов КА, полученных в результате выполнения п.3 алгоритма.

6. Выбрать тип используемых при построении НДДС триггеров, число которых не зависит от выбранного их типа и определяется размерностью n кода состояния автоматного представления НДДС. Учесть, что выбор конкретного типа триггера вводит в рассмотрение

дополнительную функцию описания КА – функцию μ возбуждения информационного входа v триггера, задаваемую в форме

$$v(k) = \mu[x(k), x(k+1)]. \quad (7.10)$$

7. Дополнить аналитическое представление функционирования НДДС в п.4 булевыми функциями, описывающими процесс формирования сигналов возбуждения информационных входов триггеров в форме

$$v(k) = \mu[x(k), \lambda[x(k), u(k)]] = \tilde{\mu}[x(k), u(k)]. \quad (7.11)$$

Для построения БФ (7.11) сконструировать таблицу возбуждения информационных входов всех триггеров выбранного типа на основе представления (7.10) и таблицы переходов КА. Построенную таблицу использовать для построения БФ (7.11) в качестве таблицы истинности. При этом следует иметь в виду, что булево представление функции $\lambda: x(k+1) = \lambda[x(k), u(k)]$, полученное в п.4, по существу является функцией возбуждения информационных входов D – триггеров.

8. На основе булевого аналитического описания НДДС осуществить проверку правильности ее функционирования.

9. Осуществить выбор элементной базы, привязать к ней аналитические описания п.4 и (7.11), и построить схемотехническую реализацию НДДС. ■

Пример 7.1

Решается задача конструирования НДДС, преобразующая входную единичную последовательность в периодическую последовательность, обеспечивающую выборку информационных разрядов из ПЗК Хэмминга (7,4) (см. Пример 6.1).

Решение

Для решения поставленной задачи конструирования ДДС воспользуемся алгоритмом 7.1.

1. Поставлена задача сформировать НДДС (конечный автомат), средствами которой входная двоичная последовательность $u(k) = 1(k):111К\ 11К$ преобразуется в выходную периодическую последовательность $y(k):1110100|1110100|К$

2. В соответствии с постановкой задачи назначаем алфавиты $Z = \{z_1, z_2\}$ входа, $S = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$ состояния и $W = \{w_1, w_2\}$ выхода описания устройства в форме АА: $\{Z, S, W, \lambda, \delta\}$, которые характеризуются соответственно мощностями $r_z = 2, n_s = 7, m_w = 2$, и задаем правила λ и δ в виде графов переходов и выходов соответственно в логике АА Мура (рисунок 7.1) и в логике АА Мили (рисунок 7.2).

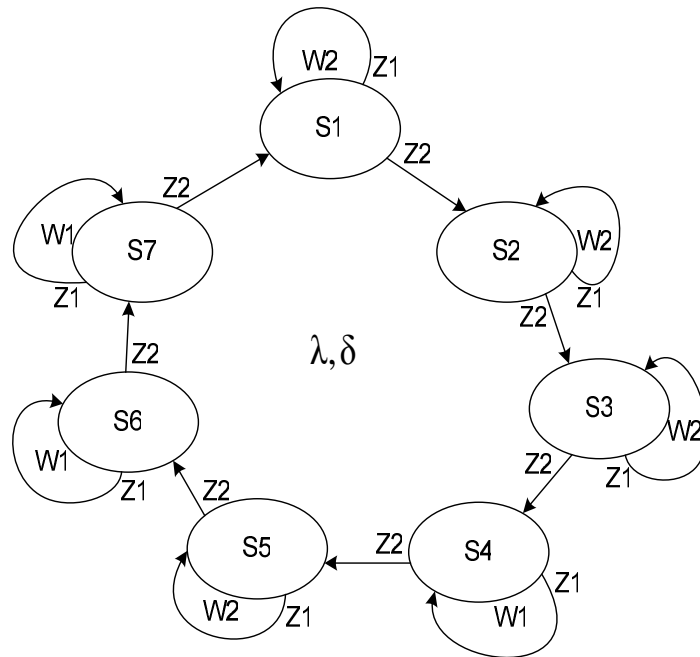


Рисунок 7.1. Граф переходов и выходов в логике АА Мура

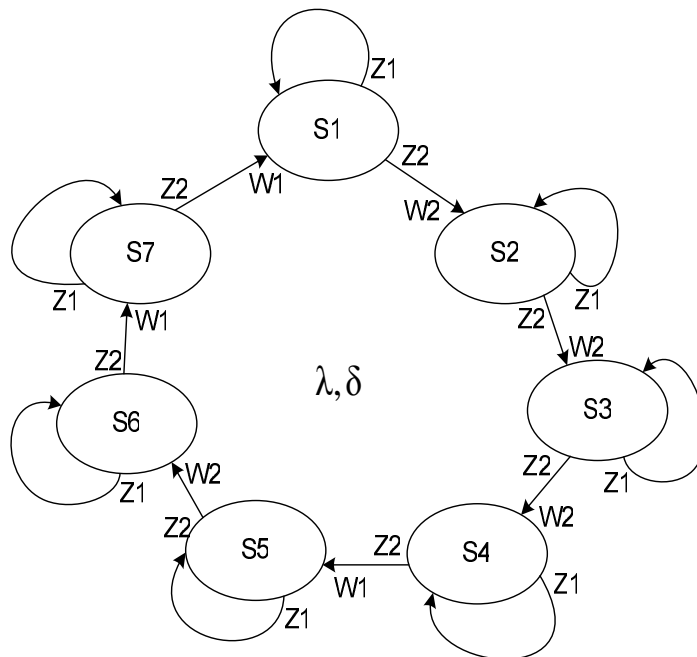


Рисунок 7.2. Граф переходов и выходов в логике АА Мили

При этом соответствующие им таблицы правила λ перехода и правила δ выхода запишутся в виде таблиц 7.1, 7.2. и 7.3..Причем таблица 7.1 в силу автоматной логики Мура совмещенная в том смысле, что совмещает процессы перехода и формирования выхода.

АА Мура: $s(k+1) = \lambda[s(k), z(k)]$, $w(k) = \delta[s(k)]$ Таблица 7.1

$w(k)$	$w_2(k)$	$w_2(k)$	$w_2(k)$	$w_1(k)$	$w_2(k)$	$w_1(k)$	$w_1(k)$
$s(k)$ $z(k)$	$s_1(k)$	$s_2(k)$	$s_3(k)$	$s_4(k)$	$s_5(k)$	$s_6(k)$	$s_7(k)$
$z_1(k)$	$s_1(k+1)$	$s_2(k+1)$	$s_3(k+1)$	$z_1(k)$	$s_1(k+1)$	$s_6(k+1)$	$s_7(k+1)$
$z_2(k)$	$s_2(k+1)$	$s_3(k+1)$	$s_4(k+1)$	$s_5(k+1)$	$s_6(k+1)$	$s_7(k+1)$	$s_1(k+1)$

АА Мили: $s(k+1) = \lambda[s(k), z(k)]$, Таблица 7.2

$s(k)$ $z(k)$	$s_1(k)$	$s_2(k)$	$s_3(k)$	$s_4(k)$	$s_5(k)$	$s_6(k)$	$s_7(k)$
$z_1(k)$	$s_1(k+1)$	$s_2(k+1)$	$s_3(k+1)$	$z_1(k)$	$s_1(k+1)$	$s_6(k+1)$	$s_7(k+1)$
$z_2(k)$	$s_2(k+1)$	$s_3(k+1)$	$s_4(k+1)$	$s_5(k+1)$	$s_6(k+1)$	$s_7(k+1)$	$s_1(k+1)$

АА Мили: $w(k) = \delta[s(k), z(k)]$ Таблица 7.3

$s(k)$ $z(k)$	$s_1(k)$	$s_2(k)$	$s_3(k)$	$s_4(k)$	$s_5(k)$	$s_6(k)$	$s_7(k)$
$z_1(k)$	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
$z_2(k)$	$w_2(k)$	$w_2(k)$	$w_2(k)$	$w_1(k)$	$w_2(k)$	$w_1(k)$	$w_1(k)$

В таблице 7.3 \emptyset – пустой символ, формируемый на переходе $s_i(k) \rightarrow s_i(k+1)$.

Нетрудно видеть, что при использовании автоматной логики Мура граф переходов и выходов $\{\lambda, \delta\}$ отображается в совмещенную таблицу, а при использовании автоматной логики Мили – в две таблицы.

3. Осуществим переход от АА: $\{Z, S, W, \lambda, \delta\}$ к конечному автомату КА: $\{U, X, Y, \lambda, \delta\}$ путем кодирования элементов алфавитов высокого уровня АА элементами двоичного поля Галуа $GF(p=2) = \{0, 1\}$, определив первоначально размерности получаемых кодов в силу соотношения (7.6), которые позволяют записать

$$\left. \begin{aligned} \dim U = r = \operatorname{argmin} \{p^r \geq r_Z\}_{p=2} &= \operatorname{argmin} \{2^r \geq r_Z\}_{r_Z=2} = 1 \\ \dim X = n = \operatorname{argmin} \{p^n \geq n_S\}_{p=2} &= \operatorname{argmin} \{2^n \geq n_S\}_{n_S=7} = 3 \\ \dim W = m = \operatorname{argmin} \{p^m \geq m_W\}_{p=2} &= \operatorname{argmin} \{2^m \geq m_W\}_{m_W=2} = 1 \end{aligned} \right\},$$

как для автоматной логики Мура, так и для автоматной логики Мили. Результаты кодирования элементов алфавитов АА Мура и Мили в силу идентичности их алфавитов сведены в единую таблицу 7.4.

Таблица кодирования алфавитов АА Мура и Мили Таблица 7.4

Z \ U	U = K{Z}	W \ Y	Y = K{W}	S \ X	X = K{S}		
	u		y		x ₁	x ₂	x ₃
z ₁	0	w ₁	0	s ₁	0	0	0
z ₂	1	w ₂	1	s ₂	0	0	1
				s ₃	0	1	1
				s ₄	0	1	0
				s ₅	1	1	0
				s ₆	1	1	1
				s ₇	1	0	1

Из таблицы 7.4 видно, что при кодировании алфавитов состояния АА использовано соседнее кодирование, осуществленное по схеме Грея. Действительно, кодовое расстояние $d(s_i, s_j)$ между состояниями s_i, s_j при $j = i + 1$ составляет величину $d(s_i, s_j) = 1$, при $i = 7$ и $j = 1$ кодовое расстояние составляет величину $d(s_i, s_j) = 2$. Это объясняется тем, что мощность алфавита состояний оказалась не полной так, как $2^n \Big|_{n=3} = 8 > n_s = 7$.

Представим правила перехода λ (7.7) и выхода δ (7.8) КА на основании кодирования алфавитов АА таблица 7.4 и таблиц 7.1 – 7.3 в виде таблиц 7.5 – 7.7 переходов и выходов КА.

КА Мура: $x(k+1) = \lambda[x(k), u(k)]$, $y(k) = \delta[x(k)]$ Таблица 7.5

y(k)	1	1	1	0	1	0	0
x(k) \ u(k)	000	001	011	010	110	111	101
0	000	001	011	010	110	111	101
1	001	011	010	110	111	101	000
	$x(k+1) = \{x_1(k+1)x_2(k+1)x_3(k+1)\}$						

КА Мили: $x(k+1) = \lambda[x(k), u(k)]$,

Таблица 7.6

$x(k) \backslash u(k)$	000	001	011	010	110	111	101
0	000	001	011	010	110	111	101
1	001	011	010	110	111	101	000
$x(k+1) = \{x_1(k+1)x_2(k+1)x_3(k+1)\}$							

КА Мили: $y(k) = \delta[x(k), u(k)]$

Таблица 7.7

$x(k) \backslash u(k)$	000	001	011	010	110	111	101
0	∅	∅	∅	∅	∅	∅	∅
1	1	1	1	0	1	0	0

4. Рассматривая таблицы 7.5 и 7.6 переходов и выходов 7.7 КА как таблицы истинности функций $\lambda: x(k+1) = \lambda[x(k), u(k)]$, и $\delta: y(k) = \delta[x(k)]$ и $\delta: y(k) = \delta[x(k), u(k)]$ относительно логических переменных $x(k), u(k)$ составим аналитические представления этих функций в булевом базисе с учетом того, что $u(k) = 1(k): 11111K 1K . (\bar{u}(k) = 0)$ Тогда получим следующие системы булевых функций для:

автоматной логики Мура:

$$y(k) = \bar{x}_1(k)\bar{x}_2(k)\bar{x}_3(k) \vee \bar{x}_1(k)\bar{x}_2(k)x_3(k) \vee \bar{x}_1(k)x_2(k)x_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k) = \\ = \bar{x}_1(k)\bar{x}_2(k) \vee \bar{x}_1(k)x_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k);$$

$$x_1(k+1) = u(k)\{\bar{x}_1(k)x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)x_3(k)\} = \\ = u(k)\{x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)\} = u(k)x_2(k)\bar{x}_3(k) \vee u(k)x_1(k)x_2(k);$$

$$x_2(k+1) = u(k)\{\bar{x}_1(k)\bar{x}_2(k)x_3(k) \vee \bar{x}_1(k)x_2(k)x_3(k) \vee \bar{x}_1(k)x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k)\} = \\ = u(k)\{\bar{x}_1(k)x_3(k) \vee x_2(k)\bar{x}_3(k)\} = u(k)\bar{x}_1(k)x_3(k) \vee u(k)x_2(k)\bar{x}_3(k);$$

$$x_3(k+1) = u(k)\{\bar{x}_1(k)\bar{x}_2(k)\bar{x}_3(k) \vee \bar{x}_1(k)\bar{x}_2(k)x_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)x_3(k)\} = \\ = u(k)\{\bar{x}_1(k)\bar{x}_2(k) \vee x_1(k)x_2(k)\} = u(k)\bar{x}_1(k)\bar{x}_2(k) \vee u(k)x_1(k)x_2(k);$$

