

# Динамика сигналов в цифровых схемах: переход от булевой алгебры к новой алгебре

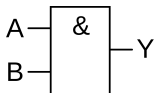
Пчелинцев Александр Николаевич

Тамбовский государственный технический университет

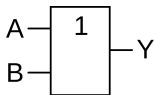
Тамбов, 2019

# Логические вентили

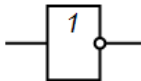
*Конъюнкция* (от лат. conjunctio – союз, связь) – логическая операция, по смыслу максимально приближенная к союзу «и». Синонимы: логическое И, логическое умножение, иногда просто И.



*Дизъюнкция* (от лат. disjunctio – разобщение), логическое сложение, логическое ИЛИ, включающее ИЛИ; иногда просто ИЛИ – логическая операция, по своему применению максимально приближённая к союзу «или» в смысле «или то, или это, или оба сразу».



*Отрицание* (инверсия, от лат. inversio – переворот, логическое «НЕ») в логике – унарная операция, результатом которой 1 тогда, когда на входе 0, 0 тогда, когда на входе 1.



# Таблицы истинности и обозначения операций

Конъюнкция

$A$	$B$	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

Дизъюнкция

$A$	$B$	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

Отрицание

$A$	$\bar{A}$
0	1
1	0

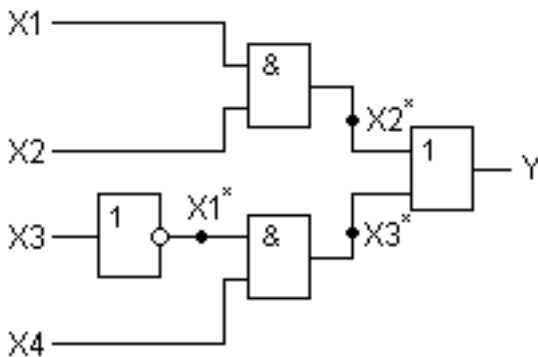
*Дизъюнктивная нормальная форма (ДНФ)* в булевой логике – форма, для которой булева формула имеет вид дизъюнкции конъюнкций булевых переменных. Любая булева формула может быть приведена к ДНФ. Коротко говоря: *ДНФ* называется дизъюнкция простых конъюнкций.

*Совершенной дизъюнктивной нормальной формой (СДНФ)* называется такая дизъюнктивная нормальная форма, у которой в каждую конъюнкцию входят все переменные данного списка (либо сами, либо их отрицания), причём в одном и том же порядке. *Конъюнктивной нормальной формой (КНФ)* называется конъюнкция простых дизъюнкций.

*Совершенной конъюнктивной нормальной формой (СКНФ)* называется такая КНФ, у которой в каждую простую дизъюнкцию входят все переменные данного списка (либо сами, либо их отрицания), причём в одинаковом порядке.

# Комбинационная логическая схема

Комбинационная схема – логическая схема, у которой состояние выхода однозначно определяется набором входных сигналов и не зависит от предыстории функционирования цифрового устройства (отсутствие элементов памяти).



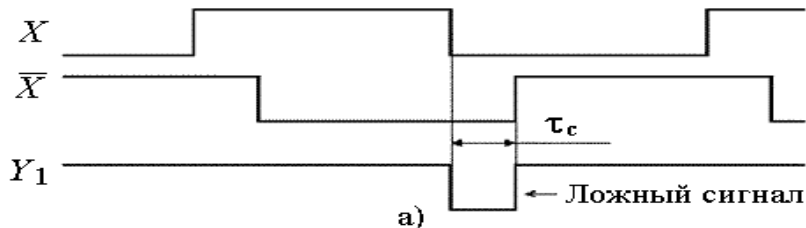
Схема, реализующая булеву функцию  $Y = X_1 \wedge X_2 \vee \overline{X_3} \wedge X_4$ .

*Риск сбоя* – возможность появления на выходе цифрового устройства сигнала, не предусмотренного алгоритмом его работы и способного привести к ложному срабатыванию управляемых устройств, для которых выходные сигналы комбинационной схемы являются входными.

