

УДК 519.1(075.8)+510.6(075:8)

Минимизация бинарных переключательных функций в базисах 2И-НЕ, 2И-2ИЛИ-НЕ

С. Ф. Тюрин, А. С. Прохоров

Пермский национальный исследовательский политехнический университет
Россия, 614990, Пермь, ул. Комсомольский пр., 29
tyurinsergfeoyandex.ru; +7-952-32-02-510

Рассматривается подход к минимизации логических функций в традиционном базисе $(\bar{x}_1 \vee \bar{x}_2)$ и в избыточном базисе $(\bar{x}_1 \vee \bar{x}_2)(\bar{x}_3 \vee \bar{x}_4)$. Показано, что поиск оптимального покрытия единиц заданной функции ищется в виде дерева импликант соответствующего базиса. Реализация в избыточном базисе равноценна по числу транзисторов, но предпочтительна по быстродействию.

Ключевые слова: минимизация; базис, импликанта; логический элемент; схема реализации мажоритарной функции в заданном базисе.

Введение

Известно, что минимизация логических (переключательных) функций выполняется в Булевом базисе И, ИЛИ, НЕ [1, 2]. Пусть задана некоторая бинарная переключательная функция своей таблицей истинности (рис. 1).

a	b	c	f(abc)
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Рис. 1. Таблица истинности некоторой бинарной переключательной функции

Минимизация в булевом базисе И, ИЛИ, НЕ, например по карте Карно (рис. 2), позволяет получить выражение

$$f(abc) = ab \vee ac \vee bc, \quad (1)$$

которое представляет собой так называемую мажоритарную функцию.

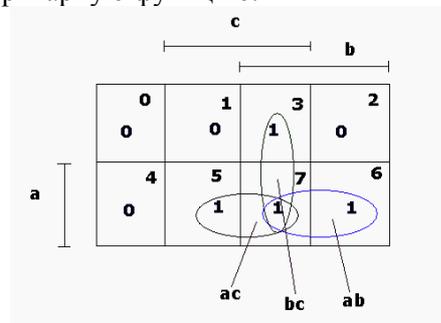


Рис. 2. Минимизация в булевом базисе И, ИЛИ, НЕ функции рис. 1 по карте Карно

В выражении (1) имеется три импликанты, но ни количество импликант, ни число переменных в импликанте при минимизации в булевом базисе не ограничено. То есть, такое представление функции является своего рода линейным (рис. 3).

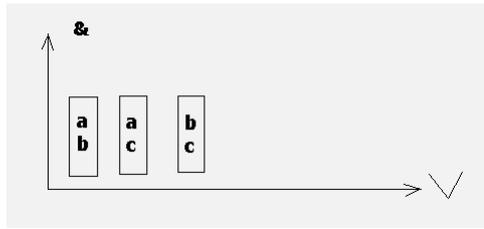


Рис. 3. Линейное представление функции в булевом базисе

После чего полученные выражения представляются в заданном базисе [3, 4]. Возникают некоторые сомнения об оптимальности полученных за два "приема" схем. Тем более, что в случае минимизации в заданном базисе, например И-НЕ, ИЛИ-НЕ, можно было бы сразу получать схему, совместив по возможности процесс её построения с процессом нахождения импликант, т.е. выполнять преобразования за один этап. Однако известно, что минимизация в произвольных базисах крайне сложна, что вызвано лавинообразно увеличивающейся комбинаторной сложностью задачи [5–8].

В [1–12] предложена концепция функционально-полного толерантного (ФПТ) логического базиса и элемента. ФПТ-элемент сохраняет свойство базисности в смысле теоремы Поста при заданной модели отказов. ФПТ-базис также предпочтителен для реализации широкого класса логических (булевых) функций по сравнению с традиционными базисами И-НЕ, ИЛИ-НЕ.

2. Минимизация логической функции в базисе $(\bar{x}_1 \vee \bar{x}_2)$

Как же минимизировать функцию рис. 1 в базисе только одной функции, например 2И-НЕ $(\bar{x}_1 \vee \bar{x}_2)$. (2)

В этом случае ограничено число дизъюнкций с инверсией не более чем одной переменной – до двух. То есть, представление функции не линейное, а древовидное, первые два яруса имеют вид (рис. 4).