автоматной логики Мура:

$$\begin{aligned} y(k) &= u(k)\{\bar{x}_1(k)\bar{x}_2(k)\bar{x}_3(k) \vee \bar{x}_1(k)\bar{x}_2(k)x_3(k) \vee \bar{x}_1(k)x_2(k)x_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k)\} = \\ &= u(k)\{\bar{x}_1(k)\bar{x}_2(k) \vee \bar{x}_1(k)x_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k)\} \end{aligned}$$

$$\begin{aligned} x_1(k+1) &= u(k)\{\bar{x}_1(k)x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)x_3(k)\} = \\ &= u(k)\{x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)\} = u(k)x_2(k)\bar{x}_3(k) \vee u(k)x_1(k)x_2(k); \end{aligned}$$

$$\begin{aligned} x_2(k+1) &= u(k)\{\bar{x}_1(k)\bar{x}_2(k)x_3(k) \vee \bar{x}_1(k)x_2(k)x_3(k) \vee \bar{x}_1(k)x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k)\} = \\ &= u(k)\{\bar{x}_1(k)x_3(k) \vee x_2(k)\bar{x}_3(k)\} = u(k)\bar{x}_1(k)x_3(k) \vee u(k)x_2(k)\bar{x}_3(k); \end{aligned}$$

$$\begin{aligned} x_3(k+1) &= u(k)\{\bar{x}_1(k)\bar{x}_2(k)\bar{x}_3(k) \vee \bar{x}_1(k)\bar{x}_2(k)x_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k) \vee x_1(k)x_2(k)x_3(k)\} = \\ &= u(k)\{\bar{x}_1(k)\bar{x}_2(k) \vee x_1(k)x_2(k)\} = u(k)\bar{x}_1(k)\bar{x}_2(k) \vee u(k)x_1(k)x_2(k); \end{aligned}$$

5. Выбираем на основе требований повышенной информационной надежности (помехозащищенности), предъявляемых к устройствам технической информатики и телемеханики для реализации НДДС преобразования кодов версию, построенную на основе автоматной логики Мура, для целей проверки правильности ее функционирования и технической реализации.

6. Для сравнимости результатов настоящего синтеза с результатами решения примера 6.1 выбираем для построения НДДС триггер типа D .

7. Дополняем булевы функции формирования выхода в автоматной логике Мура, полученными в п.4

$$y(k) = \bar{x}_1(k)\bar{x}_2(k) \vee \bar{x}_1(k)x_3(k) \vee x_1(k)x_2(k)\bar{x}_3(k),$$

булевыми функциями, описывающими процесс формирования сигналов возбуждения информационных входов D – триггеров в форме

$$v_1(k) = \mu[x(k), x(k+1)] = x_1(k+1) = u(k)x_2(k)\bar{x}_3(k) \vee u(k)x_1(k)x_2(k);$$

$$v_2(k) = \mu[x(k), x(k+1)] = x_2(k+1) = u(k)\bar{x}_1(k)x_3(k) \vee u(k)x_2(k)\bar{x}_3(k);$$

$$v_3(k) = \mu[x(k), x(k+1)] = x_3(k+1) = u(k)\bar{x}_1(k)\bar{x}_2(k) \vee u(k)x_1(k)x_2(k).$$

8. Осуществим проверку правильности функционирования аналитически спроектированной НДДС на основе ее булевого описания, приведенного в п.п.4 и 7. Результаты сведем в таблицу.

Таблица 7.8

k	$(*) (k)$											$(*) (k + 1)$		
	u	x_1	x_2	x_3	\bar{x}_1	\bar{x}_2	\bar{x}_3	x_1x_2	$\bar{x}_1\bar{x}_2$	\bar{x}_1x_3	$x_2\bar{x}_3$	x_1	x_2	x_3
0	1	0	0	0	1	1	1	0	1	0	0	0	0	1
1	1	0	0	1	1	1	0	0	1	1	0	0	1	1
2	1	0	1	1	1	0	0	0	0	1	0	0	1	0
3	1	0	1	0	1	0	1	0	0	0	1	1	1	0
4	1	1	1	0	0	0	1	1	0	0	1	1	1	1
5	1	1	1	1	0	0	0	1	0	0	0	1	0	1
6	1	1	0	1	0	1	0	0	0	0	0	0	0	0
7	1	0	0	0	1	1	1	0	1	0	0	0	0	1
8	1	0	0	1	1	1	0	0	1	1	0	0	1	1
k	0	1	2	3	4	5	6	7	8	9	10	11	12	13
$y(k)$	1	1	1	0	1	0	0	1	1	1	0	1	0	0

Таблица 7.8 позволяет констатировать правильность функционирования спроектированной НДДС (конечного автомата).

9. Теперь можно осуществить выбор элементной базы, привязать к ней аналитические (булевы) описания спроектированной НДДС и построить ее схемотехническую реализацию.

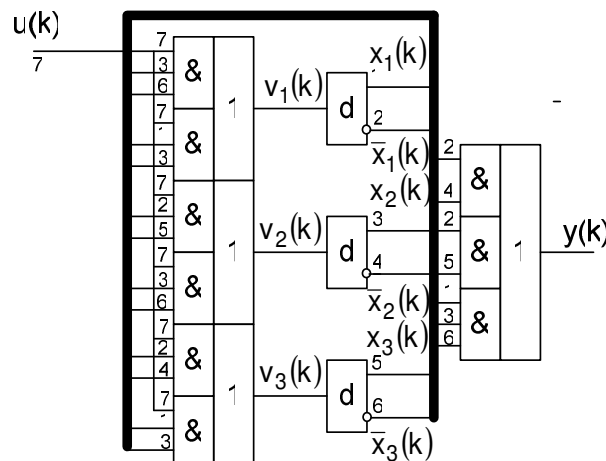


Рисунок 7.3

Примечание 7.1. Сравнивая решения одной и той же задачи преобразования двоичных кодовых последовательностей в форме НДДС (рисунок 7.3) и в форме ЛДДС (см. Пример 6.1, рисунки 6.3 и 6.4), следует констатировать, что уменьшение объема памяти

нелинейной ДДС по сравнению с линейной сопровождается заметным усложнением комбинационной схемы, в среде которой формируется выходная двоичная последовательность и сигналы возбуждения входов используемых триггеров. ■

7.2. Аналитическое описание комбинационных схем КА. Булевы функции и их свойства, базисы представления, дизъюнктивные совершенные нормальные формы (ДСНФ) и конъюнктивные совершенные нормальные формы (КСНФ), проблема редуцирования аналитического представления переключательных функций. Булевы производные, области применения

Любая динамическая система над бесконечными и конечными полями, процессы в которых параметризованы непрерывным или дискретным временем представляет собой структурное объединение блока памяти (БП) и комбинационной схемы (КС). В непрерывных системах (объектах) блок памяти реализуется на интеграторах, в дискретных системах над бесконечным полем БП реализуется на элементах задержки, а в дискретных системах над конечным (двоичном) полем Галуа $GF(p)|_{p=2} = \{0,1\}$ БП реализуется на триггерах.

Комбинационная схема динамической системы осуществляет безынерционное формирование выходного сигнала системы и сигналов возбуждения входов элементов БП, реализуя тем самым функции перехода λ и выхода δ системы. Математическое описание функций λ и δ систем над бесконечными полями строится на возможностях элементарной и высшей математики. Математическое описание этих же функций дискретных систем над двоичным полем Галуа строится на возможностях аппарата булевых функций.

Погружение в аппарат булевых функций начнем с определения.

Определение 7.1. Булевой функцией n булевых переменных $y = f(x_1, x_2 \dots x_i \dots x_n)$, (7.12)

где $\{y, x_i (i = \overline{1, n})\} \in GF(2) = \{0,1\}$, называется отображение n – мерного вектора – строки с элементами из $GF(2)$ в $GF(2)$. □

Примечание 7.7. Булева функция (7.12) может так же называться логической или переключательной функцией.

Число N_f булевых функций зависит от числа ее переменных так, что $N_f = N_f(n)$, при этом оно определяется выражением

$$N_f(n) = 2^{2^n}. \quad (7.13)$$

Рассмотрим булевы функции одной переменной $y = f(\text{row}(x_i; i = \overline{1, n}))_{n=1}$. Число N_f булевых функций одной переменной в соответствии с (7.13) равно $N_f = 2^2 = 4$. В таблице 7.9 приведены булевы функции одной переменной и даны их наименования.

Таблица 7.9

x	Наименование булевых функций одной переменной			
	константа «0»	повторение аргумента x	инверсия аргумента x	константа «1»
	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
0	0	0	1	1
1	0	1	0	1

Рассмотрим булевы функции двух переменных $y = f(\text{row}(x_i; i = \overline{1, n}))_{n=2}$. Число N_f булевых функций двух переменных в соответствии с (7.13) равно $N_f = 2^{2^2} = 2^4 = 16$. В таблице 7.10 приведены значения булевых функций двух переменных.

Таблица 7.10

x_1	x_2	значения булевых функций двух переменных $f_i(x_1, x_2); i = \overline{1, 16}$															
		f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}
0	0	0	0	1	1	1	0	1	1	0	0	0	1	0	1	1	0
0	1	0	1	1	0	0	1	1	0	0	1	0	1	1	0	1	0
1	0	0	1	0	1	0	1	1	0	1	0	1	0	0	1	1	0
1	1	1	1	1	1	1	0	0	0	0	0	1	0	1	0	1	0

Исследование булевых функций двух переменных начнем с базисных свойств этих функций. Этим базисным свойствам БФ двух переменных пять:

Свойство 1 – свойство сохранения нуля, состоящее в выполнении равенства

$$f_i(x_1, x_2)|_{x_1=x_2=0} = f_i(0, 0) = 0; \quad (7.14)$$

Свойство 2 – свойство сохранения единицы, состоящее в выполнении равенства

$$f_i(x_1, x_2)|_{x_1=x_2=1} = f_i(1, 1) = 1; \quad (7.15)$$

Свойство 3 – свойство самодвойственности (нечетности), состоящее в выполнении равенства

$$f_i(\bar{x}_1, \bar{x}_2) = \bar{f}_i(x_1, x_2); \quad (7.16)$$

Свойство 4– свойство монотонности (выпуклости), состоящее в выполнении неравенства

$$f_i(\hat{x}_1, \hat{x}_2) \geq f_i(x_1, x_2) \text{ при } \hat{x}_1 \geq x_1, \hat{x}_2 \geq x_2; \quad (7.17)$$

при этом принято считать, что справедливы неравенства $0 \geq 0; 1 \geq 0; 1 \geq 1$, более того отношение порядка переносятся на парные наборы в форме

$$\{01\} \geq \{00\}, \{10\} \geq \{00\}, \{11\} \geq \{00\}, \{11\} \geq \{01\}, \{11\} \geq \{10\},$$

причем наборы $\{01\}$ и $\{10\}$ считаются несравнимыми.

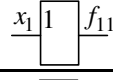
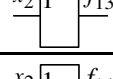
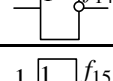
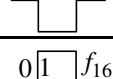
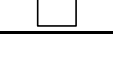
Свойство 5– свойство линейности, состоящее в выполнении равенства

$$f_i(x_1, x_2) = a \oplus b \cdot x_1 \oplus c \cdot x_2; a, b, c \in GF(2) = \{0, 1\}. \quad (7.18)$$

В таблицу 7.11 сведены булевы функции двух переменных, значения которых приведены в таблице 7.10, дополненные их наименованиями, аналитическими описаниями и условными обозначениями, а также указанием свойств, которыми обладает каждая БФ двух переменных.

Таблица 7.11

f_i	Наименование f_i	Аналитическое описание f_i	Условное обозначение f_i	Свойства БФ $f_i(x_1, x_2)$				
				1	2	3	4	5
f_1	конъюнкция	$f_1 = x_1 \cdot x_2 = x_1 x_2$		+	+	-	+	-
f_2	дизъюнкция	$f_2 = x_1 \vee x_2$		+	+	-	+	-
f_3	импликация x_1 в x_2	$f_3 = x_1 \rightarrow x_2$		-	+	-	-	-
f_4	импликация x_2 в x_1	$f_4 = x_1 \leftarrow x_2$		-	+	-	-	-
f_5	равнозначность	$f_5 = x_1 \sim x_2$		-	+	-	-	+
f_6	Неравнозначность (сумма по mod 2)	$f_6 = x_1 \oplus x_2$		+	-	-	-	+
f_7	функция Шеффера	$f_7 = x_1 / x_2$		-	-	-	-	-
f_8	функция Вебба (стрелка Пирса)	$f_8 = x_1 \downarrow x_2$		-	-	-	-	-
f_9	функция запрета x_1	$f_9 = x_1 \overrightarrow{\rightarrow} x_2$		+	-	-	-	-

f_i	Наименование f_i	Аналитическое описание f_i	Условное об-ние f_i	Свойства БФ $f_i(x_1, x_2)$				
				1	2	3	4	5
f_{10}	функция запрета x_2	$f_{10} = x_1 \overline{\leftarrow} x_2$		+	-	-	-	-
f_{11}	повторение x_1	$f_{11} = x_1$		+	+	+	+	+
f_{12}	инверсия x_1	$f_{12} = \bar{x}_1$		-	-	+	-	+
f_{13}	повторение x_2	$f_{13} = x_2$		+	+	+	+	+
f_{14}	инверсия x_2	$f_{14} = \bar{x}_2$		-	-	+	-	+
f_{15}	константа «1»	$f_{15} = 1$		-	+	-	+	+
f_{16}	константа «0»	$f_{16} = 0$		+	-	-	+	+

Рассмотрим свойства отдельных булевых функций двух переменных, сведенных в таблицу 7.11.

