УДК 519.7

ПАРАЛЛЕЛЬНЫЕ ЛОГИЧЕСКИЕ ВЫЧИСЛЕНИЯ МЕТОДАМИ МОДУЛЯРНОЙ АРИФМЕТИКИ

О.А. Финько
Краснодарский военный институт
Россия, 350035, Краснодар, Красина ул., 4
E-mail: ofinko@yandex.ru

Ключевые слова: параллельные логические вычисления, арифметические полиномы, модулярная арифметика, Китайская теорема об остатках
Key word: Parallel logical calculations, arithmetic polynomials, modular arithmetics, Chinese remainder theorem

Предложено отображение классической логики на модулярную арифметику, которое открывает новые уникальные возможности по достижению высоких уровней производительности и отказоустойчивости средств гибких логических вычислений.
PARALLEL LOGICAL CALCULATIONS BY METHODS OF MODULAR ARITHMETICS / O.A. Finko (Krasnodar military institute, 4 Krasina street, Krasnodar 350035, Russia, E-mail: ofinko@yandex.ru). Mapping classical logic on modular arithmetics is offered which opens new unique Possibilities on reaching high levels of productivity and fault tolerance of resources of flexible logical calculations.

1. Введение

Из множества алгоритмов, используемых в современных информационно-вычислительных и телекоммуникационных системах, можно выделить классы, реализация которых чаще, чем другие вызывает затруднения. К таким классам следует отнести алгоритмы, основу которых составляют *логические вычисления*. К «потребителям» логических вычислений в первую очередь следует отнести системы автоматизированного проектирования (САПР) интегральных микросхем (ИМС) и системы криптографической защиты информации [11, 14, 51, 59, 85].

Для первых острота проблемы вызвана постоянным усложнением дискретных устройств (увеличением количества вентилей, усложнением топологии ИМС, проблемой межсоединений и планаризации структур) и, следовательно, существенным возрастанием *размерности* решаемых задач [87, 102, 103].

Для вторых проблема реализации логических вычислений стала очевидной благодаря широкому внедрению криптографических методов в системы защиты электронной информации. В компьютерных системах требуется получить высокие скорости шифрования при использовании микропроцессоров

общего применения. Для высокой защищенности обработки данных необходимо обеспечить *«глобальное»* шифрование, то есть шифрование всех данных, хранимых в хранимых в энергонезависимой памяти.

В общем случае, обработка данных связана с произвольными запросами на чтение и запись данных, поэтому требуется обеспечить параллельное независимое шифрование отдельных блоков данных. Современные ЭВМ характеризуются большим объемом постоянной памяти и высокой скоростью записи и считывания. Эти факторы и определяют высокие требования, как к шифрам, так и к способам их реализации.

Перспективным способом повышения стойкости криптосистем является внесение элементов гибкости в некоторую часть струтуры криптоалгоритма. Для практической осуществимости этого способа необходимо, чтобы часть алгоритма шифрования была легко сменяемым элементом, то есть стала бы элементом секретного ключа. Такие шифры в [49] названы *недетерминированными*.

При использовании недетерминированных шифров перед сеансом шифрования выполняются предвычисления — генерация алгоритма шифрования. Наличие этапа предвычислений является фактором, существенно ужесточающим требования к *скорости* реализации логических вычислений.

В [48, 49] данная проблема решается путем разработки скоростных шифров, учитывающих специфику обработки информации в микропроцессорах общего применения или предполагающих применение предельно простых аппаратных средств. Однако, возможен и другой путь — *совершенствование методов обработки логической информации*. Второй путь более *универсален*, так как позволяет *расширить класс* эффективно реализуемых шифров на основе стандартных технических средств.

Реализация алгоритмов логического управления традиционно решалась в рамках теории конечных автоматов как задача синтеза схемы минимальной сложности. Несмотря на гибкость программной реализации, непосредственный перенос описания схемы в память ЭВМ как правило оказывается *нерациональным*, — всякое описание алгоритмов и схем на языке булевой алгебры не соответствует непрерывно возрастающим возможностям современных ЭВМ. При написании логической программы, как правило, используются лишь логические операторы и команды условного перехода. При этом, такие ресурсы ЭВМ как весь набор команд (а не только логические), аппаратно поддерживаемая разрядность целочисленных данных и др. используются крайне ограниченно.

Парадоксальность сложившейся ситуации заключается в том, что в основе построения всей вычислительной техники лежат все те же законы алгебры логики, которые с помощью этой техники не удается эффективно реализовать.

Справедливо будет отметить тот факт, что схемная реализация логических алгоритмов, все еще продолжает активно использоваться во многих практических приложениях. Этому, в частности, способствует широкое распространение ПЛИС — программируемые пользователем ИМС. Аппаратная реализация используется в системах с криптографической защитой информации с целью поддержания высокой скорости обмена данными. Логические расширители могут быть использованы для наращивания вычислительных воз-

можностей ЭВМ при ее проблемной ориентации на решение заданного класса задач. Ценой высокой скорости обработки при этом является предельное *снижение гибкости* вычислений, которая не всегда допустима (например, при реализации недетерминированных шифров).

Кроме того, из-за небольших объемов выпуска таких устройств, в стремлении снизить экономические и временные затраты, связанные с проектированием ИМС, разработчики (или заказчики) часто «жертвуют» качеством решения достаточно важных для прикладной области вопросов верификации ИМС и обеспечением технического сервиса (контроля ошибок вычислений, тестирования, восстановления работоспособности) [58]. Применение ПЛИС, разумеется, существенно упрощает разработку специализированных вычислителей, однако также очевидно, что ПЛИС — это всего лишь средство, которое отражает и все *недостатки проектирования*.

В то же время промышленностью давно освоены капиталоемкие технологии производства широкой номенклатуры высококачественных специализированных вычислителей цифровой обработки (ЦО) сигналов, лишенных указанных недостатков. Однако многие из них не могут быть эффективно использованы для реализации логических вычислений. При использовании ПЛИС желательно применять высокоразвитый научный аппарат, наиболее полно учитывающий особенности целей проектирования.

Из практики логического программирования известны примеры, когда для реализации вычислительно трудных логических операций используют приемы замены этих операций другими — менее сложными операциями для заданных условий реализации. Так, в 50-х годах XX-го столетия профессор Г. Айкен (Aiken, USA) использовал арифметическую форму задания логических функций в компьютерах MARK 3 и MARK 4 [88]. Для реализации подстановок большого размера в некоторых шифрах используют операцию возведения в дискретную степень и операцию дискретного логарифмирования в поле вычетов по заданному модулю (например, по модулю 257 в шифре SAFER).

В 70–80-х годах XX-го столетия в теории логического управления были заложены основы нового научного направления, основывающегося на *арифметических аспектах двоичной логики* [13, 42–45]. Сквозная идея развиваемого в этих работах подхода — *совместное описание систем логических функций произвольной размерности посредством арифметических полиномов*.

Последнее означает переход от двоичной логики к целочисленной арифметике (рис. **??**). Как результат такой «арифметизации» следует рассматривать обеспечение принципиально новых возможностей по применению высокоразвитых методов арифметических вычислений и парка серийной вычислительной техники, обеспеченного «отработанными» методами технического сервиса, для решения задач, связанных с реализацией параллельных логических вычислений.

Теоретическую зрелость арифметические аспекты двоичной логики приобрели в трудах сотрудника ин-та проблем управления им. В. А. Трапезникова Российской академии наук д-ра техн. наук, проф. В. Д. Малюгина [43–45, 47]. Полученные положения явились естественным продолжением прикладного направления математической логики, заложенного трудами В. Н. Шестакова, К. Шеннона, И. И. Жегалкина, М. А. Гаврилова. В настоящее время

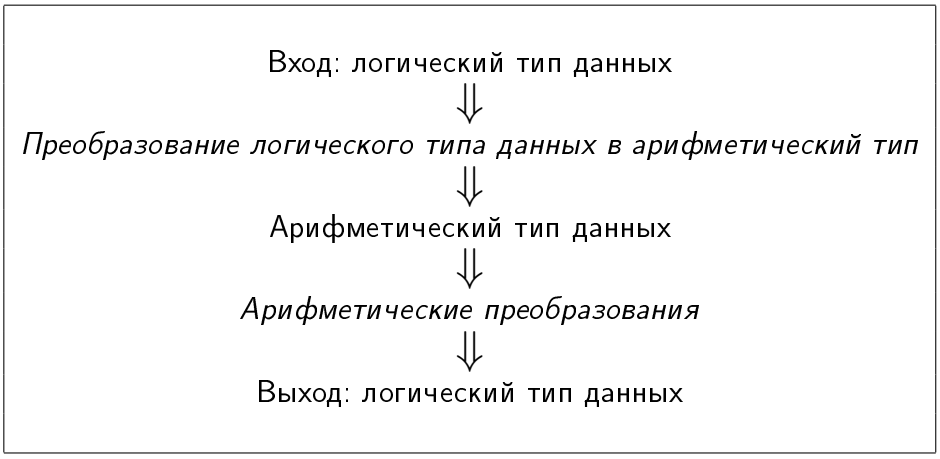

Рис. 1: Обобщенная схема логических вычислений посредством арифметических преобразований

это направление продолжает активно развиваться в работах В. П. Шмерко, А. А. Шалыто, В. А. Артюхова, Л. А. Залманова, В. Н. Кондратьева, Г. А. Кухарева, В. М. Антоненко, С. Н. Янушкевич, Н. Х. Аслановой, Р. Г. Фараджева, В. С. Выхованец и др.

В условиях достижения значительных успехов в области организации параллельных вычислений, перевод логических вычислений в плоскость арифметического счета является фундаментальным по своей сути, но еще недостаточным условием для эффективной реализации *параллельных логических вычислений* в *современных* условиях.

Наиболее ярким явлением в области организации параллельных арифметических вычислений является *модулярная арифметика* [9, 10, 12, 15, 22, 26, 27, 32, 33, 37, 38–40, 60, 61, 63, 89, 98, 99].

Уникальной особенностью средств ЦО, использующих в архитектуре и алгоритмах функционирования модулярную арифметику, является низкий, по сравнению с другими методами, *прирост аппаратурных затрат* при достижении предельного для используемой технологии изготовления микросхем уровня *производительности*. Другим не менее важным достоинством модулярной арифметики является развитость методов *контроля ошибок* вычислений и обеспечения *отказоустойчивости* средств вычислений.

Большой вклад в развитие теории модулярной машинной арифметики внесли отечественные ученые: И. Я. Акушский, Д. И. Юдицкий, В. М. Амербаев, В. А. Торгашев, М. В. Синьков, И. Т. Пак, А. А. Коляда, В. Г. Евстигнеев, Н. И. Червяков, Е. К. Лебедев, В. А. Краснобаев.

Следует отметить, что модулярная арифметика эффективна при реализации только определенных (модульных) арифметических операций, обуславливая проблемно-ориентированный характер этой арифметики. Логические операции традиционно относились к неэффективным операциям, так как для их выполнения требовался выход в вычислительную среду, основанную на «внешней» позиционной арифметике.

Однако последние достижения по «арифметизации» логических вычислений открывают уникальные возможности по реализации достоинств модулярной арифметики в области организации *параллельных логических вычис-*

лений.

Таким образом, *основной задачей дальнейших исследований будем считать получение теоретических положений, устанавливающих взаимосвязь между методами модулярной арифметики и методами реализации логических функций арифметическими полиномами.*

Решение этой задачи позволит *адаптировать* высокоразвитые математический аппарат и высокопроизводительные, отказоустойчивые технические средства ЦО, основанные на модулярной арифметике, например, средства ЦО сигнальной информации, для реализации *параллельных гибких* логических вычислений.

2. Основные положения арифметической логики

2.1. Представление булевых функций арифметическими полиномами

Пусть дана d-выходная булева функция $f(X)$ (система булевых функций — $f_1(X)$, $f_2(X)$, …, $f_d(X)$) от n переменных $X = x_1, x_2, \ldots, x_n$:

$$(1) \qquad \begin{cases} y_1 = f_1(X), \\ y_2 = f_2(X), \\ \vdots \\ y_d = f_d(X), \end{cases}$$

где y_j — значение, принимаемое j-й булевой функцией $f_j(X)$; x_i, $y_j \in \{0, 1\}$ ($i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, d$). При этом кортеж значений булевых функций $y_d \odot y_{d-1} \odot \ldots \odot y_1$, где $\odot$ — разделительный знак, интерпретируется как код целого неотрицательного числа Y, представленного в двоичной системе счисления:

$$(y_d\, y_{d-1}\, \ldots\, y_1)_2 = Y = \sum_{j=1}^{d} y_j 2^{j-1}.$$

Пример 1.
Представление Y, соответствующее системе булевых функций

$$(2) \qquad \begin{cases} f_1(X) = \overline{x_1 \oplus x_2}, \\ f_2(X) = \overline{x_1 \vee x_2}, \end{cases}$$

приведено в таблице **??** (здесь и далее $\vee$, $\wedge$, $\oplus$, $\neg$ — символы операций логического сложения, умножения, сложения по модулю 2 и инверсии соответственно).

Теорема 1.
Произвольный кортеж булевых функций $y_d \odot y_{d-1} \odot \ldots \odot y_1$ может быть представлен арифметическим полиномом

$$(3) \qquad Y = D(X) = \sum_{i=0}^{2^n-1} c_i\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n},$$

Таблица 1: Пример числового представления значений системы булевых функций, соответствующих двоичному полусумматору

x_2	x_1	y_2	y_1	Y (десятичная запись)
0	0	1	1	3
0	1	0	0	0
1	0	0	0	0
1	1	0	1	1

где здесь и далее по тексту

$$(i_1 i_2 \dots i_n)_2 = \sum_{u=1}^{n} i_u 2^{n-u} \ (i_u \in 0, 1);$$
$$x_u^{i_u} = \begin{cases} x_u, \ i_u = 1, \\ 1, \ i_u = 0; \end{cases}$$

$c_i \in \mathcal{Z}$ $(i = 0, 1, \dots, 2^n - 1)$ и притом единственным образом [43, 44, 46, 47].

Доказательство теоремы 1 дано в [47].

Принципы реализации логических вычислений посредством теоремы 1 поясняются с помощью рис. ?? и ??.

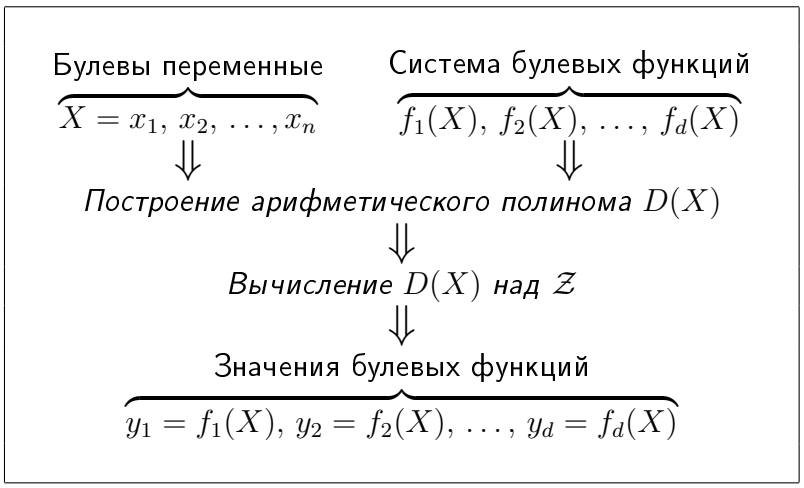

Рис. 2: Схема вычисления произвольной системы булевых функций над кольцом $\mathcal{Z}$

Схема вычисления, представленная на рис. ??, обеспечивает вычисление систем булевых функций, сведения о которых становятся известными в процессе решения задачи. Это безусловно обеспечивает высокую гибкость этого метода. Недостаток — увеличенные временные затраты, обусловленные необходимостью вычисления коэффициентов арифметического полинома. Этого недостатка лишена схема, поясняемая с помощью рис. ??. В этом случае коэффициенты арифметического полинома вычислены заранее и хранятся в памяти ЭВМ. Выбор схемы вычисления определяется особенностями конкретной решаемой задачи.

Рис. 3: Схема вычисления заданной системы булевых функций над кольцом $\mathcal{Z}$

2.2. Преобразование булевых формул в арифметические полиномы

Алгебраический метод получения арифметического полинома (**??**) заключается в реализации следующего алгоритма.

Алгоритм 1.

Шаг 1. Получение арифметического полинома $P_j(X)$ для каждой булевой функции $y_j = f_j(X)$, $j = 1, 2, \ldots, d$:

$$(4) \qquad \begin{cases} f_1(X) = P_1(X) = \sum\limits_{i=0}^{2^n-1} r_{1,\,i}\, x_1^{i_1} x_2^{i_2} \ \ldots\ x_n^{i_n}, \\ f_2(X) = P_2(X) = \sum\limits_{i=0}^{2^n-1} r_{2,\,i}\, x_1^{i_1} x_2^{i_2} \ \ldots\ x_n^{i_n}, \\ \vdots \\ f_d(X) = P_d(X) = \sum\limits_{i=0}^{2^n-1} r_{d,\,i}\, x_1^{i_1} x_2^{i_2} \ \ldots\ x_n^{i_n}. \end{cases}$$

Шаг 2. Умножение арифметических полиномов на веса:

$$(5) \qquad \begin{cases} P_1'(X) = P_1(X)2^0 = \sum\limits_{i=0}^{2^n-1} r'_{1,\,i}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \\ P_2'(X) = P_2(X)2^1 = \sum\limits_{i=0}^{2^n-1} r'_{2,\,i}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \\ \vdots \\ P_d'(X) = P_d(X)2^{d-1} = \sum\limits_{i=0}^{2^n-1} r'_{d,\,i}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \end{cases}$$

где $r'_{j,\,i} = r_{j,i} 2^{j-1} \quad (j = 1, 2, \ldots, d;\ i = 0, 1, \ldots, 2^n - 1)$.

Шаг 3. Получение искомого арифметического полинома $D(X)$ (**??**) путем суммирования коэффициентов арифметического полинома $P_j'(X)$ для

всех $j = 1, 2, \ldots, d$:

$$D(X) = \sum_{i=0}^{2^n-1} \sum_{j=1}^{d} r'_{j,\,i}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} = \sum_{i=0}^{2^n-1} c_i\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \tag{6}$$

где $c_i = \sum_{j=1}^{d} r'_{j,\,i} \quad (i = 0, 1, \ldots, 2^n - 1)$.

Структура алгоритма 1 поясняется с помощью рис. **??**.

