Г л а в а 1

Булевы формулы и булевы функции. Классификация, табулирование, свойства

Функция, у которой ее значения и значения аргументов двоичны, называется б у л е в о й.

Б у л е в ы ф у н к ц и и (БФУ) задаются в виде т а б л и ц и с т и н н о с т и (ТИ).

В настоящей работе, если это не оговаривается особо, рассматриваются одиночные, полностью определенные БФУ n переменных, заданные на 2^n входных наборах. Основной метрикой БФУ является число переменных n.

Аналитическая запись БФУ в виде формулы, использующей логические операции (базис формулы), называется б у л е в о й ф о р м у л о й (БФ).

В настоящей работе рассматриваются формулы в различных базисах, но основное внимание уделяется формулам в базисе И, ИЛИ, НЕ. Рассматриваются также формулы в таком нетрадиционном для дискретной математики [1] базисе, как И, ИЛИ, НЕРАВНОЗНАЧНОСТЬ, НЕ, который является избыточным по критерию функциональной полноты [2], но весьма удобным для сокращения длины формул и позволяющим легко декомпозировать формулы вследствие того, что для указанных двухместных операций выполняется сочетательный закон [3].

Зафиксируем базис логических операций. При этом основной метрикой БФ является число (h) символов переменных и их инверсий в правой ее части, которые будем называть буквами. Булевы формулы в фиксированном базисе из h букв могут быть разделены на два класса: бесповторные, для которых $h = n$, и повторные, для которых $h > n$.

Повторная формула в одном базисе может быть бесповторной в другом базисе, и наоборот. Например, повторная формула $f = !x_1 x_2 \vee x_1 !x_2$ в базисе {&, ∨, !}, для которой $n = 2$, $h = 4$, является бесповторной в другом базисе: $f = x_1 \oplus x_2$. Здесь «!» — символ операции НЕ (инверсии).

Бесповторные булевы формулы (ББФ) в базисе {&, ∨, !} имеют важное значение, так как они

— широко используются на практике, и в том числе в алгоритмах управления судовыми техническими средствами [3—5];

— изоморфны бесповторным параллельно-последовательным контактным схемам [5] и дихотомическим деревьям [6];

— их сравнительно «немного» [3];

— любая повторная формула из h букв может быть получена из однотипной с ней ББФ из h букв за счет переобозначений переменных, но не наоборот [3].

1.1. Классификация бесповторных булевых формул в базисе И, ИЛИ, НЕ

***PN*-классификация.** Классификация булевых формул по типам относительно двух операций эквивалентности (перестановка переменных и замена переменных их инверсиями) называется *PN*-классификацией (от английских слов «permutation» — перестановка и «negation» — отрицание).

Если в качестве представителей типов ББФ в базисе {&, ∨, !} выбрать положительно монотонные формулы, то им изоморфны соответствующие представители типов параллельно-последовательных контактных схем (П-схем) из нормально разомкнутых контактов.

В [5] выполнено подробное исследование этого класса схем и получена производящая функция для подсчета количества типов таких схем из h контактов. Количество типов T_1 этих схем и соответственно *PN*-типов ББФ при $h \leqslant 10$ приведено в табл. 1.1.

Таблица 1.1

h	1	2	3	4	5	6	7	8	9	10
T_1	1	2	4	10	24	66	180	522	1532	4624

Имея при h = const всех представителей типов, за счет перестановки входных переменных и расстановки инверсий над одиночными символами переменных могут быть получены все ББФ из h букв, из которых, в свою очередь, за счет отождествлений могут быть получены все БФ из h букв в рассматриваемом базисе.

В [3] было предложено для перечисления *PN*-типов ББФ использовать структурные полиномы, в которых числа изображают число букв в конъюнкциях, а умалчиваемые знаки произведения (в скобочных формах) и знаки суммы соответствуют операциям конъюнкции и дизъюнкции. При этом отметим, что эти полиномы отражают только структуру формул и не предназначены для вычислений по ним. Например, формуле $f = (x_1 x_2 x_3 \vee !x_4 x_5) x_6 x_7$ соответствует полином (3 + 2)2.

Табулирование полиномов при $h \leqslant 8$ выполнено автором и приведено в [7]. Например, при h = 3 таких полиномов четыре: 3, $2+1$, $1+1+1$, $(1+1)1$.

Среди всех *PN*-типов ББФ могут быть выделены *PN*-типы, соответствующие бесповторным бесскобочным формулам — бесповторным дизъюнктивным нормальным формам (ДНФ). Количество *PN*-типов T_2 таких формул при $h \leqslant 8$ приведено в табл. 1.2.

Таблица 1.2

h	1	2	3	4	5	6	7	8
T_2	1	2	3	5	7	11	15	22

Эти типы совпадают с числом разбиений числа h на слагаемые, для подсчета которых Рамануджаном была получена рекуррентная формула [8].

***N*-классификация**. В [8] приведены производящие функции для определения числа П-схем, снабженных различными ярлыками (различные индексы в обозначениях контактов). Эта классификация соответствует *N*-классификации ББФ. Число *N*-типов ББФ (T_3) при $h \leqslant 7$ приведено в табл. 1.3 [8].

Таблица 1.3

h	1	2	3	4	5	6	7
T_3	1	2	8	52	472	5504	78146

При $h = 3$ представителями *N*-типов ББФ являются следующие формулы: $x_1 x_2 x_3$, $x_1 x_2 \vee x_3$, $x_1 x_3 \vee x_2$, $x_2 x_3 \vee x_1$, $x_1 \vee x_2 \vee x_3$, $(x_1 \vee x_2) x_3$, $(x_1 \vee x_3) x_2$, $(x_2 \vee x_3) x_1$, из которых за счет расстановки инверсий над одиночными символами переменных могут быть получены все ББФ при h = 3.

***P*-классификация.** При *P*-классификации ББФ различают формулы, отличающиеся лишь структурой и расстановкой инверсий над одиночными символами переменных.

В [3] определено число *P*-типов ББФ (T_4) при $h \leqslant 6$ (табл.1.4).

Таблица 1.4

h	1	2	3	4	5	6
T_4	2	6	20	80	340	1582

При h = 2 представителями *P*-типов ББФ являются следующие формулы: $!x_1!x_2$, $!x_1x_2$, x_1x_2, $!x_1 \vee !x_2$, $!x_1 \vee x_2$, $x_1 \vee x_2$, из которых за счет перестановки переменных могут быть получены все ББФ при h = 2.

Число бесповторных булевых формул. В [3] при $h \leqslant 10$ определено число ББФ. В табл. 1.5 приведено их количество (T_5) при $h \leqslant 6$.