Простые свойства конъюнкции, дизъюнкции и инверсии сведены в таблицу 7.17.

Таблица 7.12

Аналитическое представление функции	1	2	3	4	5	6	7	8
	$x \vee x$	$x \cdot x$	$x \vee 1$	$x \cdot 1$	$x \vee 0$	$x \cdot 0$	$x \vee \bar{x}$	$x \cdot \bar{x}$
Значение функции	x	x	1	x	x	0	1	0

Примечание 7.3.

1. Функция, аналитическое представление и значение которой размещены в столбце 1, описывает свойство «тавтологии» дизъюнкции, развернутая форма которого принимает вид $x = x \vee x = x \vee x \vee x = x \vee x \vee x \vee \mathbf{K} \vee x$;

2. Функция, аналитическое представление и значение которой размещены в столбце 2, описывает свойство «тавтологии» конъюнкции, развернутая форма которого принимает вид $x = x \cdot x = x \cdot x \cdot x = x \cdot x \cdot x \cdot \mathbf{K} \cdot x$;

3. Функция, аналитическое представление и значение которой размещены в столбце 3, описывает свойство «поглощения» дизъюнкции;

4. Функция, аналитическое представление и значение которой размещены в столбце 7, описывает свойство «склеивания» дизъюнкции.

На этих свойствах построены все алгоритмы минимизации аналитического представления булевых функций, записанных в булевом базисе в виде дизъюнкций основных конъюнкций.

Функциональные свойства конъюнкции и дизъюнкции:

1. *Ассоциативность* конъюнкции и дизъюнкции:

$$1.1. x_1 \cdot (x_2 \cdot x_3) = (x_1 \cdot x_2) \cdot x_3;$$

$$1.2. x_1 \vee (x_2 \vee x_3) = (x_1 \vee x_2) \vee x_3;$$

2. *Коммутативность* элементов конъюнкции и дизъюнкции:

$$2.1. x_1 \cdot x_2 = x_2 \cdot x_1;$$

$$2.2. x_1 \vee x_2 = x_2 \vee x_1;$$

3. *Дистрибутивность* конъюнкции и дизъюнкции:

$$3.1. x_1 \cdot (x_2 \vee x_3) = (x_1 \cdot x_2) \vee (x_1 \cdot x_3);$$

$$3.2. x_1 \vee (x_2 \cdot x_3) = (x_1 \vee x_2) \cdot (x_1 \vee x_3);$$

4. *Формула де Моргана* для конъюнкции и дизъюнкции:

4.1. Словесная формулировка *формулы де Моргана* для конъюнкции звучит так: «Инверсия конъюнкции n – булевых переменных равняется дизъюнкции инверсий переменных», что аналитически принимает вид

$$\overline{x_1 \cdot x_2 \cdot \dots \cdot x_n} = \bar{x}_1 \vee \bar{x}_2 \vee \dots \vee \bar{x}_n;$$

4.2. Словесная формулировка *формулы де Моргана* для дизъюнкции звучит так: «Инверсия дизъюнкции n – булевых переменных равняется конъюнкции инверсий переменных», что аналитически принимает вид

$$\overline{x_1 \vee x_2 \vee \dots \vee x_n} = \bar{x}_1 \cdot \bar{x}_2 \cdot \dots \cdot \bar{x}_n.$$

Все перечисленные выше функциональные свойства конъюнкций и дизъюнкций традиционно в практике булевых функций не доказываются, а показываются путем фиксации совпадения значений левых и правых частей на всех наборах переменных, оформляемых в виде таблиц истинности. Проиллюстрируем этот подход на примере формул де Моргана для случая $n = 2$, так что эти формулы примут вид

$$\overline{x_1 \cdot x_2} = \bar{x}_1 \vee \bar{x}_2; \quad \overline{x_1 \vee x_2} = \bar{x}_1 \cdot \bar{x}_2.$$

Результаты вычисления четырех булевых функций сведены в таблицу 7.13.

Таблица 7.13

$x_i (i=1,2)$		$\overline{x_1 \cdot x_2} = \bar{x}_1 \vee \bar{x}_2$					$\overline{x_1 \vee x_2} = \bar{x}_1 \cdot \bar{x}_2$				
x_1	x_2	$x_1 \cdot x_2$	$\bar{x}_1 \cdot \bar{x}_2$	\bar{x}_1	\bar{x}_2	$\bar{x}_1 \vee \bar{x}_2$	$x_1 \vee x_2$	$\bar{x}_1 \vee \bar{x}_2$	\bar{x}_1	\bar{x}_2	$\bar{x}_1 \cdot \bar{x}_2$
0	0	0	1	1	1	1	0	1	1	1	1

0	1	0	1	1	0	1	1	0	1	0	0
1	0	0	1	0	1	1	1	0	0	1	0
1	1	1	0	0	0	0	1	0	0	0	0

Завершая знакомство со свойствами конъюнкций и дизъюнкций, сделаем полезное примечание.

Примечание 7.4. Введем в рассмотрение обобщенные конъюнкцию и дизъюнкцию двух переменных, принадлежащих произвольному числовому полю.

Определение 7.7. *Обобщенной конъюнкцией* $x_i \cdot x_j$ двух переменных x_i и x_j , принадлежащих произвольному числовому полю, называется функция этих переменных, удовлетворяющая аналитическому выражению

$$x_i \cdot x_j = \min\{x_i, x_j\}. \quad \square$$

Определение 7.3. *Обобщенной дизъюнкцией* $x_i \vee x_j$ двух переменных x_i и x_j , принадлежащих произвольному числовому полю, называется функция этих переменных, удовлетворяющая аналитическому выражению

$$x_i \vee x_j = \max\{x_i, x_j\}. \quad \square$$

Введенные понятия обобщенных конъюнкции и дизъюнкции двух переменных, принадлежащих произвольному числовому полю, позволяют применять методы логического синтеза устройств управления для сигнальной среды произвольной природы.

Теперь поставим задачу поиска возможностей представления произвольной булевой функции, задаваемой аналитически или таблично, в виде аналитической композиции конечного набора канонических булевых функций, приведенных в таблице 7.11, и образующих базисную систему (или просто базис). В этой связи оказываются полезными положения утверждения, приводимого без доказательства.

Утверждение 7.1. Для того, чтобы набор булевых функций двух переменных из их полного перечня, представленного таблицей 7.11 с указанием их базисных свойств, образовывал базисную систему необходимо, чтобы в его составе находилась, по крайней мере, одна БФ, которая:

- не сохраняла константу «1»;
- не сохраняла константу «0»;
- не была самодвойственной;
- не была монотонной;
- не была линейной. □

Опираясь на положения утверждения 7.1 и базисные свойства БФ двух переменных, сведенных в таблицу 7.11, можно сконструировать достаточно обширный банк базисных систем, примеры которых приводятся ниже:

1. Оптимальная базисная система, содержащая только одну БФ двух переменных – *функцию Шеффера* («И – НЕ»)

$$f_7(x_1, x_2) = x_1 / x_2 = x_1 \cdot x_2;$$

2. Оптимальная базисная система, содержащая только одну БФ двух переменных – *функцию Вебба* («ИЛИ – НЕ»)

$$f_8(x_1, x_2) = x_1 \downarrow x_2 = x_1 \vee x_2;$$

3. Базисная система, содержащая две БФ двух переменных – *конъюнкция* («И») и *инверсия* («НЕ»)

$$f_1(x_1, x_2) = x_1 \cdot x_2, f_{12,14}(x_i) = \bar{x}_i, (i = 1, 2);$$

4. Базисная система, содержащая две БФ двух переменных – *дизъюнкция* («ИЛИ») и *инверсия* («НЕ»)

$$f_2(x_1, x_2) = x_1 \vee x_2, f_{12,14}(x_i) = \bar{x}_i, (i = 1, 2);$$

5. Избыточная базисная система, именуемая базисной системой Дж. Буля или просто *булевым базисом* («И – ИЛИ – НЕ»), содержащая три БФ двух переменных – *конъюнкция* («И»), *дизъюнкция* («ИЛИ») и *инверсия* («НЕ»)

$$f_1(x_1, x_2) = x_1 \cdot x_2, f_2(x_1, x_2) = x_1 \vee x_2, f_{12,14}(x_i) = \bar{x}_i, (i = 1, 2);$$

6. Избыточная базисная система, именуемая базисной системой И.И. Жегалкина или просто *базисом Жегалкина*, содержащая три БФ двух переменных – *конъюнкция* («умножение по mod 2»), *неравнозначность* («сложение по mod 2») и *константа* «1»

$$f_1(x_1, x_2) = x_1 \cdot x_2 = x_1 \otimes x_2, f_6(x_1, x_2) = x_1 \oplus x_2, f_{15}(x_i) = 1, (i = 1, 2).$$

При сравнении базисов представления булевых функций по степени удобства их применения надо иметь в виду две сферы их использования:

1. сфера формирования *аналитического представления* БФ комбинационной схемы проектируемого конечного автомата в выбранном базисе;

2. сфера *технической реализации* комбинационной схемы КА, аналитически спроектированной в выбранном базисе.

Наиболее употребительными базисами в первой сфере – *построения аналитического описания* комбинационных схем конечных автоматов с использованием аппарата булевой алгебры являются два избыточных базиса:

– булев базис («И – ИЛИ – НЕ»), содержащий три БФ двух переменных – конъюнкцию («И»), дизъюнкцию («ИЛИ») и инверсию («НЕ»)

$$f_1(x_1, x_2) = x_1 \cdot x_2, f_2(x_1, x_2) = x_1 \vee x_2, f_{12,14}(x_i) = \bar{x}_i, (i = 1, 2);$$

– базис Жегалкина, содержащий три БФ двух переменных – конъюнкция («умножение по mod 2»), неравнозначность («сложение по mod 2») и константа «1»

$$f_1(x_1, x_2) = x_1 \cdot x_2 = x_1 \otimes x_2, f_6(x_1, x_2) = x_1 \oplus x_2, f_{15}(x_i) = 1, (i = 1, 2),$$

так, что выполняется равенство $\bar{x}_i = x_i \oplus 1$.

Применительные области этих базисов весьма различны. Так булев базис является аналитической основой всех алгоритмов синтеза комбинационных схем конечных автоматов. Базис Жегалкина в основном используется в задачах аналитического эквивалентирования конечного автомата (нелинейной двоичной динамической системы) линейным аналогом в виде линейной двоичной динамической системы расширенной по сравнению с исходным конечным автоматом памятью (размерностью).

Большим преимуществом избыточного булева базиса является возможность его «погружения» в словесное описание функционирования создаваемого устройства дискретной автоматики или информационной техники. Булев базис делает путь от словесного описания к аналитическому кратчайшим.

Проблема употребительности выбранного базиса во второй сфере – *технической реализации* комбинационной схемы КА, аналитически спроектированной в данном базисе значительно сложнее. Дело в том, что техническая среда, в которой реализуются функциональные элементы в общем представлены: электромеханическими релейно–контактными средствами, средствами дискретной пневмо– и гидро–автоматики, но основными в настоящий момент средствами являются электронные, которые в последние годы выполняются в виде интегральных микросхем (ИМС) различного уровня интеграции. Даже в универсальной технической среде, представленной ИМС, из приведенных в таблице 7.11 реализуются функциональные элементы: «И» (конъюнкция), «ИЛИ» (дизъюнкция), «И–НЕ» (функция Шеффера), «ИЛИ–НЕ» (функция Вебба), «Исключительно ИЛИ» (суммирование по mod 2), «НЕ» (инверсия), «0» (константа 0), «1» (константа 1). Функциональные элементы, реализующие остальные функции двух булевых переменных пока не осуществлены, поэтому формирование базисов, включающих в свой состав эти функции, теряет смысл.

Нетрудно видеть, что избыточные базисы Дж. Буля и И.И. Жегалкина технически реализуемы.

Для целей конструирования алгоритмов формирования аналитического представления произвольной булевой

(переключательной) функции в булевом базисе введем следующие определения.

Определение 7.4. Таблицей истинности булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ K } x_i \text{ K } x_n)$, называется таблица, один столбец которой состоит из 2^n наборов $\{x_1, x_2 \text{ K } x_i \text{ K } x_n\}$ булевых переменных, в которых $x_i \in GF(2) = \{0, 1\} (i = \overline{1, n})$, а второй – состоит из значений функции $y = f(x_1, x_2 \text{ K } x_i \text{ K } x_n)$, где $y \in GF(2) = \{0, 1\}$, на этих наборах. □

Определение 7.5. *Основной конъюнкцией* данного набора переменных $\{x_1, x_2 \text{ K } x_i \text{ K } x_n\}$ называется конъюнкция n булевых переменных, которая на данном наборе принимает *единичное* значение и нулевое на остальных. □

Определение 7.6. *Основной дизъюнкцией* данного набора переменных $\{x_1, x_2 \text{ K } x_i \text{ K } x_n\}$ называется дизъюнкция n булевых переменных, которая на данном наборе принимает *нулевое* значение и единичное на остальных. □

Введенные определения позволяют сформировать аналитическое представление булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ K } x_i \text{ K } x_n)$ в виде *дизъюнкции* основных конъюнкций или в виде *конъюнкции* основных дизъюнкций.