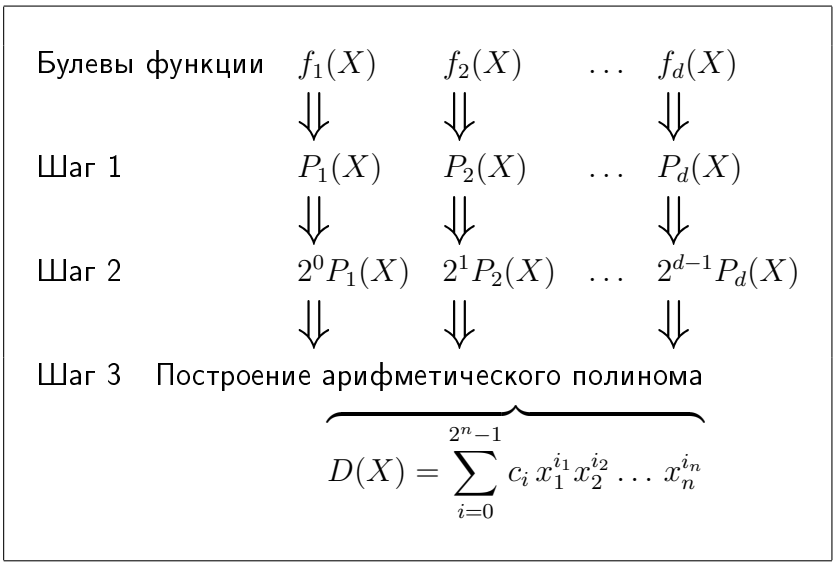

Рис. 4: Схема алгебраического метода построения арифметического полинома $D(X)$

Пример 2.
Для системы булевых функций (**??**) реализация алгоритма 1 имеет вид:

Шаг 1. Используя соотношения

$$\begin{aligned} x_1 \wedge x_2 &= x_1 x_2, \\ x_1 \vee x_2 &= x_1 + x_2 - x_1 x_2, \\ x_1 \oplus x_2 &= x_1 + x_2 - 2x_1 x_2, \\ \overline{x} &= 1 - x, \end{aligned}$$

получим

$$\begin{aligned} f_1(X) &= P_1(X) = \overline{x_1 \oplus x_2} = 1 - x_1 - x_2 + 2x_1 x_2, \\ f_2(X) &= P_2(X) = \overline{x_1 \vee x_2} = 1 - x_1 - x_2 + x_1 x_2. \end{aligned}$$

Шаг 2.

$$\begin{aligned} P'_1(X) &= 2^0(1 - x_1 - x_2 + 2x_1 x_2) = 1 - x_1 - x_2 + 2x_1 x_2, \\ P'_2(X) &= 2^1(1 - x_1 - x_2 + x_1 x_2) = 2 - 2x_1 - 2x_2 + 2x_1 x_2. \end{aligned}$$

Шаг 3.

$$\begin{aligned} D(X) &= 1+2-(1+2)x_1-(1+2)x_2+(2+2)x_1x_2= \\ &= 3-3x_1-3x_2+4x_1x_2. \end{aligned}$$

Из этого примера можно видеть, что числовой диапазон, требуемый для представления коэффициентов и результатов промежуточных вычислений арифметических полиномов может значительно превосходить числовой диапазон, достаточный для представления Y. В рассмотренном случае для представления Y достаточно двух двоичных разрядов ($0 \le Y \le 2^2-1$). В то же время для представления коэффициентов $c_1 \dots c_4$ арифметического полинома (**??**) требуется четыре двоичных разряда ($-3 \le c_{1\dots4} \le 5$), а результаты промежуточных вычислений при $x_1 = x_2 = 1$ могут принимать значения от -6 до 7.

2.3. Линейные арифметические полиномы

Важное значение для представления d-выходных булевых функций $f(X)$ имеют линейные арифметические полиномы $L(X)$, которые определяются выражением

$$U = L(X) = d_0 + \sum_{i=1}^{n} d_i\, x_i = d_0 + d_1x_1 + \ldots + d_nx_n, \tag{7}$$

где коэффициенты d_0, d_1, …, d_n — целые числа [87, 102].

При вычислении $f_j(X)$ используется оператор маскирования $\Xi^t\{U\}$ [102], служащий для определения t-го двоичного разряда (выхода) разряда (выхода) представления

$$U = (a_r \ \ldots\ a_t \ \ldots\ a_2\, a_1)_2\,,$$

то есть $\Xi^t\{U\} = a_t$.

Пример 3.

Для линеаризации $P_j(X) = x_1 + x_2 - x_1x_2$, соответствующего булевой функции $f_j(X) = x_1 \vee x_2$ используется введение дополнительной (избыточной) булевой функции $f_j^{(1)}(X)$. При этом образуется система функций:

$$\begin{aligned} f_j^{(1)}(X) &= 1 \oplus x_1 \oplus x_2, \\ f_j^{(2)}(X) &= x_1 \vee x_2. \end{aligned}$$

Тогда

$$U = L(X) = 2^1 f_j^{(2)}(X) + 2^0 f_j^{(1)}(X) = 1 + x_1 + x_2$$

и $f_j(X) = \Xi^2\{U\}$ [102].

Таким образом, для представления систем булевых функций (**??**) с помощью линейных арифметических полиномов $L(X)$ используется тот же принцип взвешивания представлений булевых функций с помощью весов 2^i

($i = 0, 1, \ldots$), что и при построении арифметических полиномов $D(X)$ (??). Однако значения i при этом выбираются с учетом введенных дополнительных булевых функций.

Пример 4.
Дана система булевых функций [102]:

$$\left\{ \begin{array}{l} f_A(X) = x_1 \wedge x_3, \\ f_B(X) = \overline{x}_1 \wedge x_2, \\ f_C(X) = \overline{x}_2 \wedge x_3, \end{array} \right. \tag{8}$$

Для обеспечения линейности результирующего арифметического полинома добавляют вспомогательные булевы функции $f_A^{(1)}(X)$, $f_B^{(3)}(X)$, $f_C^{(5)}(X)$ и получают систему булевых функций:

$$\left\{ \begin{array}{l} f_A^{(1)}(X) = x_1 \oplus x_3, \\ f_A^{(2)}(X) = f_A(X), \end{array} \right.$$

$$\left\{ \begin{array}{l} f_B^{(3)}(X) = \overline{x}_1 \oplus x_2, \\ f_B^{(4)}(X) = f_B(X), \end{array} \right.$$

$$\left\{ \begin{array}{l} f_C^{(5)}(X) = \overline{x}_2 \oplus x_3, \\ f_C^{(6)}(X) = f_C(X). \end{array} \right.$$

Далее, в соответствии с (??) и примером 3 имеем:

$$\begin{array}{rcl} U_A & = & L_A^{'}(X) = 2^1 f_A^{(1)}(X) + 2^0 f_A^{(2)}(X) = x_1 + x_3, \\ U_B & = & L_B^{'}(X) = 2^1 f_B^{(3)}(X) + 2^0 f_B^{(4)}(X) = 1 - x_1 + x_2, \\ U_C & = & L_C^{'}(X) = 2^1 f_C^{(5)}(X) + 2^0 f_C^{(6)}(X) = 1 - x_2 + x_3. \end{array}$$

Получаем линейный арифметический полином:

$$\begin{array}{rcl} U & = & L(X) = 2^0 L_A^{'}(X) + 2^2 L_B^{'}(X) + 2^4 L_C^{'}(X) = \\ & = & 20 - 4x_1 - 12x_2 + 16x_3. \end{array} \tag{9}$$

Для определения t-й булевой функции воспользуемся оператором маскирования $\Xi^t\{U\}$:

$$\left\{ \begin{array}{l} f_A(X) = \Xi^2\{U\}, \\ f_B(X) = \Xi^4\{U\}, \\ f_C(X) = \Xi^6\{U\}. \end{array} \right.$$

Отметим, что линейная форма арифметического полинома (??) достигнута за счет введения избыточных булевых функций и увеличения числового диапазона, необходимого для представления U в 2^3 раза.

2.4. Конъюнктивные преобразования

Под прямым и обратным матричным преобразованием (логическим дискретным преобразованием Фурье — ЛДПФ) понимают соответственно пару преобразований [47, 87, 86]:

$$\mathbf{C} = \mathbf{A}_{2^n}\mathbf{Y}, \tag{10}$$

$$\mathbf{Y} = \mathbf{A}_{2^n}^{-1}\mathbf{C}, \tag{11}$$

где $\mathbf{A}_{2^n}$ и $\mathbf{A}_{2^n}^{-1}$ — соответственно матрицы прямого и инверсного арифметического преобразования размерности $2^n \times 2^n$ (базис преобразования); $\mathbf{Y}$ — вектор истинности d-выходной булевой функции $f(X)$ такой, что

$$\mathbf{Y} = [\mathbf{Y}_d|\mathbf{Y}_{d-1}|\dots|\mathbf{Y}_1]^T = \left[Y^{(0)}Y^{(1)}\dots Y^{(2^n-1)}\right]^T,$$

где T — символ транспонирования; $Y^{(i)}$ — числовое значение, принимаемое d-выходной булевой функцией $f(X)$ на i-м наборе булевых аргументов обычной таблицы истинности (см. пример 1); $\mathbf{C} = [c_0\, c_1\, \dots\, c_{2^n-1}]^T$ — вектор коэффициентов арифметического полинома (**??**) или арифметический спектр булевой функции.

Матрица

$$\mathbf{A}_{2^n} = \left[\begin{array}{c|c} \mathbf{A}_{2^{n-1}} & 0 \\ \hline -\mathbf{A}_{2^{n-1}} & \mathbf{A}_{2^{n-1}} \end{array}\right]$$

является n-й кронекеровской степенью

$$\mathbf{A}_{2^n} = \bigotimes_{j=1}^{n} \mathbf{A}_1$$

базовой матрицы

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix};$$

$$\mathbf{A}_{2^n}^{-1} = \bigotimes_{j=1}^{n} \mathbf{A}_1^{-1},$$

где $\mathbf{A}_1^{-1} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ — базовая матрица обратного преобразования. Матрица $-\mathbf{A}_{2^{n-1}}$ образуется из $\mathbf{A}_{2^{n-1}}$ заменой знаков единичных элементов на противоположные.

Матричные преобразования хорошо алгоритмизируются и удобны для практического применения.

Пример 5.

Пусть задана трехвыходная булева функция, векторы принимаемых значений, которой имеют вид:

$$\begin{aligned} \mathbf{Y}_1 &= [0\ 1\ 0\ 1\ 1\ 0\ 1\ 1]^T, \\ \mathbf{Y}_2 &= [0\ 1\ 1\ 0\ 0\ 1\ 1\ 1]^T, \\ \mathbf{Y}_3 &= [0\ 1\ 1\ 0\ 1\ 0\ 0\ 1]^T, \end{aligned}$$

Тогда

$$\mathbf{Y} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 7 \\ 6 \\ 1 \\ 5 \\ 2 \\ 3 \\ 7 \end{bmatrix}.$$

Выполняя прямое ЛДПФ (??), получим

$$\mathbf{C} = \mathbf{A}_{2^3}\mathbf{Y} = \left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\ \hline -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\ -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \end{array}\right] \cdot \begin{bmatrix} 0 \\ 7 \\ 6 \\ 1 \\ 5 \\ 2 \\ 3 \\ 7 \end{bmatrix} =$$

$$= \begin{bmatrix} 0 \\ 7 \\ 6 \\ -12 \\ 5 \\ -10 \\ -8 \\ 19 \end{bmatrix} \begin{matrix} \\ x_3 \\ x_2 \\ x_2x_3 \\ x_1 \\ x_1x_3 \\ x_1x_2 \\ x_1x_2x_3 \end{matrix} .$$

Наиболее распространенной операцией криптографических алгоритмов блочного шифрования является операция *подстановки* σ, определяемая как биективное отображение конечного множества из k элементов Ω на себя:

$$\sigma = \begin{pmatrix} 1 & 2 & 3 & \dots & k \\ \sigma_1 & \sigma_2 & \sigma_3 & \dots & \sigma_k \end{pmatrix},$$

где $\sigma_i \in \{1, 2, \dots, k\}$ ($\sigma_i \neq \sigma_j$ при $i \neq j$); k — длина подстановки.

Пример 6.

Рассмотрим одну из четырех операций подстановки, S_3-блока криптографического алгоритма американского стандарта *DES*:

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ 15 & 1 & 8 & 14 & 6 & 11 & 3 & 4 & 9 & 7 & 2 & 13 & 12 & 0 & 5 & 10 \end{pmatrix}.$$

Прямое ЛДПФ (??) примет вид:

$$\mathbf{C} = \mathbf{A}_{2^4}\mathbf{Y} =$$
$$= \mathbf{A}_{2^4} \cdot \begin{bmatrix} 15 & 1 & 8 & 14 & 6 & 11 & 3 & 4 & 9 & 7 & 2 & 13 & 12 & 0 & 5 & 10 \end{bmatrix}^T =$$
$$= \begin{bmatrix} 15 \\ -14 \\ -7 \\ 20 \\ -9 \\ 19 \\ 4 \\ -24 \\ -6 \\ 12 \\ 0 \\ -7 \\ 12 \\ -29 \\ 20 \\ 28 \end{bmatrix} \begin{matrix} \\ x_4 \\ x_3 \\ x_3x_4 \\ x_2 \\ x_2x_4 \\ x_2x_3 \\ x_2x_3x_4 \\ x_1 \\ x_1x_4 \\ x_1x_3 \\ x_1x_3x_4 \\ x_1x_2 \\ x_1x_2x_4 \\ x_1x_2x_3 \\ x_1x_2x_3x_4 \end{matrix} .$$

где матрица $\mathbf{A}_{2^4}$ имеет вид:

$$\left[\begin{array}{cccccccc|cccccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
-1 & 1 & 0 & 0 & 1 & -1 & 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 \\
-1 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\
1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1
\end{array}\right]$$

Из анализа структуры матриц $\mathbf{A}_{2^n}$ и $\mathbf{A}_{2^n}^{-1}$ следует, что максимальное количество единичных элементов находится в последней строке обеих матриц. Причем количество единичных элементов с одинаковыми знаками в нижней строке матрицы $\mathbf{A}_{2^n}$ равно 2^{n-1}. Учитывая, что максимальное значение, принимаемое элементами матрицы $\mathbf{Y}$ равно $2^d - 1$ (d — количество реализуемых одновыходных булевых функций), можно сделать вывод о том, что в результирующей матрице $\mathbf{C}$ максимальное абсолютное значение может иметь коэффициент $\mathrm{abs}(c_{2^n-1}) = 2^{n-1}(2^d - 1)$. Для его представления в двоичной системе счисления с учетом необходимости представления знака числа потребуется

$$N_{\mathbf{C}} = \left\lfloor \log(2^{n-1}(2^d - 1)) + 2 \right\rfloor = n + d \tag{12}$$

двоичных разрядов ($\lfloor x \rfloor$ — наибольшее целое число, не превосходящее x).

Для линейных арифметических полиномов проблема больших коэффициентов является еще более критичной. Однако в этом случае причиной большой величины коэффициентов является, прежде всего, большое количество реализуемых булевых функций, что в свою очередь вызвано необходимостью введения избыточных булевых функций, имеющих вспомогательный (служебный) характер.

2.5. Арифметические полиномы и полиномы Жегалкина

Перепишем матричные преобразования (??) и (??) для случая арифметических полиномов (??), задающих одну булеву функцию:

$$\mathbf{R} = \mathbf{A}_{2^n}\mathbf{Y}, \tag{13}$$

$$\mathbf{Y} = \mathbf{A}_{2^n}^{-1}\mathbf{R}, \tag{14}$$

где $\mathbf{R} = [r_0\, r_1\, \ldots\, r_{2^n-1}]^T$ — вектор коэффициентов арифметического полинома (??); $\mathbf{A}_{2^n}$ и $\mathbf{A}_{2^n}^{-1}$ — соответственно матрицы прямого и обратного преобразования из отношений (??) и (??).

Для полиномома Жегалкина [1, 28, 29]

$$G(X) = \bigoplus_{i=0}^{2^n-1} g_i \wedge \left(x_1^{i_1} \wedge x_2^{i_2} \wedge \ldots \wedge x_n^{i_n}\right); \quad G(X),\ g_i \in \{0,1\}$$

могут быть заданы матричные преобразования:

$$\mathbf{G} = \mathbf{A}_{2^n} * \mathbf{Y}, \tag{15}$$

$$\mathbf{Y} = \mathbf{A}_{2^n} * \mathbf{G}, \tag{16}$$

где $\mathbf{G} = [g_0\, g_1\, \ldots\, g_{2^n-1}]^T$ — вектор коэффициентов полинома Жегалкина; $*$ — символ операции умножения матриц, в которой арифметическое сложение заменяется логическим сложением по модулю два [1].

Подставляя вектор $\mathbf{Y}$, выраженный в виде (??), в (??), можно получить [13, 46, 84]

$$\mathbf{G} = \mathbf{A}_{2^n} * \mathbf{A}_{2^n}\mathbf{R}.$$

Выражая вектор $\mathbf{Y}$ в виде (??), получим

$$\mathbf{R} = \mathbf{A}_{2^n}\mathbf{A}_{2^n} * \mathbf{G}.$$

Теорема 2.
Пусть дана система булевых функций — $f_1(X)$, $f_2(X)$, ……, $f_d(X)$ от n переменных $X = x_1, x_2, \ldots, x_n$ и соответствующее им представление полиномами Жегалкина:

$$\begin{cases} G_1(X) = \bigoplus\limits_{i=0}^{2^n-1} g_{1,i} \wedge \left(x_1^{i_1} \wedge x_2^{i_2} \wedge \ldots \wedge x_n^{i_n}\right), \\ G_2(X) = \bigoplus\limits_{i=0}^{2^n-1} g_{2,i} \wedge \left(x_1^{i_1} \wedge x_2^{i_2} \wedge \ldots \wedge x_n^{i_n}\right), \\ \vdots \\ G_d(X) = \bigoplus\limits_{i=0}^{2^n-1} g_{d,i} \wedge \left(x_1^{i_1} \wedge x_2^{i_2} \wedge \ldots \wedge x_n^{i_n}\right). \end{cases} \tag{17}$$

Тогда система булевых функций может единственным образом представлена одним логическим полиномом [47, с. 169]:

$$\begin{aligned} G^{(o)}(X) &= \bigoplus_{i=0}^{2^n-1} \left(g_{d,\,i} \ldots g_{1,\,i}\, g_{0,\,i}\right)_2 x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} = \\ &= \bigoplus_{i=0}^{2^n-1} b_i\; x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \end{aligned} \tag{18}$$

где $b_i = \left(g_{d,\,i} \ldots g_{1,\,i}\, g_{0,\,i}\right)_2 \in \{0,\, 1, \ldots, 2^d - 1\}$; $g_{j,\,i} \in \{0,\, 1\}$ — коэффициенты полиномов Жегалкина (??).

Доказательство теоремы 2.
По аналогии с шагом 1 и шагом 2 алгоритма 1 используем принцип взвешивания применительно к полиномам (??):

- умножение полиномов на веса 2^{j-1} $(j = 1,\, 2, \ldots, d)$:

$$\begin{cases} G_1(X)2^0 = \bigoplus\limits_{i=0}^{2^n-1} b'_{1,i}\; x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \\ G_2(X)2^1 = \bigoplus\limits_{i=0}^{2^n-1} b'_{2,i}\; x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \\ \vdots \\ G_d(X)2^{d-1} = \bigoplus\limits_{i=0}^{2^n-1} b'_{d,i}\; x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \end{cases} \tag{19}$$

где $b'_{j,i} = 2^{j-1} g_{j,i} = (a^{(d-1)}_{j,i} \;\; a^{(d-2)}_{j,i} \;\; \ldots \;\; a^{(1)}_{j,i} \;\; a^{(0)}_{j,i})_2$; $a^{(k)}_{j,i} \in \{0,\, 1\}$; $i = 1,\, 2,\, \ldots,\, 2^n - 1$; $k,\, j = 1,\, 2,\, \ldots,\, d$.

- получение искомого полинома (??) путем суммирования коэффициентов полиномов (??) для всех $j = 1,\, 2, \ldots, d$:

$$\begin{aligned} G^{(o)}(X) &= \bigoplus_{i=0}^{2^n-1} \bigvee_{j=1}^{d} b'_{j,i}\; x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} = \\ &= \bigoplus_{i=0}^{2^n-1} \bigvee_{j=1}^{d} (a^{(d-1)}_{j,i} \;\ldots\; a^{(1)}_{j,i} \;\; a^{(0)}_{j,i})_2 \; x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} = \\ &= \bigoplus_{i=0}^{2^n-1} b_i \; x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \end{aligned}$$

где $b_i = \sum_{j=1}^{d} g_{j,i} 2^{j-1} = (a_{d-1,\,i} \;\; a_{d-2,\,i} \;\; \ldots \;\; a_{1,\,i} \;\; a_{0,\,i})_2$; $a_j,\, i \in \{0,\, 1\}$; $j = 1,\, 2,\, \ldots,\, d$; $i = 1,\, 2,\, \ldots,\, 2^n - 1$.