*Статический риск сбоя* – риск сбоя, при котором имеются кратковременные изменения выходного сигнала схемы, который должен был бы оставаться неизменным, т.е. значение на выходе схемы изменяется чётное число раз.

*Динамический риск сбоя* – риск сбоя, при котором значение на выходе схемы изменяется нечётное число раз.

# Пример статического сбоя

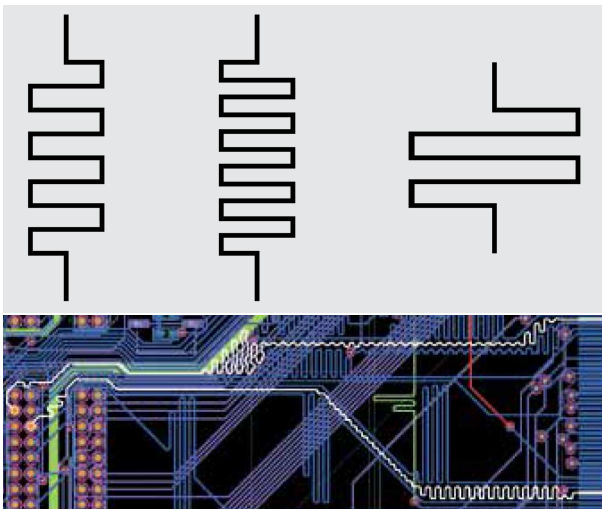


$$Y_1 = X \vee \bar{X}, Y_2 = X \wedge \bar{X}.$$

Для синхронизации работы схемы используется блок задержки, равной времени прохождения критического пути комбинационной схемы, и набор триггеров (схем, способных хранить 1 бит информации) с дополнительными элементами НЕ. Триггер необходим для сохранения состояния, предшествующему статическому сбою. В строгосамосинхронных схемах используется непосредственное определение момента окончания переходного процесса. Поэтому важно определять переходы в таблице истинности, задающей работу схемы, где возникают сбойные состояния.



# Примеры искусственных линий задержки



Зигзагообразные линии задержки.

# Переход от логического представления булевых функций к арифметическому представлению

Рассмотрим числовое множество  $\mathbb{B} = \{0; 1\}$ .

На нем определены операции отрицания, конъюнкции, дизъюнкции и производимые от них (например, импликация, альтернативная дизъюнкция и др.). Выразим эти логические операции через арифметические на множестве  $\mathbb{B}$ :

$$\begin{aligned}\bar{X} &= 1 - X, \\ X \wedge Y &= X \cdot Y, \\ X \vee Y &= X + Y - X \cdot Y.\end{aligned}\tag{1}$$

К выражениям (1) добавим правило

$$X^m = X.\tag{2}$$

Выражение для булевой функции  $Y$ , являющейся функцией входных сигналов схемы, теперь можно упростить по законам арифметических действий и правила (2).

- 1 Для построения аналитического выражения выходного сигнала схемы.
- 2 Минимизация булевых функций.
- 3 Реализация булевых операций через арифметические (советский компилятор/интерпретатор языка Бейсик).

$$X \wedge Y = \overline{\overline{X} \vee \overline{Y}}.$$

$$\begin{aligned}\overline{\overline{X} \vee \overline{Y}} &= 1 - \overline{X} \vee \overline{Y} = \\ &= 1 - (1 - X + 1 - Y - (1 - X) \cdot (1 - Y)) = \\ &= X \cdot Y = X \wedge Y.\end{aligned}$$

# Определение минимизации булевых функций

Минимизировать булеву функцию это значит построить ее кратчайшую или минимальную ДНФ или все кратчайшие или все минимальные ДНФ.

Интерес к кратчайшим и минимальным ДНФ основан на их оптимальности, которая положительно проявляется, по крайней мере, в следующих двух случаях. Во-первых, с оптимальными ДНФ легче оперировать, т.е. вычислять значения функции и подставлять в другие формулы. Во-вторых, оптимальные ДНФ более предпочтительны для построения по ним схем из логических элементов: дизъюнкторов, конъюнкторов и инверторов.