Таблица 1.5

h	1	2	3	4	5	6
T_5	2	8	64	832	15104	352265

Перечислим ББФ при $h = 2$: $!x_1!x_2$, $!x_1x_2$, $x_1!x_2$, x_1x_2, $!x_1 \vee !x_2$, $!x_1 \vee x_2$, $x_1 \vee !x_2$, $x_1 \vee x_2$.

В [3] установлен весьма интересный факт, состоящий в том, что

$$B(h) = \frac{\log T_5}{\log T_1}$$

приблизительно равно трем (по крайней мере при $h \leqslant 9$). Этот показатель может использоваться в качестве инварианта класса ББФ в рассматриваемом базисе. Отметим, что здесь и в дальнейшем, если это не оговорено особо, применяются логарифмы по основанию два.

***NPN*-классификация.** Вводя третью операцию эквивалентности — замену формулы ее отрицанием, из представителей *PN*-типов ББФ можно получить представителей *NPN*-типов этих формул. Так как при $h \geqslant 2$ среди представителей *PN*-типов ББФ нет самодвойственных, то количество *NPN*-типов ББФ (T_6) определяется при $h \geqslant 2$ делением значений, приведенных в табл. 1.1, на два (табл. 1.6).

Таблица 1.6

h	1	2	3	4	5	6	7	8
T_6	1	1	2	5	12	33	90	261

1.2. Деревья из двухвходовых элементов

1.2.1. Подсчет числа неизоморфных деревьев

В [3, 9] приведено рекуррентное соотношение для подсчета числа неизоморфных деревьев с h входами из $h-1$ двухвходовых элементов, которое имеет следующий вид:

$$R = \begin{cases} \displaystyle\sum_{i=1}^{[h/2]} R_i R_{h-i} & \begin{array}{ll} \text{при } h = 2k+1, & k \geqslant 1; \\ \text{при } h = 2k, & k = 1, 2, 3; \end{array} \\ \displaystyle\sum_{i=1}^{h/2} R_i R_{h-i} & \text{при } h = 2k, \quad k \geqslant 4, \end{cases}$$

где $[h/2]$ — символ округления числа $h/2$ до ближайшего меньшего целого.

Значения R при $h \leqslant 12$, впервые полученные в [10], указаны в табл. 1.7.

Таблица 1.7

h	1	2	3	4	5	6	7	8	9	10	11	12
R	1	1	1	2	3	6	11	23	46	98	207	451

На рис.1.1 приведено дерево, являющееся линейным каскадом при $h = 3$, а на рис. 1.2 — деревья при $h = 5$.

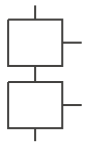

Рис. 1.1

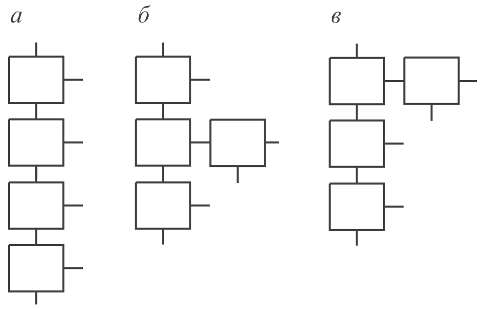

Рис. 1.2

Рассмотренные деревья могут быть использованы в качестве более «крупной» классификации ББФ. Интересно отметить, что число типов в этой классификации, например для ДНФ из пяти букв, не может быть уменьшено.

1.2.2. Покрывающие деревья

Покрывающим называется дерево, которое за счет исключения из него одного или нескольких двухвходовых элементов позволяет получить любое дерево с h входами.

В табл. 1.8 при $h \leqslant 8$ приведено число элементов в покрывающих деревьях.

Таблица 1.8

h	2	3	4	5	6	7	8
Э	1	2	4	5	8	10	15

На рис.1.3 приведено покрывающее дерево при $h = 5$.

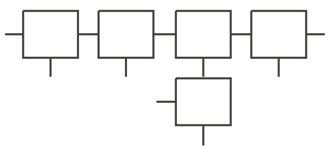

Рис. 1.3

1.2.3. Модули с простой структурой, универсальные в классе формул

Покрывающие деревья являются основой для построения модулей с простой структурой, универсальных в классе формул. Если при h = const в соответствующем покрывающем дереве каждый элемент заменить трехвходовым мажоритарным элементом (МЭ), то получится модуль, универсальный в классе формул в базисе $\{\&, \vee, !\}$ из h букв. При этом предполагается, что прямые и инверсные выходы источников информации равнодоступны.

На рис. 1.4 приведен модуль, универсальный в рассматриваемом классе формул из пяти букв (5-универсальный модуль), настроенный в качестве примера на реализацию ББФ $f = (x_1!x_2 \vee x_3)(x_4 \vee x_5)$.

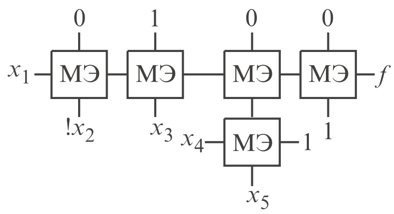

Рис. 1.4

Достоинства рассмотренных модулей состоят в простых структуре и настройке, а также в однотипности элементов, из которых они построены, а недостаток — в большом числе внешних выводов. Так, модуль на рис. 1.4 имеет 12 выводов — 11 входов и один выход.

Несколько более сложный метод построения модулей с простой структурой, основанный на наложении деревьев из элементов И и ИЛИ с различным числом входов и применении *NPN*-классификации, рассмотрен в [11]. Этот метод позволил, в частности, построить 5-универсальный модуль с 11 выводами — девятью входами и двумя выходами.

В заключение раздела отметим, что еще более сложный метод, излагаемый в гл. 7, позволяет за счет увеличения элементной сложности уменьшить число внешних выводов и построить, в частности, 5-универсальный модуль с девятью выводами [12].

1.3. *PN*-классификация бесповторных формул в базисе И, ИЛИ, НЕРАВНОЗНАЧНОСТЬ, НЕ

В [13] установлено, что число *PN*-типов ББФ в этом базисе из h букв, реализуемых линейным каскадом из $h-1$ двухвходовых элементов И, ИЛИ, НЕРАВНОЗНАЧНОСТЬ, определяется соотношением

$$T_7 = F_{2h+1},$$

где F_{2h+1} — член с номером $2h + 1$ ряда Фибоначчи, начинающегося с нуля [14].