Принимая во внимание, что в записи $b'_{j,i} = (a^{(d-1)}_{j,i} \ldots a^{(1)}_{j,i} \, a^{(0)}_{j,i})_2$ двоичные цифры $a^{(k)}_{j,i}$ имеют значения

$$a^{(k)}_{j,i} = \begin{cases} 0 & \text{при} \quad k \neq j; \\ g_{j,i} & \text{при} \quad k = j, \end{cases} \tag{20}$$

получим

$$\begin{array}{rcccccc} b'_{0,\,i} = & (0 & 0 & \ldots & 0 & a^{(0)}_{0,\,i})_2 \\ b'_{1,\,i} = & (0 & 0 & \ldots & a^{(1)}_{1,\,i} & 0)_2 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ b'_{d-1,\,i} = & (0 & a^{(d-2)}_{d-2,\,i} & \ldots & 0 & 0)_2 \\ b'_{d,\,i} = & (a^{(d-1)}_{d-1,\,i} & 0 & \ldots & 0 & 0)_2 \\ \hline b_i = & (a_{d-1,\,i} & a_{d-2,\,i} & \ldots & a_{1,\,i} & a_{0,\,i})_2, \end{array} \tag{21}$$

где

$$a_{j,\,i} = \bigvee_{k=0}^{d-1} a^{(k)}_{j,i}$$

($j = 1,\, 2,\, \ldots,\, d;\;\; i = 1,\, 2,\, \ldots,\, 2^n - 1$).

Окончательно, учитывая (??), получим

$$b_i = (g_{d-1,\,i} \;\; g_{d-2,\,i} \;\; \ldots \;\; g_{1,\,i} \;\; g_{0,\,i})_2 \,,$$

где $i = 1,\, 2,\, \ldots,\, 2^n - 1$.

□

Определение 1.
Полином (??) называется обобщенным полиномом Жегалкина [47].

2.6. Выводы

- Существующий парк вычислительной техники традиционно не ориентирован на реализацию массовых логических вычислений.
- Технический прогресс привел к существенным количественным и качественным изменениям в составе и содержании решаемых задач: интеллектуализация управления, распараллеливание процессов управления обработкой информации, возрастание объемов, ценности обрабатываемой и передаваемой информации и, как следствие, массовое применение криптографических методов защиты информации, развитие автоматизированных систем проектирования и лавинообразное возрастание сложности интегральных микросхем.
- Ключ к решению сложившегося противоречия — распараллеливание логических вычислений на основе арифметических методов представления и обработки логических данных.
- Логические функции и системы логических функций могут быть представлены в виде одного арифметического полинома, причем единственным образом.
- Существуют алгебраический и матричный методы построения арифметических полиномов. Матричный метод (разновидность логического дискретного преобразования Фурье) наиболее приспособлен для алгоритмизации методов арифметической логики.
- Существенное препятствие на пути использования методов арифметической логики — высокая сложность арифметических полиномов, большая величина коэффициентов и результатов промежуточных вычислений.
- Линеаризация арифметических полиномов — один из наиболее перспективных путей уменьшения сложности вычислений. Однако для линейных арифметических полиномов еще в более острой форме свойственна проблема больших весовых коэффициентов и результатов промежуточных преобразований.
- Перспективный путь развития арифметической логики и решения проблемы больших весовых коэффициентов и результатов промежуточных вычислений — применение методов модулярной арифметики.
- Разработка методов реализации логических вычислений, основанных на модулярной арифметике, позволит задействовать в указанных целях высокоразвитый математический аппарат и технические средства ЦО сигналов.

3. Введение в модулярные формы арифметической логики

В данной главе рассмотрим новые принципы задания булевых функций, основанные на отображении алгебры логики на арифметику конечного кольца Z_m. Основное содержание данной главы впервые апробировано и опубликовано (принято к опубликованию) в [72, 74–76, 79, 80].

3.1. Модулярные полиномиальные формы

Одномодульной арифметикой будем называть арифметику кольца вычетов Z_m, где m — значение модуля. Согласно [8, 12, 32] модулярные преобразования обладают рядом полезных свойств, позволяющих ограничить величину результатов промежуточных вычислений заданного выражения. При этом схема вычислений имеет вид (рис. **??**). Воспользуемся этой схемой для реализации логических вычислений посредством арифметических полиномов.

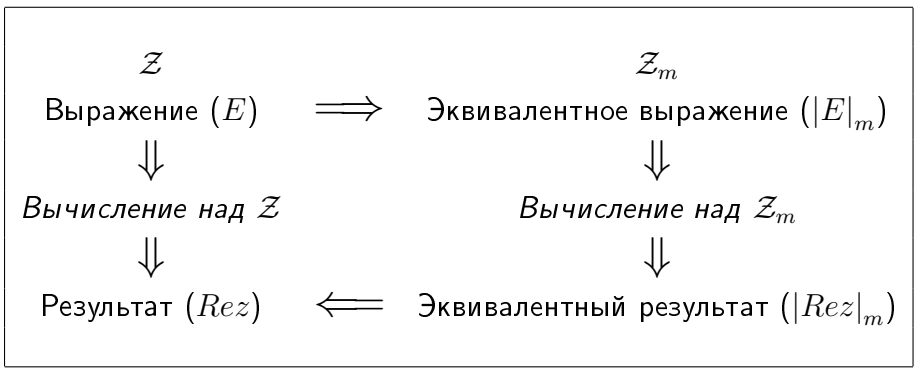

Рис. 5: Принцип ограничения величины промежуточных результатов вычисления выражения(E) над $\mathcal{Z}$ путем вычисления эквивалентного выражения $(|E|_m)$ над $\mathcal{Z}_m$ [8]

Теорема 3.
Если $m > Y_{\max}$, где $Y_{\max}$ — максимальное значение, принимаемое Y, то произвольный кортеж булевых функций может быть представлен арифметическим полиномом:

$$Y = \mu(X) = \left| \sum_{i=0}^{2^n-1} \psi_i \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m , \tag{22}$$

где $\psi_i = |c_i|_m$ $(i = 0, 1, \dots, 2^n - 1)$.

Доказательство теоремы 3.
Рассмотрим два варианта доказательств, которые определят два варианта алгоритма получения (**??**).

(i) Пусть дан арифметический полином $D(X)$ (**??**). Тогда согласно теории сравнений [18] справедливо:

$$|Y|_m = \left| \sum_{i=0}^{2^n-1} c_i \, x_1^{i_1} \dots x_n^{i_n} \right|_m = \left| \sum_{i=0}^{2^n-1} |c_i|_m \, x_1^{i_1} \dots x_n^{i_n} \right|_m . \tag{23}$$

Но при условии $Y < m$ (Y — неотрицательное) выполняется $|Y|_m = Y$. Следовательно, при $Y < m$:

$$Y = \left| \sum_{i=0}^{2^n-1} |c_i|_m \, x_1^{i_1} \dots x_n^{i_n} \right|_m = \left| \sum_{i=0}^{2^n-1} \psi_i \, x_1^{i_1} \dots x_n^{i_n} \right|_m , \tag{24}$$

где $\psi_i = |c_i|_m$.

□

Отсюда вытекает алгоритм получения (??), заключающийся в выполнении следующих шагов (рис. ??):

Шаг. 1. Построение (??) посредством алгоритма 1.

Шаг. 2. Определение (??) посредством выражения (??).

Шаг 1	c_0	c_1	$\dots$	c_{2^n-1}
	$\Downarrow$	$\Downarrow$		$\Downarrow$
Шаг 2	$\psi_0 = \lvert c_0 \rvert_m$	$\psi_1 = \lvert c_1 \rvert_m$	$\dots$	$\psi_{2^n-1} = \lvert c_{2^n-1} \rvert_m$

Рис. 6: Пояснения к первому варианту алгоритма получения (??)

(ii) Так как $f_j(X) \in \{0, 1\}$ и $m > Y$ с учетом предыдущих рассуждений, полином (??) можно представить в виде:

$$f_j(X) = \mu'_j(X) = \left| \sum_{i=0}^{2^n-1} \psi'_{j,\,i} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m , \tag{25}$$

где $\psi'_{j,\,i} = |r_{j,\,i}|_m$ ($j = 1, 2, \dots, d$; $i = 0, 1, \dots, 2^n - 1$). Далее, приведением (??) и (??) по модулю m получим:

$$\mu''_j(X) = \sum_{i=0}^{2^n-1} \psi''_{j,\,i} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}, \tag{26}$$

где $\psi''_{j,\,i} = \left|\psi'_{j,\,i} b_j\right|_m$, $b_j = |2^{j-1}|_m$, при $j = 1, 2, \dots, d$;
$i = 0, 1, \dots, 2^n - 1$;

$$Y = \mu(X) = \left| \sum_{i=0}^{2^n-1} \sum_{j=1}^{d} \psi''_{j,\,i} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m . \tag{27}$$

Преобразование (??) дает (??).

□

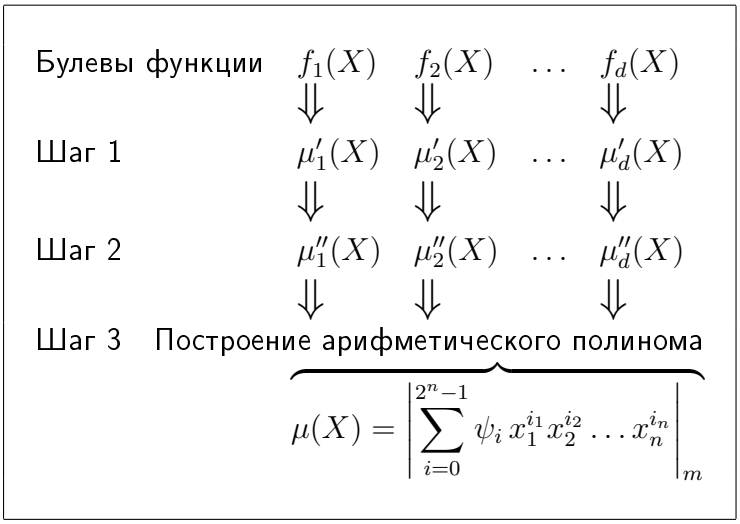

Рис. 7: Пояснения ко второму варианту алгоритма получения (??)

Таким образом, второй вариант алгоритма получения (??) состоит в выполнении трех шагов, заключающихся соответственно в построении (??), (??) и (??) (рис. ??).

Замечание 1.

$$m \geq 2^d.$$

Определение 2.
Выражение (??) будем называть представлением булевой функции $f(X)$ на основе модулярной формы арифметического полинома.

Сравнительный анализ арифметических полиномов $D(X)$ и $\mu(X)$ можно выполнить на примере некоторых элементарных булевых функций (табл. ??).

Таблица 2:

$f(X)$	$D(X)$	$\mu(X)$
$\overline{x}_i$	$1 - x_i$	$\lvert 1 + (m-1)x_i \rvert_m$
$x_1 \wedge x_2$	$x_1 x_2$	$x_1 x_2$
$x_1 \vee x_2$	$x_1 + x_2 - x_1 x_2$	$\lvert x_1 + x_2 + (m-1)x_1 x_2 \rvert_m$
$x_1 \oplus x_2$	$x_1 + x_2 - 2x_1 x_2$	$\lvert x_1 + x_2 + (m-2)x_1 x_2 \rvert_m$
$\overline{x_1 \wedge x_2}$	$1 - x_1 x_2$	$\lvert 1 + (m-1)x_1 x_2 \rvert_m$
$\overline{x_1 \vee x_2}$	$1 - x_1 - x_2 + x_1 x_2$	$\lvert 1 + (m-1)x_1 + (m-1)x_2 + x_1 x_2 \rvert_m$

Используя схему вычислений, представленную на рис. ??, схемы вычислений — рис. ?? и ?? — можно представить в виде рис. ?? и ??.

Следствие 1.
Коэффициенты ψ_i ($i = 0, 1, \dots, 2^n - 1$) арифметического полинома $\mu(X)$ (??) лежат в области целых неотрицательных чисел, а их числовой диапазон равен значению модуля m.

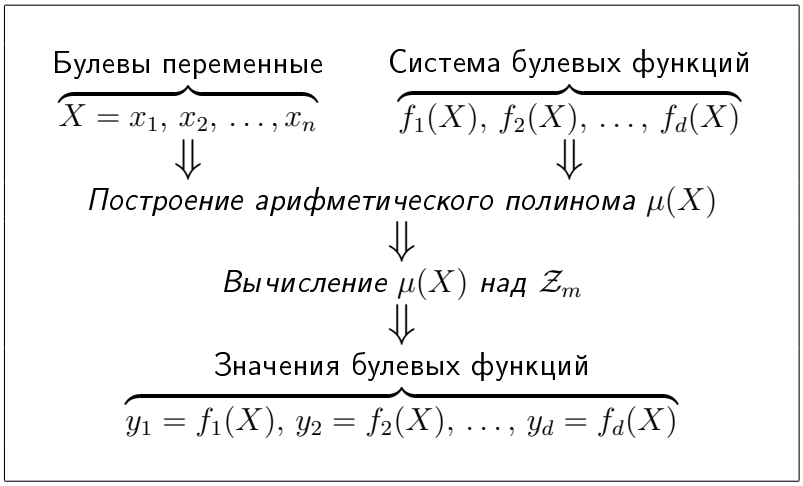

Рис. 8: Схема вычисления произвольной системы булевых функций над $\mathcal{Z}_m$

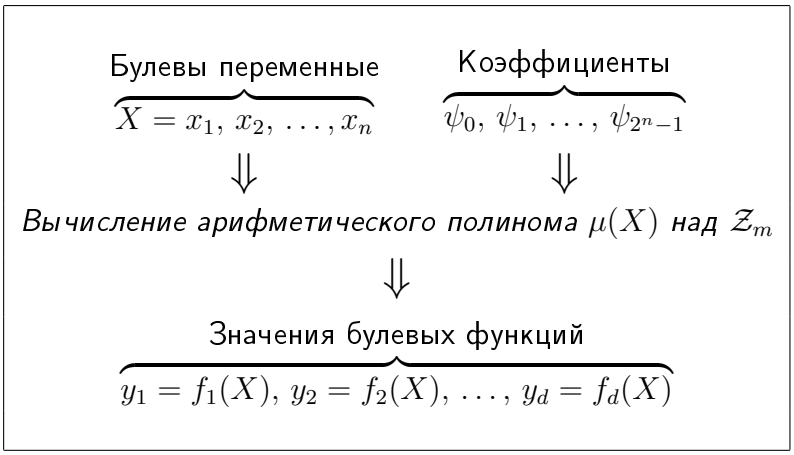

Рис. 9: Схема вычисления заданной системы булевых функций над $\mathcal{Z}_m$

Следствие 2.
Если для одной и той же системы булевых функций заданы два арифметических полинома $D(X)$ (**??**) и $\mu(X)$ (**??**), а K_1 и K_2 — количество членов этих полиномов, то $K_2 \leq K_1$.

Для пояснения следствия 2 рассмотрим следующий пример.

Пример 7.
Рассмотрим систему булевых функций (**??**), которой согласно выражению (**??**) (пример 2) соответствует арифметический полином

$$Y = D(X) = 3 - 3x_1 - 3x_2 + 4x_1x_2.$$

Применение теоремы 3 в общем случае дает:

$$Y = \mu(X) = |3 + (m-3)x_1 + (m-3)x_2 + 4x_1x_2|_m .$$

При $m = 4$ получим

$$\mu(X) = |3 + x_1 + x_2|_4 .$$

Таким образом, следствие 2 указывает на то, что модулярная форма арифметического полинома (**??**) как минимум не усложняет полиномиальной

формы представления систем булевых функций по показателям K_1 и K_2, а как максимум — позволяет уменьшить сложность арифметических полиномов за счет сокращения коэффициентов, кратных m. Следовательно, значение модуля m может выбираться не только по критерию собственной минимальности, но и по критерию минимальности K_2.

Лемма 1.

Если кортеж булевых функций (??) задан линейным арифметическим полиномом (??), то при $m > U_{\max}$ справедлива модулярная форма линейного арифметического полинома:

$$\begin{aligned} U = \lambda(X) &= \left| \omega_0 + \sum_{i=1}^{n} \omega_i x_i \right|_m = \\ &= |\omega_0 + \omega_1 x_1 + \ldots + \omega_n x_n|_m \,, \end{aligned} \tag{28}$$

где $\omega_j = |d_j|_m$ $(j = 0, 1, \ldots, n)$.

Доказательство леммы 1.

Рассмотрим арифметический полином

$$\begin{aligned} U &= \left| \omega_0^\star \sum_{i=1}^{s} \omega_i^\star z_i \right|_m = \\ &= |\omega_0^\star + \omega_1^\star z_1 + \ldots + \omega_n^\star z_n|_m \,, \end{aligned}$$

который при подстановке $\omega_0^\star = \omega_0$, $\omega_i^\star = \omega_i$, $z_i = x_i$, $s = n$ соответствует полиному (??). Пусть $U = Y$, $\omega_0^\star = \psi_0$, $\omega_i^\star = \psi_i$, $z_i = x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}$, $s = 2^n - 1$, тогда доказательство (??) сводится к повторению п. (i) доказательства теоремы 3. □

Замечание 2.

Значения параметра t оператора $\Xi^t\{U\}$ при переходе от (??) к (??) не изменяются.

Определение 3.

Выражение (??) будем называть представлением булевой функции на основе модулярной формы линейного арифметического полинома.

Пример 8.

Для системы булевых функций (??), заданной линейным арифметическим полиномом (??), параметр t оператора $\Xi^t\{U\}$ имеет максимальное значение $t_{\max} = 6$ и $U_{\max} = 36$. Выберем $m = 2^6 > 36$. Тогда

$$U = |20 + 60x_1 + 52x_2 + 16x_3|_{64} \,.$$

Пусть $x_1 \odot x_2 \odot x_3 = 0 \odot 1 \odot 1$. Следовательно,
$U = |88|_{64} = (24)_{10} = (0\,1\,1\,0\,0\,0)_2$. Окончательно имеем:

$$\begin{aligned} f_A(X) &= \Xi^2\{(0\,1\,1\,0\,0\,0)_2\} = 0, \\ f_B(X) &= \Xi^4\{(0\,1\,1\,0\,0\,0)_2\} = 1, \\ f_C(X) &= \Xi^6\{(0\,1\,1\,0\,0\,0)_2\} = 0. \end{aligned}$$

Связь оператора $\Xi^t\{U\}$ с модулярной арифметикой устанавливается отношением:

$$\Xi^t\{U\} = \left| \left\lfloor \frac{U}{2^t} \right\rfloor \right|_2 .$$

Замечание 3.
Если для получения U используются избыточные булевы функции с номерами, превышающими $t_{\max}$ — максимальное значение параметра t оператора $\Xi^t\{U\}$, то модулю m можно присвоить значение $2^{t_{\max}}$. В этом случае вместо U в (??) следует писать $u = |U|_{2^{t_{\max}}}$, при этом $u \le U$.

Для пояснения замечания 3 рассмотрим следующий пример.