# Простейший пример минимизация булевых функций

$$\begin{aligned}f(X_1, X_2, X_3) &= (X_1 \vee X_2) \wedge (X_1 \vee X_3) = \\&= (X_1 + X_2 - X_1 X_2)(X_1 + X_3 - X_1 X_3) = \\&= X_1^2 + X_1 X_2 - X_1^2 X_2 + X_1 X_3 + X_2 X_3 - X_1 X_2 X_3 - \\&\quad - X_1^2 X_3 - X_1 X_2 X_3 + X_1^2 X_2 X_3 = \\&= X_1 + \cancel{X_1 X_2} - \cancel{X_1 X_2} + \cancel{X_1 X_3} + X_2 X_3 - X_1 X_2 X_3 - \\&\quad - \cancel{X_1 X_3} - \cancel{X_1 X_2 X_3} + \cancel{X_1 X_2 X_3} = \\&= X_1 + X_2 X_3 - X_1(X_2 X_3) = X_1 \vee X_2 \wedge X_3.\end{aligned}$$

# Таблица истинности

Пусть булева функция задана следующей таблицей истинности.

X1	X2	X3	X4	f
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

СКНФ:

$$\begin{aligned} f(X_1, X_2, X_3, X_4) = & (X_1 \vee X_2 \vee X_3 \vee X_4) \wedge (X_1 \vee X_2 \vee X_3 \vee \overline{X_4}) \wedge \\ & \wedge (X_1 \vee X_2 \vee \overline{X_3} \vee X_4) \wedge (X_1 \vee \overline{X_2} \vee X_3 \vee X_4) \wedge \\ & \wedge (\overline{X_1} \vee X_2 \vee X_3 \vee X_4). \end{aligned}$$

# Карта Карно

Поменяем местами последний и предпоследний столбец, чтобы получить карту Карно.

		X3 X4					
		00	01	11	10		
X1 X2	00	0	0	1	0		
	01	0	1	1	1		
11	1	1	1	1			
10	0	1	1	1			

Используем эту карту для минимизации булевой функции.



# Минимизация с помощью карты Карно

X1 X2 \ X3 X4		X3 X4			
		00	01	11	10
X1 X2	00	0	0	1	0
	01	0	1	1	1
	11	1	1	1	1
	10	0	1	1	1

Diagram illustrating the minimization of a Boolean function using a Karnaugh map. The map shows six prime implicants (S1 to S6) circled in different colors:

- S1 (Red):  $X_1 \wedge X_2 \wedge X_3$
- S2 (Green):  $X_1 \wedge X_2$
- S3 (Blue):  $X_3$
- S4 (Yellow):  $X_2 \wedge X_3$
- S5 (Grey):  $X_3 \wedge X_4$
- S6 (Pink):  $X_2 \wedge X_4$

- 1 Все области содержат  $2^n$  клеток;
- 2 Области S1, S2, S3, S4, S5 и S6 максимально большие;
- 3 Получаем рациональный вариант (ДНФ):

$$\begin{aligned}f(X_1, X_2, X_3, X_4) &= S1 \vee S2 \vee S3 \vee S4 \vee S4 \vee S5 \vee S6 = \\ &= X_3 \wedge X_4 \vee X_1 \wedge X_2 \vee X_2 \wedge X_4 \vee X_1 \wedge X_4 \vee X_1 \wedge X_3 \vee X_2 \wedge X_3.\end{aligned}$$