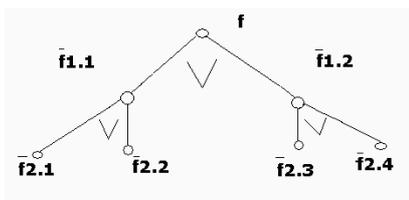


Рис. 4. Древовидное (два уровня) представление – корнем вверх – функции в базисе И-НЕ $(\bar{x}_1 \vee \bar{x}_2)$

Следует заметить, что если бы базис был не 2И-НЕ, а 3И-НЕ, то дерево (рис. 4) было бы не бинарное, а тернарное.

Продолжаем, для третьего уровня:

$$f = \overline{f_{3.1} \vee f_{3.2} \vee f_{3.3} \vee f_{3.4} \vee f_{3.5} \vee f_{3.6} \vee f_{3.7} \vee f_{3.8}} \quad (3)$$

Необходимо определить все подфункции этого уровня. То есть, найти минимальное дерево из всех возможных деревьев!

3. Проверка покрытия заданной функции одним элементом $(\bar{x}_1 \vee \bar{x}_2)$

Будем получать всевозможные выражения вида

$$(\bar{f}_{i,j} \vee \bar{f}_{i,j+1}); f \in \{a, b, c, 0, 1\}, \quad (4)$$

т.е. вначале из переменных $f(abc)$ констант 0, 1 и строить соответствующие дополнительные столбцы таблицы истинности – получим рис. 5.

a	b	c	f(abc)	\bar{a}	\bar{b}	\bar{c}	$a \vee \bar{b}$	$\bar{b} \vee c$	$\bar{a} \vee \bar{c}$
0	0	0	1	1	1	1	1	1	1
0	0	1	1	1	1	0	1	1	1
0	1	0	0	1	0	1	1	1	1
0	1	1	1	1	0	0	1	0	1
1	0	0	0	0	1	1	1	1	1
1	0	1	1	0	1	0	1	1	0
1	1	0	1	0	0	1	0	1	1
1	1	1	1	0	0	0	0	0	0

Рис. 5. Расширенная таблица истинности функции, представленной рис. 1 для выражений вида (5)

$$(\bar{f}_{i.1} \vee \bar{f}_{i.2}); f \in \{a, b, c, 0, 1\}. \quad (5)$$

Ни в одном дополнительном столбце (рис. 5) единицы не соответствуют единицам функции $f(abc)$. То есть, одним элементом заданную функцию не реализовать. Кроме того, ни один дополнительный столбец (рис. 5) не имплицирует $f(abc)$.

4. Проверка возможности реализации заданной функции двумя элементами $(\bar{x}_1 \vee \bar{x}_2)$

Полагаем, в первую очередь необходимо проверить, не покрываются ли единицы функции ее переменными и дизъюнкциями двух таких конъюнкций. Для этого требуется 2 элемента, что можно назвать "простым" или "примитивным" представлением, так как некоторые элементы используются как простые инверторы. То есть, иными словами,

$$(\bar{f}_{i,1} \vee \bar{f}_{i,2}); f \in \{a, b, c, 0, 1, \bar{a}, \bar{b}, \bar{c}\}. \quad (6)$$

Получим рис. 6.

a	b	c	f(abc)	\bar{a}	\bar{b}	\bar{c}	$\overline{a \vee b}$	$\overline{b \vee c}$	$\overline{a \vee c}$
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0
0	1	0	0	0	1	0	0	0	0
0	1	1	1	0	1	1	0	1	0
1	0	0	0	1	0	0	0	0	0
1	0	1	1	1	0	1	0	0	1
1	1	0	0	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	1

Рис. 6. Проверка "простых" покрытий, реализуемых двумя элементами ($\bar{x}_1 \vee \bar{x}_2$)

Ни один столбец рис. 6 не покрывает своими единицами единиц функции f(abc). Таким образом, двумя элементами (когда один – общий инвертор) заданную функцию реализовать невозможно. Но столбцы $\overline{a \vee b}$, $\overline{b \vee c}$, $\overline{a \vee c}$ – импликанты f(abc)! Они покрывают все единицы f(abc) и совершенно равноценны – каждый покрывает две единицы из четырех, но двумя столбцами не обойтись. Легко, однако, видеть, что дизъюнкция столбцов $\overline{a \vee b}$, $\overline{b \vee c}$, $\overline{a \vee c}$ покрывает единицы функции, но в том-то и дело, что ее нельзя использовать в нашем случае!