Пример 8.
Для системы булевых функций

$$\begin{aligned} f_A(X) &= \overline{x_1 \wedge x_2 \wedge x_3} = \Xi^3\{6 - x_1 - x_2 - x_3\}, \\ f_B(X) &= \overline{x_1 \oplus x_2 \oplus x_3} = \Xi^1\{1 + x_1 + x_2 + x_3\} \end{aligned}$$

линейный арифметический полином $L(X)$ имеет вид:

$$U = 2^3 f_B(X) + 2^0 f_A(X) = 14 + 7x_1 + 7x_2 + 7x_3.$$

Так как $t_{\max} = 4$, то в соответствии с замечанием 3 получим $m = 16$ и

$$u = \lambda(X) = |14 + 7x_1 + 7x_2 + 7x_3|_{16}\,.$$

Пусть $x_1 \odot x_2 \odot x_3 = 1 \odot 1 \odot 1$, тогда $u = |35|_{16} = (2)_{10} = (0\,0\,1\,0)_2$ и

$$\begin{aligned} f_A(X) &= \Xi^3\{(0\,0\,1\,0)_2\} = 0, \\ f_B(X) &= \Xi^4\{(0\,0\,1\,0)_2\} = 0. \end{aligned}$$

Т. е. вместо шести разрядов, необходимых для представления U в соответствии с (??) и (??), замечание 3 позволяет обойтись только четырьмя разрядами.

Таким образом основным свойством модулярной формы полинома (??) является уменьшение числового диапазона, требуемого для его вычисления. Прежде чем сделать более точную оценку числового диапазона, рассмотрим принципы реализации матричных преобразований, основанных на модулярной арифметике.

3.2. Конъюнктивные модулярные преобразования

Теорема 4.
Если для d-выходной булевой функции $f(X)$ задана пара ЛДПФ (??) и (??) и $m > Y_{\max}$, где $Y_{\max}$ — максимальное значение, принимаемое Y, то справедлива следующая модулярная форма преобразований:

$$\boldsymbol{\Psi} = \mathbf{A}_{2^n}\mathbf{Y} \pmod m, \tag{29}$$

$$\mathbf{Y} = \mathbf{A}_{2^n}^{-1}\boldsymbol{\Psi} \pmod m, \tag{30}$$

где $\mathbf{A}_{2^n}$ и $\mathbf{A}_{2^n}^{-1}$ — соответственно матрицы прямого и инверсного арифметического преобразования; $\mathbf{Y}$ и $\boldsymbol{\Psi}$ — соответственно вектор истинности булевой функции $f(X)$ и вектор коэффициентов модулярной формы арифметического полинома $\mu(X)$ (??). Запись $\bmod m$ означает, что арифметические операции, используемые при произведении матриц $\mathbf{A}_{2^n}$ и $\mathbf{A}_{2^n}^{-1}$ на вектор-столбец $\mathbf{Y}$ или $\boldsymbol{\Psi}$, выполняются по модулю m.

Для доказательства теоремы 4 необходимо учесть взаимооднозначность матричной (??), (??) и полиномиальной (??) формам представления системы

булевых функций [47]. Тогда справедливость (??) и (??) вытекает из справедливости (??).

□

Полученная пара преобразований имеет много общего с теоретико-числовыми преобразованиями методов цифровой обработки сигналов (number theoretic transforms) [35, 41, 98].

Определение 4.
Преобразования (??) и (??) будем соответственно называть модулярной формой прямого и обратного матричного арифметического преобразования или логическими теоретико-числовыми преобразованиями (ЛТЧП, logical number theoretic transforms).

Учитывая, что $|-1|_m = m-1$, выражение (??) можно переписать в другой форме:

$$\mathbf{\Psi} = \mathbf{M}_{2^n}\mathbf{Y} \pmod m, \tag{31}$$

где $\mathbf{M}_{2^n} = |\mathbf{A}_{2^n}|_m$. Запись $|\mathbf{A}_{2^n}|_m$ означает, что отрицательные элементы (единицы) матрицы $\mathbf{A}_{2^n}$ заменяются на $m-1$.

Для уменьшения вычислительных затрат к (??) и (??) могут быть применены методы факторизации матриц [90]

$$\mathbf{A}_{2^n} = \mathbf{A}_{2^n}^{(1)}\mathbf{A}_{2^n}^{(2)}\ldots\mathbf{A}_{2^n}^{(n)}$$

и

$$\mathbf{A}_{2^n}^{-1} = \widetilde{\mathbf{A}}_{2^n}^{(1)}\widetilde{\mathbf{A}}_{2^n}^{(2)}\ldots\widetilde{\mathbf{A}}_{2^n}^{(n)},$$

предложенные в [47]. При этом ожидается снижение вычислительной сложности преобразования в

$$\xi = \frac{3^n - 2^n}{n2^{n-1}}$$

раз [47].

Пример 10.
Продемонстрируем применение ЛТЧП (??) и (??) к двухвыходной булевой функции (??) с матрицей истинности, заданной табл. ?? (см. для сопоставления пример 2):

$$\mathbf{\Psi} = \mathbf{A}_{2^2}\mathbf{Y}\ (\mathbf{mod}\,2^2) = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ \hline -1 & 1 & 0 & 0 \\ 1 & -1 & -1 & 1 \end{array}\right] \cdot \begin{bmatrix} 3 \\ 0 \\ 0 \\ 1 \end{bmatrix} \pmod{2^2} =$$

$$= \begin{bmatrix} 3 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{matrix} \\ x_2 \\ x_1 \\ x_1x_2 \end{matrix},$$

$$\mathbf{Y} = \mathbf{A}_{2^2}^{-1}\mathbf{\Psi}\ (\mathrm{mod}\,2^2) = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \hline 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{array}\right] \cdot \begin{bmatrix} 3 \\ 1 \\ 1 \\ 0 \end{bmatrix} \pmod{2^2} =$$

$$= \begin{bmatrix} 3 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

Пример 11.
Применение прямого ЛТЧП (??) при $m = 2^3$ к трехвыходной булевой функции из примера 5 дает результат:

$$\begin{aligned} \mathbf{\Psi} &= \mathbf{M}_{2^3}\mathbf{Y} \pmod{2^3} = \\ &= \left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 7 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 7 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 7 & 7 & 1 & 0 & 0 & 0 & 0 \\ \hline 7 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 7 & 0 & 0 & 7 & 1 & 0 & 0 \\ 1 & 0 & 7 & 0 & 7 & 0 & 1 & 0 \\ 7 & 1 & 1 & 7 & 1 & 7 & 7 & 1 \end{array}\right] \cdot \begin{bmatrix} 0 \\ 7 \\ 6 \\ 1 \\ 5 \\ 2 \\ 3 \\ 7 \end{bmatrix} \pmod{2^3} = \\ &= \begin{bmatrix} 0 \\ 7 \\ 6 \\ 4 \\ 5 \\ 6 \\ 0 \\ 3 \end{bmatrix} \begin{array}{l} \\ x_3 \\ x_2 \\ x_2x_3 \\ x_1 \\ x_1x_3 \\ x_1x_2 \\ x_1x_2x_3 \end{array}. \end{aligned}$$

Пример 12.
Для примера 6 реализации первой операции подстановки блока S_3 криптографического алгоритма стандарта DES прямое ЛТЧП (??) при $m = 2^4$ примет вид:

$$\begin{gathered} \mathbf{\Psi} = \mathbf{M}_{2^4}\mathbf{Y} \pmod{2^4} = \\ = \mathbf{M}_{2^4} \cdot \left[\begin{array}{cccccccccccccccc} 15 & 1 & 8 & 14 & 6 & 11 & 3 & 4 & 9 & 7 & 2 & 13 & 12 & 0 & 5 & 10 \end{array}\right]^T = \\ = \left[\begin{array}{cccccccccccccccc} 15 & 2 & 9 & 4 & 7 & 3 & 4 & 8 & 10 & 12 & 0 & 9 & 12 & 3 & 4 & 12 \end{array}\right]^T. \end{gathered}$$

Таким образом, значения коэффициентов вектора $\mathbf{\Psi}$ лежат в интервале $0 \le \psi_i < m$, $i = 0, 1, \ldots, 2^n - 1$, который сохраняется и при выполнении промежуточных вычислений (сравним
с $-28 \le c_i \le 28$ в примере 5).

По аналогии с ЛДПФ в качестве оценки сложности ЛТЧП выберем размер матрицы-спектра. Для представления элементов матрицы $\mathbf{\Psi}$ потребуется $N_{\mathbf{\Psi}} = \lceil \log_2 m \rceil$ ($\lceil x \rceil$ — наименьшее целое число равное или превышающее x) двоичных разрядов или, при $m = 2^d$, $N_{\mathbf{\Psi}} = d$ двоичных разрядов, что в

$$\frac{N_{\mathbf{C}}}{N_{\mathbf{\Psi}}} = \frac{n}{d} + 1 \tag{32}$$

раз меньше по сравнению с количеством разрядов $N_{\mathbf{C}}$, необходимых для представления элементов матрицы $\mathbf{C}$ (??).

Так как $N_{\mathbf{C}}$ и $N_{\mathbf{\Psi}}$ — это максимальные размерности (количество двоичных разрядов) коэффициентов арифметических полиномов (??) и (??) соответственно, то оценка (??) применима и к арифметическому полиному (??).

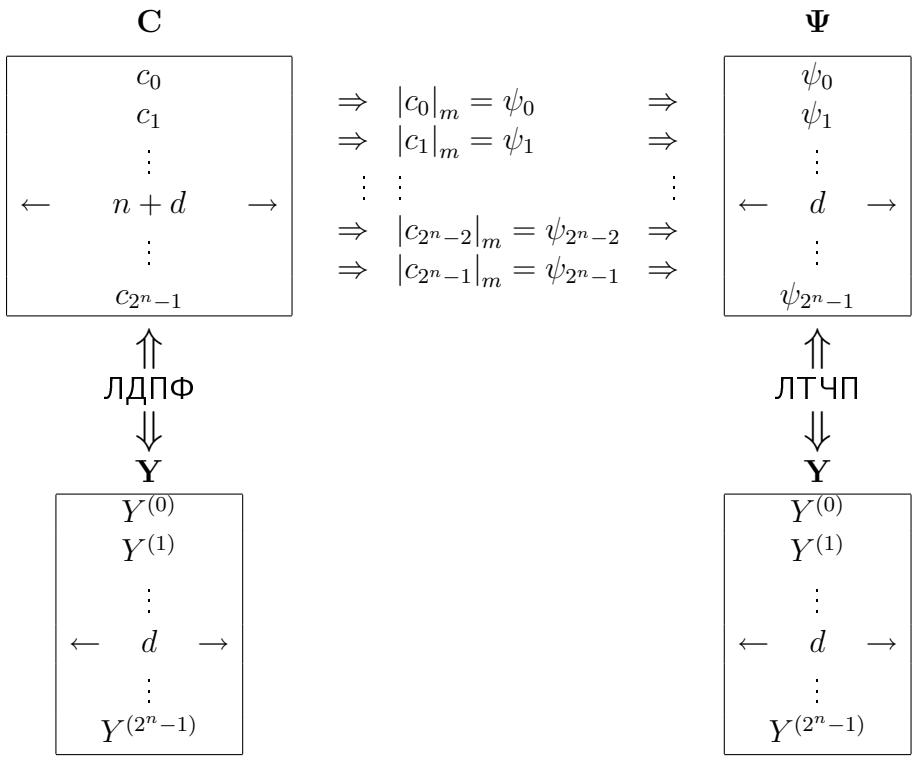

Рис. 10: Геометрическая интерпретация ЛТЧП и его связь с ЛДПФ

На рис. **??** представлена геометрическая интерпретация получаемого выигрыша в виде представления матриц $\mathbf{Y}$, $\mathbf{C}$ и $\mathbf{\Psi}$ (здесь ширина матриц означает количество двоичных символов, которые необходимы для представления элементов матриц-столбцов, ПЛДПФ и ОЛДПФ — соответственно прямое и обратное ЛДПФ, а ПЛТЧП и ОЛТЧП — соответственно прямое и обратное ЛТЧП).

Однако этот выигрыш не удается сохранить для линейных арифметических полиномов, для которых числовой диапазон представления коэффициентов гарантированно можно уменьшить только в два раза — за счет переноса вычислений в область неотрицательных чисел (в некоторых случаях может быть использовано также замечание 3. Препятствием для дальнейшего уменьшения, используемого числового диапазона является большая величина модуля m.

3.3. Выводы

- Предложен метод реализации систем булевых функций посредством модулярной формы арифметического полинома, которая классифицирована как *арифметическое обобщение полинома Жегалкина* [1, 28].
- Установлено, что модулярная форма арифметического полинома позволяет существенно ослабить проблему больших коэффициентов и ограничить величину результатов промежуточных вычислений.
- Предложено прямое аналитическое преобразование систем булевых функций непосредственно в модулярную форму арифметического полинома.

- Установлена взаимосвязь между немодулярной и модулярной формами арифметического полинома.
- Построены прямое и обратное матричные преобразования между системами булевых функций и модулярной формой арифметических полиномов. Полученные матричные преобразования определены как *логические теоретико-числовые преобразования.*
- Показано, что логические теоретико-числовые преобразования могут иметь структуру, аналогичную быстрым конъюнктивным преобразованиям, предложенным В. Малюгиным.

4. Многомерные формы, основанные на Китайской теореме

В данной главе введенные ранее модулярные арифметико-логические формы развиваются в многомерные отображения в кольце $Z_{m_1, m_2, ..., m_v}$. Основное содержание данной главы впервые апробировано и опубликовано в [70, 72, 74–76, 79, 80].

4.1. Полиномиальные многомерные формы, основанные на Китайской теореме об остатках

При моделировании реальных цифровых устройств значения коэффициентов линейных арифметических полиномов могут превышать величину 2^{100} [87, 102]. В таких случаях требуются более радикальные пути уменьшения числовых диапазонов.

Пусть модуль m для (??) и (??) обладает свойством

$$m = m_1 m_2 \dots m_k,$$

причем $\gcd(m_i, m_j) = 1$; $i, j = 1, 2, \dots, v$; $i \neq j$ (здесь и далее $\gcd(a, b)$ — наибольший общий делитель a и b). Тогда в соответствии с Китайской теоремой об остатках Y можно взаимно однозначно отобразить в v-мерный вектор $\{Y\} = (\phi_1, \phi_2, \dots, \phi_v)$, где $\phi_k = |Y|_{m_k}$ $(k = 1, 2, \dots, v)$ [8, 12, 32]. При этом $Y \in Z_m$. Применение для каждого вычета ϕ_k $(k = 1, 2, \dots, v)$ рассмотренного выше подхода позволяет получить следующие положения.

Теорема 5.
Если $m > Y_{\max}$, причем $m = \prod_{k=1}^{v} m_k$ и $\gcd(m_i, m_j) = 1$ $(i, j = 1, 2, \dots, v; i \neq j)$, то произвольный кортеж булевых функций может быть однозначно представлен системой модулярных форм арифметических полиномов:

$$(33) \qquad \begin{cases} \phi_1 = \mu_1(X) = \left| \sum\limits_{i=0}^{2^n-1} \psi_{i,1}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_{m_1}, \\ \phi_2 = \mu_2(X) = \left| \sum\limits_{i=0}^{2^n-1} \psi_{i,2}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_{m_2}, \\ \vdots \\ \phi_v = \mu_v(X) = \left| \sum\limits_{i=0}^{2^n-1} \psi_{i,v}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_{m_v}, \end{cases}$$

где $\psi_{i,\,k} = |c_i|_{m_k}$ $(i = 0,\, 1,\, \ldots,\, 2^n - 1;\ k = 1,\, 2,\, \ldots,\, v)$.

Доказательство теоремы 5.

По аналогии с доказательством теоремы 3 рассмотрим два варианта доказательств, определяющих два варианта построения алгоритма получения (??).

(i) Так как (??) справедливо для любых значений модуля $m > 1$, то справедливы и v записей (??) для различных значений модуля $m > 1$:

$$
\begin{aligned}
|Y|_{m_1} &= \left| \sum_{i=0}^{2^n-1} c_i\, x_1^{i_1} x_2^{i_2} \ldots\, x_n^{i_n} \right|_{m_1}, \\
|Y|_{m_2} &= \left| \sum_{i=0}^{2^n-1} c_i\, x_1^{i_1} x_2^{i_2} \ldots\, x_n^{i_n} \right|_{m_2}, \\
&\vdots \\
|Y|_{m_v} &= \left| \sum_{i=0}^{2^n-1} c_i\, x_1^{i_1} x_2^{i_2} \ldots\, x_n^{i_n} \right|_{m_v}
\end{aligned}
$$

или

$$
\begin{aligned}
|Y|_{m_1} &= \left| \sum_{i=0}^{2^n-1} |c_i|_{m_1}\, x_1^{i_1} x_2^{i_2} \ldots\, x_n^{i_n} \right|_{m_1}, \\
|Y|_{m_2} &= \left| \sum_{i=0}^{2^n-1} |c_i|_{m_2}\, x_1^{i_1} x_2^{i_2} \ldots\, x_n^{i_n} \right|_{m_2}, \\
&\vdots \\
|Y|_{m_v} &= \left| \sum_{i=0}^{2^n-1} |c_i|_{m_v}\, x_1^{i_1} x_2^{i_2} \ldots\, x_n^{i_n} \right|_{m_v}.
\end{aligned}
\tag{34}
$$

На основании Китайской теоремы об остатках система вычетов

$$\phi_1 = |Y|_{m_1},\ \phi_2 = |Y|_{m_2},\ \ldots,\ \phi_v = |Y|_{m_v}$$

имеет единственное решение Y, если выполнены условия: $Y_{\max} < \prod_{k=1}^{v} m_k$; $\gcd(m_i,\, m_j) = 1$ $(i,\, j = 1,\, 2,\, \ldots,\, v;\ i \neq j)$. Так как эти условия в теореме 5 определены, то и (??) имеет единственное решение Y.

□

Из данного доказательства следует, что построить (??) можно посредством выполнения двух шагов:

Шаг. 1. Построение (??) посредством алгоритма 1.

Шаг. 2. Построение системы (??) посредством (??).

(ii) Используем (??) для построения системы арифметических полиномов по v заданным модулям, отвечающим требованиям однозначности пред-

ставления (??):

$$P_j(X) = \begin{cases} \mu'_{j,1}(X) = \left| \sum\limits_{i=0}^{2^n-1} \psi'_{j,i,1}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_{m_1}, \\ \mu'_{j,2}(X) = \left| \sum\limits_{i=0}^{2^n-1} \psi'_{j,i,2}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_{m_2}, \\ \vdots \\ \mu'_{j,v}(X) = \left| \sum\limits_{i=0}^{2^n-1} \psi'_{j,i,v}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_{m_v}, \end{cases} \tag{35}$$

где $\psi'_{j,i,k} = |r_{j,i}|_{m_k}$ ($j = 1, 2, \dots, d$; $i = 1, 2, \dots, 2^n - 1$; $k = 1, 2, \dots, v$). Аналогично используя (??), получим:

$$\begin{cases} \mu''_{j,1}(X) = \sum\limits_{i=0}^{2^n-1} \psi''_{j,i,1}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}, \\ \mu''_{j,2}(X) = \sum\limits_{i=0}^{2^n-1} \psi''_{j,i,2}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}, \\ \vdots \\ \mu''_{j,v}(X) = \sum\limits_{i=0}^{2^n-1} \psi''_{j,i,v}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}, \end{cases} \tag{36}$$

где

$$\psi''_{j,i,k} = \left|\psi'_{j,i,k} b_{j,k}\right|_{m_k};$$
$$b_{j,1} = \left|2^{j-1}\right|_{m_1},\ b_{j,2} = \left|2^{j-1}\right|_{m_2},\ \dots\ b_{j,s} = \left|2^{j-1}\right|_{m_s}$$

($j = 1, 2, \dots, d$; $i = 0, 1, \dots, 2^n - 1$; $k = 1, 2, \dots, v$). Окончательно, обобщая (??) для модулей $m_1, m_2, \dots, m_v$ получают (??) с помощью выражений

$$\psi_{j,k} = \left| \sum_{j=1}^{d} \psi''_{j,i,k} \right|_{m_k} \tag{37}$$

($i = 0, 1, \dots, 2^n - 1$; $k = 1, 2, \dots, v$).