# Минимизация с помощью символьных вычислений в пакете Maxima

```
wdMaxima 12.04.0 [ example.wdm* ]
Файл Правка Cell Maxima Уравнения Алгебра Анализ Упростить Графики Численные расчеты Помощь
[ (i1) d(x1,x2,x3,x4):=x1+x2+x3+x4-x1*x2-x1*x3-x2*x3-x1*x4-x2*x4-x3*x4+x1*x2*x3+x1*x2*x4+x1*x3*x4+x2*x3*x4-x1*x2*x3*x4;
[ (i2) n(x):=1-x^5
[ (i3) e1:expand(d(x1,x2,x3,x4)*d(x1,x2,x3,n(x4))*d(x1,x2,n(x3),x4)*d(x1,n(x2),x3,x4)*d(n(x1),x2,x3,x4));
[ (i4) e2:ev(e1,[x1^2=x1,x2^2=x2,x3^2=x3,x4^2=x4,x1^3=x1,x2^3=x2,x3^3=x3,x4^3=x4,x1^4=x1,x2^4=x2,x3^4=x3,x4^4=x4,x1^5=x1,x2^5=x2,x3^5=x3,x4^5=x4]);
[ (o4) 3 x1 x2 x3 x4 - 2 x2 x3 x4 - 2 x1 x3 x4 + x3 x4 - 2 x1 x2 x4 + x2 x4 + x1 x4 - 2 x1 x2 x3 + x2 x3 + x1 x3 + x1 x2
-->
x1*x2
x1*x3
x2*x3
x1*x4
x2*x4
x3*x4

-x1*x2 * x1*x3 = -x1*x2*x3
-x1*x3 * x2*x3 = -x1*x2*x3
-x1*x2 * x2*x3 = -x1*x2*x3

-x1*x2 * x1*x3 - x1*x3 * x2*x3 - x1*x2 * x2*x3 = -3*x1*x2*x3
x1*x2 * x1*x3 * x2*x3 = x1*x2*x3
```

Арифметическое представление дизъюнкции 4-х переменных:

$$\begin{aligned} X_1 \vee X_2 \vee X_3 \vee X_4 = & X_1 + X_2 + X_3 + X_4 - X_1 X_2 - X_1 X_3 - X_2 X_3 - X_1 X_4 - \\ & - X_2 X_4 - X_3 X_4 + X_1 X_2 X_3 + X_1 X_2 X_4 + X_1 X_3 X_4 + X_2 X_3 X_4 - \\ & - X_1 X_2 X_3 X_4. \end{aligned}$$

<http://maxima.sourceforge.net/project.html>

$$d(x_1, x_2, x_3, x_4) := x_1 + x_2 + x_3 + x_4 - x_1 * x_2 - x_1 * x_3 - x_2 * x_3 -$$

$$x_1 * x_4 - x_2 * x_4 - x_3 * x_4 + x_1 * x_2 * x_3 + x_1 * x_2 * x_4 + x_1 * x_3 * x_4 +$$

$$x_2 * x_3 * x_4 - x_1 * x_2 * x_3 * x_4$$

$$n(x) := 1 - x$$

$$e1: \text{expand}(d(x_1, x_2, x_3, x_4) * d(x_1, x_2, x_3, n(x_4)) *$$

$$d(x_1, x_2, n(x_3), x_4) * d(x_1, n(x_2), x_3, x_4) *$$

$$d(n(x_1), x_2, x_3, x_4))$$

$$e2: \text{ev}(e1, [x_1^2 = x_1, x_2^2 = x_2,$$

$$x_3^2 = x_3, x_4^2 = x_4,$$

$$x_1^3 = x_1, x_2^3 = x_2,$$

$$x_3^3 = x_3, x_4^3 = x_4,$$

$$x_1^4 = x_1, x_2^4 = x_2,$$

$$x_3^4 = x_3, x_4^4 = x_4,$$

$$x_1^5 = x_1, x_2^5 = x_2,$$

$$x_3^5 = x_3, x_4^5 = x_4]);$$

$$> 3 * x_1 * x_2 * x_3 * x_4 - 2 * x_2 * x_3 * x_4 - 2 * x_1 * x_3 * x_4 + x_3 * x_4 -$$

$$2 * x_1 * x_2 * x_4 + x_2 * x_4 + x_1 * x_4 - 2 * x_1 * x_2 * x_3 + x_2 * x_3 +$$

$$x_1 * x_3 + x_1 * x_2$$

$x_1 * x_2$

$x_1 * x_3$

$x_2 * x_3$

$x_1 * x_4$

$x_2 * x_4$

$x_3 * x_4$

$$-x_1 * x_2 * x_1 * x_3 = -x_1 * x_2 * x_3$$

$$-x_1 * x_3 * x_2 * x_3 = -x_1 * x_2 * x_3$$

$$-x_1 * x_2 * x_2 * x_3 = -x_1 * x_2 * x_3$$

$$-x_1 * x_2 * x_1 * x_3 - x_1 * x_3 * x_2 * x_3 -$$

$$x_1 * x_2 * x_2 * x_3 = -3 * x_1 * x_2 * x_3$$

$$x_1 * x_2 * x_1 * x_3 * x_2 * x_3 = x_1 * x_2 * x_3$$

# Ввод в булеву алгебру параметра «время»

Как известно, единичная ступенчатая функция или функция Хевисайда определена на области действительных чисел и возвращает число, принадлежащее множеству  $\mathbb{B}$ :

