

Зайцев Николай Геннадьевич

Аспирант кафедры физической электроники ТУСУРа

Телефон (3822) 413936

Эл. почта: kolan317_1@mail.ru

P.E. Troyan, N.G. Zaicev

Automation of measurement of the characteristics of mos-structures with use of trunks ISA and PCI

The summary: The development cycles of the device for automatic measurement volt-farad of the characteristics MOS - structures are considered on the basis of the COMPUTER. The problems met by development of the device and variants of their elimination are stated.

УДК 681.3.06: 62-507

С.В. Шидловский

Исследование функциональных возможностей изотропной структуры, обеспечивающей вычисление неупорядоченных булевых функций

В статье рассматривается построение управляющих устройств на основе многофункциональных логических модулей, обеспечивающих с помощью перенастройки структуры устройства реализацию алгоритмов управления из заданного класса.

Введение

Специализированные процессоры, способные автоматически настраивать параметры цифровых регуляторов под динамические свойства объекта управления, получили в настоящее время довольно широкое распространение [1]. Одним из актуальных направлений является разработка и построение этих спецпроцессоров на основе многофункциональных логических модулей (МЛМ).

В работах [2, 3] рассмотрены F -, S - и T -структуры, реализующие вычисление только упорядоченных булевых функций и не обеспечивающие вычисление неупорядоченных булевых функций, определяемых приведенной в [4] классификацией. В связи с этим возникает необходимость в разработке новых структур с более широкими возможностями.

Вычисление неупорядоченных булевых функций

Пусть задана произвольная функция n аргументов в дизъюнктивно нормальной форме (ДНФ), и пусть она имеет k конъюнкций. Очевидно, что длина конъюнкций не может превышать n букв, поскольку n — это длина минтерма (члена сокращенной дизъюнктивно нормальной формы (СДНФ)). В работе [5] отмечено, что аргументы каждой из конъюнкций, входящих в указанную функцию ДНФ, можно упорядочить. Следовательно, заданную функцию n аргументов можно представить дизъюнкцией неповторных упорядоченных функций с пропусками аргументов

$$f = \lambda_1 + \lambda_2 + \dots + \lambda_k = \bigvee_{i=1}^k \lambda_i, \quad (1)$$

где λ_i — конъюнкции заданной функции f .

Исследуем три варианта вычисления функции (1):

- 1) комбинационной схемой, представленной изотропной либо квазиизотропной матрицей (т.е. составленной из S - и T -ячеек);
- 2) многотактным автоматом путем последовательного вычисления за k тактов, когда в течение каждого такта вычисляется значение только одной конъюнкции функции (1);
- 3) линейной изотропной структурой, построенной из МЛМ, ориентированных на вычисление как упорядоченных, так и неупорядоченных булевых функций.

Рассмотрим первый вариант. Изотропная матрица строится на основе $S(T)$ -структур, где символ $S(T)$ обозначает, что структура состоит из S - либо T -ячеек (рис. 1). По схеме видно, что вместо матрицы можно рассматривать линейную $S(T)$ -структуру, состоящую из $n \times k$ ячеек, где входы $A_1, A_2, \dots, A_{n+1}, A_{2n+1}, \dots, A_{(k-1)n+1}$ соединяются между собой и подключаются к устройству, моделирующему вход A_1 . Входы $A_2, A_{n+2}, A_{2n+2}, \dots, A_{(k-1)n+2}$ также соединяются между собой и объявляются входом аргумента A_2 и т.д. до A_n , которому соответствует соединение $A_n, A_{2n}, \dots, A_{kn}$. Такую структуру целесообразно применять, если основным требованием, предъявляемым к устройству, является быстродействие.

Рассмотрим второй (многотактный) вариант вычисления функции (1). Его суть состоит в том, что все настроечные коды для каждой конъюнкции λ_i записываются в запоминающее устройство (ЗУ), откуда последовательно поступают на настроечные входы $S(T)$ -структуры (рис. 2).