В [3] отмечено, что при $h \geqslant 3$

$$T_7 < 3^{h-1}.$$

В [14] найдено при $h \leqslant 6$ число *PN*-типов формул рассматриваемого класса T_8, а в [3] выполнено их табулирование при $h \leqslant 4$.

В [9] приведена верхняя оценка T_9 числа *PN*-типов этих формул для произвольных h, которая имеет следующий вид:

$$T_9 = RT_7.$$

В табл. 1.9 приведены значения T_7, T_8 и T_9 при $h \leqslant 8$.

Таблица 1.9

h	1	2	3	4	5	6	7	8
T_7	1	3	8	21	55	144	377	987
T_8	1	3	8	29	101	405	?	?
T_9	1	3	8	42	165	864	4147	22791

1.4. Свойства бесповторных формул в базисе И, ИЛИ, НЕ

1.4.1. Ранг функций

Число единиц в столбце значений булевой функции называется ее р а н г о м.

Приведем несколько утверждений относительно рангов функций:

— если БФУ имеет нечетный ранг, то она существенно зависит от всех своих переменных [15];

— ранг функции, несущественно зависящей от некоторых переменных, — четное число;

— ранг ББФ в базисе $\{\&, \vee, !\}$, существенно зависящей от всех переменных, нечетен, но не наоборот, так, например, повторная БФ $f = !x_1!x_2x_3 \vee x_1x_2$ имеет ранг три [3];

— существуют различные *PN*-типы ББФ в этом базисе, имеющие одинаковые ранги, например формулы $f_1 = x_1x_2 \vee x_3x_4$ и $f_2 = x_1(x_2 \vee x_3 \vee x_4)$ имеют ранг семь;

— самодвойственные функции, существенно зависящие от n переменных, не реализуются ББФ в указанном базисе, так как функции этого класса имеют четный ранг, равный 2^{n-1} [9].

1.4.2. Вычисление числа единиц по формуле

Таблично-аналитический метод подсчета числа единиц для бесповторных булевых формул предложен в [3].

В [16] для решения этой задачи используется одна из аксиом теории вероятностей. Так как в таблице истинности $P(x_i = 0) = P(x_i = 1) = 1/2$, определим вероятность того, что БФУ принимает значение, равное единице, в виде дроби

$$P = \frac{r}{2^n},$$

в которой числитель определяет ранг функции. Таким образом, «вероятностный» метод позволяет определить невероятностную характеристику БФУ.

В [16] было предложено преобразовывать ББФ в базисе $\{\&, \vee, !\}$ в ББФ в базисе И—НЕ, избавляясь тем самым от неортогонализованных операций ИЛИ, нарушающих аксиому о независимости событий. По преобразованной формуле записывается формула для вычисления искомой вероятности, что весьма трудоемко.

Более простой метод, использующий эту аксиому, был предложен автором и А.Ю.Еремеевым в [17], который позволяет применять ее непосредственно к заданной ББФ. При этом для подформулы вида $f_1 = x_i x_j$ записываются соотношения

$$P_{ij} = P_i P_j\,; \quad Q_{ij} = 1 - P_{ij}\,,$$

а для подформулы $f_2 = x_i \vee x_j$ — соотношения

$$Q_{ij} = Q_i Q_j\,; \quad P_{ij} = 1 - Q_{ij}\,,$$

а затем на основе этих соотношений по заданной ББФ определяются значения P и r.

Пример 1.1. Определить ранг функции, заданной ББФ $f = x_1x_2 \vee \vee x_3x_4$.

Выполним следующие вычисления:

$$P_{12} = P_1 P_2 = \frac{1}{4}; \quad Q_{12} = 1 - P_{12} = \frac{3}{4};$$

$$P_{34} = P_3 P_4 = \frac{1}{4}; \qquad Q_{34} = 1 - P_{34} = \frac{3}{4};$$

$$Q = Q_{12} Q_{34} = \frac{9}{16}; \quad P = 1 - Q = \frac{7}{16}.$$

Таким образом, $r = 7$.

Еще более простой метод решения сформулированной задачи может быть предложен на основе построения по ББФ, за счет ортогонализации всех дизъюнкций путем «домножения», ортогональной скобочной формулы (ОСФ). Для уменьшения трудоемкости метода ОСФ должна, по возможности, содержать минимальное число букв. По ОСФ за счет замены $x_{\rm i}$ на $P_{\rm i}$, $!x_i$ на Q_i ($P_i = Q_i = 1 / 2$) строится искомая формула для подсчета вероятности числа единиц.

Пример 1.2. Решить задачу, рассмотренную в предыдущем примере.

Построим ОСФ:

$$f = x_1 x_2 \vee x_3 x_4 = x_1 x_2 \vee !(x_1 x_2)\, x_3 x_4 = x_1 x_2 \vee (!x_1 \vee !x_2)\, x_3 x_4 =$$
$$= x_1 x_2 \vee (!x_1 \vee x_1 !x_2)\, x_3 x_4.$$

При этом

$$P = P_1 P_2 + (Q_1 + P_1 Q_2) P_3 P_4 =$$
$$= \left(\frac{1}{2}\right)\left(\frac{1}{2}\right) + \left(\left(\frac{1}{2}\right) + \left(\frac{1}{2}\right)\left(\frac{1}{2}\right)\right)\left(\frac{1}{2}\right)\left(\frac{1}{2}\right) = \frac{7}{16}.$$

Следовательно, $r = 7$.

Пример 1.3. Подсчитать число единиц в таблице истинности для ББФ

$$f = \Big(\Big((x_1 \vee x_2)\, x_3 \vee x_4\Big)\, x_5 \vee x_6\Big)\, x_7 \vee x_8 .$$

Построим ОСФ с минимальным числом букв:

$$f = \Big(\Big((x_1\, !x_2 \vee x_2)\, x_3\, !x_4 \vee x_4\Big)\, x_5\, !x_6 \vee x_6\Big)\, x_7\, !x_8 \vee x_8 .$$

При этом

$$P = \Big(\Big((P_1 Q_2 + P_2) P_3 Q_4 + P_4\Big) P_5 Q_6 + P_6\Big) P_7 Q_8 + P_8 = \frac{171}{256}.$$

Таким образом, $r = 171$.

Для упрощения построения ОСФ для повторных в рассматриваемом базисе формул ортогонализацию дизъюнкций в них будем проводить не только за счет «домножения», но и за счет проведения разложения Шеннона (разд.3.1.1) по повторным переменным.