$$h(t) = \begin{cases} 1, & t \geq 0, \\ 0, & t < 0. \end{cases}$$

Очевидно следующее утверждение: любой сигнал в логической схеме, включающий переход из одного логического состояния в другое, можно представить как сумму-разность функций Хевисайда, взятых с соответствующим аргументом.

Для функции  $h$  имеет место правило

$$\prod_{i=1}^n h(t - \tau_i) = h\left(t - \max_{i=1, n} \tau_i\right), \quad (3)$$

где  $\tau_i$  – момент времени, когда происходит изменение сигнала. Добавим формулу (3) к (1) и (2).

# Задержки в логических элементах схемы

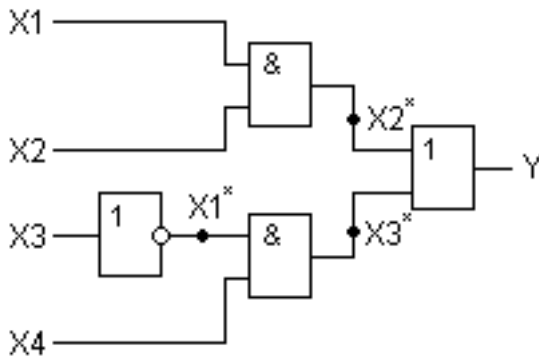
Теперь, зная аналитическое выражение для входных сигналов логической схемы, можно найти вид функции выходного сигнала. Задержку сигнала в логическом элементе удобно моделировать как разность аргумента функции Хевисайда и длительности задержки, т.к. для существующих логических элементов в основном задержки по фронту (переход из 0 в 1) и спаду (переход из 1 в 0) примерно одинаковы.

Таким образом, *любой реальный логический элемент схемы можно моделировать как последовательное соединение звена чистого запаздывания по каждому входу и идеального логического элемента (здесь запаздывание равно длительности задержки).*

- 1 Пусть исследуемая схема функционирует в соответствии с некоторым логическим выражением, задаваемым ДНФ.
- 2 Задаемся функциями входных сигналов, представляющие собой переходы в таблице истинности, выражаемые через функцию Хевисайда.
- 3 Идем по пути следования сигналов в логической схеме с целью поиска выражения для выходного сигнала схемы, применяя правила (1) – (3).
- 4 Если в полученном выражении присутствует разность функций Хевисайда, то мы имеем статический сбой; если присутствует функция Хевисайда с задерживающим аргументом, то сбой динамический.

# Пример анализа логической схемы

Исследуем переход из набора 1111 в набор 1001 ( $15 \rightarrow 9$ ) таблицы истинности для схемы, показанной на следующем рисунке.



Схема, реализующая булеву функцию  $Y = X_1 \wedge X_2 \vee \overline{X_3} \wedge X_4$ .



# Пример анализа логической схемы

$$\begin{aligned}X_1 &= 1, \\X_2(t) &= 1 - h(t - 5), \\X_3(t) &= 1 - h(t - 5), \\X_4 &= 1.\end{aligned}$$

Предположим, что все элементы имеют одинаковые задержки, равные  $\tau$ . Тогда

$$\begin{aligned}X_1^*(t) &= 1 - (1 - h(t - 5 - \tau)) = h(t - 5 - \tau), \\X_2^*(t) &= 1 \cdot (1 - h(t - 5 - \tau)) = 1 - h(t - 5 - \tau), \\X_3^*(t) &= X_1^*(t - \tau) \cdot 1 = h(t - 5 - 2\tau), \\Y(t) &= X_2^*(t - \tau) + X_3^*(t - \tau) - X_2^*(t - \tau) \cdot X_3^*(t - \tau) = \\&= 1 - h(t - 5 - 2\tau) + h(t - 5 - 3\tau) - \\&- (1 - h(t - 5 - 2\tau)) \cdot h(t - 5 - 3\tau) = 1 - h(t - (5 + 2\tau)) + \\&+ h(t - (5 + 2\tau)) \cdot h(t - (5 + 3\tau)) = 1 - h(t - (5 + 2\tau)) + \\&+ h(t - (5 + 3\tau)).\end{aligned}$$