Определение 7.7. Представление булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ K } x_i \text{ K } x_n)$ в виде *дизъюнкции* основных конъюнкций именуется *дизъюнктивной совершенной нормальной формой* (ДСНФ) этой функции. □

Определение 7.8. Представление булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ K } x_i \text{ K } x_n)$ в виде *конъюнкции* основных дизъюнкций именуется *конъюнктивной совершенной нормальной формой* (КСНФ) этой функции. □

Определение 7.9. *Дизъюнктивной* совершенной нормальной формой (ДСНФ) булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ K } x_i \text{ K } x_n)$ называется дизъюнкция основных конъюнкций тех наборов переменных, на которых функция принимает *единичное* значение. □

Определение 7.10. *Конъюнктивной* совершенной нормальной формой (КСНФ) булевой (переключательной) функции n булевых

переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ называется конъюнкция основных дизъюнкций тех наборов переменных, на которых функция принимает нулевое значение. \square

Введенные определения ДСНФ и КСНФ булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ позволяют построить алгоритмы их формирования.

Алгоритм 7.2

формирования ДСНФ аналитического представления булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$

1. Задать булеву (переключательную) функцию n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ с помощью таблицы истинности.

2. Дополнить таблицу истинности столбцом основных конъюнкций всех наборов переменных.

3. Выделить наборы переменных, на которых булева (переключательная) функция n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ принимает единичные значения.

4 Сформировать аналитическое представление булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ в виде ДСНФ, то есть дизъюнкции основных конъюнкций тех наборов переменных, на которых функция принимает единичное значение. \blacksquare

Алгоритм 7.3

формирования КСНФ аналитического представления булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$

1. Задать булеву (переключательную) функцию n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ с помощью таблицы истинности.

2. Дополнить таблицу истинности столбцом основных дизъюнкций всех наборов переменных.

3. Выделить наборы переменных, на которых булева (переключательная) функцию n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ принимает нулевые значения.

4. Сформировать аналитическое представление булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ в виде КСНФ, то есть конъюнкции основных дизъюнкций тех наборов переменных, на которых функция принимает нулевое значение. ■

Проиллюстрируем процедуру формирования аналитического представления БФ в форме ДСНФ и КСНФ на примере функций заданных изначально как таблицей истинности, так и аналитически в виде композиции функций, приведенных в таблице 7.11.

Пример 7.2

Рассмотрим задачу формирования аналитического представления в булевом базисе в форме ДСНФ и КСНФ булевой функции $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n) \Big|_{n=3} = f(x_1, x_2, x_3)$, заданной таблицей истинности (таблица 7.14).

Таблица 7.14

наборы переменных			значения БФ $y = f(x_1, x_2, x_3)$	основные конъюнкции	основные дизъюнкции
x_1	x_2	x_3			
0	0	0	1	$\bar{x}_1 \bar{x}_2 \bar{x}_3$	$x_1 \vee x_2 \vee x_3$
0	0	1	0	$\bar{x}_1 \bar{x}_2 x_3$	$x_1 \vee x_2 \vee \bar{x}_3$
0	1	0	1	$\bar{x}_1 x_2 \bar{x}_3$	$x_1 \vee \bar{x}_2 \vee x_3$
0	1	1	1	$\bar{x}_1 x_2 x_3$	$x_1 \vee \bar{x}_2 \vee \bar{x}_3$
1	0	0	0	$x_1 \bar{x}_2 \bar{x}_3$	$\bar{x}_1 \vee x_2 \vee x_3$
1	0	1	1	$x_1 \bar{x}_2 x_3$	$\bar{x}_1 \vee x_2 \vee \bar{x}_3$
1	1	0	1	$x_1 x_2 \bar{x}_3$	$\bar{x}_1 \vee \bar{x}_2 \vee x_3$
1	1	1	0	$x_1 x_2 x_3$	$\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3$

Решение

1. Следуя п.1 алгоритмов 7.2 и 7.3, фиксируем задание БФ в виде таблицы истинности.

2. Следуя п.2 алгоритмов 7.2 и 7.3, составим основные конъюнкции и основные дизъюнкции на всех наборах переменных, которые разместим в совмещенной таблице 7.14, образованной таблицей истинности БФ $y = f(x_1, x_2, x_3)$ и двумя столбцами основных конъюнкций и дизъюнкций.

3. Следуя п.3 алгоритмов 7.2 и 7.3, сформируем аналитическое представление БФ $y = f(x_1, x_2, x_3)$ в булевом базисе в форме ДСНФ как дизъюнкции основных конъюнкций тех наборов переменных, на которых БФ принимает единичное значение

$$(ДСНФ) \quad y = f(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 x_2 \bar{x}_3 \vee \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 x_3 \vee x_1 x_2 \bar{x}_3,$$

а также в форме КСНФ как конъюнкции основных дизъюнкций тех наборов переменных, на которых БФ принимает нулевое значение

$$(КСНФ) \quad y = f(x_1, x_2, x_3) = (x_1 \vee x_2 \vee \bar{x}_3) (\bar{x}_1 \vee x_2 \vee x_3) (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3). \quad \blacksquare$$

Пример 7.3

Рассмотрим задачу формирования аналитического представления в булевом базисе в форме ДСНФ и КСНФ булевой функции $y = f(x_1, x_2, \dots, x_n) \Big|_{n=3} = f(x_1, x_2, x_3)$, заданной аналитически в виде композиции булевых функций, не входящих в булев базис. В качестве примера рассматривается БФ

$$y = f(x_1, x_2, x_3) = \{(x_1 x_2 \vee x_3) \rightarrow (x_1 \oplus x_2 x_3)\}.$$

Решение

1. Следуя п.1 алгоритмов 7.2 и 7.3, сформируем задание БФ в виде таблицы истинности, вычислив значения термов $(x_1 x_2 \vee x_3) = z_1$ и $(x_1 \oplus x_2 x_3) = z_2$ и определив на полученных наборах переменных (z_1, z_2) функцию двух переменных $f_3(z_1, z_2) = z_1 \rightarrow z_2$ «импликация из z_1 в z_2 », сведя полученные результаты в таблицу 7.15.

Таблица 7.15

наборы переменных			$y = f(x_1, x_2, x_3) = \{(x_1 x_2 \vee x_3) \rightarrow (x_1 \oplus x_2 x_3)\}.$						
x_1	x_2	x_3	$(x_1 x_2 \vee x_3)$	\vee	x_3	\rightarrow	$(x_1 \oplus x_2 x_3)$	\oplus	$x_2 x_3$
0	0	0	0	0	0	1	0	0	0
0	0	1	0	1	1	0	0	0	0
0	1	0	0	0	0	1	0	0	0
0	1	1	0	1	1	1	0	1	1
1	0	0	0	0	0	1	1	1	0
1	0	1	0	1	1	1	1	1	0
1	1	0	1	1	0	1	1	1	0
1	1	1	1	1	1	0	1	0	1

2. Следуя п.2 алгоритмов 7.2 и 7.3, составим основные конъюнкции и основные дизъюнкции на всех наборах переменных, которые изыдем из двух столбцов основных конъюнкций и дизъюнкций совмещенной таблицы 7.14, что позволяет сделать факт совпадения наборов переменных таблиц 7.14 и 7.15.

3. Следуя п.3 алгоритмов 7.2 и 7.3, сформируем аналитическое представление БФ $y = f(x_1, x_2, x_3)$ в булевом базисе в форме ДСНФ как дизъюнкции основных конъюнкций тех наборов переменных, на которых БФ принимает единичное значение (ДСНФ)

$$y = f(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 x_2 \bar{x}_3 \vee \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 \bar{x}_3 \vee x_1 \bar{x}_2 x_3 \vee x_1 x_2 \bar{x}_3,$$

а также в форме КСНФ как конъюнкции основных дизъюнкций тех наборов переменных, на которых БФ принимает нулевое значение

$$(КСНФ) \quad y = f(x_1, x_2, x_3) = (x_1 \vee x_2 \vee \bar{x}_3)(\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3). \quad \blacksquare$$

Сравнивая две формы аналитического представления булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2, \dots, x_n)$ в булевом базисе в форме ДСНФ и КСНФ, следует отметить, что более употребительной оказалась форма ДСНФ. Все алгоритмы преобразования БФ, минимизация аналитического их представления сориентированы на представления БФ в форме ДСНФ. Это связано с тем, что свойства дизъюнкции «склеивание» и «поглощение» являются основой всех алгоритмов минимизации аналитического представления БФ в булевом базисе.

Ниже будем полагать представление БФ в форме ДСНФ основным.

Методы минимизации аналитического представления булевых (переключательных) функций в булевом базисе в форме ДСНФ.

В пособии рассматриваются два наиболее употребляемых метода:

- метод Квайна – Мак–Класки,
- метод карт Карно.

Метод Квайна – Мак–Класки минимизации аналитического представления булевых (переключательных) функций в булевом базисе в форме ДСНФ реализуется в виде процедуры, представленной в виде алгоритма 7.4.

Алгоритм 7.4

минимизации аналитического представления булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ в форме ДСНФ методом Квайна – Мак–Класки

1. Разбить основные конъюнкции, входящие в ДСНФ, на группы по критерию числа *прямых* булевых переменных, входящих в основные конъюнкции, начиная с числа, равного единице.
2. Применить процедуру склеивания $a(x)x_i \vee a(x)\bar{x}_i = a(x)$ к конъюнкциям соседних групп.
3. Переменные, по которым произошло склеивание в сравниваемых основных конъюнкциях, заменить значком $(-)$ – прочерк.
4. Полученные конъюнкции со значком вместо переменной разделить на группы, по правилу, описанному в п.1, произведя далее деления каждой группы на подгруппы по критерию совпадения позиций значка в конъюнкции.
5. Применить процедуру склеивания к конъюнкциям одностипных по позиции значка подгрупп соседних групп.
6. Выполнить действия, описанные в п.3.
7. Процедуру, описанную в п.п.4–6 проводить до тех пор, пока не будет исчерпан банк сравниваемых конъюнкций, содержащих значки вместо склеившихся булевых переменных.
8. Списать с конъюнкций, содержащих значки вместо склеившихся булевых переменных, минимальные конъюнкции, представляющие собой конъюнкции не клеящихся переменных.
9. Построить таблицу реализации исходных основных конъюнкций минимальными конъюнкциями, в которой число столбцов определяется числом основных конъюнкций, входящих в исходную ДСНФ, а число строк – числом минимизированных конъюнкций. Зафиксировать каким–либо значком факт вхождения минимальной конъюнкции в исходные основные конъюнкции.
10. Построить варианты представления исходной булевой функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ в виде дизъюнкции минимальных конъюнкций, которые в совокупности входят в основные конъюнкции исходной ДСНФ, тем самым реализуя их. Выбрать из построенных вариантов вариант с минимальным числом членов, именуемый *минимальная дизъюнктивная нормальная форма* (ДНФ). ■

Пример 7.4

Проиллюстрируем метод Квайна – Мак–Класки минимизации на примере ДСНФ булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ при $n = 4$, имеющей аналитическое представление

$$y = f(x_1, x_2, x_3, x_4) = \bar{x}_1\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1\bar{x}_2x_3\bar{x}_4 \vee \bar{x}_1x_2\bar{x}_3x_4 \vee \bar{x}_1x_2x_3\bar{x}_4 \vee x_1\bar{x}_2\bar{x}_3x_4 \vee x_1\bar{x}_2x_3\bar{x}_4 \vee x_1x_2\bar{x}_3x_4 \vee x_1x_2x_3\bar{x}_4$$

Решение

1. Следуя п.1 алгоритма 7.4, разобьём основные конъюнкции, входящие в ДСНФ, на группы по критерию числа *прямых* булевых переменных, входящих в основные конъюнкции, начиная с числа, равного единице, в результате получим группы, приведенные в таблице 7.16 (второй столбец).

2. Следуя п.п.2 – 8 алгоритма 7.4, получим минимизированные конъюнкции, последовательность и результаты получения которых также приведены в таблице 7.16.

Таблица 7.16

основные конъюнкции	конъюнкции, разбитые на группы	результаты первого склеивания, разбитые на две подгруппы		результаты второго склеивания	минимальные конъюнкции (МК)
$\bar{x}_1\bar{x}_2\bar{x}_3x_4$	$\bar{x}_1\bar{x}_2\bar{x}_3x_4$	$-\bar{x}_2\bar{x}_3x_4$	$\bar{x}_1 - \bar{x}_3x_4$	$--\bar{x}_3x_4$	\bar{x}_3x_4
$\bar{x}_1\bar{x}_2x_3\bar{x}_4$	$\bar{x}_1\bar{x}_2x_3\bar{x}_4$	$-\bar{x}_2x_3\bar{x}_4$	$\bar{x}_1 - x_3\bar{x}_4$	$--x_3\bar{x}_4$	$x_3\bar{x}_4$
$\bar{x}_1x_2\bar{x}_3x_4$					
$\bar{x}_1x_2x_3\bar{x}_4$	$\bar{x}_1x_2\bar{x}_3x_4$	$-x_2\bar{x}_3x_4$	$x_1 - \bar{x}_3x_4$	$x_1x_2 - x_4$	$x_1x_2x_4$
$x_1\bar{x}_2\bar{x}_3x_4$	$\bar{x}_1x_2x_3\bar{x}_4$	$-x_2x_3\bar{x}_4$	$x_1 - x_3\bar{x}_4$	$x_1x_2x_3 -$	$x_1x_2x_3$
$x_1\bar{x}_2x_3\bar{x}_4$	$x_1\bar{x}_2\bar{x}_3x_4$				
$x_1x_2\bar{x}_3x_4$	$x_1\bar{x}_2x_3\bar{x}_4$	$x_1x_2 - x_4$	$x_1x_2x_3 -$		
$x_1x_2x_3\bar{x}_4$					
$x_1x_2x_3x_4$	$x_1x_2\bar{x}_3x_4$				
	$x_1x_2x_3\bar{x}_4$				
	$x_1x_2x_3x_4$				

3. Следуя п.9 алгоритма 7.4, построим таблицу реализаций основных конъюнкций, входящих в исходную ДСНФ, минимальными конъюнкциями (МК), в результате чего получим таблицу 7.17.