Выберем первый столбец $\overline{a \vee b}$, ибо он допустим, и попробуем определить покрытие в нашем базисе для оставшихся единиц:

$$f = \overline{a \vee b} \vee f_{3,5} \vee f_{3,6} \vee f_{3,7} \vee f_{3,8}. \quad (7)$$

То есть необходимы уже два элемента с базисом ($\bar{x}_1 \vee \bar{x}_2$) и теперь следует допустимым образом комбинировать столбцы $\overline{b \vee c}$, $\overline{a \vee c}$. Как комбинировать? Все тем же способом

$$f_{3,5} \vee f_{3,6} \vee f_{3,7} \vee f_{3,8}, \quad (8)$$

так как "левое плечо" ($\bar{x}_1 \vee \bar{x}_2$) уже занято, теперь необходимо получить выражение под "правой" инверсией (рис. 7).

a	b	c	f(abc)	$\overline{b \vee c}$	$\overline{a \vee c}$	$\overline{b \vee c \vee a \vee c}$	$\overline{b \vee c \vee a \vee c}$
0	0	0	0	0	0	1	0
0	0	1	0	0	0	1	0
0	1	0	0	0	0	1	0
0	1	1	1	1	0	1	1
1	0	0	0	0	0	1	0
1	0	1	1	0	1	1	1
1	1	0	0	0	0	1	0
1	1	1	1	1	1	0	1

Рис. 7. Проверка покрытий, реализуемых комбинацией $\overline{b \vee c}$, $\overline{a \vee c}$, $\overline{b \vee c \vee a \vee c}$. (9)

Видно, что простая инверсия столбца

$$\overline{b \vee c \vee a \vee c}$$

приведет к получению требуемого покрытия. Таким образом, выражение

$$\overline{b \vee c \vee a \vee c} \quad (10)$$

также является импликантой и совместно с $\overline{a \vee b}$ покрывает все единицы функции f(abc). Это не что иное, как еще 4 элемента. Итак, получили 6 элементов (24 транзистора):

$$\overline{a \vee b} \vee \overline{b \vee c \vee a \vee c} \quad \therefore \quad (11)$$

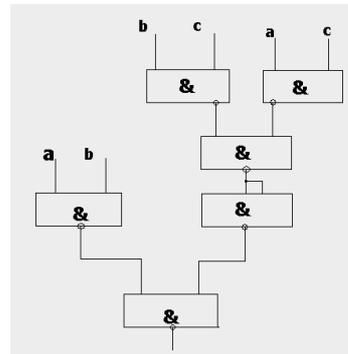


Рис. 8. Схема реализации мажоритарной функции в базисе 2И-НЕ

По существу, рис. 8 – это дерево корнем вниз. Имеется 4 уровня, на каждом уровне реализуется функция ($\bar{x}_1 \vee \bar{x}_2$). Быстродействие такой схемы -4τ , где τ – задержка одного элемента 2И-НЕ (4 транзистора).

Таким образом, минимизация в базисе ($\bar{x}_1 \vee \bar{x}_2$) сводится к нахождению всех импликант вида:

$$(\bar{f}_{i,1} \vee \bar{f}_{i,2}); f \in \{\bar{a}, \bar{b}, \bar{c}, 0, 1, \bar{a}, \bar{b}, \bar{c}, \bar{f}_j, \bar{f}_j\}, \quad (12)$$

т. е. построенных из переменных, их инверсий, констант и уже полученных импликант. При этом решается задача оптимального покрытия импликантами единичных наборов функции.

5. Минимизация логической функции в базисе ($\bar{x}_1 \vee \bar{x}_2$)($\bar{x}_3 \vee \bar{x}_4$)

Возникает вопрос: как же минимизировать функцию рис. 1 в избыточном базисе? Например:

$$(\bar{x}_1 \vee \bar{x}_2)(\bar{x}_3 \vee \bar{x}_4). \quad (13)$$

В этом случае ограничено число конъюнкций до двух, число переменных в конъюнкции

юнкции также не более двух, да еще обязательна инверсия! Искомое выражение представляет собой не ДНФ, а вложенные друг в друга выражения вида

$$(\bar{f}_{i,1} \vee \bar{f}_{i,2})(\bar{f}_{i,3} \vee \bar{f}_{i,4}). \quad (14)$$

То есть представление функции также древовидное, но "дерево" уже другое (рис. 5).