# Пример анализа логической схемы

Таким образом, мы получили статический сбой – в результирующее выражение входит разность функций Хевисайда.

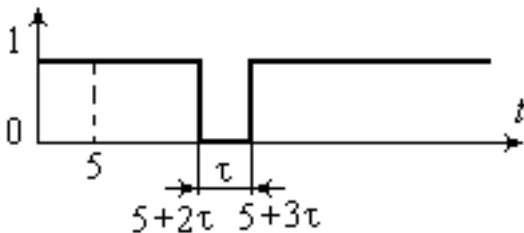
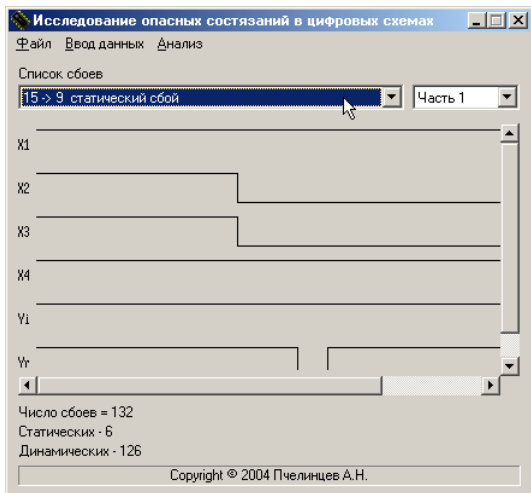


График сигнала  $Y(t)$ .

# Поиск сбойных состояний с помощью программы




Количество сбойных состояний равно 132 из  $A_{2^4}^2 = 2^4(2^4 - 1) = 240$  переходов таблицы истинности (55%).

Пчелинцев А. Анализ цифровых схем на риски сбоя, Habr, 2013.  
<https://habr.com/ru/post/202052/>

 MichaelBorisov 14 ноября 2013 в 03:18   +1 

Спасибо за интересную статью.

Вопрос к вам, автор. Вы могли бы рассказать о том, в каких схемах обычно требуется избавиться от сбоев, и как это достигается? Исходя из моего скромного опыта разработок на FPGA, сбои присутствуют в логических схемах повсюду и спасает от них лишь синхронность работы устройства.

 zhogar 14 ноября 2013 в 08:32   0 

в каких схемах обычно требуется избавиться от сбоев

В безопасных системах автоматии и телемеханики, в медицине, в реакторных системах, в различных АСУ ТП.

как это достигается

Путем способности системы к переходу в предопределенное безопасное состояние в случае своего собственного отказа.

 zhogar 14 ноября 2013 в 08:37   0 

Путем повышения надежности за счет дублирования блоков структурных элементов схемы в сочетании с программной частью (если это MCU).

FPGA – программируемая пользователем вентильная матрица (ППВМ, англ. field-programmable gate array, FPGA).

Microcontroller Unit (MCU) – микроконтроллер (микросхема, содержащая процессор, память и периферийные устройства).

ППВМ – полупроводниковое устройство, которое может быть сконфигурировано проектировщиком цифровых схем после изготовления. ППВМ программируются путём изменения логики работы принципиальной схемы, например, с помощью исходного кода на языке проектирования (типа Verilog), на котором можно описать эту логику работы микросхемы.

ППВМ могут быть модифицированы практически в любой момент в процессе их использования. Они состоят из конфигурируемых логических блоков, подобных переключателям с множеством входов и одним выходом (логические вентили). В цифровых схемах такие переключатели реализуют базовые двоичные операции И, ИЛИ, НЕ и ИСКЛЮЧАЮЩЕЕ ИЛИ.