С целью получения в формуле для вычисления P величины равной 2^n при необходимости ОСФ корректируется: подформулы в ней умножаются на выражения вида $!x_i \vee x_i$, где x_i — отсутствующая в подформуле переменная, которая не позволяет получить в ортогональной ДНФ (ОДНФ) хотя бы одну конъюнкцию, состоящую из n букв. При этом отметим, что в формуле для подсчета P выражению $!x_i \vee x_i$ соответствует выражение $Q_i + P_i$, записываемое в окончательной форме не как единица, а как дробь $2/2$.

Пример 1.4. Определить ранг БФУ, заданной формулой $f = (x_1 \vee x_2) x_3 \vee x_1 x_2$.

Выполним разложение Шеннона по переменной x_3:

$$f = !x_3 x_1 x_2 \vee x_3 (x_1 \vee x_2).$$

Построим ОСФ:

$$f = !x_3 x_1 x_2 \vee x_3 (x_1 \vee !x_1 x_2).$$

Если по ОСФ построить ОДНФ, то в ней две конъюнкции содержат по три буквы, и поэтому корректировка ОСФ не требуется. При этом

$$P = Q_3 P_1 P_2 + P_3 (P_1 + Q_1 P_2) = \frac{4}{8}.$$

Следовательно, $r = 4$.

Пример 1.5. Определить ранг БФУ, заданной формулой $f = (!x_1 \oplus x_2) \vee (x_3 \oplus !x_4)$.

Запишем БФУ в виде $f = !x_1 !x_2 \vee x_1 x_2 \vee !x_3 !x_4 \vee x_3 x_4$ и разложим ее по переменной x_1:

$$f = !x_1 (!x_2 \vee !x_3 !x_4 \vee x_3 x_4) \vee x_1 (x_2 \vee !x_3 !x_4 \vee x_3 x_4).$$

Остаточные повторные формулы ортогонализуем «домножением»:

$$f = !x_1 (!x_2 \vee x_2 !x_3 !x_4 \vee x_2 x_3 x_4) \vee$$
$$\vee x_1 (x_2 \vee !x_2 !x_3 !x_4 \vee !x_2 x_3 x_4).$$

ОДНФ в этом случае содержит конъюнкции из четырех букв, и поэтому корректировка ОСФ не требуется. При этом

$$P = Q_1 (Q_2 + P_2 Q_3 Q_4 + P_2 P_3 P_4) +$$
$$+ P_1 (P_2 + Q_2 Q_3 Q_4 + Q_2 P_3 P_4) = \frac{12}{16}.$$

Следовательно, $r = 12$.

Пример 1.6. Определить ранг БФУ, заданной формулой $f = !x_1 x_2 \vee !x_1 x_3 \vee x_2 x_4$.

Выполним разложение Шеннона по переменной x_2:

$$f = !x_2\,!x_1 x_3 \vee x_2\,(!x_1 \vee x_4).$$

Построим ОСФ:

$$f = !x_2\,!x_1 x_3 \vee x_2\,(!x_1 \vee x_1 x_4).$$

ОДНФ этой функции не содержит конъюнкций из четырех букв, и поэтому откорректируем ОСФ, например следующим образом:

$$f = !x_2\,!x_1 x_3\,(!x_4 \vee x_4) \vee x_2\,(!x_1 \vee x_1 x_4).$$

При этом

$$P = Q_2 Q_1 P_3\,(!Q_4 + P_4) + P_2\,(Q_1 + P_1 P_4) = \frac{8}{16}.$$

Следовательно, $r = 8$.

1.4.3. Свойства бесповторных пороговых формул

Для *PN*-типов бесповторных пороговых формул (БПФ) могут быть выбраны такие представители, у которых все единицы в столбцах значений расположены подряд, начиная со «дна» ТИ.

Инвариантами класса *PN*-типов БПФ, существенно зависящих от n переменных, являются нечетные числа, соответствующие их рангам: 1, 3, 5, ..., $2^n - 1$.

Число *PN*-типов БПФ, существенно зависящих от n переменных, при $n \geqslant 1$ равно 2^{n-1}.

Представитель любого *PN*-типа БПФ из h букв реализуется каскадом из $h - 1$ двухвходовых элементов И и ИЛИ, каскадом из $h - 1$ МЭ и каскадом из двухвходовых элементов И—НЕ (ИЛИ—НЕ), в котором число элементов определяется соотношением [9]:

$$h - 1 \leqslant Э \leqslant 2(h - 1).$$

В [9] показано, что если функция $\Phi(x_1, ..., x_{n-1})$ реализуется бесповторной пороговой формулой, то этому классу принадлежит также формула $f(x_1, ..., x_n) = \Phi(x_1, ..., x_{n-1}) * x_n$, где знак $*$ соответствует операциям $\{\&, \vee\}$.

Отсюда следует, что произвольная (в том числе скобочная) БПФ может быть представлена в виде:

$$f(x_1, \ldots, x_n) = \Big(\Big(\ \ldots (x_1 * x_2) * x_3 \ldots * x_{n-2}\Big) * x_{n-1}\Big) * x_n.$$

Бесповторные пороговые формулы n переменных с наибольшей суммой весов и порога имеют вид [9]:

$$f_1(x_1,\dots,x_n)=$$
$$=\Big(\dots(x_1x_2\vee x_3)x_4\vee x_5\Big)x_6\vee\dots\vee x_{n-1}\Big)x_n \qquad \text{при } n=2k;$$
$$f_2(x_1,\dots,x_n)=$$
$$=\Big(\dots(x_1\vee x_2)x_3\vee x_4\Big)x_5\vee\dots\vee x_{n-1}\Big)x_n \qquad \text{при } n=2k+1.$$

Пороговая реализация функций f_1 и f_2 имеет вид: $(p_1, p_2, \dots, p_n; T)$, где $p_i=F_i$; $T=F_{n+1}$; p_i — вес i-й переменной, T — порог функции; F_i — i-е число ряда Фибоначчи, $F_1=F_2=1$. Этот результат был независимо получен в [9, 18].

Для БПФ f_1 и f_2 справедливо соотношение [9]

$$T+\sum_{i=1}^{n}p_i=F_{n+3}-1.$$

Верхние оценки параметров БПФ удовлетворяют соотношениям:

$$\max p_i\leqslant F_n,\ \max T\leqslant F_{n+1},$$

где максимум берется по всем формулам класса.

Простой метод определения весов и порога для произвольной БПФ по построенному для нее линейному бинарному графу с максимальным числом путей изложен в разд. 15.5. При этом значение пометки Акерса входа i-й условной вершины равно p_i, а число нулевых путей в графе равно T. Этот метод впервые описан в [19].

1.5. Отношения покрытия

Будем считать, что функция $Q_2(X)$ покрывает функцию $Q_1(X)$, если на каждом входном наборе X выполняется соотношение $Q_2\geqslant Q_1$. При этом отношение покрытия будем обозначать так же ($Q_2\geqslant Q_1$).