Таблица 7.17

МК	ОСНОВНЫЕ КОНЪЮНКЦИИ								
	$\bar{x}_1\bar{x}_2\bar{x}_3x_4$	$\bar{x}_1\bar{x}_2x_3\bar{x}_4$	$\bar{x}_1x_2\bar{x}_3x_4$	$\bar{x}_1x_2x_3\bar{x}_4$	$x_1\bar{x}_2\bar{x}_3x_4$	$x_1\bar{x}_2x_3\bar{x}_4$	$x_1x_2\bar{x}_3x_4$	$x_1x_2x_3\bar{x}_4$	$x_1x_2x_3x_4$
\bar{x}_3x_4	×		×		×		×		
$x_3\bar{x}_4$		×		×		×		×	
$x_1x_2x_4$							×		×
$x_1x_2x_3$								×	×

4. Следуя п.10 алгоритма 7.4, построим два варианта представления исходной булевой функции $y = f(x_1, x_2, x_3, x_4)$ в виде дизъюнкции минимальных конъюнкций, которые в совокупности входят в основные конъюнкции исходной ДСНФ, тем самым реализуя их:

$$1) y = f(x_1, x_2, x_3, x_4) = \bar{x}_3x_4 \vee x_3\bar{x}_4 \vee x_1x_2x_4,$$

$$2) y = f(x_1, x_2, x_3, x_4) = \bar{x}_3x_4 \vee x_3\bar{x}_4 \vee x_1x_2x_3. \quad \blacksquare$$

Метод карт Карно минимизации аналитического представления булевых (переключательных) функций в булевом базисе в форме ДСНФ реализуется в виде процедуры, представленной в виде алгоритма 7.5.

Алгоритм 7.5

минимизации аналитического представления булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ в форме ДСНФ методом карт Карно

1. Разбить булевы переменные на две группы, каждая мощностью \bar{n} , если исходное число n четное ($n = 2\bar{n}$). Если число n нечетное ($n = 2\bar{n} + 1$), то на две группы, одна мощностью \bar{n} , а другая – $(\bar{n} + 1)$.

2. Сформировать таблицу (карту Карно) с числом строк, равным $2^{\bar{n}}$, и числом столбцов также равным $2^{\bar{n}}$ в случае, если исходное число n четное ($n = 2\bar{n}$). Если число n нечетное ($n = 2\bar{n} + 1$), то сформировать таблицу (карту Карно) с числом строк, равным $2^{\bar{n}}$, и числом столбцов также равным $2^{(\bar{n}+1)}$.

3. Закодировать строки карты Карно, начиная с верхней строки, соседними \bar{n} – разрядными двоичными кодами по схеме Грея, чем

обеспечивается кодовое расстояние между кодами соседних строк, равное единице так, что коды соседних строк отличаются только в одном разряде. Коды разместить в непосредственной близости от обоих концов строк.

4. Закодировать столбцы карты Карно, начиная с левого столбца, соседними $\bar{n}, (\bar{n} + 1)$ – разрядными двоичными кодами по схеме Грея, чем обеспечивается кодовое расстояние между кодами соседних столбцов, равное единице так, что коды соседних столбцов отличаются только в одном разряде. Коды разместить в непосредственной близости от обоих концов столбцов.

5. Сформировать на кодах строк и столбцов согласованные с ними основные конъюнкции, разместив первые рядом слева и справа с кодами строк, а вторые – над кодами над верхним краем карты и под кодами под нижним краем карты.

6. Нанести единичные значения минимизируемой функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$, для основных конъюнкции исходных наборов, скомпонованных объединением основных конъюнкций строк и столбцов карты Карно.

7. Полностью покрыть массив единичных значений минимизируемой БФ прямоугольными контурами, так чтобы число покрытых единичных значений представляло собой степень числа два: $2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8 \dots 2^n$, при этом допускается перекрытие покрывающих прямоугольных контуров.

8. Выбрать варианты покрытия, которые составлены минимальным числом контуров максимального размера.

9. Оценить путем обхода контуров склеиваемые булевы переменные с учетом того, что на контуре, покрывающем одну единицу склеиваемых переменных *нет*, на контуре покрывающем две единицы число склеиваемых переменных равно *единице*; на контуре, покрывающем четыре единицы склеиваемых переменных *две*; в общем случае – на контуре, покрывающем 2^m единицы склеиваемых переменных m .

10. С учетом положений п.9 списать с карты Карно варианты минимизированного представления булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ в форме дизъюнкции минимальных конъюнкций

Пример 7.5

Проиллюстрируем метод карт Карно минимизации на примере ДСНФ булевой функции n булевых переменных $y = f(x_1, x_2 \text{ К } x_i \text{ К } x_n)$ при $n = 4$, имеющей аналитическое представление

$$y = f(x_1, x_2, x_3, x_4) = \bar{x}_1\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1\bar{x}_2x_3\bar{x}_4 \vee \bar{x}_1x_2\bar{x}_3x_4 \vee \bar{x}_1x_2x_3\bar{x}_4 \vee x_1\bar{x}_2\bar{x}_3x_4 \vee x_1\bar{x}_2x_3\bar{x}_4 \vee x_1x_2\bar{x}_3x_4 \vee x_1x_2x_3\bar{x}_4$$

Решение

1. Выполним п.п.1– 6 алгоритма 7.5 в результате построим 4×4 карту Карно с закодированными строками и столбцами кодами по схеме Грея, с соответствующими этим кодам основные конъюнкции и нанесенными на нее единичными значениями минимизируемой БФ $y = f(x_1, x_2, x_3, x_4)$, приведенную в форме таблицы 7.17.

2. Следуя п.7 алгоритма 7.5, полностью покроем массив единичных значений минимизируемой БФ в таблице 7.17 прямоугольными контурами.

3. Следуя п.п.8 – 10 алгоритма 7.5, спишем с карты Карно (таблица 7.17) варианты минимизированного представления булевой (переключательной)

Таблица 7.17

	$\bar{x}_3\bar{x}_4$ 00	\bar{x}_3x_4 01	x_3x_4 11	$x_3\bar{x}_4$ 10	
$\bar{x}_1\bar{x}_2$; 00		1		1	00; $\bar{x}_1\bar{x}_2$
\bar{x}_1x_2 ; 01		1		1	01; \bar{x}_1x_2
x_1x_2 ; 11		1	1	1	11; x_1x_2
$x_1\bar{x}_2$; 10		1		1	10; $x_1\bar{x}_2$
	00 $\bar{x}_3\bar{x}_4$	01 \bar{x}_3x_4	11 x_3x_4	10 $x_3\bar{x}_4$	

функции $n = 4$ булевых переменных $y = f(x_1, x_2, x_3, x_4)$ в форме дизъюнкции минимальных конъюнкций

1) $y = f(x_1, x_2, x_3, x_4) = \bar{x}_3x_4 \vee x_3\bar{x}_4 \vee x_1x_2x_4,$

2) $y = f(x_1, x_2, x_3, x_4) = \bar{x}_3x_4 \vee x_3\bar{x}_4 \vee x_1x_2x_3.$ ■

Заканчивая рассмотрение методов минимизации аналитического представления булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \dots x_i \dots x_n)$ в форме ДСНФ, следует сказать, что они характеризуются различной применительной практикой. Так программные реализации процедур минимизации БФ в форме ДСНФ в основном строятся на базе метода метод Квайна – Мак–Класки. Метод карт Карно является ручным и используется при минимизации аналитического представления булевой (переключательной) функции n булевых переменных $y = f(x_1, x_2 \dots x_i \dots x_n)$ в форме ДСНФ при n не более шести.

Булевы производные булевых функций, области применения

Булевы (селлерсовские) производные булевых функций ввел в конце шестидесятых годов прошлого столетия Ф. Селлерс. Он использовал булевы производные (математически точнее *разности*) булевых функций для обнаружения ошибок в функционировании дискретных устройств, аналитическое представление которых задается с помощью аппарата БФ. Со временем Ф. Селлерс переносит предложенный аппарат на задачу обнаружения ошибок в работе ЭВМ. Однако аппарат селлерсовского дифференцирования, за некоторым исключением остается за пределами массовой технической литературы, проблемно ориентированной на разработки устройств дискретной автоматики и технической информатики (УДАТИ).

Задача параграфа – привлечь внимание студентов как будущих разработчиков УДАТИ к возможностям аппарата селлерсовского дифференцирования и предложить инструментарий для исследования аналитических их описаний в классе НДДС (автоматных) представлений. С этой целью сформулируем основные положения аппарата селлерсовского дифференцирования.

Определение 7.11. Частной производной Селлерса (ЧПС) порядка булевой функции $f(x) = f(x_1, x_2, \dots, x_i, \dots, x_n)$ n булевых переменных по булевой переменной x_i называется булева функция $\frac{\partial f(x)}{\partial x_i}$, задаваемая выражением

$$\frac{\partial f(x)}{\partial x_i} = f(x_1, x_2, \dots, x_i, \dots, x_n) \oplus f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n). \quad \square \quad (7.18)$$

Вычисление частной производной Селлерса от БФ

$$f(x) = f(x_1, x_2, \dots, x_i, \dots, x_n)$$

по переменной x_i может быть произведено несколькими способами.

Первый способ основан на определении ЧПС (7.18).

Второй способ использует метод карт Карно, в соответствии с которым строятся две карты Карно для булевых функций $f(x_1, x_2, \dots, x_i, \dots, x_n)$ и $f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n)$, которые суммируются по модулю два, что приводит к карте Карно для частной производной. Этот способ позволяет получать минимальное представление ЧПС.

Третий способ использует разложение К. Шеннона, которое для (7.18) позволяет записать

$$\frac{\partial f(x)}{\partial x_i} = f(x_1, x_2, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) \oplus f(x_1, x_2, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n). \quad (7.19)$$

Четвертый способ, в развитие третьего способа, использует представление БФ $f(x_1, x_2, \dots, x_i, \dots, x_n)$ с помощью таблицы истинности, наборы переменных в которой представлены в форме, имеющей x_i в качестве переменной младшего разряда набора. В этом случае смена значения с x_i на \bar{x}_i , приводящая к смене значения БФ, свидетельствует о единичном значении ЧПС на этом наборе, а отсутствие смены значения БФ – о нулевом значении ЧПС. Следует заметить, что последний способ позволяет оценивать значимость переменной x_i в БФ, определяемую весом ЧПС на всех наборах переменных.

Для вычисления частных производных Селлера от БФ полезно использовать их свойства, которые могут быть установлены непосредственно из определения.

Свойство 7.1. Булевы (селлерсовские) производные от БФ на соседних наборах по булевой переменной, по которой берется производная, равны. \square

Свойство 7.7. Равенство нулю частной селлерсовской производной порядка $k > 1$ произвольной БФ по одной и той же переменной x_i

$$\left. \frac{\partial^k f(x)}{\partial x_i^k} \right|_{k>1} = 0. \quad \square (7.20)$$

Доказательство. Свойство является следствием определения 7.11, примененного к булевой функции типа ЧПС первого порядка от исходной БФ по той же переменной. \blacksquare

Свойство 7.3. (Инвариантность ЧПС относительно инверсии)

$$\frac{\partial f}{\partial x_i} = \frac{\partial \bar{f}}{\partial x_i} = \frac{\partial f}{\partial \bar{x}_i} = \frac{\partial \bar{f}}{\partial \bar{x}_i}. \quad \square (7.21)$$

Свойство 7.4. (Правило дифференцирования констант)

$$\frac{\partial I}{\partial x_i} = \frac{\partial \theta}{\partial x_i} = 0. \quad \square (7.22)$$

Свойство 7.5. («Тривиальные свойства» дифференцирования)

$$\frac{\partial x_i}{\partial x_i} = \frac{\partial x_i}{\partial \bar{x}_i} = \frac{\partial \bar{x}_i}{\partial x_i} = \frac{\partial \bar{x}_i}{\partial \bar{x}_i} = 1. \quad \square (7.23)$$

Свойство 7.6. Если БФ $f(x_1, x_2, \dots, x_i, \dots, x_n)$ представима в форме конъюнкции функций, одна из которых не зависит от x_i :

$$\begin{aligned} f(x_1, x_2, \dots, x_i, \dots, x_n) &= \\ &= f_1(x_j, \quad j = \overline{1, n}, j \neq i) \cdot f_2(x_1, x_2, \dots, x_i, \dots, x_n) = f_1(x) \cdot f_2(x), \end{aligned}$$

то

$$\frac{\partial f(x)}{\partial x_i} = \frac{\partial \{f_1(x) f_2(x)\}}{\partial x_i} = f_1(x) \frac{\partial f_2(x)}{\partial x_i}. \quad \square (7.24)$$

Свойство 7.7. Если БФ $f(x)$ представима в виде конъюнкции БФ $\varphi_1(x)$ и $\varphi_2(x)$:

$$f(x) = \varphi_1(x) \cdot \varphi_2(x),$$

то

$$\frac{\partial f(x)}{\partial x_i} = \frac{\partial \varphi_1(x)}{\partial x_i} \varphi_2(x) \oplus \varphi_1(x) \frac{\partial \varphi_2(x)}{\partial x_i} \oplus \frac{\partial \varphi_1(x)}{\partial x_i} \cdot \frac{\partial \varphi_2(x)}{\partial x_i}. \quad \square (7.25)$$

Свойство 7.8.