□

Алгоритм получения (??) состоит в выполнении трех шагов, заключающихся соответственно в построении систем полиномов (??), (??) и собственно (??) при помощи (??).

Достоинством первых вариантов алгоритмов формирования арифметических полиномов (??) и (??) является простота и наглядность. Особенностью вторых вариантов является формирование соответствующих арифметических полиномов, обеспечивающих модулярность преобразований на всех этапах реализации алгоритма, что дает „удержание» промежуточных результатов вычислений в заданном числовом диапазоне.

Определение 5.
Систему арифметических полиномов (??) будем называть полиномиальной многомерной формой представления булевых функций, основанной на Китайской теореме об остатках.

Замечание 4.
Модулярные формы (??) и (??) связаны отношениями:

$$\{Y\} = (\phi_1, \phi_2, \ldots, \phi_v),$$
$$(\psi_{i,1}, \psi_{i,2}, \ldots, \psi_{i,v}) = \{\psi_i\} = |c_i|_m \ (i = 0, 1, \ldots, 2^n - 1).$$

Для каждого арифметического полинома системы (??) справедливы и следствия 1 и 2 (при этом вместо m необходимо рассматривать соответствующий модуль m_j $(j = 1, 2, \ldots, v)$.

Лемма 2.
Если кортеж булевых функций (??) задан линейным арифметическим полиномом $L(X)$ (??), то при $m > U_{max}$, где $m = \prod_{k=1}^{v} m_k$, причем $\gcd(m_i, m_j) = 1$ $(i, j = 1, 2, \ldots, v;\ i \neq j)$, справедлива следующая модулярная форма линейного арифметического полинома:

$$(38) \qquad \begin{cases} \phi_1 = \lambda_1(X) = |\omega_{0,1} + \omega_{1,1}x_1 + \ldots + \omega_{n,1}x_n|_{m_1}, \\ \phi_2 = \lambda_2(X) = |\omega_{0,2} + \omega_{1,2}x_1 + \ldots + \omega_{n,2}x_n|_{m_2}, \\ \vdots \\ \phi_v = \lambda_v(X) = |\omega_{0,v} + \omega_{1,v}x_1 + \ldots + \omega_{n,v}x_n|_{m_v}, \end{cases}$$

где $\omega_{j,k} = |d_i|_{m_k}$ $(j = 0, 1, \ldots, n;\ k = 1, 2, \ldots, v)$.

Справедливость (??) следует из применения доказательства справедливости (??) для каждого номера модуля (??) в отдельности и из Китайской теоремы об остатках.

Определение 6.
Систему арифметических полиномов (??) будем называть линейной полиномиальной многомерной формой представления булевых функций, основанной на Китайской теореме об остатках.

Замечание 5.
Модулярные формы (??) и (??) связаны отношениями:

$$\{U\} = (\phi_1, \phi_2, \ldots, \phi_v),$$
$$(\omega_{j,1}, \omega_{j,2}, \ldots, \omega_{j,v}) = \{\omega_j\} = \{\omega_j\} = |d_j|_m \ (j = 0, 1, \ldots, n).$$

Замечание 3 к (??) не применимо.

Для упрощения изложения в дальнейшем не будем различать числа Y и U.

Решение системы равенств

$$\begin{cases} Y = \phi_1 \pmod{m_1}, \\ Y = \phi_2 \pmod{m_2}, \\ \vdots \\ Y = \phi_v \pmod{m_v} \end{cases}$$

дает Китайская теорема об остатках. Для этого будем использовать запись

$$(39) \qquad Y = \mathbf{CRT}_{k=1}^{v} \phi_k \pmod{m_k}.$$

В современной трактовке Китайской теоремы об остатках для вычисления (??) используется формула

$$Y = \mathbf{CRT}_{k=1}^{v}\phi_k \pmod{m_k} = \left|\phi_1 B_1 + \phi_2 B_2 + \ldots + \phi_v B_v\right|_m , \tag{40}$$

где $B_k = q_k m m_k^{-1}$; q_k находится из $q_k m m_k^{-1} \equiv 1 \pmod{m_k}$
$(k = 1, 2, \ldots, v)$.

Рассмотрим далее два примера реализации систем булевых функций с использованием Китайской теоремы об остатках.

Пример 13.
Рассмотрим реализацию универсального многополюсника ((2, 16)-полюсника), арифметическая форма выходных функций которого имеет вид [47]:

$$\begin{array}{ll}
f_1 = \overline{x_1 \oplus x_2} = 1 - x_1 - x_2 + 2x_1x_2, & f_9 = \overline{x_1} \wedge x_2 = x_2 - x_1x_2, \\
f_2 = \overline{x_1 \vee x_2} = 1 - x_1 - x_2 + x_1x_2, & f_{10} = x_1 \wedge x_2 = x_1x_2, \\
f_3 = \overline{x_1 \wedge \overline{x_2}} = 1 - x_1 + x_1x_2, & f_{11} = \overline{x_1} = 1 - x_1, \\
f_4 = \overline{\overline{x_1} \wedge x_2} = 1 - x_2 + x_1x_2, & f_{12} = x_1, \\
f_5 = \overline{x_1 \wedge x_2} = 1 - x_1x_2, & f_{13} = \overline{x_2} = 1 - x_2, \\
f_6 = x_1 \oplus x_2 = x_1 + x_2 - 2x_1x_2, & f_{14} = x_2, \\
f_7 = x_1 \vee x_2 = x_1 + x_2 - x_1x_2, & f_{15} = 1, \\
f_8 = x_1 \wedge \overline{x_2} = x_1 - x_1x_2, & f_{16} = 0.
\end{array}$$

Результирующий арифметический полином в соответствии с (??) имеет вид [47]:

$$D(X) = 21535 + 1241x_1 + 4437x_2.$$

Максимальное значение, принимаемое полином при $x_1 = 1$ и $x_2 = 1$, равно 27213. Поэтому выберем попарно простые модули $m_1 = 11$, $m_1 = 13$, $m_1 = 15$ и $m_1 = 16$ такие, что $M = 11 \cdot 13 \cdot 15 \cdot 16 = 34320 > 27213$. Используя один из алгоритмов, вытекающих из доказательства теоремы 5, получим модулярные формы арифметических полиномов в соответствии с (??):

$$\begin{aligned}
\mu_1(X) &= \left|8 + 9x_1 + 4x_2 + 0x_1x_2\right|_{11}, \\
\mu_2(X) &= \left|7 + 6x_1 + 4x_2 + 0x_1x_2\right|_{13}, \\
\mu_3(X) &= \left|10 + 11x_1 + 12x_2 + 0x_1x_2\right|_{15}, \\
\mu_4(X) &= \left|15 + 9x_1 + 5x_2 + 0x_1x_2\right|_{16}.
\end{aligned}$$

Далее легко построить таблицу расчетных значений $\phi_1, \phi_2, \phi_3, \phi_4$ и, с помощью (??), — Y для всех наборов булевых переменных (табл. ??).

Кортежи значений выходов (2, 16)-полюсника $f_{16} \odot \ldots \odot f_1$ на всех наборах булевых переменных $x_1 \odot x_2$ получаются из представлений Y в двоичной системе счисления:

$$\begin{aligned}
(21535)_{10} &= (0101010000011111)_2, \\
(25972)_{10} &= (0110010101110100)_2, \\
(22776)_{10} &= (0101100011111000)_2, \\
(27213)_{10} &= (0110101001001101)_2.
\end{aligned}$$

Таблица 3: Таблица расчетных значений модулярных форм арифметических полиномов на всех наборах булевых переменных

$x_1 \odot x_2$	ϕ_1	ϕ_2	ϕ_3	ϕ_4	Y
$0 \odot 0$	8	7	10	15	21535
$0 \odot 1$	1	11	7	4	25972
$1 \odot 0$	6	0	6	8	22776
$1 \odot 1$	10	4	3	13	27213

Пример 14.
Рассмотрим линейный арифметический полином [102]:

$$Y = L(X) = 8 + 137x_1 - 7x_2 + 129x_3 + 136x_4 + 64x_5,$$

реализующий систему булевых функций:

$$\begin{aligned} f_1(X) &= x_1 \oplus x_2 \oplus x_3, \\ f_2(X) &= x_1 \oplus \overline{x}_2 \oplus x_3, \\ f_3(X) &= x_5, \\ f_4(X) &= x_1 \oplus x_3 \oplus x_4. \end{aligned}$$

Причем

$$\begin{aligned} f_1(X) &= \Xi^1\{Y\}, \\ f_2(X) &= \Xi^4\{Y\}, \\ f_3(X) &= \Xi^7\{Y\}, \\ f_4(X) &= \Xi^8\{Y\}. \end{aligned}$$

Так как $t_{\max} = 8$ и $Y_{\max} = 474$, выберем модули $m_1 = 7$, $m_2 = 8$, $m_3 = 9$, обеспечивающие выполнение условия

$$m = m_1 m_2 m_3 = 504 > 474.$$

Тогда согласно (**??**):

$$\begin{aligned} \lambda_1(X) &= |1 + 4x_1 + 3x_3 + 3x_4 + x_5|_7 , \\ \lambda_2(X) &= |x_1 + x_2 + x_3|_8 , \\ \lambda_3(X) &= |8 + 2x_1 + 2x_2 + 3x_3 + x_4 + x_5|_9 . \end{aligned}$$

Пусть $x_1 \odot x_2 \odot x_3 \odot x_4 \odot x_5 = 1 \odot 0 \odot 0 \odot 1 \odot 1$. Тогда

$$\begin{aligned} \lambda_1(X) &= 2, \\ \lambda_2(X) &= 1, \\ \lambda_3(X) &= 3. \end{aligned}$$

Используя формулу (**??**) и учитывая, что $B_1 = 288$, $B_2 = 441$, $B_3 = 280$, получим

$$Y = |2 \cdot 288 + 1 \cdot 441 + 3 \cdot 280|_{504} = |1857|_{504} = (101011001)_2.$$

Несмотря на классический вид формулы (??) она не всегда удобна для практического использования, в частности, из-за необходимости обеспечения большого числового диапазона. Другие, более приемлемые методы восстановления позиционной формы числа, будут рассмотрены в разделе 8.

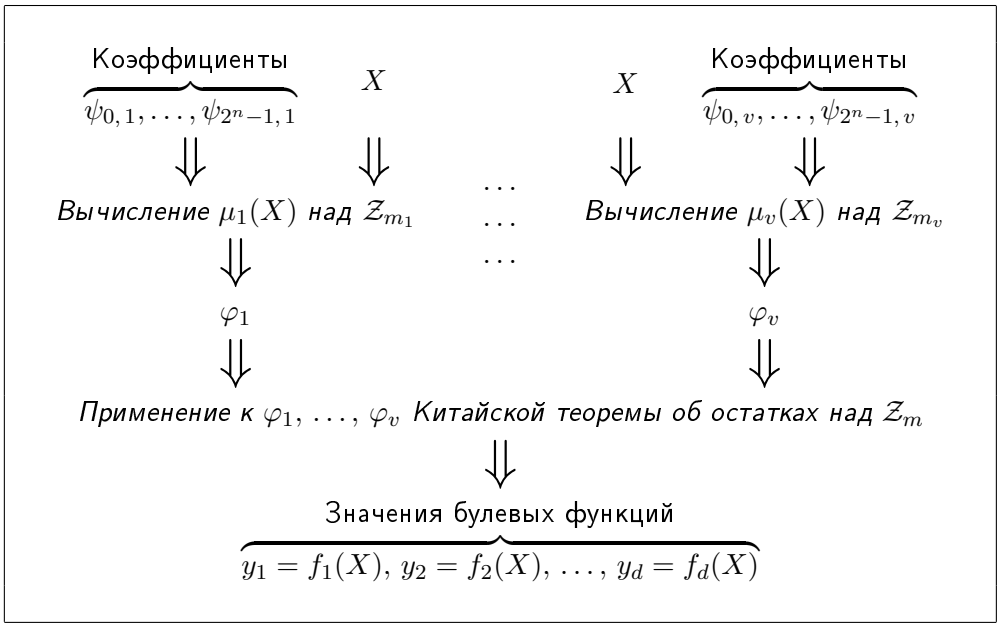

Рис. 11: Схема параллельного вычисления заданной системы булевых функций над $Z_{m_1, m_2, \ldots, m_v}$ на основе Китайской теоремы об остатках

Схема, поясняющая принцип реализации булевых функций посредством модулярных форм арифметических полиномов, основанных на Китайской теореме об остатках, представлена на рис. ??.

4.2. Конъюнктивные многомерные преобразования, основанные на Китайской теореме об остатках

Лемма 3.
Если для d-выходной булевой функции $f(X)$ задана пара ЛТЧП (??) и (??) и $m > Y_{max}$, причем $m = \prod_{k=1}^{v} m_k$ и $\gcd(m_i, m_j)$ $(i, j = 1, 2, \ldots, v;\ i \neq j)$, то справедлива следующая модулярная арифметико-логическая форма преобразований:

$$(41) \qquad \begin{cases} \mathbf{\Psi}_1 = \mathbf{A}_{2^n} \mathbf{\Phi}_1 \pmod{m_1}, \\ \mathbf{\Psi}_2 = \mathbf{A}_{2^n} \mathbf{\Phi}_2 \pmod{m_2}, \\ \vdots \\ \mathbf{\Psi}_v = \mathbf{A}_{2^n} \mathbf{\Phi}_v \pmod{m_v}; \end{cases}$$

$$(42) \qquad \begin{cases} \mathbf{\Phi}_1 = \mathbf{A}_{2^n}^{-1} \mathbf{\Psi}_1 \pmod{m_1}, \\ \mathbf{\Phi}_2 = \mathbf{A}_{2^n}^{-1} \mathbf{\Psi}_2 \pmod{m_2}, \\ \vdots \\ \mathbf{\Phi}_v = \mathbf{A}_{2^n}^{-1} \mathbf{\Psi}_v \pmod{m_v}, \end{cases}$$

где $\mathbf{A}_{2^n}$ и $\mathbf{A}_{2^n}^{-1}$ — соответственно матрицы прямого и инверсного арифметического преобразования;

$$\mathbf{\Phi}_k \ = \ [\phi_{0,k}\, \phi_{1,k}\, \ldots\, \phi_{2^n-1,k}]^T; \quad \phi_{i,k} = \left|Y^{(i)}\right|_{m_k}; \quad k = 1, \ldots, v;$$

$$\mathbf{\Psi}_k \;=\; [\psi_{0,\,k}\,\psi_{1,\,k}\,\ldots\,\psi_{2^n-1,\,k}]^T;\quad k=1,\,\ldots,\,v.$$

Доказательство леммы 3 следует 1) из взаимооднозначности связи матричной (??) и (??) и полиномиальной (??) форм представления булевых функций [47] и 2) из доказательства справедливости полиномиальной формы представления (??), основанной на Китайской теореме об остатках.

Определение 7.
Системы матричных преобразований (??) и (??) будем называть конъюнктивными многомерными преобразованиями, основанными на Китайской теореме об остатках или ЛТЧП, основанными на Китайской теореме об остатках (ЛТЧП КТО).

Продемонстрируем применение ЛТЧП КТО и некоторые полезные свойства этого преобразования, связанные с сокращением длин соответствующих арифметических полиномов, на примере получения спектров коэффициентов арифметических полиномов, соответствующих операции подстановки криптографического алгоритма стандарта DES (пример 6).

Пример 15.
Для подстановки

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ 15 & 1 & 8 & 14 & 6 & 11 & 3 & 4 & 9 & 7 & 2 & 13 & 12 & 0 & 5 & 10 \end{pmatrix}.$$

в соответствии с леммой 3 необходимо, чтобы $m_1 m_2 \geq 2^4$.

Для выбора модулей, целесообразных по критерию сложности получаемых арифметических полиномов, построим таблицу ?? (для рассматриваемой подстановки), где коэффициент уменьшения сложности полинома определим как $\frac{K}{2^n}$; K — количество ненулевых коэффициентов. Учитывая требование обеспечения попарной простоты модулей, выберем $m_1 = 3$, $m_2 = 7$. Тогда

$$\begin{cases} \mathbf{\Psi}_1 = \mathbf{A}_{2^4}\mathbf{\Phi}_1 \pmod 3, \\ \mathbf{\Psi}_2 = \mathbf{A}_{2^4}\mathbf{\Phi}_2 \pmod 7, \end{cases}$$

где

$$\mathbf{\Phi}_1 = [\;0\;\;1\;\;2\;\;2\;\;0\;\;2\;\;0\;\;1\;\;0\;\;1\;\;2\;\;1\;\;0\;\;0\;\;2\;\;1\;]^T \pmod 3,$$
$$\mathbf{\Phi}_2 = [\;1\;\;1\;\;1\;\;0\;\;6\;\;4\;\;3\;\;4\;\;2\;\;0\;\;2\;\;6\;\;5\;\;0\;\;5\;\;3\;]^T \pmod 7.$$

Тогда

$$\mathbf{\Psi}_1 = [\;3\;\;1\;\;2\;\;2\;\;0\;\;1\;\;1\;\;0\;\;0\;\;0\;\;0\;\;2\;\;0\;\;1\;\;2\;\;1\;]^T \pmod 3,$$
$$\mathbf{\Psi}_2 = [\;1\;\;0\;\;0\;\;6\;\;5\;\;5\;\;4\;\;4\;\;1\;\;12\;\;0\;\;0\;\;5\;\;5\;\;6\;\;0\;]^T \pmod 7.$$

Замечание 6.
ЛТЧП КТО (??) и (??) связаны с ЛТЧП (??) и (??) следующими отношениями

$$\mathbf{\Psi} = \mathbf{CRT}_{k=1}^{v}\mathbf{\Psi}_k \pmod{m_k},$$

$$\mathbf{Y} = \mathbf{CRT}_{k=1}^{v}\mathbf{\Phi}_k \pmod{m_k}.$$

Структура графов матричных преобразований для каждого номера k модуля m_k ЛТЧП КТО аналогична структурам графов ЛТЧП. Следовательно,

Таблица 4: Оценка зависимости сложности арифметического полинома от значения модуля

Значение модуля	Длина полинома (сложность)	Коэффициент уменьшения
2	7	0,56
3	9	0,47
4	8	0,5
5	12	0,25
6	11	0,31
7	11	0,31
8	14	0,12
9	14	0,12
10	13	0,19
11	15	0,06
12	13	0,19
13	15	0,06
14	13	0,19
15	15	0,06
16	15	0,06

к ЛТЧП КТО применимы методы факторизации матриц $\mathbf{A}_{2^n}$ и $\mathbf{A}_{2^n}^{-1}$ (методы быстрых конъюнктивных преобразований), применимые и к ЛДПФ [47] и ЛТЧП.

На рис. **??** показана геометрическая интерпретация ЛТЧП КТО и его взаимосвязь с ЛДПФ и ЛТЧП. Согласно этой диаграмме смысл ЛТЧП КТО сводится к разложению каждой из матриц $\mathbf{Y}$ и $\mathbf{C}$ на v матриц меньшей «ширины» — l_1, l_2, …, l_v, где $l_k = \lceil \log_2 m_k \rceil$, что позволяет упростить преобразование для каждой из полученных матриц $\mathbf{\Psi}_k$ или $\mathbf{\Phi}_k$ в отдельности. Полученные результаты затем восстанавливаются с помощью Китайской теоремы об остатках. При этом спектр $\mathbf{\Psi}$ является матрицей ЛТЧП по модулю $m = \prod_{k=1}^{v} m_k$.