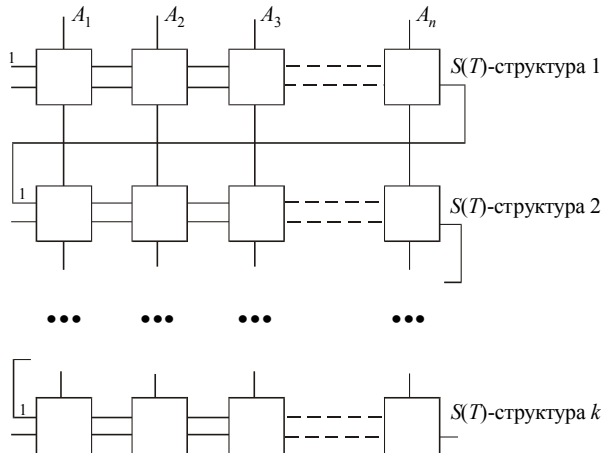


Рис. 1 — Изотропная матрица

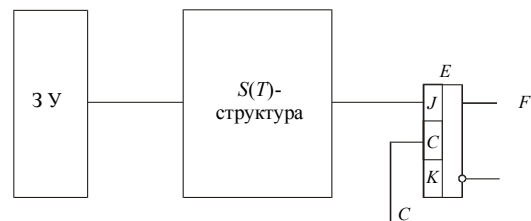


Рис. 2 — Многотактный автомат

С каждым тактом смены настроечных кодов на синхровход триггера E поступает импульс. Если на каком-либо настроечном коде выходной сигнал триггера E принимает единичное значение, то триггер E переходит в единичное состояние, что сигнализирует о равенстве единице функции f . Если же после полного просмотра ЗУ триггер E остался в нулевом состоянии, это значит, что функция на данном наборе значений аргументов равна нулю. ЗУ можно организовать различными способами. Условимся считать, что каждому настроечному коду соответствует определённый адрес ЗУ. Тогда можно пронумеровать конъюнкции λ_i и считать, что номер этой конъюнкции совпадает с адресом, по которому в ЗУ находится её настроечный код. Если не учитывать код β , которым задаётся распределение инверсий над аргументами, то длина выходного кода, хранящегося по заданному адресу ЗУ, равна $2n$. Рассмотрим пример для $n = 7$. Пусть дана функция

$$f = A_1 A_2 + A_1 A_5 A_6 + A_1 A_3 A_7 + A_1 A_6 A_7 + A_4 A_5 A_7. \tag{2}$$

Эта функция не является неповторной, но каждая ее конъюнкция есть неповторная функция с пропусками:

$$\lambda_1 = A_1 A_2; \quad \lambda_2 = A_1 A_5 A_6; \quad \lambda_3 = A_1 A_3 A_7; \quad \lambda_4 = A_1 A_6 A_7; \quad \lambda_5 = A_4 A_5 A_7.$$

В ЗУ будут записаны следующие коды

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	
$S =$	0	0	1	1	1	1	1	
$\alpha =$	0	1	0	0	0	0	0	по адресу 1
$S =$	0	1	1	1	0	0	1	
$\alpha =$	0	0	0	0	0	1	0	по адресу 2
$S =$	0	1	0	1	1	1	0	
$\alpha =$	0	0	0	0	0	0	1	по адресу 3
$S =$	0	1	1	1	1	0	0	
$\alpha =$	0	0	0	0	0	0	1	по адресу 4
$S =$	1	1	1	0	0	1	0	
$\alpha =$	0	0	0	0	0	0	1	по адресу 5

Значение функции (2) будет определено не более чем за пять тактов. Функцию (2) можно вычислить и за четыре такта, если учесть, что первая и последняя конъюнкции образуют неповторную упорядоченную функцию с пропусками аргументов A_3, A_6

$$\lambda_1 = A_1 A_2 + A_4 A_5 A_7,$$

S - и α -коды которой имеют вид

	A_1	A_2	A_3	A_4	A_5	A_6	A_7
$S =$	0	0	1	1	1	1	1
$\alpha =$	0	1	0	0	0	0	0

Под S - и α -кодами, согласно [6], будем понимать следующее. Поставим в соответствие каждому логическому аргументу неповторной упорядоченной функции двоичный разряд и единицей условимся обозначать последний справа аргумент каждой конъюнкции. Все остальные аргументы будем обозначать нулями. В результате получим n -разрядное двоичное число, где n — число аргументов функции. Условимся называть такое число α -кодом.

В S -структуре код, содержащий информацию об удаленных аргументах и задаваемый с помощью S -триггеров, будем называть S -кодом.