Если БФ $f(x)$ представима в виде дизъюнкции БФ $\varphi_1(x)$ и $\varphi_2(x)$:

$$f(x) = \varphi_1(x) \vee \varphi_2(x), \quad (7.26)$$

то

$$\frac{\partial f(x)}{\partial x_i} = \frac{\partial \varphi_1(x)}{\partial x_i} \bar{\varphi}_2(x) \oplus \bar{\varphi}_1(x) \frac{\partial \varphi_2(x)}{\partial x_i} \oplus \frac{\partial \varphi_1(x)}{\partial x_i} \cdot \frac{\partial \varphi_2(x)}{\partial x_i}. \quad \square (7.27)$$

Свойство 7.9. Если БФ $f(x)$ представима в виде суммы по модулю два БФ $\varphi_1(x)$ и $\varphi_2(x)$:

$$f(x) = \varphi_1(x) \oplus \varphi_2(x),$$

то

$$\frac{\partial f(x)}{\partial x_i} = \frac{\partial \varphi_1(x)}{\partial x_i} \oplus \frac{\partial \varphi_2(x)}{\partial x_i}. \quad \square (7.28)$$

Свойство 7.10. Если БФ $f(x)$ представима в форме конъюнкции ее переменных:

$$f(x) = \bigg\&_{j=1}^n x_j, \quad (7.29)$$

то

$$\frac{\partial f(x)}{\partial x_i} = \bigotimes_{\substack{j=1 \\ j \neq i}}^n x_j. \quad \square(7.30)$$

Свойство 7.11. Если БФ $f(x)$ представима в форме дизъюнкции ее переменных:

$$f(x) = \bigvee_{j=1}^n x_j, \quad (7.31)$$

то

$$\frac{\partial f(x)}{\partial x_i} = \bigotimes_{\substack{j=1 \\ j \neq i}}^n \bar{x}_j. \quad \square(7.32)$$

Свойство 7.17. Если $f(x)$ является сложной БФ, задаваемой в форме

$$f(x) = f(x, \varphi(x)), \quad (7.33)$$

то

$$\begin{aligned} \frac{\partial f(x)}{\partial x_i} &= f(x_1, x_2, \dots, x_i, \dots, x_n, \varphi(x_1, x_2, \dots, x_i, \dots, x_n)) \oplus \\ &\oplus f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n, \varphi(x_1, x_2, \dots, \bar{x}_i, \dots, x_n)). \end{aligned} \quad \square(7.34)$$

Рассмотрим далее понятие частных смешанных производных Селлера высокого порядка БФ и их свойства.

Определение 7.13. Двукратной смешанной производной Селлера булевой функции $f(x) = f(x_1, x_2, \dots, x_i, \dots, x_j, \dots, x_n)$ от n булевых переменных называется БФ, задаваемая выражением

$$\frac{\partial^2 f(x)}{\partial x_j \partial x_i} = \frac{\partial^2 f(x)}{\partial x_j \partial x_i} = \frac{\partial}{\partial x_i} \left(\frac{\partial f(x)}{\partial x_j} \right) = \frac{\partial}{\partial x_j} \left(\frac{\partial f(x)}{\partial x_i} \right). \quad \square(7.35)$$

Определение 7.14. m -кратной смешанной производной Селлера по m переменным $x_{i_1}, x_{i_2}, \dots, x_{i_m}$ булевой функции

$f(x) = f(x_1, \dots, x_{i_1}, \dots, x_{i_2}, \dots, x_{i_m}, \dots, x_n)$ называется БФ, задаваемая выражением

$$\begin{aligned} \frac{\partial^m f(x)}{\partial x_{i_1} \partial x_{i_2} \dots \partial x_{i_m}} &= \frac{\partial}{\partial x_{i_1}} \left(\frac{\partial}{\partial x_{i_2}} \left(\Lambda \left(\frac{\partial f(x)}{\partial x_{i_m}} \right) \right) \right) = \\ &= \frac{\partial}{\partial x_{i_m}} \left(\frac{\partial}{\partial x_{i_{(m-1)}}} \left(\Lambda \left(\frac{\partial f(x)}{\partial x_{i_1}} \right) \right) \right). \end{aligned} \quad \square(7.36)$$

Определение 7.15. Производная m -го порядка $\frac{\partial^m f(x)}{\partial (x_{i_1} x_{i_2} \dots x_{i_m})}$ от булевой функции $f(x) = f(x_1, x_2, \dots, x_n)$ по кортежу переменных

$(x_{i_1}, x_{i_2}, \dots, x_{i_m})$ определяет условия, при которых функция $f(x)$ изменяет свое значение при **одновременном** изменении значений переменных из набора булевых переменных. \square

Основными сферами использования аппарата булевых (селлерсовских) производных булевых функций являются:

– обнаружение избыточных переменных в составе набора булевых переменных, которые использованы при булевом описании комбинационной схемы конечного автомата (нелинейной двоичной динамической системы);

– ранжирование булевых переменных по степени их востребованности в процессе функционирования НДДС (КА).

В связи со сказанным введем в рассмотрение следующие определения.

Определение 7.15. Булева переменная x_i булевой функции $f(x) = f(x_1, x_2, \dots, x_i, \dots, x_n)$ n булевых переменных считается *избыточной* (не востребованной), если булева производная по булевой переменной x_i $\frac{\partial f(x)}{\partial x_i} = f(x_1, x_2, \dots, x_i, \dots, x_n) \oplus f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n)$ на всех

наборах переменных принимает нулевое значение. \square

Определение 7.16. Булева переменная x_i булевой функции $f(x) = f(x_1, x_2, \dots, x_i, \dots, x_n)$ n булевых переменных считается *наиболее востребованной*, если булева производная по булевой переменной x_i $\frac{\partial f(x)}{\partial x_i} = f(x_1, x_2, \dots, x_i, \dots, x_n) \oplus f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n)$ принимает *единичное*

значение на наибольшем числе наборов переменных по сравнению с другими булевыми переменными из этих наборов. \square

Пример 7.6

Проиллюстрируем процедуру вычисления булевой (селлерсовской) производной булевой функции $f(x) = f(x_1, x_2, \dots, x_i, \dots, x_n)$ n булевых переменных по всем булевым переменным x_i ($i = \overline{1, n}$) первым способом, опирающимся на определение булевой производной $\frac{\partial f(x)}{\partial x_i} = f(x_1, x_2, \dots, x_i, \dots, x_n) \oplus f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n)$, и свойство 7.1

булевых производных, на всех наборах переменных на примере БФ трех переменных, заданной таблицей истинности, представленной в таблице 7.4.

Решение

Результаты вычисления булевых производных по всем трем переменным БФ представлены в форме таблицы 7.18.

Таблица 7.18

наборы переменных			значения БФ $y = f(x_1, x_2, x_3)$	булевы (селлерсовские) производные БФ $y = f(x_1, x_2, x_3)$		
x_1	x_2	x_3		$\partial f(x)/\partial x_3$	$\partial f(x)/\partial x_2$	$\partial f(x)/\partial x_1$
0	0	0	1	1	0	1
0	0	1	0	1	1	1
0	1	0	1	0	0	0
0	1	1	1	0	1	1
1	0	0	0	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	0
1	1	1	0	1	1	1

Вывод: Все булевы переменные БФ одинаково востребованы.

7.3. Триггеры: функции перехода и функции возбуждения входов триггеров

Предыдущий параграф был посвящен проблемам построения аналитического описания комбинационной схемы конечного автомата (НДДС), настоящий параграф посвящен проблемам описания функционирования второго базового компонента КА – блока памяти, который над двоичным полем строится на *триггерах*. Начнем изучение этой проблемы с определения.

Определение 7.17. *Триггером* называется конечный автомат *первого порядка*, абстрактным аналогом которого является абстрактный автомат с логикой Мура, аналитически описываемый двумя функциями:

– функцией перехода

$$\lambda : x(k+1) = \lambda\{x(k), v(k), c(k)\}; \quad (7.37)$$

– функцией возбуждения информационного (ых) входа триггера

$$\mu : v(k) = \mu\{x(k), x(k+1)\}. \quad (7.38)$$

В выражениях (7.37), (7.38) $x(k), x(k+1)$ – соответственно исходное состояние триггера и состояние перехода, $v(k), c(k)$ – соответственно сигнал на информационном(ых) входе и сигнал на синхровходе триггера. ■

Триггеры различаются по:

– по виду функций $\lambda\{x(k), v(k), c(k)\}$ перехода и $\mu\{x(k), x(k+1)\}$ возбуждения информационного (ых) входа триггера: если эти функции линейные, то триггеры именуется *линейными*, если эти функции нелинейные, то триггеры именуется *нелинейными*;

– по присутствию в функционировании триггера синхросигнала: если $c(k) \neq 0$, то триггер именуется *синхронным*, если $c(k) = 0$, то триггер именуется *асинхронным*;

– по числу информационных входов: если информационный вход один, то триггер именуется *одновходовым*, если информационных входов два, то триггер именуется *двухходовым*.

В схемотехнике современных устройств дискретной автоматики и техники преобразования кодов и кодовых последовательностей выделяют четыре базовых типа триггеров: *D* – линейный синхронный одновходовый триггер, именуемый *D* – триггером; *T* – линейный асинхронный одновходовый триггер, именуемый *T* – триггером; *JK* – нелинейный синхронный двухходовый триггер, именуемый *JK* – триггером; *RS* – нелинейный асинхронный двухходовый триггер, именуемый *RS* – триггером.

В таблицах 7.19а и 7.19б приведены основные пользовательские характеристики перечисленных базовых триггеров в виде графического изображения, таблиц перехода, аналитического представления функций перехода и возбуждения информационных входов триггеров.

Таблица 7.19а

Линейные триггеры современной схемотехники						
<i>D</i> – триггер				<i>T</i> – триггер		
Таблицы истинности функций $\lambda : x(k+1) = \lambda\{x(k), v(k), c(k)\}$						
<i>C</i> (<i>k</i>)	$v_D(k)$	$x(k)$	$x(k+1)$	$v_T(k)$	$x(k)$	$x(k+1)$
0	0	0	0	0	0	0
		1	1			
	1	0	0		1	1
		1	1			
1	0	0	0	1	0	1
		1	0			
	1	0	1		1	0
		1	1			
Аналитическое представление функций $\lambda : x(k+1) = \lambda\{x(k), v(k), c(k)\}$						
$\lambda : \begin{aligned} x(k+1) &= \bar{c}(k)x(k) \vee c(k)v_D(k) = \\ &= v_D(k) \Big _{c(k)=1} \end{aligned}$				$\lambda : x(k+1) = x(k) \oplus v_T(k)$		
Аналитическое представление функций $\mu : v(k) = \mu\{x(k), x(k+1)\}$						
$\mu : v_D(k) = x(k+1) \Big _{c(k)=1}$				$\mu : v_T(k) = x(k) \oplus x(k+1)$		

Таблица 7.19б

Нелинейные триггеры современной схемотехники												
<i>RS</i> – триггер				<i>JK</i> – триггер								
Таблицы истинности функций $\lambda : x(k+1) = \lambda\{x(k), v(k), c(k)\}$												
$v_S(k)$	$v_R(k)$	$x(k)$	$x(k+1)$	$C(k)$	$v_J(k)$	$v_K(k)$	$x(k)$	$x(k+1)$				
0	0	0	0	0	0	0	0	0				
		1	1				1	1				
0	1	0	0		0	1	0	0	0			
		1	0					1	1			
1	0	0	1		1	0	0	0	0			
		1	1					1	1			
1	1	0	×		1	1	1	0	0			
		1	×					1	1			
Комбинация входных сигналов, выделенная заливкой является запрещенной, средствами схемотехники она должна не допускаться.				1	0	0	0	0				
							1	1				
					0	1	0	0	0	0	0	0
							1	0	1	0		
					1	0	0	1	0	0	0	1
							1	1	1	1		
Аналитическое представление функций $\lambda : x(k+1) = \lambda\{x(k), v(k), c(k)\}$				$\lambda : x(k+1) = \bar{v}_K(k)x(k) \vee v_J(k)\bar{x}(k) \Big _{c(k)=1}$								
				$\lambda : x(k+1) = v_S(k)\bar{v}_R(k) \vee \bar{v}_R(k)x(k)$								
Таблицы истинности функций $\mu : v(k) = \mu\{x(k), x(k+1)\}$												
$x(k)$	$x(k+1)$	$v_S(k)$	$v_R(k)$	$x(k)$	$x(k+1)$	$v_J(k)$	$v_K(k)$	$c(k)$				
0	0	0	$\nabla\{0;1\}$	0	0	0	$\nabla\{0;1\}$	1				
0	1	1	0	0	1	1	$\nabla\{0;1\}$					
1	0	0	1	1	0	$\nabla\{0;1\}$	1					
1	1	$\nabla\{0;1\}$	0	1	1	$\nabla\{0;1\}$	0					
Аналитическое представление функций $\mu : v(k) = \mu\{x(k), x(k+1)\}$ при $\nabla\{0;1\} = 0$												
$\mu : v_S(k) = x(k+1)\bar{x}(k); v_R(k) = \bar{x}(k+1)x(k)$				$\mu : v_J(k) = x(k+1)\bar{x}(k); v_K(k) = x(k+1)\bar{x}(k); c(k) = 1$								

Анализ таблицы 7.19а показывает, что *D*– и *T*–триггеры действительно линейные, так как их функции перехода обнаруживают, что состояние перехода $x(k+1)$ линейно связано с исходным состоянием $x(k)$ и с сигналом на информационном входе этих

триггеров. Более того, обнаруживается, что T – триггер, представляет собой D – триггер, охваченный единичной обратной связью так, что если D – триггер описывается передаточной функцией $\Phi_D(d) = d$, то T – триггер – $\Phi_T(d) = d/(1 + d)$.