Взаимосвязь ЛТЧП КТО и ЛТЧП поясняется с помощью рис. **??**.

Будем исходить из предположения, что преобразования для всех модулей ЛТЧП КТО выполняются параллельно. Тогда наибольшую сложность будет иметь преобразование ЛТЧП КТО для наибольшего модуля m_v. Максимальная «ширина» (необходимое количество двоичных разрядов для представления элементов матрицы) соответствует матрице $\mathbf{\Psi}_v$ по наибольшему модулю m_v и составит

$$l_v = \lceil \log_2 m_v \rceil$$

двоичных разрядов. Учитывая, что для большинства практических задач значение l_v не превышает $6 \div 7$, выигрыши в сравнении с ЛДПФ ЛТЧП соответственно составят

$$\frac{N_{\mathbf{C}}}{l_v} \approx \frac{n+d}{6 \div 7} \text{ и } \frac{N_{\mathbf{\Psi}}}{l_v} \approx \frac{d}{6 \div 7} \text{ раз.}$$

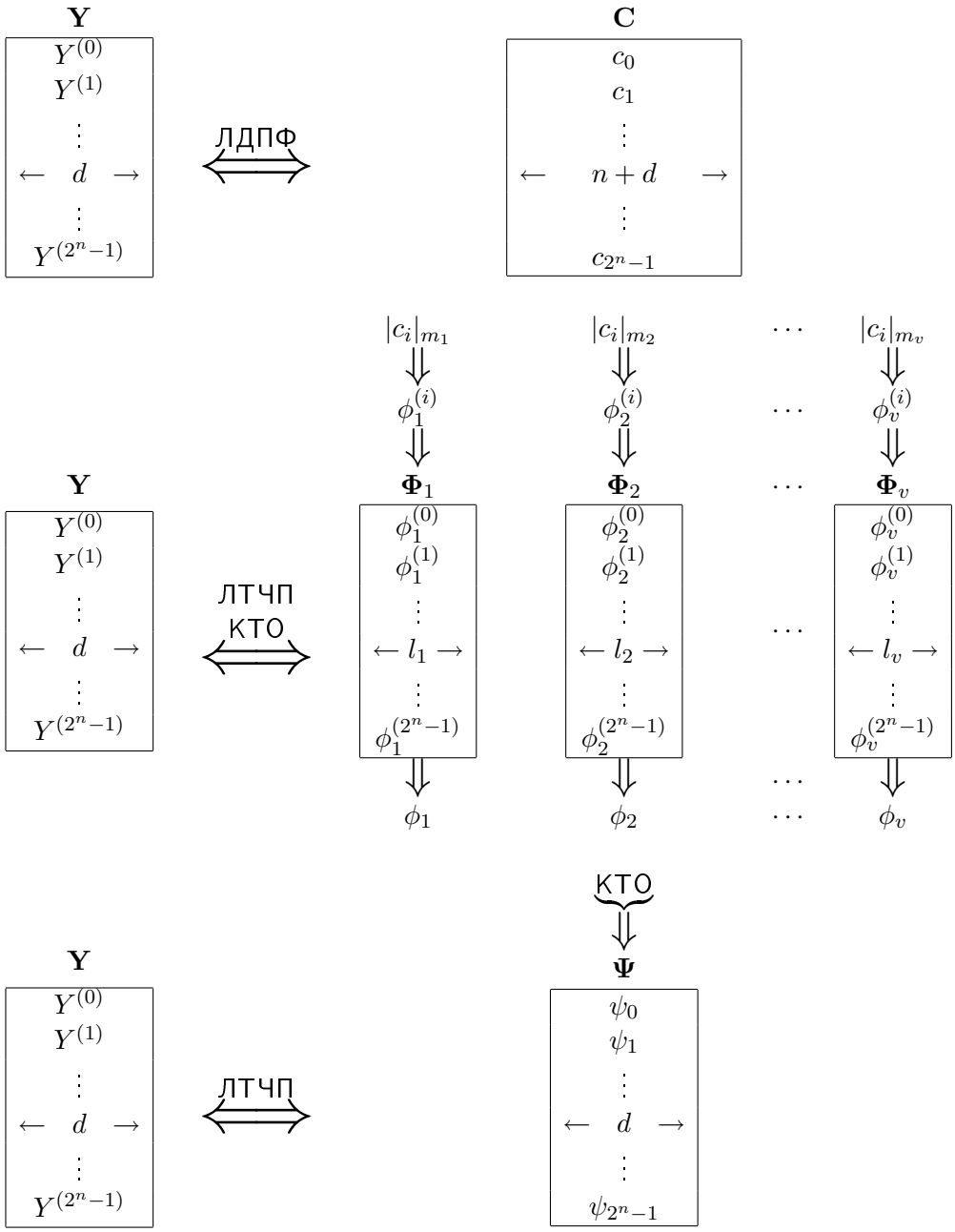

Рис. 12: Геометрическая интерпретация ЛТЧП КТО и его взаимосвязь с ЛДПФ и ЛТЧП

Например, при $n = d = 40$ и $m_v = 128$ выигрыши будут соответственно 11, 4 и 5, 7 раз. Например, набор из 17 модулей (табл. **??**) обеспечивает представление коэффициентов арифметического полинома (**??**) или (**??**) с верхней границей — 2^{105}. Полученные оценки сохраняют силу и для модулярных форм полиномиальных преобразований, основанных на Китайской теореме об остатках. При этом для сравнения с линейным арифметическим полиномом необходимо использовать отношение $\frac{d^\star}{6 \div 7}$, где $d^\star$ — общее количество (с учетом избыточных) реализуемых булевых функций или максимальное количество разрядов, необходимых для представления коэффициентов полинома. Относительно $d^\star = 2^{100}$ линейного арифметического полинома выигрыш в количестве используемых двоичных разрядных цифр в

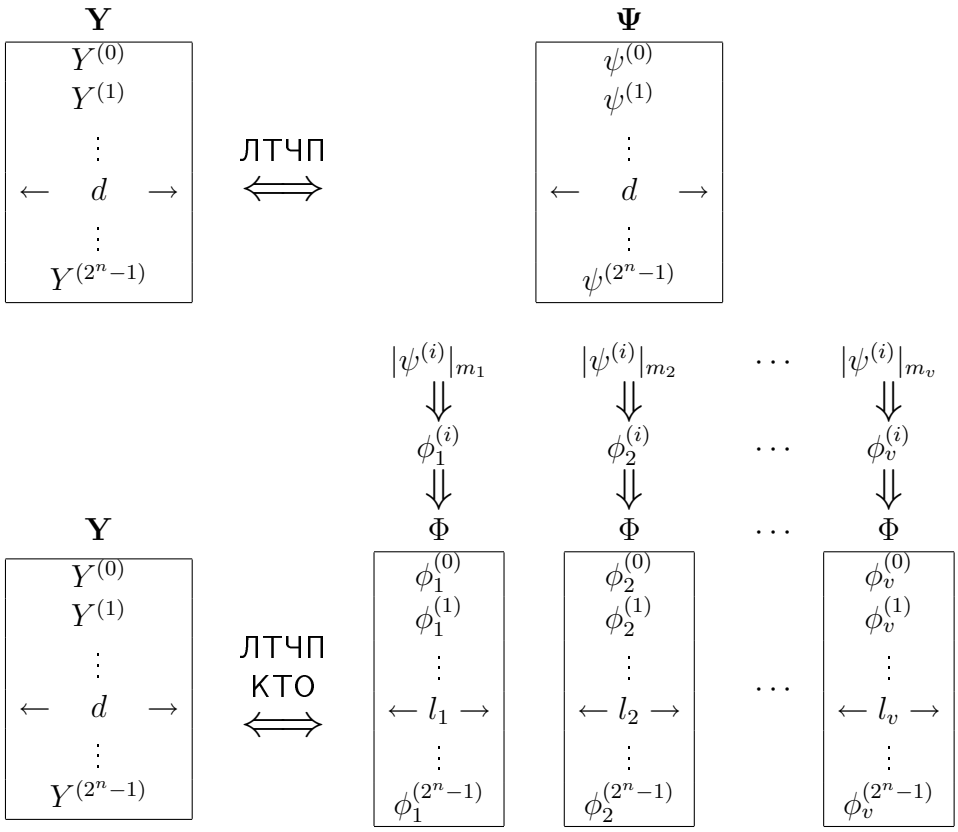

Рис. 13: Геометрическая интерпретация взаимосвязи ЛТЧП и ЛТЧП КТО

Таблица 5:

m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
67	71	73	79	83	89	91	97	101	103
m_{11}	m_{12}	m_{13}	m_{14}	m_{15}	m_{16}	m_{17}			
107	109	111	113	115	127	128			

пределах одного канала (модуля) преобразования составит 14, 3 раза.

4.3. Выводы

- Предложен метод реализации систем булевых функций посредством арифметической полиномиальной формы, основанной на Китайской теореме об остатках, которая представляет собой систему модулярных форм арифметических полиномов по системе специальным образом выбранных модулей.

- Показано, что предложенный метод, основанный на Китайской теореме об остатках, позволяет полностью решить проблему больших коэффициентов как нелинейных, так и линейных арифметических полиномов, при условии обеспечения распараллеливания вычислений по системе выбранных модулей.

- Предложено прямое аналитическое преобразование систем булевых функций непосредственно в арифметическую полиномиальную форму, основанную на Китайской теореме об остатках.

- Установлена взаимосвязь между немодулярной, модулярной и арифметической полиномиальной формой, основанной на Китайской теореме об остатках.

- Построены прямое и обратное матричные преобразования между системами булевых функций и арифметической полиномиальной формой, основанной на Китайской теореме об остатках. Полученные матричные преобразования определены как *логические теоретико-числовые преобразования, основанные на Китайской теореме об остатках.*

- Указано, что по аналогии с быстрыми конъюнктивными преобразованиями к логическим теоретико-числовым преобразованиям, основанным на Китайской теореме об остатках, могут быть применены методы факторизации матриц.

5. Организация вычислений в больших числовых диапазонах

Необходимость реализации систем логических функций больших размерностей ставит задачу организации параллельных вычислений в больших числовых диапазонах — $2^{100} \dots 2^{1000}$ и более. Здесь под *большим* числовым диапазоном понимается числовой диапазон, который на несколько порядков превышает *типовой* числовой диапазон. Эффективное решение задачи таких вычислений состоит в организации логических вычислений на основе *многопроцессорных* или *многомашинных* вычислительных систем.

5.1. Принцип больших модулей

Если сопоставить каждому модулю вычислитель, который бы функционировал по этому модулю, то совокупность таких вычислителей, объединенных связями, может представлять собой вычислительную систему, функционирующую в числовом диапазоне, равному произведению этих модулей. Ясно, что количество таких вычислителей должно быть минимизировано (без существенной потери производительности). Добиться этого можно, если значения модулей, сопоставимы с числовыми диапазонами, аппаратурно поддерживаемыми этими вычислителями (например $\approx 2^k$, где $k = 16, 32, 64 \dots$).

Изучению принципов построения таких вычислительных систем был посвящен ряд работ [23, 30, 31]. В частности в [30, 31] исследовались принципы вычислений в квадратичных больших и сверхбольших числовых диапазонах на основе модулярной арифметики, заданной квадратами модулей.

Достоинством принципа больших модулей является возможность использования для организации вычислительной системы не только специализированных, но и *серийных* — «универсальных» вычислителей, а также *IBM-совместимых PC.*

Несмотря на общий параллелизм вычислений, образованный организацией параллельного функционирования некоторой совокупности вычислителей, в рассмотренном случае *не задействуется* параллелизм вычислений на уровне *каждого* вычислителя.

5.2. Принцип групп модулей

В ЦО сигналов широко используются специализированные вычислители для реализации теоретико-числовых преобразований и числовой свертки. Возьмем за основу этот вычислитель (фрагмент структуры) для организации многопроцессорной вычислительной системы, предназначенной для реализации систем логических функций большой размерности.

5.2.1. Использование полностью параллельных вычислителей. Рассмотрим принцип построения вычислительной системы, на основе специализированных вычислителей, каждый из которых функционирует по системе из s модулей. На входы специализированных вычислителей помимо коэффициентов модулярных форм арифметических полиномов подаются результаты вычисления конъюнкций булевых переменных или *непосредственно булевы переменные* в случае использования *линейных форм* арифметических полиномов.

Пусть задана система модулей

$$m_1,\, m_2,\, \ldots,\, m_v,$$

свойства которых удовлетворяют теореме 5. Поставим им в соответствие вычеты

$$\phi_1,\, \phi_2,\, \ldots,\, \phi_v,$$

определяющие значение искомого результата

$$Y = \mathbf{CRT}_{i=1}^{v}\phi_i \pmod{m_i}.$$

Разобьем систему модулей $m_1,\, m_2,\, \ldots,\, m_v$ на t групп по $s = \frac{v}{t}$ модулей:

$$\overbrace{m_1,\, m_2,\, \ldots,\, m_s}^{1};$$
$$\overbrace{m_{s+1},\, m_{s+2},\, \ldots,\, m_{2s}}^{2};$$
$$\vdots$$
$$\overbrace{m_{ts-s},\, m_{ts-s+1},\, \ldots,\, m_{st}}^{t}.$$

В соответствие группам модулей сопоставим t групп вычетов

$$\overbrace{\phi_1,\, \phi_2,\, \ldots,\, \phi_s}^{1};$$
$$\overbrace{\phi_{s+1},\, \phi_{s+2},\, \ldots,\, \phi_{2s}}^{2};$$
$$\vdots$$
$$\overbrace{\phi_{ts-s},\, \phi_{ts-s+1},\, \ldots,\, \phi_{st}}^{t}.$$

Таким образом, вычислительная система может быть построена на основе t специализированных вычислителей, каждый из которых функционирует по группе s модулей. Результат вычислений всех специализированных вычислителей преобразуется в позиционную двоичную форму (значения булевых функций) с помощью выходного специализированного вычислителя, ориентированного на реализацию Китайской теоремы об остатках.

Важным свойством данной вычислительной системы является высокий уровень параллелизма, образованный как на уровне организации специализированных вычислителей, так и на уровне функционирования каждого специализированного вычислителя.

5.2.2. Использование промежуточных устройств восстановления числа. Для того чтобы для построения вычислительной системы использовались специализированные вычислители, структура которых бы максимально приближалась к структуре *серийных специализированных вычислителей ЦО сигналов*, функционирующих в модулярной арифметике, выполнение части алгоритма восстановления позиционной формы числа можно распределить среди специализированных вычислителей за счет использования в них встроенных *устройств восстановления позиционной формы числа по группе модулей функционирования*. При этом достигается *уменьшение количества связей* в вычислительной системе, а также *сокращение объема оборудования* выходного специализированного вычислителя, ориентированного на восстановление позиционной формы числа.

Применим к каждой группе вычетов Китайскую теорему об остатках:

$$(43)\qquad \begin{cases} \phi_1^* = \mathbf{CRT}_{i=1}^{s}\phi_i \pmod{m_i}, \\ \phi_2^* = \mathbf{CRT}_{i=s+1}^{2s}\phi_i \pmod{m_i}, \\ \vdots \\ \phi_t^* = \mathbf{CRT}_{i=ts-s}^{v}\phi_i \pmod{m_i}. \end{cases}$$

При этом отметим, что

$$\phi_1^* = |Y|_{m_1^*}, \quad \phi_2^* = |Y|_{m_2^*}, \quad \ldots, \quad \phi_t^* = |Y|_{m_t^*},$$

где

$$m_1^* = \prod_{i=1}^{s} m_i, \quad m_2^* = \prod_{i=s+1}^{2s} m_i, \quad \ldots, \quad m_t^* = \prod_{i=ts-s}^{v} m_i.$$

Поэтому

$$Y = \mathbf{CRT}_{i=1}^{t}\phi_i^* \pmod{m_i^*}.$$

Таким образом, схема вычисления Y может быть представлена в соответствии с рис. **??**.

Принцип поэтапного восстановления позиционной формы числа в литературе по модулярной арифметике известен. Мы лишь адаптировали этод принцип к условиям нашей задачи.

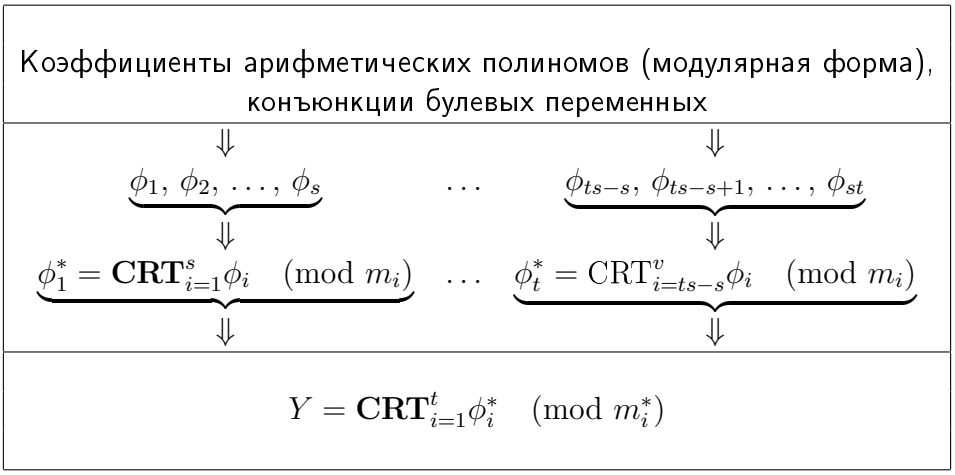

Рис. 14: Схема вычисления Y, использующая принцип разбиения системы модулей на группы модулей

5.3. Многоступенчатые многомерные формы

5.3.1. Полиномиальные многоступенчатые многомерные формы. Рассмотренный выше принцип организации многопроцессорной вычислительной системы требует использования набора специализированных вычислителей, каждый из которых функционирует по «индивидуальному» набору модулей. Это не всегда удобно. Например, в некоторых случаях требуется повысить однородность оборудования за счет применения набора *одинаковых* специализированных вычислителей.

В предыдущем случае это препятствие возможно обойти путем применения специальных приемов выполнения модулярных операций (по произвольному модулю). Однако такое решение достигается как правило за счет введения дополнительных временных и аппаратурных затрат и сужает область допустимых технических решений. Поэтому применять вычислительные схемы по произвольному модулю целесообразно в случаях, когда имеется некоторый резерв производительности, но требуется обеспечить высокие показатели надежности и живучести вычислительной системы.

В теории модулярной арифметики известен метод сведения вычислений над большими вычетами к вычислениям над маленькими вычетами, предусматривающее вторичное (или многократное) кодирование вычетов, основанное на Китайской теореме об остатках [9]. Этот метод, как правило, предлагается как дополнительное средство для уменьшения объема табличных операционных устройств и не получил широкого распространения.

Однако, в нашем случае этот метод приобретает особую целесообразность в силу необходимости организации вычислений в *больших* числовых диапазонах, которые не используются в обычных сферах приложения модулярной арифметки (цифровая обработка сигналов). Более того, как будет показано далее, благодаря отсутствию операций арифметического умножения требования к величине вторичного числового диапазона могут быть существенно снижены (имеется в виду числовой диапазон, достаточный для представления не только конечных, но и промежуточных результатов вычислений, которые при операциях умножения значительно превышают промежуточные результаты, получаемые при сложении).

Принцип многоступенчатого кодирования поясним на примере двухступенчатого кодирования.

Теорема 6.