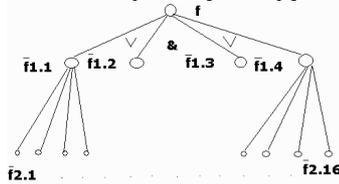


Рис. 9. Древовидное представление функции в базисе $(\bar{x}_1 \vee \bar{x}_2)(\bar{x}_3 \vee \bar{x}_4)$

На рис. 9 индекс подфункции

$$f_{i,j,k,m} \quad (15)$$

указывает ярус дерева (i), позицию (j, = 1...4), верхний ярус (k), к какой позиции верхнего яруса подключается (m), причем $i \neq k$. В таком представлении пара соседних листов объединяется конъюнкцией, между двумя крайними парами – дизъюнкция, а в вершине – инверсия, из которой исходят эти пары.

6. Проверка покрытия заданной функции одним элементом

$$(\bar{x}_1 \vee \bar{x}_2)(\bar{x}_3 \vee \bar{x}_4)$$

Нам необходимо получить всевозможные выражения вида

$$(\bar{f}_{i,1} \vee \bar{f}_{i,2})(\bar{f}_{i,3} \vee \bar{f}_{i,4}); f \in \{a,b,c,0,1\}, \quad (16)$$

т.е. вначале из переменных $f(abc)$ констант 0, 1 и строить соответствующие дополнительные столбцы таблицы истинности (рис. 10).

a	b	c	f(abc)	a	b	c	a∨b	b∨c	a∨c	ab	bc	ac	(a∨b)c	(b∨c)a	(a∨c)b
0	0	0	0	1	1	1	1	1	1	0	0	0	1	1	1
0	0	1	0	1	1	0	1	1	1	0	0	0	1	1	1
0	1	0	0	1	0	1	1	1	1	0	0	1	1	1	0
0	1	1	1	1	0	0	1	0	1	0	0	0	0	0	0
1	0	0	0	0	1	1	1	1	1	0	1	0	1	0	1
1	0	1	1	0	1	0	1	1	0	0	0	0	0	0	0
1	1	0	1	1	0	0	1	0	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0

a	b	c	f(abc)	(a∨c)(b∨c)	(a∨b)(a∨c)	(a∨b)(b∨c)
0	0	0	0	1	1	1
0	0	1	0	1	1	1
0	1	0	0	1	1	1
0	1	1	1	0	1	0
1	0	0	0	1	1	1
1	0	1	1	0	0	1
1	1	0	1	1	0	0
1	1	1	1	0	0	0

Рис. 10. Расширенная таблица истинности функции, представленной рис. 1 для выражений вида

$$(\bar{f}_{i,1} \vee \bar{f}_{i,2})(\bar{f}_{i,3} \vee \bar{f}_{i,4}); f \in \{a,b,c,0,1\}$$

Ни один столбец рис. 10 не покрывает своими единицами единиц функции $f(abc)$. То есть, одним элементом заданную функцию не реализовать.

7. Проверка возможности реализации заданной функции двумя элементами

$$(\bar{x}_1 \vee \bar{x}_2)(\bar{x}_3 \vee \bar{x}_4)$$

В целом, следует проверить, не покрываются ли единицы функции ее переменными, конъюнкциями этих переменных длиной 2 и дизъюнкциями двух таких конъюнкций – для этого требуется 2 элемента. Это можно назвать "простым" или "примитивным" представлением (рис. 11).

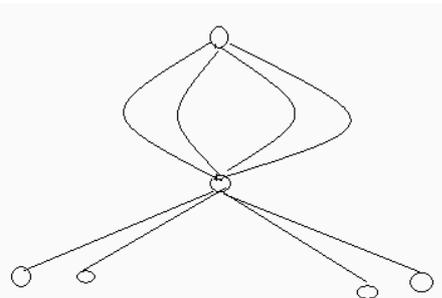


Рис. 11. "Простое" или "примитивное" представление некоторой функции из двух элементов $(\bar{x}_1 \vee \bar{x}_2)(\bar{x}_3 \vee \bar{x}_4)$, один из которых используется как инвертор

Получим рис. 12.

a	b	c	f(abc)	a	b	c	a∨b	b∨c	a∨c	ab	bc	ac
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	1
0	1	0	0	0	1	0	0	0	0	0	1	0
0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	0	0	0	0	0	1	1	0
1	0	1	1	1	0	1	0	0	0	1	1	1
1	1	0	1	1	1	0	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1

a	b	c	f(abc)	(a∨b)c	(b∨c)a	(a∨c)b	(a∨b)(b∨c)	(b∨c)(a∨c)	(a∨b)(a∨c)
0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0
0	1	0	0	0	0	1	0	0	0
0	1	1	1	1	1	1	1	1	0
1	0	0	0	0	1	0	0	0	0
1	0	1	1	1	1	1	0	1	1
1	1	0	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1

Рис.12. Проверка "простых" покрытий, реализуемых двумя элементами $(\bar{x}_1 \vee \bar{x}_2)(\bar{x}_3 \vee \bar{x}_4)$

Ни один столбец рис. 12 не покрывает своими единицами единиц функции $f(abc)$.