Пусть имеется выражение

$$f=Q_0\,!\Phi_1!\Phi_2\vee Q_1\,!\Phi_1\Phi_2\vee Q_2\Phi_1!\Phi_2\vee Q_3\Phi_1\Phi_2.$$

Вынесение за скобки незначительно уменьшает число букв и не исключает инверсий функций Φ_j:

$$f=!\Phi_1(Q_0!\Phi_2\vee Q_1\Phi_2)\vee\Phi_1(Q_2!\Phi_2\vee Q_3\Phi_2).$$

1. В [3] показано, что если выполняются отношения

$$Q_0 \leqslant Q_1\,,\ Q_0 \leqslant Q_2\,,\ Q_0 \leqslant Q_3\,,\ Q_1 \leqslant Q_3\,,\ Q_2 \leqslant Q_3\,,$$

то выражение для f даже в ДНФ может быть существенно упрощено и оно не будет содержать инверсий функций Φ_j:

$$f = Q_0 \vee Q_1\Phi_2 \vee Q_2\Phi_1 \vee Q_3\Phi_1\Phi_2\,.$$

Пример 1.7. Если $Q_0 = 0,\ Q_1 = !x_1x_2\,,\ Q_2 = x_1\,!x_2\,,\ Q_3 = 1,$ то приведенные соотношения выполняются и

$$f = !x_1x_2\Phi_2 \vee x_1\,!x_2\Phi_1 \vee \Phi_1\Phi_2 = (!x_1x_2 \vee \Phi_1\,)\Phi_2 \vee x_1\,!x_2\Phi_1\,.$$

2. При выполнении отношений вида

$$Q_0 \leqslant Q_1 \leqslant Q_2 \leqslant Q_3$$

может появиться возможность дальнейшего упрощения.

Пример 1.8. Если $Q_0 = 0,\ Q_1 = x_1x_2\,,\ Q_2 = x_1 \vee x_2\,,\ Q_3 = 1,$ то последние соотношения выполняются и

$$f = x_1x_2\Phi_2 \vee (x_1 \vee x_2\,)\Phi_1 \vee \Phi_1\Phi_2 = (x_1 \vee \Phi_2\,)\Phi_1 \vee x_2\Phi_2\,.$$

3. Если $Q_0 = 0,\ Q_1 = !Q,\ Q_2 = Q,\ Q_3 = 1,$ то

$$f = !Q\Phi_2 \vee Q\Phi_1 = \mathrm{MX}(\Phi_2\,, \Phi_1\,; Q)\,,$$

где MX — функция мультиплексирования.

4. Если выполняются отношения

$$Q_0 = Q_1 \leqslant Q_2 \leqslant Q_3\,,$$

то

$$f = Q_1 \vee \Phi_1\,(Q_2 \vee Q_3\Phi_2\,)\,.$$

5. Если функция *f* содержит только три типа фрагментов — Q_1, Q_2, Q_3, для которых выполняются отношения покрытия, то при $Q_0 = 0$ справедливо, что $f = Q_1\Phi_2 \vee \Phi_1\,(Q_2 \vee Q_3\Phi_2\,)$, а при $Q_0 = Q_1$ получается более простое выражение $f = Q_1 \vee \Phi_1\,(Q_2 \vee Q_3\Phi_2\,)$.

6. Если функция *f* содержит три типа фрагментов, для которых

$$Q_0 = 0 \leqslant Q_1 = Q_2 = Q \leqslant Q_3 = 1,$$

то функция f является мажоритарной функцией трех переменных:

$$f = (\Phi_1 \vee \Phi_2\,)Q \vee \Phi_1\Phi_2 = \Phi_1 \# \Phi_2 \# Q.$$

7. Если функция *f* содержит только два типа фрагментов — Q_0 и Q_1, то

$$f = Q_0!\Phi \vee Q_1\Phi = \mathrm{MX}(Q_0\,, \mathrm{Q}_1\,; \Phi)\,.$$

8. При $Q_0 \leqslant Q_1$ справедливо, что $f = Q_0 \vee Q_1 \Phi$. Например, при $Q_0 = x_1 x_2 \leqslant Q_1 = x_1 \vee x_2$ справедливо, что $f = x_1 x_2 \vee (x_1 \vee \vee x_2) \Phi = x_1 \# x_2 \# \Phi$. При $Q_0 \geqslant Q_1$ справедливо, что $f = Q_0 !\Phi \vee \vee Q_1$.

9. Выполнение отношения покрытия между функциями Q_0 и Q_1 позволяет их реализовывать совместно. Например, при $Q_0 = !x_1 x_2 \leqslant \leqslant Q_1 = x_1 \oplus x_2 = Q_0 \vee x_1 !x_2$, а при $Q_0 = !x_1 x_2 \leqslant Q_1 = x_1 \vee x_2 = = Q_0 \vee x_1$.

Более подробно отношения покрытия, введенные в [3], рассмотрены в разд. 7.2.2.

1.6. Метод совместной реализации булевых функций

Известен [20] «гарвардский» метод совместной реализации булевых функций, заданных таблицей истинности. Пусть в качестве примера требуется реализовать две БФУ — f_1 и f_2, зависящие от переменных $x_1, \ldots, x_n$. Суть метода состоит в том, что столбец значений одной из этих функций, например f_1, переносится в левую часть ТИ, а функция f_2 рассматривается как не полностью определенная БФУ, заданная на 2^n входных наборах из 2^{n+1} наборов. Минимизируя эту функцию, в ряде случаев удается получить формулу

$$f_2 = \Phi_1 (f_1, x_1, \ldots, x_n),$$

существенно более простую, чем формула

$$f_2 = \Phi_1 (x_1, \ldots, x_n).$$

Обратим внимание, что сложность совместной реализации зависит от того, какая из двух функций переносится в левую часть ТИ, что и было отмечено при участии автора настоящей работы в [20].

Построим формулы для f_1 и f_2 в отдельности и перенесем в левую часть «простейшую» из них. Такой подход будем использовать в качестве эвристического правила для выбора БФУ, подлежащей переносу.

Пример 1.9. Реализовать «гарвардским» методом БФУ f_1 и f_2, заданные табл. 1.10.

Из табл.1.10 следует, что

$$f_1 = (x_1 \vee x_2) x_3 \vee x_1 x_2 = x_1 \# x_2 \# x_3,$$
$$f_2 = !x_1 x_2 x_3 \vee x_1 (x_2 \oplus x_3).$$

Считая, что функция f_1 простейшая, перенесем столбец f_1 в левую часть ТИ (табл. 1.11).