Пусть даны модули $m_1 < m_2 < \ldots < m_v$ такие, что $\gcd(m_i, m_j)$ $(i \neq j;\ i, j = 1, \ldots, v)$;

$$\prod_{i=1}^{v} m_i = m \geq 2^d,$$

где d — количество реализуемых булевых функций, а также даны модули $\widetilde{m}_1, \widetilde{m}_2, \ldots, \widetilde{m}_l$ такие, что $\gcd(\widetilde{m}_i, \widetilde{m}_j)$ $(i \neq j;\ i, j = 1, \ldots, l;)$,

$$\prod_{i=1}^{l} \widetilde{m}_i > 2^n(m_v - 1), \tag{44}$$

где n — количество булевых переменных. Тогда произвольный кортеж булевых функций может быть однозначно представлен системой модулярных форм арифметических полиномов:

$$\begin{cases}
\begin{cases}
\widetilde{\phi}_{1,1} = \widetilde{\mu}_{1,1}(X) = \left| \sum\limits_{i=0}^{2^n-1} \widetilde{\psi}_{1,i,1}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{\widetilde{m}_1}, \\
\widetilde{\phi}_{1,2} = \widetilde{\mu}_{1,2}(X) = \left| \sum\limits_{i=0}^{2^n-1} \widetilde{\psi}_{1,i,2}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{\widetilde{m}_2}, \\
\vdots \\
\widetilde{\phi}_{1,l} = \widetilde{\mu}_{1,l}(X) = \left| \sum\limits_{i=0}^{2^n-1} \widetilde{\psi}_{1,i,l}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{\widetilde{m}_l};
\end{cases} \\
\begin{cases}
\widetilde{\phi}_{2,1} = \widetilde{\mu}_{2,1}(X) = \left| \sum\limits_{i=0}^{2^n-1} \widetilde{\psi}_{2,i,1}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{\widetilde{m}_1}, \\
\widetilde{\phi}_{2,2} = \widetilde{\mu}_{2,2}(X) = \left| \sum\limits_{i=0}^{2^n-1} \widetilde{\psi}_{2,i,2}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{\widetilde{m}_2}, \\
\vdots \\
\widetilde{\phi}_{2,l} = \widetilde{\mu}_{2,l}(X) = \left| \sum\limits_{i=0}^{2^n-1} \widetilde{\psi}_{2,i,l}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{\widetilde{m}_l};
\end{cases} \\
\vdots \\
\begin{cases}
\widetilde{\phi}_{v,1} = \widetilde{\mu}_{v,1}(X) = \left| \sum\limits_{i=0}^{2^n-1} \widetilde{\psi}_{v,i,1}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{\widetilde{m}_1}, \\
\widetilde{\phi}_{v,2} = \widetilde{\mu}_{v,2}(X) = \left| \sum\limits_{i=0}^{2^n-1} \widetilde{\psi}_{v,i,2}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{\widetilde{m}_2}, \\
\vdots \\
\widetilde{\phi}_{v,l} = \widetilde{\mu}_{v,l}(X) = \left| \sum\limits_{i=0}^{2^n-1} \widetilde{\psi}_{v,i,l}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{\widetilde{m}_l}.
\end{cases}
\end{cases} \tag{45}$$

где $\widetilde{\psi}_{k,i,l} = |\psi_{k,i}|_{m_j}$; $i = 0, 1, \ldots, 2^n - 1$; $k = 1, 2, \ldots, v$; $j = 1, 2, \ldots, l$; $\psi_{k,i}$ — коэффициенты полиномиальной арифметической формы, основанной на Китайской теореме об остатках (**??**).

Доказательство теоремы 6.
Пусть дан результат вычисления полинома $\mu_k(X)$ — вычет $0 \le \phi_{k,1} < m_k$. На основании Китайской теоремы об остатках вычет $0 \le \phi_{k,1} < m_k$ может быть представлен системой вычетов $\widetilde{\phi}_{k,1}$, $\widetilde{\phi}_{k,2}, \ldots, \widetilde{\phi}_{k,l}$ по системе попарно простых модулей $\widetilde{m}_1$, $\widetilde{m}_2$, $\ldots$, $\widetilde{m}_l$. Это представление будет единственным, если

$$\prod_{i=1}^{l} \widetilde{m}_i \ge m_k \quad (k = 1, 2, \ldots, v). \tag{46}$$

Тогда так как, согласно теореме 5 заданная система булевых функций может быть единственным способом представлена системой вычетов $\phi_1, \phi_2, \ldots, \phi_k$, то система вычетов

$$\begin{array}{rcccc}
\phi_1 = & (\widetilde{\phi}_{1,1}, & \widetilde{\phi}_{1,2}, & \ldots, & \widetilde{\phi}_{1,l}); \\
\phi_2 = & (\widetilde{\phi}_{2,1}, & \widetilde{\phi}_{2,2}, & \ldots, & \widetilde{\phi}_{2,l}); \\
\vdots & \vdots & \vdots & & \vdots \\
\phi_v = & (\widetilde{\phi}_{v,1}, & \widetilde{\phi}_{v,2}, & \ldots, & \widetilde{\phi}_{v,l})
\end{array}$$

также единственным образом соответствует заданной системе булевых функций.

Примем во внимание то, что максимальный результат, получаемый при вычислении полинома

$$\sum_{i=0}^{2^n-1} \psi_{i,k}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n},$$

с учетом $0 \le \psi_{i,k} < m_k$ равен

$$2^n(m_k - 1).$$

Но, так как $\max(m_1, m_2, \ldots, m_v) = m_v$, то условие (??) должно быть заменено на условие (??). Если (??) выполняется, то и представление (??) единственно. □

Определение 8.
Систему арифметических полиномов (??) будем называть двухступенчатой полиномиальной формой представления булевых функций, основанной на Китайской теореме об остатках.

Из (??) следует, что при использовании нелинейных арифметических полиномов величина числового диапазона, обеспечиваемая второй ступенью модулярного кодирования может быть *существенной*. Уменьшить его можно, если применять Китайскую теорему после более коротких участков цепочки вычислений. Однако это резко снизит производительность вычислений.

«Естественным» путем уменьшения числового диапазона второй ступени модулярного кодирования является применение идеи многоступенчатого кодирования к *линейным* арифметическим полиномам.

Лемма 4.
Пусть кортеж булевых функций (??) задан линейным арифметическим полиномом $L(X)$ (??). Даны модули

$$m_1 < m_2 < \ldots < m_v$$

такие, что $\gcd(m_i,\, m_j)$ $(i \neq j;\ i,\, j = 1,\, \ldots,\, v)$;

$$\prod_{i=1}^{v} m_i = m > U_{max}.$$

Также даны модули

$$\widetilde{m}_1,\, \widetilde{m}_2,\, \ldots,\, \widetilde{m}_l$$

такие, что $\gcd(\widetilde{m}_i,\, \widetilde{m}_j)$ $(i \neq j;\ i,\, j = 1,\, \ldots,\, l;)$,

$$\prod_{i=1}^{l} \widetilde{m}_i > (n+1)(m_v - 1), \tag{47}$$

где n — количество булевых переменных. Тогда произвольный кортеж булевых функций может быть однозначно представлен системой модулярных форм линейных арифметических полиномов:

$$\begin{cases}
\widetilde{\phi}_{1,\,1} = \widetilde{\lambda}_{1,\,1}(X) = |\widetilde{\omega}_{1,\,0,\,1} + \widetilde{\omega}_{1,\,1,\,1}x_1 + \ldots + \widetilde{\omega}_{1,\,n,\,1}x_n|_{\widetilde{m}_1}, \\
\widetilde{\phi}_{1,\,2} = \widetilde{\lambda}_{1,\,2}(X) = |\widetilde{\omega}_{1,\,0,\,2} + \widetilde{\omega}_{1,\,1,\,2}x_1 + \ldots + \widetilde{\omega}_{1,\,n,\,2}x_n|_{\widetilde{m}_2}, \\
\vdots \\
\widetilde{\phi}_{1,\,l} = \widetilde{\lambda}_{1,\,l}(X) = |\widetilde{\omega}_{1,\,0,\,l} + \widetilde{\omega}_{1,\,1,\,l}x_1 + \ldots + \widetilde{\omega}_{1,\,n,\,l}x_n|_{\widetilde{m}_l}; \\
\widetilde{\phi}_{2,\,1} = \widetilde{\lambda}_{2,\,1}(X) = |\widetilde{\omega}_{2,\,0,\,1} + \widetilde{\omega}_{2,\,1,\,1}x_1 + \ldots + \widetilde{\omega}_{2,\,n,\,1}x_n|_{\widetilde{m}_1}, \\
\widetilde{\phi}_{2,\,2} = \widetilde{\lambda}_{2,\,2}(X) = |\widetilde{\omega}_{2,\,0,\,2} + \widetilde{\omega}_{2,\,1,\,2}x_1 + \ldots + \widetilde{\omega}_{2,\,n,\,2}x_n|_{\widetilde{m}_2}, \\
\vdots \\
\widetilde{\phi}_{2,\,l} = \widetilde{\lambda}_{2,\,l}(X) = |\widetilde{\omega}_{2,\,0,\,l} + \widetilde{\omega}_{2,\,1,\,l}x_1 + \ldots + \widetilde{\omega}_{2,\,n,\,l}x_n|_{\widetilde{m}_l}; \\
\vdots \\
\widetilde{\phi}_{v,\,1} = \widetilde{\lambda}_{v,\,1}(X) = |\widetilde{\omega}_{v,\,0,\,1} + \widetilde{\omega}_{v,\,1,\,1}x_1 + \ldots + \widetilde{\omega}_{v,\,n,\,1}x_n|_{\widetilde{m}_1}, \\
\widetilde{\phi}_{v,\,2} = \widetilde{\lambda}_{v,\,2}(X) = |\widetilde{\omega}_{v,\,0,\,2} + \widetilde{\omega}_{v,\,1,\,2}x_1 + \ldots + \widetilde{\omega}_{v,\,n,\,2}x_n|_{\widetilde{m}_2}, \\
\vdots \\
\widetilde{\phi}_{v,\,l} = \widetilde{\lambda}_{v,\,l}(X) = |\widetilde{\omega}_{v,\,0,\,l} + \widetilde{\omega}_{v,\,1,\,l}x_1 + \ldots + \widetilde{\omega}_{v,\,n,\,l}x_n|_{\widetilde{m}_l}.
\end{cases} \tag{48}$$

где $\widetilde{\omega}_{k,\,i,\,l} = |\omega_{k,\,i}|_{m_j}$;
$i = 0,\, 1,\, \ldots,\, 2^n - 1;\ k = 1,\, 2,\, \ldots,\, v;\ j = 1,\, 2,\, \ldots,\, l;\ \omega_{k,\,i}$ — коэффициенты линейной полиномиальной арифметической формы, основанной на Китайской теореме об остатках (**??**).

Доказательство леммы 4.
Если условие (**??**) выполнено, то на основании справедливости леммы 2 система вычетов

$$\begin{array}{lllll}
\phi_1 = & (\widetilde{\phi}_{1,\,1}, & \widetilde{\phi}_{1,\,2}, & \ldots, & \widetilde{\phi}_{1,\,l}); \\
\phi_2 = & (\widetilde{\phi}_{2,\,1}, & \widetilde{\phi}_{2,\,2}, & \ldots, & \widetilde{\phi}_{2,\,l}); \\
\quad \vdots & \vdots & \vdots & & \vdots \\
\phi_v = & (\widetilde{\phi}_{v,\,1}, & \widetilde{\phi}_{v,\,2}, & \ldots, & \widetilde{\phi}_{v,\,l})
\end{array}$$

единственна. Максимальный результат, получаемый при вычислении линейного полинома

$$\widetilde{\omega}_{0,\,k} + \widetilde{\omega}_{1,\,k}x_1 + \ldots + \widetilde{\omega}_{n,\,k}x_n$$

равен

$$(n+1)(m_k-1).$$

Так как условие (??) оговорено, то лемма 4 справедлива.

□

Определение 9.
Систему арифметических полиномов (??) будем называть двухступенчатой линейной полиномиальной формой представления булевых функций, основанной на Китайской теореме об остатках.

Следствие 3.

$$Y = \mathbf{CRT}_{i=1}^{v}\phi_i \pmod{m_i},$$

где

$$\phi_1 = \left|\widetilde{\phi}_1\right|_{m_1}, \quad \phi_2 = \left|\widetilde{\phi}_2\right|_{m_2}, \quad \ldots, \quad \phi_v = \left|\widetilde{\phi}_v\right|_{m_v};$$

$$\begin{aligned}
\widetilde{\phi}_1 &= \mathbf{CRT}_{i=1}^{l}\widetilde{\phi}_{1,i} \pmod{\widetilde{m}_i},\\
\widetilde{\phi}_2 &= \mathbf{CRT}_{i=1}^{l}\widetilde{\phi}_{2,i} \pmod{\widetilde{m}_i},\\
&\vdots\\
\widetilde{\phi}_v &= \mathbf{CRT}_{i=1}^{l}\widetilde{\phi}_{v,i} \pmod{\widetilde{m}_i}.
\end{aligned}$$

Следствие 4.

$$Y = \left|\widetilde{\phi}_1 B_1 + \widetilde{\phi}_2 B_2 + \ldots + \widetilde{\phi}_v B_v\right|_m,$$

где B_k $(k = 1, 2, \ldots v)$ — ортогональные базисы из выражения (??).

Следствие 5.
Из (??) и (??) следует, что результат вычисления модулярной формы линейного полинома второй ступени (??) в

$$\frac{2^n}{n+1}$$

раз меньше по сравнению с результатом вычисления модулярной формы полинома второй ступени (??). Поэтому применение модулярной формы линейного полинома более целесообразно.

Замечание 7.
Использование в теореме 6 и лемме 4 единой системы модулей $\widetilde{m}_1, \widetilde{m}_2, \ldots, \widetilde{m}_l$ позволяет при технической реализации использовать комплект из одинаковых специализированных вычислителей.

5.3.2. Конъюнктивные многоступенчатые многомерные формы. Существование полиномиальных модулярных многоступенчатых форм дает основание для построения многоступенчатых теоретико-числовых преобразований.

Лемма 5.
Для d-выходной булевой функции $f(X)$ справедлива пара матричных преобразований:

$$
\begin{array}{l}
\left\{\begin{array}{ll}
\widetilde{\mathbf{\Psi}}_{1,1} = \mathbf{A}_{2^n}\widetilde{\mathbf{\Phi}}_{1,1} & (\mathrm{mod}\ \widetilde{m}_1),\\
\widetilde{\mathbf{\Psi}}_{1,2} = \mathbf{A}_{2^n}\widetilde{\mathbf{\Phi}}_{1,2} & (\mathrm{mod}\ \widetilde{m}_2),\\
\vdots & \\
\widetilde{\mathbf{\Psi}}_{1,l} = \mathbf{A}_{2^n}\widetilde{\mathbf{\Phi}}_{1,l} & (\mathrm{mod}\ \widetilde{m}_l);
\end{array}\right.\\
\left\{\begin{array}{ll}
\widetilde{\mathbf{\Psi}}_{1,1} = \mathbf{A}_{2^n}\widetilde{\mathbf{\Phi}}_{1,1} & (\mathrm{mod}\ \widetilde{m}_1),\\
\widetilde{\mathbf{\Psi}}_{1,2} = \mathbf{A}_{2^n}\widetilde{\mathbf{\Phi}}_{1,2} & (\mathrm{mod}\ \widetilde{m}_2),\\
\vdots & \\
\widetilde{\mathbf{\Psi}}_{1,l} = \mathbf{A}_{2^n}\widetilde{\mathbf{\Phi}}_{1,l} & (\mathrm{mod}\ \widetilde{m}_l);
\end{array}\right.\\
\vdots\\
\left\{\begin{array}{ll}
\widetilde{\mathbf{\Psi}}_{1,1} = \mathbf{A}_{2^n}\widetilde{\mathbf{\Phi}}_{1,1} & (\mathrm{mod}\ \widetilde{m}_1),\\
\widetilde{\mathbf{\Psi}}_{1,2} = \mathbf{A}_{2^n}\widetilde{\mathbf{\Phi}}_{1,2} & (\mathrm{mod}\ \widetilde{m}_2),\\
\vdots & \\
\widetilde{\mathbf{\Psi}}_{1,l} = \mathbf{A}_{2^n}\widetilde{\mathbf{\Phi}}_{1,l} & (\mathrm{mod}\ \widetilde{m}_l)
\end{array}\right.
\end{array}
\qquad (49)
$$

(прямое преобразование);

$$
\begin{array}{l}
\left\{\begin{array}{ll}
\widetilde{\mathbf{\Phi}}_{1,1} = \mathbf{A}^{-1}_{2^n}\widetilde{\mathbf{\Psi}}_{1,1} & (\mathrm{mod}\ \widetilde{m}_1),\\
\widetilde{\mathbf{\Phi}}_{1,2} = \mathbf{A}^{-1}_{2^n}\widetilde{\mathbf{\Psi}}_{1,2} & (\mathrm{mod}\ \widetilde{m}_2),\\
\vdots & \\
\widetilde{\mathbf{\Phi}}_{1,l} = \mathbf{A}^{-1}_{2^n}\widetilde{\mathbf{\Psi}}_{1,l} & (\mathrm{mod}\ \widetilde{m}_l);
\end{array}\right.\\
\left\{\begin{array}{ll}
\widetilde{\mathbf{\Phi}}_{1,1} = \mathbf{A}^{-1}_{2^n}\widetilde{\mathbf{\Psi}}_{1,1} & (\mathrm{mod}\ \widetilde{m}_1),\\
\widetilde{\mathbf{\Phi}}_{1,2} = \mathbf{A}^{-1}_{2^n}\widetilde{\mathbf{\Psi}}_{1,2} & (\mathrm{mod}\ \widetilde{m}_2),\\
\vdots & \\
\widetilde{\mathbf{\Phi}}_{1,l} = \mathbf{A}^{-1}_{2^n}\widetilde{\mathbf{\Psi}}_{1,l} & (\mathrm{mod}\ \widetilde{m}_l);
\end{array}\right.\\
\vdots\\
\left\{\begin{array}{ll}
\widetilde{\mathbf{\Phi}}_{1,1} = \mathbf{A}^{-1}_{2^n}\widetilde{\mathbf{\Psi}}_{1,1} & (\mathrm{mod}\ \widetilde{m}_1),\\
\widetilde{\mathbf{\Phi}}_{1,2} = \mathbf{A}^{-1}_{2^n}\widetilde{\mathbf{\Psi}}_{1,2} & (\mathrm{mod}\ \widetilde{m}_2),\\
\vdots & \\
\widetilde{\mathbf{\Phi}}_{1,l} = \mathbf{A}^{-1}_{2^n}\widetilde{\mathbf{\Psi}}_{1,l} & (\mathrm{mod}\ \widetilde{m}_l)
\end{array}\right.
\end{array}
\qquad (50)
$$

(обратное преобразование),
где $\mathbf{A}_{2^n}$ и $\mathbf{A}^{-1}_{2^n}$ — соответственно матрицы прямого и инверсного арифметического преобразования;

$$
\begin{aligned}
\widetilde{\mathbf{\Phi}}_{k,j} &= \left[\widetilde{\phi}_{0,k,j},\, \widetilde{\phi}_{1,k,j},\, \ldots,\, \widetilde{\phi}_{2^n-1,k,j}\right]^T;\\
\widetilde{\mathbf{\Psi}}_{k} &= \left[\widetilde{\psi}_{k,0,j},\, \widetilde{\psi}_{k,1,j},\, \ldots,\, \widetilde{\psi}_{k,2^n-1,j}\right]^T;\\
\widetilde{\phi}_{i,k,j} &= \left|\phi^{(i)}_k\right|_{\widetilde{m}_j};\\
\phi^{(i)}_k &= |Y^{(i)}|_{m_k};
\end{aligned}
$$

$$\begin{aligned} i &= 0, 1, \ldots, 2^n - 1; \\ k &= 1, \ldots, v; \\ j &= 1, \ldots, l. \end{aligned}$$

Доказательство леммы 5 следует 1) из взаимооднозначности связи матричной (??) и (??) и полиномиальной (??) форм представления булевых функций [47] и 2) из доказательства справедливости полиномиальной формы представления (??).