Метод декомпозиции для вычисления произвольных булевых функций

Пусть задана произвольная функция f , представленная в ДНФ и зависящая от n аргументов. Подвергнем ее операции декомпозиции следующим образом:

$$f = \varphi_1 + \varphi_2 + \dots + \varphi_n, \quad (3)$$

где φ_i ($i = 1, 2, \dots, n$) — неповторные упорядоченные функции с пропусками или без пропусков аргументов.

Чтобы функцию f вычислить за наименьшее число тактов и с наименьшими затратами памяти, необходимо найти минимальный список функций φ_i , дизъюнкция которых дает функцию f . В результате решения данной задачи разработан метод нахождения минимального списка функций φ_i , которые в дальнейшем будем называть φ -импликантами.

Суть метода состоит в следующем. Находим конъюнкцию, содержащую наименьший номер аргумента. Если их несколько, то выбираем из них первый произвольным образом. Если выбранная конъюнкция состоит более чем из одного аргумента, то она оканчивается аргументом с номером δ_1 . Затем находим конъюнкцию, содержащую первым аргумент, индекс которого является ближайшим к δ_1 . Это будет вторая конъюнкция, которая войдет в φ -импликанту. Вторая конъюнкция оканчивается аргументом с индексом δ_2 . Аналогично находим третью, четвертую конъюнкции и т.д., пока не получим конъюнкцию, оканчивающуюся аргументом с номером δ_i , таким, что других конъюнкций, первый аргумент которых превышает δ_i , в заданной функции f нет.

Закодируем двоичными числами все найденные φ -импликанты. Для этого запишем в ряд конъюнкции, образующие функцию (3). Каждой из них поставим в соответствие двоичный разряд и единицами будем отмечать конъюнкции, входящие в ту или иную φ -импликанту. Перечислив все φ -импликанты, получим таблицу двоичных чисел, где каждому числу будет соответствовать определенная φ -импликанта. Обозначим φ -импликанты символами $\varphi_1, \varphi_2, \dots, \varphi_n$. Затем составим булево уравнение, представляющее собой конъюнкцию скобочных выражений, где в каждой скобке записана дизъюнкция φ -импликант. Число скобочных выражений в уравнении равно числу конъюнкций заданной функции, т.е. каждой конъюнкции заданной функции соответствует скобочное выражение в уравнении. Чтобы найти $\varphi_1, \varphi_2, \dots, \varphi_n$, входящие в скобочные выражения, достаточно обратить внимание на единицы в соответствующей колонке полученной таблицы.

Представим булево уравнение в ДНФ. После минимизации получим дизъюнкцию конъюнкции символов $\varphi_1, \varphi_2, \dots, \varphi_n$. Каждая конъюнкция даёт одно решение задачи декомпозиции.

Для иллюстрации метода рассмотрим пример. Дана функция

$$f = A_1 A_2 A_3 + A_1 A_4 A_6 + A_4 A_7 A_8 A_9 + A_5 A_7 A_8 + A_9 A_{10} A_{12} A_{14} + A_9 A_{13} + A_{10} A_{11} A_{12} + A_{13} A_{14},$$

которая не является неповторной, следовательно, её необходимо подвергнуть операции декомпозиции. Наиболее простой вариант декомпозиции состоит в том, что вычисление ведётся отдельно по каждой конъюнкции. При этом потребуются 8 тактов. Выясним, сколько

тактов потребуется при минимальном варианте. Составляем табл. 1 φ -импликант, где полный список φ -импликант, выраженных через исходные аргументы заданной функции, имеет вид

$$\begin{aligned}\varphi_1 &= A_1 A_2 A_3 + A_4 A_7 A_8 A_9 + A_{10} A_{11} A_{12} + A_{13} A_{14} ; \\ \varphi_2 &= A_1 A_2 A_3 + A_5 A_7 A_8 + A_9 A_{10} A_{12} A_{14} ; \\ \varphi_3 &= A_1 A_2 A_3 + A_5 A_7 A_8 + A_9 A_{13} ; \\ \varphi_4 &= A_1 A_2 A_3 + A_5 A_7 A_8 + A_{10} A_{11} A_{12} + A_{13} A_{14} ; \\ \varphi_5 &= A_1 A_4 A_6 + A_9 A_{10} A_{12} A_{14} ; \\ \varphi_6 &= A_1 A_4 A_6 + A_9 A_{13} ; \\ \varphi_7 &= A_1 A_4 A_6 + A_{10} A_{11} A_{12} + A_{13} A_{14} .\end{aligned}$$