Таблица 1.10

x_1	x_2	x_3	f_1	f_2	f_3	f_4	f_5	f_6
0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	1
0	1	0	0	0	0	0	1	1
0	1	1	1	1	0	1	0	1
1	0	0	0	0	0	0	1	1
1	0	1	1	1	0	1	0	1
1	1	0	1	1	0	1	0	1
1	1	1	1	0	1	1	1	0

Таблица 1.11

x_1	x_2	x_3	f_1	f_2
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	0

Минимизируя функцию f_2 с применением карты Карно, получим:

$$f_2 = f_1 (!x_1 \vee !x_2 \vee !x_3) = f_1!(x_1 x_2 x_3).$$

Предложим для решения той же задачи менее трудоемкий метод, суть которого состоит в записи уравнения

$$f = f_1 \oplus f_2$$

и решении его, оставляя в правой части «простейшую» формулу, например f_1. При этом

$$f_2 = f_1 \oplus f.$$

Пример 1.10. Реализовать БФУ f_1 и f_2 из предыдущего примера с помощью предлагаемого метода.

Используя табл.1.10, построим БФУ $f_3 = f_1 \oplus f_2 = x_1 x_2 x_3$. Таким образом, $f_2 = f_1 \oplus x_1 x_2 x_3$. Эта БФ содержит меньшее число логических операций по сравнению с БФ, найденной в предыдущем примере.

Пример 1.11. Реализовать параллельный одноразрядный двоичный сумматор (функции f_4 и f_5 табл. 1.10).

При этом

$$f_4 = (x_1 \vee x_2) x_3 \vee x_1 x_2, \; f_5 = x_1 \oplus x_2 \oplus x_3,$$

а суммарное число логических операций равно шести.

Применяя «гарвардский» метод, получим:

$$f_4 = (x_1 \vee x_2) x_3 \vee x_1 x_2, \; f_5 = !f_4 (x_1 \vee x_2 \vee x_3) \vee x_1 x_2 x_3$$

или

$$f_5 = x_1 \oplus x_2 \oplus x_3, \; f_4 = x_1 x_2 \vee x_3 !f_5.$$

В первом случае число логических операций равно одиннадцати (после факторизации — девять), а во втором — шести.

Используя предлагаемый метод, запишем $f_6 = f_4 \oplus f_5$ (табл.1.10). Так как функция f_6 является «зеркальной» (разд. 10.7), то из [21] следует, что

$$f_6 = (x_1 \oplus x_2) \vee (x_1 \oplus x_3).$$

Так как $f_4 = f_6 \oplus f_5$, то

$$\Phi = x_1 \oplus x_2, \; f_5 = \Phi \oplus x_3, \; f_4 = \big(\Phi \vee (x_1 \oplus x_3)\big) \oplus f_5.$$

Эти соотношения содержат лишь пять логических операций, из которых четыре однотипны.

Предложенный метод, как и «гарвардский», может применяться не только для двух, но и для большего числа булевых формул.

Пример 1.12. Использовать предлагаемый метод для реализации системы БФ (СБФ):

$$f_1 = x_1 x_2, \; f_2 = x_1 \vee x_2, \; f_3 = x_1 \oplus x_2.$$

Построим для заданной системы ТИ и сформируем по ней БФ $f = f_1 \oplus f_2 \oplus f_3$, которая равна нулю. Из соотношения $f_1 \oplus f_2 \oplus$ $\oplus f_3 = 0$, в частности, следует, что $f_3 = f_1 \oplus f_2$.

Предложенный подход может быть также использован и для упрощения внесения изменений в реализованную СБФ [22]. Еще один метод совместной реализации БФУ изложен в разд.16.8.

1.7. Новые соотношения для преобразований булевых функций

В [23] получены соотношения:

$$f(x_1, \ldots, x_n) = \big(!x_i f(x_i = 0) \vee x_i\, !f(x_i = 1)\big) \oplus x_i;$$
$$f(x_1, \ldots, x_n) = \big(!x_i\, !f(x_i = 0) \vee x_i f(x_i = 1)\big) \oplus !x_i.$$

Эти соотношения названы в [24] преобразованиями Артюхова—Шалыто.

Они неоднократно применяются в настоящей работе при решении различных задач логического проектирования. Рассмотрим несколько из таких задач.

1. На основе использования полученных преобразований может быть предложен подход для упрощения настройки мультиплексоров (МХ) и постоянных запоминающих устройств (ПЗУ), программируемых по таблицам истинности.

Предположим, что настройка нулем предпочтительнее, чем настройка единицей, а столбец значений реализуемой полностью определенной функции f содержит q_0 нулей и q_1 единиц.

Если $q_0 > q_1$, то целесообразно применять соотношения:

$$f_1 = f,\ f_1 = f \oplus 0. \quad (1.1)$$

Если $q_0 < q_1$, то целесообразно использовать соотношения:

$$f_1 = !f,\ f_1 = f \oplus 1. \quad (1.2)$$

Предположим, что $q_0 = q_1$, но число нулей в верхней и нижней половинах различно.

Если при этом число нулей в верхней половине больше, чем в нижней, то целесообразно применять соотношения:

$$f_1 = !x_1 f(x_1 = 0) \vee x_1 !f(x_1 = 1),\ f = f_1 \oplus x_1, \quad (1.3)$$

а в противном случае — соотношения:

$$f_1 = !x_1 !f(x_1 = 0) \vee x_1 f(x_1 = 1),\ f = f_1 \oplus !x_1. \quad (1.4)$$

Применительно к МХ этот подход впервые описан в [23] и использован в [25]. Из изложенного следует, что все приведенные выше соотношения могут быть реализованы схемой на рис.1.5.

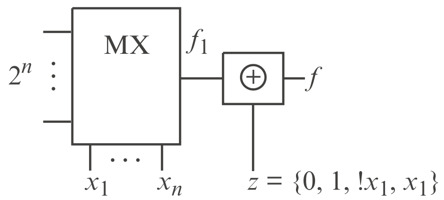

Рис. 1.5

Этот подход может быть применен для каждого из выходов ПЗУ с целью уменьшения «травмирования» этого устройства при программировании.

Пример 1.13. Реализовать мажоритарную функцию трех переменных «два и более из трех» $f(x_1, x_2, x_3) = \left| 0001\ 0111 \right|^T$ с помощью схемы на рис.1.5, в которой $n = 3$. Здесь T — символ операции «транспонирование».

При программировании МХ непосредственно по столбцу значений (соотношения (1.1)) число единиц равно четырем, а $z = 0$.