Анализ таблицы 7.19б показывает, что RS – и JK – триггеры действительно нелинейные, так как их функции перехода обнаруживают, что состояние перехода $x(k+1)$ нелинейно связано с исходным состоянием $x(k)$ и с сигналами на информационных входах этих триггеров.

Анализ таблиц 7.19а и 7.19б показывает, что все асинхронные триггеры (T – и RS – типов) могут быть поставлены в режим синхронных, если информационные входы этих триггеров возбуждать сигналами вида $\{c(k)v_T(k)\}$ и $\{c(k)v_S(k)\}, \{c(k)v_R(k)\}$, формируемых с помощью двухвходовых конъюнкторов.

Анализ таблиц 7.19а и 7.19б обнаруживает, что путем наложения таблицы истинности функции перехода одного триггера на таблицу истинности функции перехода другого триггера можно выявить возможность реализации одного триггера средствами другого. Таких «реализационных пар» достаточно, приведем примеры некоторых из них.

Так D – триггер может быть реализован средствами:

T – триггера, охваченного с помощью сумматора по mod 2 единичной обратной связью, то есть с помощью сигнала $\{v_T(k) = v_D(k) \oplus x(k)\}$;

RS – триггера, который с помощью сигналов $\{v_S(k) = v_D(k); v_R(k) = \bar{v}_D(k)\}$;

JK – триггера с помощью сигналов $\{c(k) = c(k); v_J(k) = v_D(k); v_K(k) = \bar{v}_D(k)\}$.

В свою очередь T – триггер может быть реализован средствами:

D – триггера, охваченного с помощью сумматора по mod 2 единичной обратной связью, то есть с помощью сигнала $\{v_D(k) = v_T(k) \oplus x(k)\}$;

D – триггера с помощью сигналов $\{c(k) = v_T(k); v_D(k) = \bar{x}(k)\}$;

RS – триггера, который с помощью сигналов $\{v_S(k) = v_D(k); v_R(k) = \bar{v}_D(k)\}$ ставится в режим D – триггера, с последующим охватом с помощью сумматора по mod 2 единичной обратной связью, то есть с помощью сигналов $\{v_S(k) = v_T(k) \oplus x(k); v_R(k) = \overline{(v_T(k) \oplus x(k))}\}$;

JK – триггера с помощью сигналов $\{c(k) = v_T(k); v_J(k) = v_K(k) = 1\}$.

Примечание 7.5. Нетрудно видеть, что устройства дискретной автоматики и техники преобразования кодов и кодовых последовательностей, спроектированные на основе модельных представлений в форме линейных двоичных динамических систем, в силу приведенных реализационных представлений могут быть построены с использованием как линейных, так и нелинейных триггеров.

7.4. Реализация рекуррентных устройств помехозащитного кодирования и декодирования в логике произвольных триггеров

Задача, вынесенная в заголовок параграфа, может быть решена двумя способами.

Первый способ опирается на алгоритмы 6.11 и 6.12 синтеза соответственно ЛДДС рекуррентного помехозащитного кодирования и ЛДДС рекуррентного помехозащитного декодирования, а также примечание 7.5. При этом предпочтение следует отдать использованию универсального JK – триггера потому, что реализация синхронного D – триггера его средствами требует минимального числа дополнительных схмотехнических компонентов в виде одного инвертора. Таким образом, реализация рекуррентного устройства помехозащитного кодирования в логике произвольных триггеров может быть осуществлено с помощью алгоритма 7.6.

Алгоритм 7.6

синтеза устройства рекуррентного помехозащитного кодирования в логике произвольных триггеров

1. Выполнить п.п. 0 – 11 алгоритма 6.11;
2. Выбрать тип триггера из состава базовых типов триггеров: D – линейный синхронный, T – линейный асинхронный триггер, RS – нелинейный асинхронный триггер, JK – нелинейный триггер, используемый в синхронном и асинхронном режимах;
3. Если выбранный триггер D – типа то, перейти к п.4, если нет, то выбранный триггер в соответствии с рекомендациями параграфа 7.3 поставить в режим D – триггера;
4. Построить схмотехническую реализацию устройства рекуррентного кодирования в логике выбранного триггера. ■

Алгоритм 7.7

синтеза устройства рекуррентного помехозащитного декодирования в логике произвольных триггеров

1. Выполнить п.п. 1 – 6 алгоритма 6.12;
2. Выбрать тип триггера из состава базовых типов триггеров: D – линейный синхронный, T – линейный асинхронный триггер, RS – нелинейный асинхронный триггер, JK – нелинейный триггер, используемый в синхронном и асинхронном режимах;
3. Если выбранный триггер D – типа то, перейти к п.4, если нет, то выбранный триггер в соответствии с рекомендациями параграфа 7.3 поставить в режим D – триггера;
4. Построить схемотехническую реализацию устройства рекуррентного декодирования в логике выбранного триггера. ■

Заканчивая рассмотрение решения задачи реализации рекуррентных устройств помехозащитного кодирования и декодирования в логике произвольных триггеров *первым способом*, следует заметить, что первый способ не приводит к необходимости агрегирования алгоритмов синтеза ЛДДС с алгоритмом автоматного синтеза дискретных устройств преобразования кодов и кодовых последовательностей.

Второй способ решения задачи построения рекуррентных устройств помехозащитного кодирования и декодирования в логике произвольных триггеров реализуется в виде двухфазной процедуры. В *первой фазе* осуществляется синтез линейных рекуррентного кодирующего и декодирующего устройств с помощью алгоритмов 6.11 и 6.17. Во *второй фазе*, когда получено векторно-матричное представление функций перехода и выхода устройств, осуществляется «погружение» этих устройств в автоматную среду, свободную от ограничения на тип используемой триггерной логики. Таким образом, решение задачи достигается агрегированием алгоритмов линейного и автоматного синтеза ДДС помехозащитного кодирования и декодирования.

Содержательной базой такого агрегирования является то обстоятельство, что кодирование алфавитов *входа, состояния и выхода*, предусмотренное п.3 алгоритма 7.1 автоматного синтеза устройства преобразования кодов, уже произведено при построении линейных векторно-матричных представлений ЛДДС КУ (6.92) с матрицами $\{B_{ку}, A, N\}$ и (6.93) с матрицами $\{\bar{A}, C\}$ в соответствии с п.8 и п.9 алгоритма синтеза ЛДДС КУ 6.11. Аналогичное действие осуществлено при построении линейных векторно-матричных представлений ЛДДС ДКУ (6.78) с матрицами $\{B_{оку}, A\}$ в соответствии с п.3 алгоритма 6.17.

Если эти векторно-матричные соотношения использовать для формирования таблиц истинности функций перехода и выхода КУ и ДКУ, то конструирование устройств рекуррентного кодирования и декодирования можно продолжить в рамках алгоритма автоматного синтеза 7.1, начиная с п.4.

Следует заметить, что коммутация структуры ЛДДС КУ в случае автоматного представления может заменяться использованием двухразрядного кода внешнего входа КУ $U = [u, u_y]$, в котором старший разряд u является *информационным*, значения которого определяются значениями $a(k)$ – информационной помехонезащищенной кодовой последовательности. Младший разряд u_y – *управляющий*, принимающий значение «0» в течение первых k тактов работы КУ, и значение «1» – в течение последних m тактов. Таким образом за цикл формирования ПЗК у за n –тактов двумерный вектор внешнего входа КУ $U = [u, u_y]$ получает реализации: $[0\ 0]$, $[1\ 0]$, $[0\ 1]$, комбинация $[1\ 1]$ является не реализуемой.

В результате агрегирования алгоритмов 6.11 синтеза ЛДДС КУ и алгоритма 7.1 автоматного синтеза устройства преобразования кодов построен

Алгоритм 7.8

синтеза устройства рекуррентного помехозащитного кодирования в логике произвольных триггеров

1. Выполнить п.п.0–9 алгоритма 6.11;
2. Сформировать код входного алфавита КА (7.5) $U = [u, u_y]$;
3. Выполнить п.3 алгоритма 7.1 и получить таблицу реализации функции перехода λ вида $\lambda : x(k) \times U(k) \Rightarrow x(k+1)$ НДДС–КУ (7.5) для наборов $U = [0M]$; $U = [1M]$ с использованием (6.92), для набора $U = [0M]$ – с помощью (6.93);
4. Выполнить п.п. 4–9 алгоритма 7.1. ■

В результате агрегирования алгоритмов 6.12 синтеза ЛДДС ДКУ и алгоритма 7.1 автоматного синтеза устройства преобразования кодов построен

Алгоритм 7.9

синтеза устройства рекуррентного помехозащитного декодирования в логике произвольных триггеров

1. Выполнить п.п.1–4 алгоритма 6.12;

2. Выполнить п.3 алгоритма 7.1 и получить таблицу реализации функции перехода λ вида $\lambda : x(k) \times u(k) \Rightarrow x(k+1)$ НДДС–ДКУ (7.5) с помощью (6.78);

3. Выполнить п.п. 4–9 алгоритма 7.1. ■

Пример 7.7

Просинтезировать рекуррентное кодирующее устройство, формирующее помехозащищенный код (7,4) с образующим многочленом $g(x) = x^3 + x + 1$ в форме НДДС в логике произвольных триггеров.

Решение

Решение поставленной задачи осуществляем с использованием алгоритма 7.8, в соответствии с которым:

1. Выполним п.1 алгоритма 7.8, в результате чего для матричных компонентов $(A, B_{кy}, N)$ и (\bar{A}, C) устройства рекуррентного кодирования, задаваемого парой векторно-матричных описаний (6.92) и (6.93), получим:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}; B_{кy} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}; N = [1]; \bar{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}; C = [1 \ 0 \ 0].$$

2. Выполним п.2 алгоритма 7.8, в результате чего в рассмотрение вводится код входного алфавита КА (7.5) $U = [u, u_y]$.

3. Выполним п.3 алгоритма 7.8, в результате чего получим численные реализации функции перехода λ вида $\lambda : x(k) \times U(k) \Rightarrow x(k+1)$ НДДС–КУ (7.5) для наборов $U(k) = [u(k) \mathbf{M}_y(k)]$: $U = [0\mathbf{M}]$; $U = [1\mathbf{M}]$ с использованием (6.92), для набора $U = [0\mathbf{M}]$ – с помощью (6.93), сведенные в таблицу 7.20

$$\lambda : x(k+1) = Ax(k) + B_{кy}u(k); x(k+1) = \bar{A}x(k) \quad \text{Таблица 7.20}$$

$x^T(k)$	000	001	010	011	100	101	110	111	
$U(k) = [0\mathbf{M}]$	000	010	100	110	011	001	111	101	$x^T(k+1)$
$U(k) = [1\mathbf{M}]$	011	001	111	101	000	010	100	110	
$U(k) = [0\mathbf{M}]$	000	010	100	110	000	010	100	110	

4. Выполним п.п. 4–9 алгоритма 7.1., в результате чего:
– сформируем совмещенную таблицу значений функций возбуждения входов триггеров на основе их аналитических

представлений, приведенных в таблицах 7.19а и 7.19б, и значений выхода кодирующего устройства, представляющего собой помехозащищенный код; результаты выполнения этого пункта приведены в таблице 7.21;

Таблица 7.21

x^T [k]	x^T [k +1]	$U =$ u_{y^y}	$y[k]$	Возбуждаемые входы триггеров											
				D			T			RS			JK		
000	000	00	0												
	011	10	1		D_2	D_3		T_2	T_3		S_2	S_3		J_2	J_3
	000	01	0												
001	010	00	0		D_2			T_2	T_3		S_2	R_3		J_2	K_3
	001	10	1			D_3									
	010	01	0		D_2			T_2	T_3		S_2	R_3		J_2	K_3
010	100	00	0	D_1			T_1	T_2		S_1	R_2		J_1	K_2	
	111	10	1	D_1	D_2	D_3	T_1		T_3	S_1		S_3	J_1		J_3
	100	01	0	D_1			T_1	T_2		S_1	R_2		J_1	K_2	
011	110	00	0	D_1	D_2		T_1		T_3	S_1		R_3	J_1		K_3
	101	10	1	D_1		D_3	T_1	T_2		S_1	R_2		J_1	K_2	
	110	01	0	D_1	D_2		T_1		T_3	S_1		R_3	J_1		K_3
100	011	00	0		D_2	D_3	T_1	T_2	T_3	R_1	S_2	S_3	K_1	J_2	J_3
	000	10	1				T_1			R_1			K_1		
	000	01	1				T_1			R_1			K_1		
101	001	00	0			D_3	T_1			R_1			K_1		
	010	10	1		D_2		T_1	T_2	T_3	R_1	S_2	R_3	K_1	J_2	K_3
	010	01	1		D_2		T_1	T_2	T_3	R_1	S_2	R_3	K_1	J_2	K_3
110	111	00	0	D_1	D_2	D_3			T_3			S_3			J_3
	100	10	1	D_1				T_2			R_2			K_2	
	100	01	1	D_1				T_2			R_2			K_2	
111	101	00	0	D_1		D_3		T_2			R_2			K_2	
	110	10	1	D_1	D_2				T_3			R_3			K_3
	110	01	1	D_1	D_2				T_3			R_3			K_3

– выберем тип триггера рекуррентного кодирующего устройства, в качестве которого принимается триггер JK – типа;