Таблица 1

 φ -ИМПЛИКАНТЫ

$A_1 A_2 A_3$	$A_1 A_4 A_6$	$A_4 A_7 A_8 A_9$	$A_5 A_7 A_8$	$A_9 A_{10} A_{12} A_{14}$	$A_9 A_{13}$	$A_{10} A_{11} A_{12}$	$A_{13} A_{14}$	
1	0	1	0	0	0	1	1	φ_1
1	0	0	1	1	0	0	0	φ_2
1	0	0	1	0	1	0	0	φ_3
1	0	0	1	0	0	1	1	φ_4
0	1	0	0	1	0	0	0	φ_5
0	1	0	0	0	1	0	0	φ_6
0	1	0	0	0	0	1	1	φ_7

Составляем булево уравнение. Прежде всего рассмотрим колонку таблицы, где записана конъюнкция $A_4 A_7 A_8 A_9$, так как в ней только одна единица. Это значит, что конъюнкция $A_4 A_7 A_8 A_9$ войдёт в какую-либо функцию искомого их списка только в единственном случае — когда в список будет включена φ -импликанта φ_1 . Очевидно, что эта φ -импликанта войдёт во все варианты декомпозиции. Далее, поскольку импликанта φ_1 введена в список, следовательно, в него будут введены и конъюнкции $A_1 A_2 A_3$, $A_{10} A_{11} A_{12}$, $A_{13} A_{14}$. Вычеркиваем из таблицы верхнюю строку и колонки, обозначенные единицами в этой строке. Построим новую таблицу (табл. 2). По ней строим булево уравнение в виде конъюнкции дизъюнкций:

$$(\varphi_5 + \varphi_6 + \varphi_7)(\varphi_2 + \varphi_3 + \varphi_4)(\varphi_2 + \varphi_5)(\varphi_3 + \varphi_6) = 1.$$

Раскроем скобки:

$$\begin{aligned} & (\varphi_5 + \varphi_2 \varphi_6 + \varphi_2 \varphi_7)(\varphi_3 + \varphi_2 \varphi_6 + \varphi_4 \varphi_6) = \\ & = \varphi_3 \varphi_5 + \varphi_2 \varphi_3 \varphi_6 + \varphi_2 \varphi_3 \varphi_7 + \varphi_2 \varphi_5 \varphi_6 + \varphi_2 \varphi_6 + \varphi_2 \varphi_6 \varphi_7 + \varphi_4 \varphi_5 \varphi_6 + \varphi_2 \varphi_4 \varphi_6 + \varphi_2 \varphi_4 \varphi_7. \end{aligned}$$

Выполнив все операции поглощения, получим минимальную ДНФ

$$\varphi_3 \varphi_5 + \varphi_2 \varphi_6 + \varphi_2 \varphi_3 \varphi_7 + \varphi_4 \varphi_5 \varphi_6 = 1.$$

Отсюда следует, что существует четыре варианта искомого списка функций, из которых два варианта соответствуют случаю, когда список содержит три функции. Эти два минимальных варианта имеют вид

$$\begin{cases} \varphi_1 = A_1 A_2 A_3 + A_4 A_7 A_8 A_9 + A_{10} A_{11} A_{12} + A_{13} A_{14}, \\ \varphi_3 = A_1 A_2 A_3 + A_5 A_7 A_8 + A_9 A_{13}, \\ \varphi_5 = A_1 A_4 A_6 + A_9 A_{10} A_{12} A_{14}; \end{cases} \quad (4)$$

$$\begin{cases} \varphi_1 = A_1 A_2 A_3 + A_4 A_7 A_8 A_9 + A_{10} A_{11} A_{12} + A_{13} A_{14}, \\ \varphi_2 = A_1 A_2 A_3 + A_5 A_7 A_8 + A_9 A_{10} A_{12} A_{14}, \\ \varphi_6 = A_1 A_4 A_6 + A_9 A_{13}. \end{cases} \quad (5)$$

В списке (4) конъюнкция $A_1 A_2 A_3$ встречается дважды. Одну из них можно удалить, например, из функции φ_3 . Аналогично удаляем эту конъюнкцию и из функции φ_2 списка (5).