Так как в этом случае $q_0 = q_1$, а в верхней половине число нулей больше, чем в нижней, то программирование МХ выполним на основе соотношений (1.3). При этом $f_1 = \left| 0001\ 1000 \right|^T$, а $z = x_1$. Таким образом, в данном случае число единиц в коде настройки уменьшилось до двух.

2. Построить параллельный одноразрядный двоичный сумматор-вычитатель, в котором обеспечивается компромисс между однородностью используемых элементов и их числом.

Известно [26], что двоичный сумматор описывается системой БФ:

$$f_1 = x_1 (x_2 \vee x_3) \vee x_2 x_3, \; f_2 = x_1 \oplus x_2 \oplus x_3,$$

а двоичный вычитатель — СБФ:

$$f_3 = !x_1 (x_2 \vee x_3) \vee x_2 x_3, \; f_4 = x_1 \oplus x_2 \oplus x_3.$$

Наименее трудоемким является построение СБФ:

$$F_1 = (z \oplus x_1)(x_2 \vee x_3) \vee x_2 x_3, \; F_2 = x_1 \oplus x_2 \oplus x_3,$$

где $z = \{0, 1\}$, которая, однако, не обеспечивает указанный компромисс.

Применяя к функции f_1 первое преобразование по переменной x_2, получим:

$$f_1 = (!x_1 x_2 !x_3 \vee x_1 !x_2 x_3) \oplus x_2 = \Phi \oplus x_2.$$

Функция Ф является «зеркальной», и поэтому на основании [21]

$$\Phi = (x_1 \oplus x_2)(x_2 \oplus x_3).$$

Следовательно, $f_1 = (x_1 \oplus x_2)(x_2 \oplus x_3) \oplus x_2$.

Применяя к функции f_3 первое преобразование по переменной x_3, получим:

$$f_3 = (!x_1 x_2 !x_3 \vee x_1 !x_2 x_3) \oplus x_3 = \Phi \oplus x_3.$$

Таким образом, сумматор-вычитатель может быть реализован СБФ:

$$F_1 = (x_1 \oplus x_2)(x_2 \oplus x_3) \oplus z, \quad F_2 = x_1 \oplus x_2 \oplus x_3,$$

где $z = \{x_2, \; x_3\}$.

Эта СБФ, полученная в [27], была использована в [28]. В этой работе заданное устройство реализуется схемой из пяти двухвходовых логических элементов, четыре из которых однотипны.

Обращая внимание на тот факт, что исходные формулы f_1 и f_3 отличаются на инверсию первой буквы, из изложенного выше можно записать:

$$F_1 = (z \oplus x_1 \oplus x_2)(x_2 \oplus x_3) \oplus x_2, \; F_2 = x_1 \oplus x_2 \oplus x_3,$$

где $z = \{0, 1\}$.

По этой СБФ может быть построена схема из шести двухвходовых элементов, которая более проста, чем схема, приведенная в [29]. Полностью однородная схема из двухвходовых элементов И—НЕ (ИЛИ—НЕ) содержит более шести элементов.

3. Рассмотрим вопрос о применении предложенных преобразований с целью минимизации числа букв в БФ.

Пример 1.14. Булева функция «точно два из трех» $f(x_1, x_2, x_3) = \left| 0001\ 0110 \right|^T$ реализуется формулой:

— в базисе {&, ∨, !} из восьми букв, получаемой с использованием карты Карно,

$$f = !x_1 x_2 x_3 \vee x_1 (!x_2 x_3 \vee x_2 !x_3);$$

— в базисе {&, ⊕} из семи букв, получаемой на основе полинома Жегалкина [20],

$$f = x_1 (x_2 \oplus x_3 \oplus x_2 x_3) \oplus x_2 x_3;$$

— в базисе {&, ∨, ⊕, !} из шести букв, получаемой из первой БФ,

$$f = !x_1 x_2 x_3 \vee x_1 (x_2 \oplus x_3);$$

— в базисе {&, ⊕, !} из шести букв, получаемой из предыдущей формулы при замене операции ∨ на операцию ⊕ между ортогональными подформулами,

$$f = !x_1 x_2 x_3 \oplus x_1 (x_2 \oplus x_3);$$

— в базисе {&, ∨, ⊕} из пяти букв, получаемой с помощью разложения Рида (разд.3.1.1) [30] по переменной x_1,

$$f = x_1 (x_2 \vee x_3) \oplus x_2 x_3;$$

— в базисе {&, ∨, ⊕, !} из пяти букв, получаемой как с помощью разложения Рида по переменной $!x_3$, так и с помощью первого из предлагаемых преобразований, выполняемого последовательно по переменным x_2 и x_1,

$$f = (x_1 \vee x_2) !x_3 \oplus x_1 \oplus x_2.$$

Полученная формула может быть реализована каскадной схемой из двухвходовых элементов, в то время как предыдущая формула такой схемой не реализуется.

Пример 1.15. Булева функция $f(x_1, x_2, x_3, x_4) = \left| 0100\ 0011\ 0011\ 1110 \right|^T$ реализуется формулой:

— в базисе {&, ∨, !} из 13 букв, получаемой с применением карты Карно,

$$f = (!x_1 !x_2 x_4 \vee x_1 x_2) !x_3 \vee (!x_1 x_2 \vee x_1 !x_2 \vee x_2 !x_4) x_3;$$

— в базисе {&, ⊕} из 12 букв, получаемой на основе полинома Жегалкина,

$$f = x_1 \big((x_2 \oplus x_3) \oplus x_2 x_3 \big) (1 \oplus x_4) \oplus (1 \oplus x_1 \oplus x_2 \oplus x_3) x_4;$$

— в базисе $\{\&, \vee, \oplus, !\}$ из 11 букв, получаемой из первой БФ,

$$f = (!x_1 \, !x_2 x_4 \vee x_1 x_2)!x_3 \vee \big((x_1 \oplus x_2) \vee x_2 \, !x_4\big) x_3 ;$$

— в базисе $\{\&, \vee, \oplus, !\}$ из 10 букв, получаемой из второй БФ,

$$f = x_1 \big((x_2 \oplus x_3) \oplus x_2 x_3\big)!x_4 \oplus (!x_1 \oplus x_2 \oplus x_3) x_4 ;$$

— в базисе $\{\&, \vee, \oplus, !\}$ из 8 букв, получаемой с помощью разложения Рида по переменной $!x_1$,

$$f = !x_1 (!x_2 \oplus x_3)!x_4 \oplus \big((x_2 \oplus x_3) \vee x_2 \, !x_4\big) ;$$