– рассматривая таблицу 7.21 как таблицу истинности для булевой функции выхода кодирующего устройства и для булевых функций сигналов возбуждения информационных входов JK – триггера на наборах переменных $\{x(k)U(k)\} = \{x(k)u(k)u_y(k)\}$ составим аналитическое представление для этих функций, которые в минимизированной форме принимают вид

$$\begin{aligned}
 y(k) &= \bar{u}_y(k)u(k) \vee u_y(k)\bar{u}(k)x_1(k); \\
 v_{J1}(k) &= \bar{x}_1(k)x_2(k)(\bar{u}_y(k) \vee \bar{u}(k)u_y(k)), \\
 v_{K1}(k) &= x_1(k)\bar{x}_2(k)(\bar{u}_y(k) \vee \bar{u}(k)u_y(k)); \\
 v_{J2}(k) &= \bar{u}_y(k)\bar{u}(k)\bar{x}_2(k)(\bar{x}_1(k)x_3(k) \vee x_1(k)\bar{x}_3(k)) \vee u_y(k)\bar{u}(k)\bar{x}_2(k)x_3(k) \vee \\
 &\quad \vee \bar{u}(k)u_y(k)\bar{x}_2(k)(\bar{x}_1(k)\bar{x}_3(k) \vee x_1(k)x_3(k)) \\
 v_{K2}(k) &= u(k)\bar{u}_y(k)x_2(k)(x_1(k)\bar{x}_3(k) \vee \bar{x}_1(k)x_3(k)) \vee \bar{u}(k)u_y(k)x_2(k)\bar{x}_3(k) \vee \\
 &\quad \vee \bar{u}(k)\bar{u}_y(k)x_2(k)(\bar{x}_1(k)\bar{x}_3(k) \vee x_1(k)x_3(k)) \\
 v_{J3}(k) &= \bar{x}_1(k)\bar{x}_3(k)u(k)\bar{u}_y(k) \vee x_1(k)\bar{x}_3(k)\bar{u}(k)\bar{u}_y(k), \\
 v_{K3}(k) &= \bar{x}_1(k)x_3(k)\bar{u}(k) \vee x_1(k)x_3(k)(u(k)\bar{u}_y(k) \vee \bar{u}(k)u_y(k)). \quad \blacksquare
 \end{aligned}$$

7.5 Синтез конечных автоматов на основе граф – схем алгоритмов функционирования устройств дискретной автоматики

Рассмотрим теперь возможности автоматного синтеза ДДС (КА) с использованием граф-схем алгоритмов (ГСА) их функционирования, для построения их нелинейного модельного представления «вход-состояние-выход» (ВСВ). Автоматный синтез конечных автоматов с использованием ГСА обладает заметной пользовательской общностью, так как позволяет синтезировать как ДДС устройств преобразования кодов, так и ДДС устройств дискретной автоматики общетехнического назначения. Причем для его реализации достаточно вербального (словесного) описания функционирования указанных выше устройств.

Алгоритм 7.10

автоматного синтеза модельного представления (ВСВ) НДДС с использованием ГСА описаний

1. Сформулировать постановку задачи функционирования устройства дискретной автоматики или преобразования кодов, решаемой средствами синтезируемой нелинейной ДДС.

2. Построить граф–схему алгоритма функционирования ДДС с вербальным описанием операторов и логических условий изменения хода процесса функционирования ДДС (реализации ГСА) на основе ее словесного описания или анализа временной диаграммы устройства

дискретной автоматики с учетом того обстоятельства, что ГСА является направленным графом, использующим вершины трех типов: начальную/конечную операторную, рабочие операторные и условные;

В операторные вершины вписать вербальные конструкции в виде инфинитивов или отглагольных существительных, несущих информацию о необходимости выполнения конкретного действия, с учетом того, что начальная и конечная вершины имеют соответственно только выход или только вход, а рабочая операторная вершина имеет один вход и один выход;

В условные вершины вписать вербальные логические условия, с помощью которых осуществляется управление последовательностью действий проектируемой НДДС. Условные вершины имеют один вход и два выхода, причем если вершина моделирует процесс (состояние) ожидания выполнения условия, то один из ее выходов соединяется с ее входом;

Проконтролировать корректность составленной ГСА путем проверки наличия хотя бы одной ветви с выхода произвольной (операторной или условной) вершины, ведущей к входу конечной вершины, и проверки отсутствия ветвей с выхода каждой произвольной вершины графа к входам более чем одной вершины. При контроле дополнительно учесть, что ГСА допускает размещение одного и того же условия в различных условных вершинах графа и разрешает выполнение одного и того же действия в различных операторных вершинах графа.

3. Составить формальную версию ГСА путем замены вербальных конструкций операторных вершин на элементы алфавита высокого уровня $w_j, j = 0, m_w - 1$ символьного представления действий (операций, команд), и вербальных конструкций, вписанных в условные вершины, на элементы $z_i, i = 1, r_z$, алфавита символьного представления условий, имеющих бинарную реализацию в форме «да – нет».

4. «Погрузить» сформированную в п.3 алгоритма формальную версию ГСА синтезируемой ДДС в среду абстрактных автоматов (АА) с учетом следующих обстоятельств;

Если АА строится в автоматной логике абстрактного автомата Мура, то всем операторным вершинам w_j присваиваются состояния s_{k+1} , причем начальная w_0 и конечная $w_{k=m_w-1}$ вершины объединены в одну, которой присваивается состояние s_1 ;

Если АА строится в автоматной логике абстрактного автомата Мили, то состояние s_1 присваивается входу первой условной вершины, непосредственно следующей за начальной операторной вершиной. Это

же состояние присваивается конечной операторной вершине. Остальные состояния $s_k, k = \overline{2, n_s}$ присваиваются входам всех условных вершин, непосредственно следующих за операторными вершинами графа;

Обратить внимание на то, что АА, реализующий ГСА в автоматной логике Мура, характеризуется числом состояний n_s , совпадающим с числом операторных вершин, в то время как АА, реализуемый в автоматной логике Мили, характеризуется числом состояний n_s , в общем случае не совпадающим с числом операторных вершин, при этом возможны такие ГСА, где число состояний меньше числа операторных вершин. На этапе погружения формальной ГСА в автоматную среду на паре автоматных логик Мили/Мура осуществить начальную минимизацию автоматной реализации НДДС. Зафиксировать результат погружения формальной версии ГСА в автоматную среду в форме АА, задаваемого с помощью макровектора (7.1) с функциями перехода и выхода в форме (7.2) – (7.4). Построенная в этом пункте ГСА именуется «отмеченной»

5. В соответствии с выбранной логикой функционирования АА построить граф переходов и выходов АА, чем завершается погружения ГСА функционирования проектируемого устройства в автоматную среду, что позволяет далее следовать алгоритму 7.1 канонического синтеза конечных автоматов.

6. Выполнить п.п.3–7 алгоритма 7.1 применительно к АА в выбранной логике. При этом следует иметь ввиду, что выходы $y_l (l = \overline{1, m})$ есть сигналы разрешения выполнения действий, вписанных в операторную вершину так, что в соответствии с автоматной логикой Мура выходы определяются кодами состояния автомата, а в соответствии с автоматной логикой Мили выходы формируются на переходе из состояния в состояния и формируются в виде конъюнкции кода исходного состояния и сигнала управления этим переходом в виде конъюнкции переменных $z_j, j = \overline{1, r_z}$ условных вершин, соединяющих исходное состояние с состоянием перехода. ■

Примечание 7.6. При выполнении п. 5 алгоритма 7.10 в фазе кодирования следует отметить, что кодирование алфавитов состояния и выхода осуществляется в полном соответствии с п. 2 алгоритма 7.1. Кодирование элементов алфавита Z не производится, так как управление исполнением графа осуществляется конъюнкцией логических переменных $z_j, j = \overline{1, r_z}$, потому что логические переменные z_j имеют бинарную природу так, что «да» – «нет» можно заменить на «1» – «0», т.е. положить $z_j \in \{0; 1\}$.

□

Примечание 7.7. При составлении системы булевых функций (БФ), предусмотренных п.5 алгоритма 7.1 применительно к конструированию функций возбуждения триггеров, они строятся в виде дизъюнкций основных конъюнкций, которые формируются на кодах исходных состояний $x(k)$ и управляющих сигналов, считываемых с условных вершин и связывающих исходное состояние с состоянием перехода $x(k+1)$. БФ формирования выходов, в случае использования логики абстрактных автоматов Мура, конструируются посредством дизъюнкций основных конъюнкций, представляющих собой исходные состояния автомата. В случае использования абстрактных автоматов Мили булевы функции строятся по той же схеме, что и булевы функции возбуждения.

□

Пример 7.8. Ставится задача синтеза нелинейной ДДС, функционирование которой задается отмеченными ГСА в соответствии с автоматной логикой Мура (рисунок 7.4) и автоматной логикой Мили (рисунок 7.5).

Таким образом в решении поставленной задачи опускается этап вербального описании ГСА и осуществляется с этапа отмеченных формальных ГСА, построенных в двух автоматных логиках, чем обеспечивается возможность непосредственного перехода к п.5 алгоритма 7.10.

Более того выполнение п.5 алгоритма 7.10, состоящего в выполнении п.п.3–7 алгоритма 7.1, может быть осуществлено агрегированным образом так, что результатом п.5 становится таблица «переходов – выходов – возбуждаемых входов триггеров», которая характеризуется наличием следующих столбцов: состояния перехода $s_i(k+1)$; кода состояния перехода $K\{s_i(k+1)\} = \{x(k+1)\}_i$; исходные состояния $s_v(k)$; коды исходных состояний $K\{s_v(k)\} = \{x(k)\}_v$; управляющее воздействия в виде конъюнкций $U = \&u_j$ логических переменных, размещенных в условных вершинах, соединяющих состояния $s_i(k+1)$ и $s_v(k)$; выходов $y_l (l = \overline{0, m-1})$, а также столбцов «возбуждаемых входов триггеров», при этом выход y_0 , связанный с начальной – конечной вершиной w_0 является пустым и в таблице «переходов – выходов – возбуждаемых входов триггеров» в графе «выходы» представляется прочерком. В отмеченном графе автомата Мили выбираются только пути, связывающие два состояния, на которых обязательно формируется выход, остальные пути сопровождаются пустым выходом, потому игнорируются и в таблицу не вносятся.

Таблицы «переходов – выходов – возбуждаемых входов триггеров» для двух автоматных логик Мура (ГСА рисунок 7.4) и Мили (ГСА рисунок 7.5) проектируемой НДДС (КА) оформлен в форме таблиц 7.22 и 7.23.

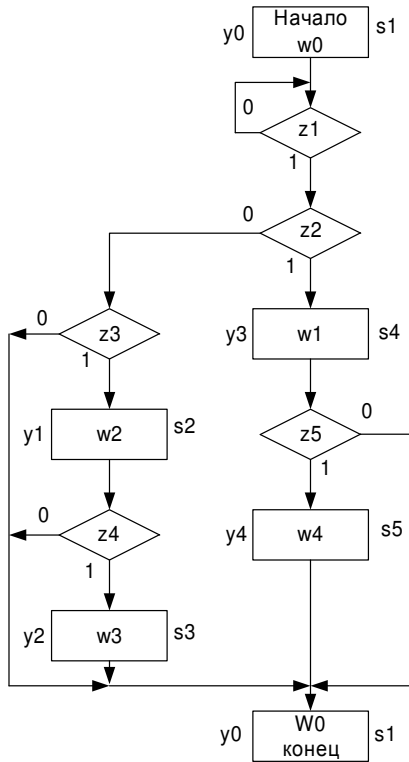


Рисунок 7.4

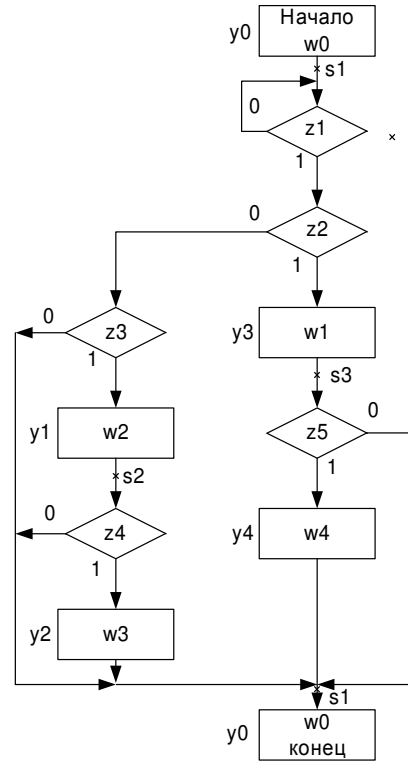


Рисунок 7.5

Система булевых функций, описывающих спроектированный КА (НДДС) по приведенной ГСА в автоматной логике Мура и реализуемый на D -триггерах принимает вид:

$$y_1 = \bar{x}_1 \bar{x}_2 x_3; y_2 = \bar{x}_1 x_2 x_3; y_3 = \bar{x}_1 x_2 \bar{x}_3; y_4 = x_1 x_2 \bar{x}_3;$$

$$v_{D1} = \bar{x}_1 x_2 \bar{x}_3 z_5; v_{D2} = \bar{x}_1 x_2 \bar{x}_3 z_5 \vee x_1 x_2 \bar{x}_3 z_1 z_2 \vee \bar{x}_1 \bar{x}_2 x_3 z_4;$$

$$v_{D3} = \bar{x}_1 \bar{x}_2 \bar{x}_3 z_1 \bar{z}_2 z_3 \vee \bar{x}_1 \bar{x}_2 x_3 z_4.$$

Система булевых функций, описывающих спроектированный КА (НДДС) по приведенной ГСА в автоматной логике Мили и реализуемый на T -триггерах в свою очередь принимает вид:

$$y_1 = x_1 x_2 z_5; y_2 = \bar{x}_1 x_2 z_4; y_3 = \bar{x}_1 x_2 z_1 z_2; y_4 = x_1 x_2 z_5;$$

$$v_{T1} = \bar{x}_1 x_2 z_1 z_2 \vee x_1 x_2 z_5; v_{T2} = \bar{x}_1 x_2 z_4 \vee x_1 x_2 z_5 \vee x_1 x_2 z_5.$$