Таким образом, двумя элементами (когда один – общий инвертор) заданную функцию реализовать невозможно.

Легко, однако, видеть, что дизъюнкция столбцов $\overline{a \vee b}$, $\overline{b \vee c}$, $\overline{a \vee c}$ покрывает единицы функции, но к сожалению, ее нельзя использовать в нашем случае! Здесь мы пока не использовали возможность инверсий некоторых или всех входных переменных, т.е. строго говоря, это должно выглядеть следующим образом:

$$(\overline{f_{i,1} \vee f_{i,2}})(\overline{f_{i,3} \vee f_{i,4}}); f \in \{a, b, c, 0, 1, \overline{a}, \overline{b}, \overline{c}\}. \quad (17)$$

Полагаем, необходимо рис. 10, 12 еще дополнить всеми вариантами использования инверсий переменных.

8. Проверка возможности реализации заданной функции тремя элементами $(\overline{x_1 \vee x_2})(\overline{x_3 \vee x_4})$

Теперь следует комбинировать столбцы, т.е. получать выражения вида $(\overline{f_{j,1} \vee f_{j,2}})(\overline{f_{j,3} \vee f_{j,4}})$ из полученных ранее столбцов вида

$$(\overline{f_{i,1} \vee f_{i,2}})(\overline{f_{i,3} \vee f_{i,4}}); f \in \{a, b, c, 0, 1, \overline{a}, \overline{b}, \overline{c}\}. \quad (18)$$

Легко увидеть, что конъюнкция \overline{ab} и $\overline{(a \vee b)c}$ покрывает все единицы $f(abc)$.

Значит, можно реализовать ее в виде $(\overline{ab})(\overline{a \vee b})(\overline{c})$ всего тремя элементами! (рис.13):

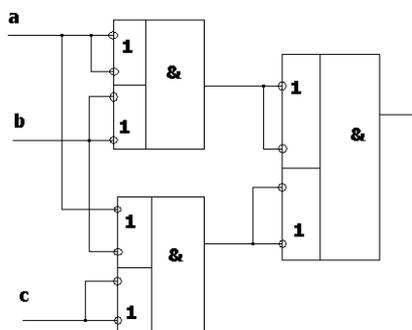


Рис. 13. Реализация $f(abc) = \overline{ab}(a \vee b)(c)$ тремя элементами с базисом $(\overline{x_1 \vee x_2})(\overline{x_3 \vee x_4})$

В каждом элементе $(\overline{x_1 \vee x_2})(\overline{x_3 \vee x_4})$ восемь транзисторов [9, 10], итого 24 транзистора, как и в схеме рис. 6, однако быстродействие в количестве элементов - $2\tau^2$, где τ^2 – быстродействие одного элемента с базисом $(\overline{x_1 \vee x_2})(\overline{x_3 \vee x_4})$, в котором сигнал проходит 2 транзистора, итого быстродействие – 4 транзистора, а в схеме рис. 6 – 8 транзисторов.

Дело в том, что в схемах с базисом $(\overline{x_1 \vee x_2})(\overline{x_3 \vee x_4})$ [11] цепи подключения + и общей шины одинаковы и содержат 2 последовательных транзистора n -МОП, p -МОП, а в элементах с базисом $(\overline{x_1 \vee x_2})$ имеется один n -МОП транзистор, но два последовательных p -МОП.

Заключение

Таким образом, оптимальное покрытие логических функций в традиционном базисе $(\bar{x}_1 \vee \bar{x}_2)$ и в избыточном базисе $(\bar{x}_1 \vee \bar{x}_2)(\bar{x}_3 \vee \bar{x}_4)$ в отличие от "линейного" булева базиса И, ИЛИ, НЕ может быть найдено только в виде дерева соответствующих импликант. Построение всевозможных различных допустимых для данного базиса импликант, а также минимизация соответствующих деревьев представляет собой сложную комбинаторную задачу. Оптимальная реализация мажоритарной функции в избыточном базисе $(\bar{x}_1 \vee \bar{x}_2)(\bar{x}_3 \vee \bar{x}_4)$ равноценна реализации в базисе $(\bar{x}_1 \vee \bar{x}_2)$ по числу транзисторов, но предпочтительна по быстродействию. Целесообразна автоматизация поиска оптимального покрытия.