В большинстве современных микропроцессоров функции логических блоков фиксированы и не могут изменяться. Принципиальное отличие ППВМ состоит в том, что и функции блоков, и конфигурация соединений между ними могут меняться с помощью специальных сигналов, посылаемых схеме.



ППВМ фирмы Altera

РАЗНОЕ

К.т.н. Н.И. Литвиненко (ФГУП НИИ "Аргон")

N.I. Litvinenko

### НУЖНА ЛИ НОВАЯ АЛГЕБРА ВЗАМЕН БУЛЕВОЙ ДЛЯ СИНТЕЗА КОНЕЧНЫХ АВТОМАТОВ ?

### DO WE REALLY NEED NEW TOOLS TO INSTEAD OF BOOLEAN ALGEBRA ?

*Почти десятилетия в отечественной научно-технической литературе дискутируется необходимость замены булевой алгебры, как инструмента синтеза конечных автоматов, новой алгеброй. Основу дискуссии составляет мнение, что булева алгебра не обеспечивает синтез безрисковых конечных автоматов. В статье предпринята попытка показать, как профессиональное использование булевой алгебры синтезирует конечные автоматы без лишних рисков.*

Keywords: Boolean algebra, digital circuits, risk prone circuits.

Беспокойство участников этой дискуссии понятно по двум причинам:

- 1) традиционный синтез конечных автоматов (КА) [3] (рис. 1), не имеет средств, при помощи которых разработчик может убедиться о том, свободна ли от рисков разработанная им схема.
- 2) нет методологии синтеза безрисковых КА.

Шаг 1. Составление уравнения:  $F = ab + \bar{a}b + \bar{b}c + \bar{a}c$ .

Шаг 2. Минимизация уравнения (методом диаграммы Вейча или карт Карно).

Таблица 1

ab → c ↓	00	01	11	10
0	0	0	1	0
1	0	0	1	1

$$F = bc + ab + ac \quad (1)$$

Шаг 3. Реализация уравнения (1) на двучинных логических элементах (ДЛЭС).

Рис. 1

Традиционная последовательность синтеза КА

РАЗНОЕ

Выявить наличие рисков в схеме можно трудоёмким процессом построения временной диаграммы (рис.2). Риск схемы показан точечными отрезками.

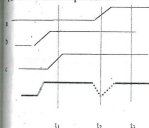


Таблица 2

AB → C ↓	00	01	11	10
0	0	1	0	0
1	0	1	1	1

$$F = \bar{a}b + ac$$

Рис. 2

Поведение КА ( $F = \bar{a}b + ac$ ) во времени

В прямоугольниках показано, на каком временном отрезке рис. 2 какой терм поддерживает значение функции. На рис. 3 показано как устранить риск КА, поведение которого задано табл. 2. Задержки срабатывания ДЛЭС условно вынесены в серые прямоугольники и приняты разными. Поэтому изменение входного вектора (a,b,c) в момент времени  $t_1$  реализуется немедленно в виде внутреннего состояния схемы  $S(t)$ . Реальная задержка вынесена на рисунке в элемент  $\Delta t$ . Эта задержка равна наибольшей величине в  $\{\Delta t_1, \Delta t_2, \Delta t_3\}$ .

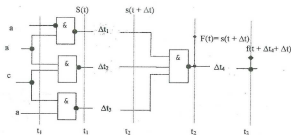


Рис. 3

Устранение риска в КА, который описан в табл. 2

РАЗНОЕ

Вводим два понятия:

-схема находится в устойчивом (стабильном состоянии), если  $s(t+\Delta t) = S(t)$ ;

-схема находится в неустойчивом (переходном) состоянии, если  $s(t+\Delta t) \subset S(t)$ .

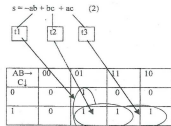
Аналогичные определения действительны и для  $F(t) \Rightarrow s(t+\Delta t)$ , и для  $f(t + \Delta t_1 + \Delta t_2)$ .