Таблица 2

φ -импликанты после декомпозиции

$A_1 A_4 A_6$	$A_5 A_7 A_8$	$A_9 A_{10} A_{12} A_{14}$	$A_9 A_{13}$	
0	1	1	0	φ_2
0	1	0	1	φ_3
0	1	0	0	φ_4
1	0	1	0	φ_5
1	0	0	1	φ_6
1	0	0	0	φ_7

В связи с тем что в функции φ_3 системы (4) конъюнкцию $A_1 A_2 A_3$ можно было бы и не вычёркивать, можно сделать вывод: в общем случае представление минимального списка φ -импликант α - и S -кодами неоднозначно.

Однотактное вычисление неупорядоченных булевых функций

Как отмечалось выше, существует третий путь вычисления неупорядоченных булевых функций — с помощью линейных изотропных сред на многофункциональных логических модулях, обеспечивающих вычисление как упорядоченных, так и неупорядоченных булевых функций. Применение МЛМ подобного типа сопряжено с трудностями их разработки. Покажем реализацию разработанного МЛМ, ориентированного на вычисление вышеназванных булевых функций, представленных классами 1–6 [4] и подклассом J , определяемым следующим образом.

Пусть дана функция, явно или неявно зависящая от аргументов x_1, x_2, \dots, x_n . Запишем эти аргументы в порядке возрастания их индексов слева направо. Аргумент, имеющий наименьший индекс, будем называть минимальным, а наибольший — максимальным. Диапазоном функции назовем замкнутый интервал, границы которого образуют индексы минимального и максимального аргументов. Условимся, что интервалы двух различных функций пересекаются, если минимальный аргумент одной из функций входит в интервал другой.

Если функция f представлена в виде

$$f = l * \omega * q,$$

то она входит в подкласс J , где l и ω — упорядоченные функции с пересекающимися диапазонами; q — упорядоченная функция, минимальный аргумент которой не входит в диапазон функций l и ω .

В общем случае функция q может быть тождественно равна нулю; $*$ — знак конъюнкции или дизъюнкции. Функции l и ω могут быть любого порядка.

Логическая схема разработанного МЛМ описывается следующей системой булевых формул:

$$\begin{cases} f_1 = y_1 (\bar{z}_1 z_2 \vee \bar{z}_1 z_3 \bar{z}_4 \vee x \bar{z}_3) \vee (z_1 \vee z_2) x \bar{z}_3; \\ f_2 = (\bar{z}_1 x \vee z_3 x \vee z_2 \vee \bar{z}_2 y_1 \vee \bar{z}_1 \bar{z}_3 \vee \bar{z}_1 z_4) y_2 \vee x z_4 \vee y_1 z_1 z_2, \end{cases}$$

где x, y_1, y_2 — информационные входы; z_1, z_2, z_3, z_4 — настроечные входы; f_1, f_2 — выходы МЛМ.

Многофункциональный логический модуль, который в дальнейшем будем называть H -ячейкой, реализует следующие системы булевых формул:

- | | | | |
|--|--|--|--|
| 1) при $z_4 = 0, z_3 = 0, z_2 = 0, z_1 = 0$; | 9) при $z_4 = 1, z_3 = 0, z_2 = 0, z_1 = 0$; | | |
| 2) при $z_4 = 0, z_3 = 0, z_2 = 0, z_1 = 1$. | 10) при $z_4 = 1, z_3 = 0, z_2 = 0, z_1 = 1$; | | |
| 1) $\begin{cases} f_1 = y_1 x, \\ f_2 = y_2; \end{cases}$ | 2) $\begin{cases} f_1 = x, \\ f_2 = y_1 y_2; \end{cases}$ | 9) $\begin{cases} f_1 = y_1 x, \\ f_2 = y_2 \vee x; \end{cases}$ | 10) $\begin{cases} f_1 = x, \\ f_2 = y_1 y_2 \vee x; \end{cases}$ |
| 3) при $z_4 = 0, z_3 = 0, z_2 = 1, z_1 = 0$; | 11) при $z_4 = 1, z_3 = 0, z_2 = 1, z_1 = 0$; | | |
| 4) при $z_4 = 0, z_3 = 0, z_2 = 1, z_1 = 1$; | 12) при $z_4 = 1, z_3 = 0, z_2 = 1, z_1 = 1$; | | |
| 3) $\begin{cases} f_1 = y_1 \vee x, \\ f_2 = y_2; \end{cases}$ | 4) $\begin{cases} f_1 = x, \\ f_2 = y_1 \vee y_2; \end{cases}$ | 11) $\begin{cases} f_1 = y_1 \vee x, \\ f_2 = y_2 \vee x; \end{cases}$ | 12) $\begin{cases} f_1 = x, \\ f_2 = y_1 \vee y_2 \vee x; \end{cases}$ |
| 5) при $z_4 = 0, z_3 = 1, z_2 = 0, z_1 = 0$; | 13) при $z_4 = 1, z_3 = 1, z_2 = 0, z_1 = 0$; | | |
| 6) при $z_4 = 0, z_3 = 1, z_2 = 0, z_1 = 1$; | 14) при $z_4 = 1, z_3 = 1, z_2 = 0, z_1 = 1$; | | |
| 5) $\begin{cases} f_1 = y_1, \\ f_2 = y_2 x; \end{cases}$ | 6) $\begin{cases} f_1 = 0, \\ f_2 = x y_2; \end{cases}$ | 13) $\begin{cases} f_1 = 0, \\ f_2 = y_2 \vee x; \end{cases}$ | 14) $\begin{cases} f_1 = 0, \\ f_2 = x; \end{cases}$ |
| 7) при $z_4 = 0, z_3 = 1, z_2 = 1, z_1 = 0$; | 15) при $z_4 = 1, z_3 = 1, z_2 = 1, z_1 = 0$; | | |
| 8) при $z_4 = 0, z_3 = 1, z_2 = 1, z_1 = 1$; | 16) при $z_4 = 1, z_3 = 1, z_2 = 1, z_1 = 1$; | | |
| 7) $\begin{cases} f_1 = y_1, \\ f_2 = y_2; \end{cases}$ | 8) $\begin{cases} f_1 = 0, \\ f_2 = y_1 \vee y_2; \end{cases}$ | 15) $\begin{cases} f_1 = y_1, \\ f_2 = y_2 \vee x; \end{cases}$ | 16) $\begin{cases} f_1 = 0, \\ f_2 = y_1 \vee y_2 \vee x. \end{cases}$ |

Проиллюстрируем работу изотропных сред (H -структур), построенных на H -ячейках, на следующих примерах.

Пример 1. Реализация бесповторной неупорядоченной функции вида $f_1 = (x_1 x_3 + x_4) x_2$ представлена на рис. 3, где для каждой ячейки указаны настроечные коды, функция f_1 принадлежит подклассу J , поскольку $l = x_1 x_3 + x_4$, $\omega = x_2$, $q \equiv 0$, а минимальный аргумент ω -функции входит в диапазон функции l . Нетрудно убедиться в том, что функция f_1 является неупорядоченной, так как среди всех формул, получаемых путем тождественных преобразований, упорядоченные записи отсутствуют.

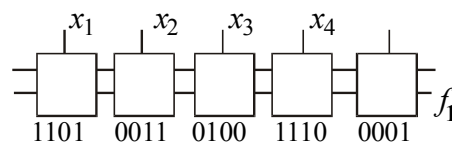


Рис. 3 — К примеру 1

Пример 2. Аналогично для бесповторной неупорядоченной ни при каких тождественных преобразованиях функции $f_2 = x_1 x_3 + x_2 x_4 + x_6 x_8 + x_9$, содержащей пропуска аргументов x_5 и x_7 , имеем $l = x_1 x_4$; $\omega = x_2 x_4$; $q = x_6 x_8 + x_9$.

Структура и настроечные коды ячеек приведены на рис. 4.