— в базисе $\{\&, \vee, \oplus, !\}$ из 6 букв, получаемой с помощью второго из предлагаемых преобразований, выполняемого по переменной $!x_1$,

$$f = !x_1 \oplus \big((!x_1 \oplus x_3) x_4 \vee (x_2 \oplus x_3)\big) .$$

Выводы

1. Определено количество типов бесповторных булевых формул (ББФ) в базисе И, ИЛИ, НЕ в различных классификациях.
2. Выполнено табулирование представителей *PN*-типов этих формул при $h \leqslant 8$.
3. Определено рекуррентное соотношение для подсчета неизоморфных деревьев из двухвходовых элементов.
4. Рассмотрены покрывающие деревья и на их основе предложены модули, универсальные в указанном классе формул с простой структурой.
5. Рассмотрена *PN*-классификация ББФ в базисе И, ИЛИ, НЕ-РАВНОЗНАЧНОСТЬ, НЕ и выполнено их табулирование при $h \leqslant 4$.
6. Установлено, что ББФ в базисе И, ИЛИ, НЕ, существенно зависящие от всех переменных, имеют нечетный ранг.
7. Предложены методы подсчета числа единиц при формульном задании булевой функции.
8. Определены свойства бесповторных пороговых формул и установлена их связь с числами Фибоначчи.
9. Определены отношения покрытия, выполнение которых позволяет записывать булевы формулы в форме, аналогичной разложению Шеннона, которая не содержит переменных с инверсиями.
10. Предложен метод совместной реализации булевых функций, обладающий меньшей трудоемкостью по сравнению с «гарвардским» методом.
11. Получены новые соотношения для преобразования булевых функций и на примерах показана целесообразность их применения при решении ряда задач логического проектирования.

Литература

1. *Яблонский С. В.* Введение в дискретную математику. М.: Наука, 1979.

2. *Яблонский С. В., Гаврилов Г. П., Кудрявцев В. Б.* Функции алгебры логики и классы Поста. М.: Наука, 1966.

3. *Артюхов В. Л., Копейкин Г. А., Шалыто А. А.* Настраиваемые модули для управляющих логических устройств. Л.: Энергоиздат, 1981.

4. *Лазарев В. Г., Наумчук О. Ф., Саввин Г. Г., Сагалович Ю. Л.* О переписи типовых релейно-контактных схем автоматической телефонии // Проблемы передачи информации. 1963. № 8.

5. *Шеннон К. Э.* Работы по теории информации и кибернетике. М.: Изд-во иностр. лит., 1963.

6. *Кнут Д.* Искусство программирования для ЭВМ. Т. 3. Сортировка и поиск. М.: Мир, 1978.

7. *Артюхов В. Л.* Логические методы синтеза дискретных систем. Л.: Ин-т повышения квалификации руководящих работников и специалистов судостроительной промышленности (ИПК СП), 1974.

8. *Риордан Дж.* Введение в комбинаторный анализ. М.: Изд-во иностр. лит., 1963.

9. *Артюхов В. Л., Розенблюм Л. Я., Шалыто А. А.* Логические возможности некоторых типов каскадных структур // Сети связи и дискретные устройства управления. М.: Наука, 1976.

10. *Артюхов В. Л., Копейкин Г. А., Шалыто А. А.* О классификации бесповторных булевых функций / Теория автоматов и ее приложения. Материалы советско-болгарского семинара. М.: Ин-т проблем передачи информации, 1973.

11. *Попов Ю. А., Воронин А. Т., Сладков А. Б.* Вопросы проектирования схем с высоким уровнем интеграции на основе КНС технологии // Дискретные системы. Т.1. Симпозиум IFAC. Рига: Зинатне, 1974.

12. *Артюхов В. Л., Копейкин Г. А., Шалыто А. А.* Вопросы выбора и применения многофункциональных логических модулей // Дискретные системы. Т.1. Симпозиум IFAC. Рига: Зинатне, 1974.

13. *Levy S. Y., Winder R., Mott T. H.* A note on tributary switching networks // IEEE Trans. on Computers. 1964. N 2.

14. *Воробьев Н. Н.* Числа Фибоначчи. М.: Наука, 1978.

15. *Чистов В. П., Битюцкий В. П.* Функциональная полнота в ленточных однородных структурах // Изв. АН СССР. Техн. кибернетика. 1971. № 3.

16. *Пупырев Е. И.* Использование аксиомы теории вероятностей в задачах анализа булевых функций // Автоматика и телемеханика. 1976. № 7.

17. *Шалыто А. А.* Реализация алгоритмов судовых управляющих логических систем при использовании микропроцессорной техники. Л.: ИПК СП, 1988.

18. *Fridman A. D., Menon P. R.* Theory and design of switching circuits. Computer Science Press., Inc., 1975.

19. *Кондратьев В. Н., Шалыто А. А.* Реализация булевых функций одним линейным арифметическим полиномом с маскированием // Автоматика и телемеханика. 1996. № 1.

20. *Поспелов Д. А.* Логические методы анализа и синтеза схем. М.: Энергия, 1974.

21. *Шалыто А. А.* Многофункциональный логический модуль. А. с. СССР № 1283744 // Бюл. изобр. 1987. № 2.

22. *Шалыто А. А.* SWITCH-технология. Алгоритмизация и программирование задач логического управления. СПб.: Наука, 1998.

23. *Артюхов В. Л., Шалыто А. А.* Судовые управляющие логические системы. Л.: ИПК СП, 1983.

24. *Артюхов В. Л., Шалыто А. А.* Реализация булевых формул однородными мультиплексорными и мажоритарными каскадами // Изв. РАН. Теория и системы управления. 1996. № 5.

25. *Артюхов В. Л., Шалыто А. А.* Многофункциональный логический модуль. А. с. СССР № 1149244 // Бюл. изобр. 1985. № 13.

26. *Мищенко В. А., Козюминский В. Д., Семашко А. Н.* Многофункциональные автоматы и элементная база цифровых ЭВМ. М.: Радио и связь, 1981.

27. *Шалыто А. А.* Программная реализация алгоритмов логического управления судовыми системами. Л.: ИПК СП, 1989.

28. *Шалыто А. А.* Многофункциональный логический модуль. А. с. СССР № 1290290 // Бюл. изобр. 1987. № 6.

29. *Шалыто А. А.* Комбинационный двоичный сумматор-вычитатель. А. с. СССР № 1264166 // Бюл. изобр. 1986. № 38.

30. *Reed I. S.* Class of multiple error correcting codes and decoding scheme // IRE Trans. Inform. Theory. 1954. IT-4.