После изменения входного вектора схема приходит в устойчивое состояние в моменты времени, обозначенные на рис. 3 как  $s(t+\Delta t)$  и  $f(t + \Delta t_1 + \Delta t_2)$ . Эту стабильность обеспечивают изолированные друг от друга термы ( $\sim ab$ ) и ( $ac$ ), табл. 2.

В промежутках времени  $[t_1, t_2]$  и  $[t_2, t_3]$  нет терма, обеспечивающего конкретное состояние схемы. Поэтому на рис. 3 появился дополнительный терм на ДЛЭ ( $bc$ ), который выполняет функцию защиты КА от "риска", табл. 3. В этой таблице в овалах показаны термы, которые реализуют значения выходной функции во времена  $t_1, t_2, t_3$ .

**Правило 1.** Для устранения риска в КА термы, образующие дизъюнкцию уравнения КА на карте Карно (диаграмме Вейча), должны быть связаны между собой.

Таблица 3



Следовательно, КА, поведение которого описано в табл. 1, не содержит явлений риска, т.к. все его термы связаны. Поскольку диаграммы Вейча и карты Карно имеют естественное ограничение на число переменных, для целей анализа на наличие риска в схеме можно использовать неориентированный граф. В таком графе вершины представляют собой переменные, а рёбра и циклы – термы

РАЗНОЕ

(конъюнкции), которые в дизъюнктивном представлении описывают уравнение КА.

На рис. 4 КА, уравнение (1), представлен в виде графа. Рёбра этого графа эквивалентны термам табл. 1.

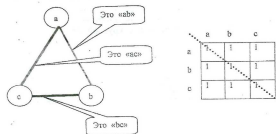


Рис. 4

Граф и матрица инцидентий вершин для КА по формуле (1)

**Правило 2.** Для устранения риска в КА должен быть связан граф, подграфы которого образуют конъюнкции дизъюнкции булева уравнения КА.

Разработчик может проверить схему на наличие (отсутствие) в ней ситуаций риска, построив для синтезируемого КА граф или матрицу инцидентий вершин графа, как показано на рис. 4.

Матрица инцидентий вершин графа, помимо целей анализа, может служить в качестве данных для целей минимизации исходного уравнения и порождения дополнительных термов, при помощи которых достигается связность графа.

### Литература

1. Воробьев Н.В. Риски сбоя в комбинационных схемах – "Chip News", 1998, № 2, p. 26-30.
2. Пчелинцев А. Размышления на тему «Новая алгебра». – "Chip News", 2005, № 9, p.64-65.
3. Булева алгебра и конечные автоматы. Пер. с франц. Под ред. П.П. Пархоменко. М., Мир, 1969.

Статья поступила 11.12.2009



- 1 Артюхов В.Л., Кондратьев В.Н., Шалыто А.А. Реализация булевых функций арифметическими полиномами // Автоматика и телемеханика, 1988, № 4. – С. 138-147.
- 2 Heidtmann K.D. Arithmetic spectrum applied to fault detection for combinational networks // IEEE Trans. on Comp., 1991, Vol. 40, Iss. 3, pp. 320-324.
- 3 Воробьев Н.В. Риски сбоя в комбинационных схемах // Chip News: Инженерная микроэлектроника, 1998. № 2. – С. 26-30.
- 4 Воробьев Н.В. Методы анализа комбинационных схем на риски сбоя // Chip News: Инженерная микроэлектроника, 1998. № 3. – С. 42-44.
- 5 Пчелинцев А.Н., Касьянов А.Н. Анализ опасных состояний в комбинационных цифровых схемах при автоматизированном проектировании // Вестник ТГТУ, 2005. Т. 11. № 2. – С. 368-371.
- 6 Пчелинцев А. Размышления на тему «Новая алгебра» // Chip News: Инженерная микроэлектроника, 2005. № 9. – С. 64-65; англ. пер.: Pchelintsev A.N. The failure risk analysis of digital circuits, 2013, arXiv: 1303.4051. <https://arxiv.org/abs/1303.4051>
- 7 Литвиненко Н.И. Нужна ли новая алгебра взамен булевой для синтеза конечных автоматов? // Радиопромышленность, 2010. № 2. – С. 152-155

СПАСИБО ЗА ВНИМАНИЕ!