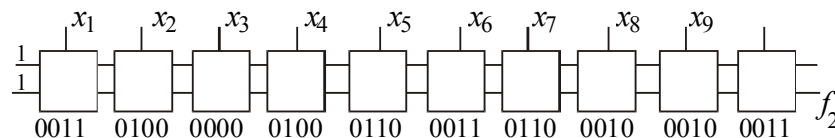


Рис. 4 — К примеру 2

Пример 3. Рассмотрим случай реализации функции $f_3 = x_1 x_2 + x_1 x_3 + x_2 x_3 + x_4 + x_5$. Функция f_3 является повторной, так как не существует никаких тождественных преобразований, в результате которых получалась бы запись с однократным вхождением в нее каждо-

го аргумента. Настраиваемые коды для каждой ячейки изотропной среды, реализующей функцию f_3 , представлены на рис. 5.

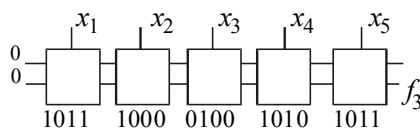


Рис. 5 — К примеру 3

Пример 4. Реализация неупорядоченной функции высокого порядка

$$f_4 = \{(x_1x_6 + x_8)x_{10} + x_{11}\}x_{12} + x_{13} \{ \{(x_2x_3x_4 + x_5)x_7x_9 + x_{14}\} \} + [(x_{15}x_{16} + x_{17})x_{18} + x_{20}] x_{21}$$

с пропуском аргумента x_{19} , настраиваемыми кодами для каждой ячейки представлена на рис. 6.

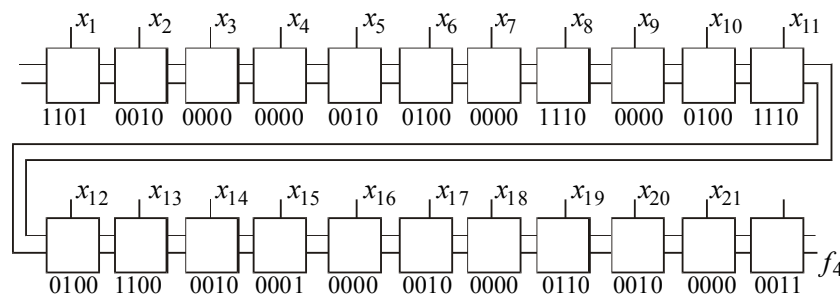


Рис. 6 — К примеру 4

Заключение

Предложены оригинальные многофункциональные логические модули для построения управляющих устройств, обеспечивающих реализацию алгоритмов управления из заданного класса. По результатам проведенного исследования можно сделать следующие выводы.

1. Изотропные среды, построенные на H -ячейках, обеспечивают реализацию как класса неповторных упорядоченных произвольных нормальных формул из h букв (в том числе любых скобочных), так и формул из класса неупорядоченных и повторных формул, определяемых подклассом J .

2. На основе рассмотренных изотропных сред можно построить различные проблемно-ориентированные комплексы, которые могут настраиваться на реализацию конкретного алгоритма из заданного класса.

Литература

1. Артюхов В.Л., Копейкин Г.А., Шалыто А.А. Настраиваемые модули для управляющих логических устройств. —Л.: Энергоатомиздат, 1981. — 168 с.
2. Шидловский С.В., Фоминых Ю.И. Исследование функциональных возможностей однородной структуры, обеспечивающей вычисление неповторных ДНФ булевых функций // Вестник Сибирского отделения АН ВШ. —2001. —№ 1(7). — С. 50–59.
3. Шидловский С.В. Многофункциональный логический модуль для реализации операций удаления аргументов из булевых функций //Радиотехнические устройства, информационные технологии и системы управления: Материалы региональной науч.-техн. конф. студентов и молодых специалистов. — Томск, 2001. — С. 15–16.
4. Шидловский С.В. Перестраиваемые структуры на многофункциональных логических модулях //Информационные системы: Труды постоянно действующей науч.-техн. школы-семинара студентов, аспирантов и молодых специалистов «Информационные системы мониторинга окружающей среды». Вып. 2. —Томск: ТУСУР, 2003. — С. 105–117.
5. Арсенев Ю.Н., Журавлев В.М. Проектирование систем логического управления на микропроцессорных средствах. —М.: Высш. шк., 1991. — 319 с.
6. Шидловский С.В., Светлаков А.А. Исследование функциональных возможностей многофункционального логического модуля, реализующего операции удаления аргументов из булевых функций //Вестник Сибирского отделения АН ВШ. — 2002. — № 1(8). — С. 72–78.