Список литературы

1. Аляев Ю.А., Тюрин С.Ф. Дискретная математика и математическая логика: учеб. для студентов вузов. М., 2006.
2. Тюрин С.Ф., Аляев Ю.А. Дискретная математика: Практическая дискретная математика и математическая логика: учеб. пособие. М., 2010.
3. Тюрин С.Ф., Громов О.А., Гладышева П.В. Автоматизированный синтез цифровых комбинационных схем в функционально-полном толерантном базисе и в остаточных базисах // Дискуссия теоретиков и практиков. 2010. № 2. С. 181–196.
4. Тюрин С.Ф., Греков А.В., Громов О.А. Программа автоматизированного синтеза цифровых комбинационных схем в функционально-полном толерантном базисе и в нетривиальном остаточном базисе // Народное хозяйство. Вопросы инновационного развития. 2011. № 2. С. 258–263.
5. Чебурахин В.Ф. Математическое моделирование и синтез вычислительных и управляющих логических устройств. URL: <http://www.dslib.net/mat-modelirovanie/cheburahin.html> (дата обращения: 24.10.2013).
6. Лупанов О.Б. О возможностях синтеза схем из произвольных элементов / Сб. ст. по математ. логике и ее приложениям к некоторым вопросам кибернетики: тр. МИАН СССР, 51, Изд-во АН СССР, М., 1958. С. 158–173.
7. Лупанов О.Б. Асимптотические оценки сложности управляющих систем. М.: Изд-во МГУ, 1984.
8. Минимизация функций в произвольном базисе. URL: <http://www.cyberforum.ru/mathematical-logic-sets/thread443495.html> (дата обращения: 29.9.2013).
9. Тюрин С.Ф., Громов О.А. Функционально-полный толерантный элемент. Патент РФ № 2438234. Оpubл. БИ № 36 27.12.2011.
10. Тюрин С.Ф., Громов О.А., Греков А.В. Функционально-полный толерантный элемент. Патент РФ № 2449469. Оpubл. БИ № 12 27.04.2012.
11. Тюрин С.Ф., Громов О.А., Греков А.В., Сулейманов А.А. Функционально-полный толерантный элемент. Патент РФ № 2496227. Оpubл. БИ № 29 20.10.2013.
12. Тюрин С.Ф., Набатов А.В., Громов О.А., Греков А.В., Карлов Д.А. Программируемое логическое устройство. Патент РФ № 2503993. Оpubл. БИ № 1 10.01.2014.
13. Дудкин Ю.П., Тюрин С.Ф., Южаков А.А., Громов О.А. Функционально-полный толерантный элемент. Патент РФ № 2541854. Оpubл. БИ № 5 20.02.2015.
14. Тюрин С.Ф. Программируемое логическое устройство. Патент РФ № 2544750. Оpubл. БИ № 8 20.03.2015.
15. Тюрин С.Ф., Городилов А.Ю., Вихорев Р.В. Программируемое логическое устройство. Патент РФ № 2547229. Оpubл. БИ № 10 10.04.2015.
16. Тюрин С.Ф., Громов О.А. Базисный элемент программируемых логических интегральных схем // Вестник Ижевского государственного технического университета. 2010. № 3. С. 122–126.

Minimization of binary switching functions in bases 2AND-NOT, 2AND-2OR-NOT

S. F. Tyurin, A. S. Prokhorov

Perm National Research Polytechnic University; 29, Komsomolsky prospekt, Perm, 614990, Russia
tyurinsergfe0@yandex.ru; +7 952-320-02-510

This paper considers the approach to minimization of logic functions in the classical 2NOR basis and in the 2OR-2AND-NOT basis. It is shown that search for the optimal coverage of units of the given function is conducted in the form of a tree of implicants of the corresponding basis. Implementation in the redundant basis is equivalent with regard to the number of transistors but preferable in terms of speed.

Keywords: *logic functions minimization; logic element with the redundancy basis; functional complete tolerant logic element (FCTLE); implicant; basis.*