□

Определение 10.
Системы матричных преобразований (??) и (??) будем называть конъюнктивными двухступенчатыми многомерными преобразованиями, основанными на Китайской теореме об остатках или двухступенчатыми ЛТЧП, основанными на Китайской теореме об остатках (двухступенчатыми ЛТЧП КТО).

Таким образом, метод применения многоступенчатых многомерных арифметико-логических форм позволяет значительно повысить однородность вычислительной системы (то есть строить ее на одинаковых специализированных вычислителях).

5.4. Выводы

- Рассмотрены принципы построения *многопроцессорных вычислительных систем*, обеспечивающих интенсивную реализацию логических функций большой размерности в *больших числовых диапазонах*.

- Принцип *больших модулей* обеспечивает *доступность* технической реализации на основе серийных универсальных и специализированных микропроцессоров и 32—64-х разрядных IBM-совместимых PC.

- Принцип *групп модулей* позволяет задействовать для построения параллельной вычислительной системы специализированные параллельные вычислители, идентичные специализированным вычислителям, применяемым в ЦО сигналов для реализации ТЧП. При этом достигается *высокий уровень параллелизма*, обеспечиваемый параллельной организацией параллельных специализированных вычислителей.

- Предложены *полиномиальные* и *конъюнктивные* многомерные многоступенчатые арифметико-логические формы. Они реализуют принцип больших модулей с обеспечением параллельного характера вычислений «внутри» вычетов. При этом как и при использовании принципа групп модулей для построения многопроцессорной системы применяются параллельные специализированные вычислители, обеспечивая, тем самым, чрезвычайно высокий уровень параллелизма. Однако в отличие от принципа групп модулей достигается более высокая однородность оборудования за счет применения идентичных специализированных вычислителей.

6. Обобщение модулярных форм на k-значную логику

Глава посвящена обобщению арифметической логики на формы и функции многозначной логики. Общая схема погружения функции алгебры логики в полиномиальное кольцо сохраняется. Однако при этом возникает специфика, которая состоит в том, что коэффициенты целозначных полиномов теперь могут быть рациональными числами. Как будет показано далее этот факт необходимо учитывать при выборе модулей модулярного представления форм и функций многозначной алгебры логики.

6.1. Полиномиальная арифметика k-значной логики

Под многозначной $f^{(k)}(X)$ функцией алгебры логики (ФАЛ) n переменных $X = x_1, x_2, \ldots, x_n$ будем понимать логическую функцию, заданную на множестве $\{0, 1, \ldots, k-1\}$, значения аргументов которой принадлежат этому же множеству, где k — значность ФАЛ.

Число всевозможных ФАЛ равно k^{k^n}.

Как известно, любую многозначную ФАЛ можно представить в виде арифметического полинома [36]:

$$P^{(k)}(X) = \sum_{i=0}^{k^n-1} p_i^{(k)} x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \tag{51}$$

где $p_i^{(k)}$ — коэффициенты АП, такие, что $p_i^{(k)} \in \mathcal{R}$ ($\mathcal{R}$ — множество рациональных чисел); $X = x_1, x_2, \ldots, x_n$ — аргументы ФАЛ $x_u \in \{0, 1, \ldots, k-1\}$ $(u = 1, 2, \ldots, n)$; $(i_1\, i_2 \ldots i_n)_k$ — представление параметра i в k-ичной системе счисления:

$$\begin{aligned} (i_1\, i_2 \ldots i_n)_k &= \sum_{u=1}^{n} i_u k^{n-u} \ (i_u \in \{0, 1, \ldots, k-1\}); \\ x_u^{i_u} &= \begin{cases} x_u, & i_u \neq 0, \\ 1, & i_u = 0. \end{cases} \end{aligned}$$

Так же как и в случае вычисления булевых функций для k-значных ФАЛ можно построить две схемы вычисления посредством арифметических полиномов (рис. ?? и ??). В отличие от схем, представленных на рис. ?? и ??, вычисление производится над множеством рациональных чисел $\mathcal{R}$.

Аналогично двоичной логике в k-ичной логике могут быть определены алгебраический и матричный методы построения арифметического полинома (??) [36].

Прямое и обратное матричное преобразование (ЛДПФ — в дальнейшем k-ЛДПФ) определяется соответственно выражениями

$$\mathbf{P}^{(k)} = N_k^{-1} \mathbf{K}_{k^n} \mathbf{S}, \tag{52}$$

$$\mathbf{S} = \mathbf{K}_{k^n}^{-1} \mathbf{P}^{(k)}, \tag{53}$$

где N_k — нормализующий множитель; $\mathbf{K}_{k^n}$ и $\mathbf{K}_{k^n}^{-1}$ — соответственно матрицы прямого и инверсного арифметического преобразования размерности $k^n \times k^n$ (базис преобразования); $\mathbf{S}$ — вектор истинности k-значной ФАЛ;

$$\mathbf{S} = \left[S^{(0)} S^{(1)} \ldots S^{(k^n-1)}\right]^T,$$

Рис. 15: Схема вычисления произвольной k-значной ФАЛ над $\mathcal{R}$

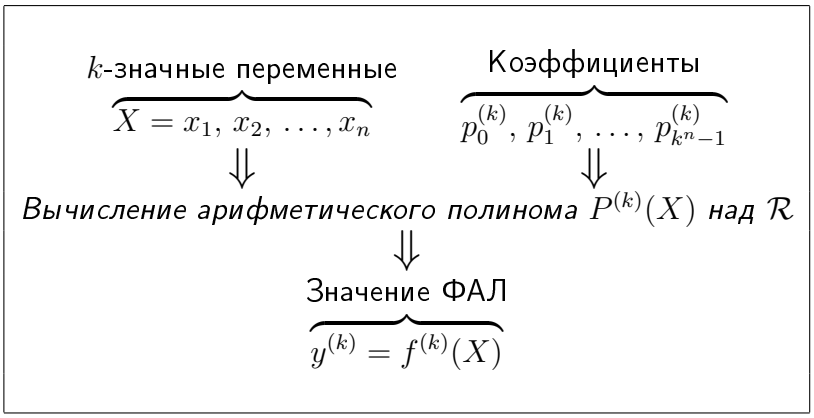

Рис. 16: Схема вычисления заданной k-значной ФАЛ над $\mathcal{R}$

где $S^{(i)}$ — числовое значение, принимаемое k-значной ФАЛ на i-м наборе переменных; вектор коэффициентов (спектр) арифметического полинома (**??**):

$$\mathbf{P}^{(k)} = \left[p_0^{(k)} p_1^{(k)} \dots p_{k^n-1}^{(k)}\right].$$

Матрицы $\mathbf{K}_{k^n}$ и $\mathbf{K}_{k^n}^{-1}$ как и базис преобразования двоичной логики определяется кронекеровским возведением в степень:

$$\mathbf{K}_{k^n} = \bigotimes_{j=1}^{n} \mathbf{K}_k; \quad \mathbf{K}_{k^n}^{-1} = \bigotimes_{j=1}^{n} \mathbf{K}_k^{-1},$$

где $\mathbf{K}_k$ и $\mathbf{K}_k^{-1}$ — базовые матрицы прямого и обратного преобразования (см. табл. **??** — для $k = 2 \dots 6$).

Пример 16.

Пусть $k = 3$ и вектор принимаемых значений ФАЛ при $n = 2$ имеет вид: $\mathbf{S} = [2\ 0\ 2\ 1\ 1\ 1\ 0\ 1\ 2]^T$. Тогда прямое преобразование (**??**) примет вид:

$$\mathbf{P}^{(3)} = \frac{1}{4}\mathbf{K}_{3^2}\mathbf{S} =$$

Таблица 6:

k	$\mathbf{K}_k$	$\mathbf{K}_k^{-1}$
2	$\begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$
3	$\begin{bmatrix} 2 & 0 & 0 \\ -3 & 4 & -1 \\ 1 & -2 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \end{bmatrix}$
4	$\begin{bmatrix} 6 & 0 & 0 & 0 \\ -11 & 18 & -9 & 2 \\ 6 & -15 & 12 & -3 \\ -1 & 3 & -3 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 \\ 1 & 3 & 9 & 27 \end{bmatrix}$
5	$\begin{bmatrix} 24 & 0 & 0 & 0 & 0 \\ -50 & 96 & -72 & 32 & -6 \\ 35 & -104 & 114 & -56 & 11 \\ -10 & 36 & -48 & 28 & -6 \\ 1 & -4 & 6 & -4 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 & 16 \\ 1 & 3 & 9 & 27 & 81 \\ 1 & 4 & 16 & 64 & 256 \end{bmatrix}$
6	$\begin{bmatrix} 120 & 0 & 0 & 0 & 0 & 0 \\ -274 & 660 & -600 & 400 & -150 & 24 \\ 225 & -770 & 1070 & -780 & 305 & -20 \\ -85 & 355 & -590 & 490 & -205 & 35 \\ 15 & -70 & 130 & -120 & 55 & -10 \\ -1 & 5 & -10 & 10 & -5 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 & 16 & 32 \\ 1 & 3 & 9 & 27 & 81 & 243 \\ 1 & 4 & 16 & 64 & 256 & 1024 \\ 1 & 5 & 25 & 125 & 625 & 3125 \end{bmatrix}$

$$= \frac{1}{4} \begin{bmatrix} 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -6 & 8 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & -4 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ -6 & 0 & 0 & 8 & 0 & 0 & -2 & 0 & 0 \\ 9 & -12 & 3 & -12 & 16 & -4 & 3 & -4 & 1 \\ -3 & 6 & -3 & 4 & -8 & 4 & -1 & 2 & -1 \\ 2 & 0 & 0 & -4 & 0 & 0 & 2 & 0 & 0 \\ -3 & 4 & -1 & 6 & -8 & 2 & -3 & 4 & -1 \\ 1 & -2 & 1 & -2 & 4 & -2 & 1 & -2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 2 \\ 0 \\ 2 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 2 \end{bmatrix}$$

$$= \frac{1}{4} \begin{bmatrix} 8 \\ -16 \\ 8 \\ -4 \\ 22 \\ -12 \\ 0 \\ -6 \\ 4 \end{bmatrix} \begin{matrix} \\ x_2 \\ x_2^2 \\ x_1 \\ x_1x_2 \\ x_1x_2^2 \\ x_1^2 \\ x_1^2x_2 \\ x_1^2x_2^2 \end{matrix} .$$

Таким образом на основании (**??**) алгебраическая форма будет иметь вид:

$$\begin{aligned} P^{(3)}(X) &= \frac{1}{4}(8 - 16x_2 + 8x_2^2 - 4x_1 + 22x_1x_2 - 12x_1x_2^2 + \\ &+ 0x_1^2 - 6x_1^2x_2 + 4x_1^2x_2^2) = \\ &= 2 - 4x_2 + 2x_2^2 - x_1 + \frac{11}{2}x_1x_2 - 3x_1x_2^2 - \end{aligned}$$

$$- \quad \frac{3}{2}x_1^2x_2 + x_1^2x_2^2.$$

Например, пусть $x_1 = 1$, $x_2 = 2$, тогда значение ФАЛ определится как:

$$\begin{aligned} P^{(3)}(X) &= \frac{1}{4}(8 - 16 \cdot 2 + 8 \cdot 2^2 - 4 \cdot 1 + 22 \cdot 1 \cdot 2 - 12 \cdot 1 \cdot 2^2 - \\ &- 6 \cdot 1^2 \cdot 2 + 4 \cdot 1^2 \cdot 2^2) = \frac{1}{4}4 = 1. \end{aligned}$$

Обратное преобразование (**??**) будет иметь вид:

$$\mathbf{S} = \mathbf{K}_{3^2}^{-1}\mathbf{P}^{(3)} =$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 1 & 2 & 4 & 1 & 2 & 4 \\ 1 & 0 & 0 & 2 & 0 & 0 & 4 & 0 & 0 \\ 1 & 1 & 1 & 2 & 2 & 2 & 4 & 4 & 4 \\ 1 & 2 & 4 & 2 & 4 & 8 & 4 & 8 & 16 \end{bmatrix} \cdot \begin{bmatrix} 2 \\ -4 \\ 2 \\ -1 \\ \frac{11}{2} \\ -3 \\ 0 \\ -\frac{3}{2} \\ 1 \end{bmatrix} \begin{matrix} \\ x_2 \\ x_2^2 \\ x_1 \\ x_1x_2 \\ x_1x_2^2 \\ x_1^2 \\ x_1^2x_2 \\ x_1^2x_2^2 \end{matrix} =$$

$$= \begin{bmatrix} 2 \\ 0 \\ 2 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 2 \end{bmatrix}.$$

Недостатком представления (**??**) является возможность принятия коэффициентами $p_i^{(k)}$ ($i = 0, 1, \ldots, k^n - 1$) как неотрицательных, так и отрицательных значений, что требует удваивания числового диапазона по сравнению с использованием неотрицательных коэффициентов, а их абсолютные значения и, тем более абсолютные значения промежуточных результатов вычисления (**??**), могут многократно превышать значение k.

Известно представление многозначной ФАЛ с помощью логического полинома [4]:

$$F^{(k)}(X) = \left| \sum_{i=0}^{k^n-1} f_i^{(k)} \, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_k, \tag{54}$$

где $f_i^{(k)}$ — коэффициенты, такие что $f_i^{(k)} \in \{0, 1, \ldots, k-1\}$;

$$x_u^{i_u} = \begin{cases} x_u, \ i_u \neq 0, \\ 1, \ i_u = 0. \end{cases}$$

Недостатком (**??**) является необходимость выполнения вычислений по «жестко» заданному модулю k (k — *простое число*), обязательно совпадающему со значностью ФАЛ, что не обеспечивает требуемую свободу выбора средств вычисления ФАЛ.

6.2. Модулярные формы k-значной логики

Теорема 7.
Пусть $m \geq k$, где k — значность логики и m — простое, тогда произвольная ФАЛ может быть представлена арифметическим полиномом:

$$\mu^{(k)}(X) = \left| \sum_{i=0}^{k^n-1} \rho_i \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m , \tag{55}$$

где $\rho_i, h_u^{i_u} \in \mathcal{Z}_m$;

$$x_u^{i_u} = \begin{cases} x_u, \ i_u \neq 0, \\ 1, \ i_u = 0. \end{cases}$$

Доказательство теоремы 7 строится аналогично доказательству теоремы 3.

Замечание 8.
Арифметические полиномы (??) и (??) связаны отношением $\rho_i = |p_i^{(k)}|_m$.

Определение 11.
Выражение (??) будем называть представлением ФАЛ k-значной логики на основе модулярной формы арифметического полинома.

Примитивный алгоритм (рис. ??) получения (??), который состоит из двух шагов и заключается в 1) вычислении арифметического полинома (??) и 2) определении арифметического полинома (??) на основании замечания 8.

Шаг 1	$p_0^{(k)}$	$p_1^{(k)}$	$\dots$	$p_{k^n-1}^{(k)}$
	$\Downarrow$	$\Downarrow$		$\Downarrow$
Шаг 2	$\rho_0 = \lvert p_0^{(k)} \rvert_m$	$\rho_1 = \lvert p_1^{(k)} \rvert_m$	$\dots$	$\rho_{k^n-1} = \lvert p_{k^n-1}^{(k)} \rvert_m$

Рис. 17: Пояснения к варианту построения алгоритма получения (??) на основании замечания 8

К (??) справедливы следствия 1 и 2.

Основные схемы вычисления k-значной ФАЛ посредством модулярной формы арифметического полинома представлены на рис. ?? и ??, из которых следует, что вычисления из поля рациональных чисел $\mathcal{R}$ перенесены в простое поле $PG(m)$ целых чисел.

Полученный результат можно наглядно продемонстрировать на следующих примерах.

Пример 17.
Выберем значения $k = 2, 3, 4$ и $n = 2$. Определим модуль $m = 5$, величина которого удовлетворяет всем выбранным значениям k (то есть $m \geq k$). Используем полиномы (??) и (??) для представления следующих элементарных ФАЛ:

$\overline{x} = (k-1) - x$ — инверсия;
$\hat{x} = |x+1|_k$ — циклическое отрицание ($\overline{x} = x \oplus 1$, при $k = 2$);
$x_1 \vee x_2 = \max(x_1, x_2)$ — дизъюнкция;

Рис. 18: Схема вычисления произвольной k-значной ФАЛ над $PG(m)$

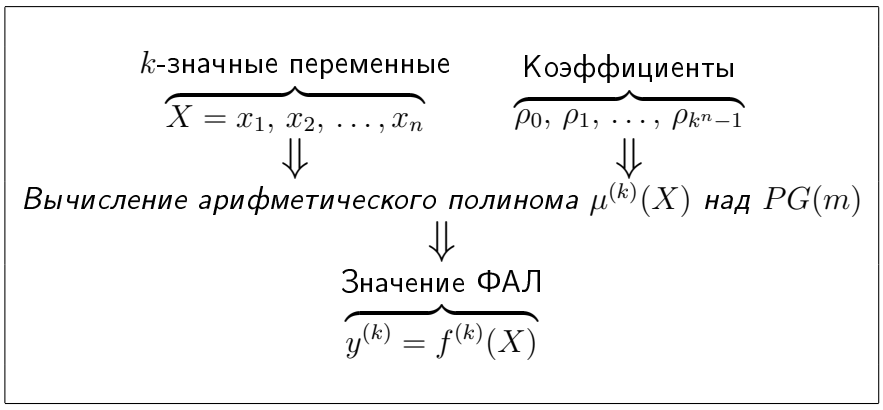

Рис. 19: Схема вычисления заданной k-значной ФАЛ над $PG(m)$

$x_1 \wedge x_2 = \min(x_1, x_2)$ — конъюнкция;
$|x_1 + x_2|_k$ — сложение по модулю k ($x_1 \oplus x_2$, при $k = 2$);
$|x_1 x_2|_k$ — умножение по модулю k ($x_1 \wedge x_2$, при $k = 2$);
$x_1 \uparrow x_2 = |(x_1 \vee x_2) + 1|_k$ — функция Вебба ($\overline{x_1 \vee x_2} = (x_1 \vee x_2) \oplus 1$ — «стрелка Пирса», при $k = 2$);
$x_1 \downarrow x_2 = \overline{x_1 \wedge x_2}$ — «штрих Шеффера», при $k = 2$.

Результаты использования (??) и (??) для $k = 2, 3, 4$ представлены соответственно в табл. ??—??.

Для пояснения принципа преобразования представления ФАЛ из (??) в (??) рассмотрим следующий пример.

Пример 18.
При $k = 3$ получить модулярную форму арифметического полинома (??) на основе данного арифметического полинома вида (??) ФАЛ $|x_1 + x_2|_k$ для произвольного m и $m = 5$ (табл. ??):

$$|x_1 + x_2|_3 =$$

Таблица 7:

$f(\mathbf{X})$	$P(\mathbf{X})$	$\mu(\mathbf{X})$ при $m=5$
$\overline{x}_i$	$1-x_i$	$\lvert 1+4x_i\rvert_5$
$x_1 \wedge x_2$	x_1x_2	$\lvert x_1x_2\rvert_5$
$x_1 \vee x_2$	$x_1+x_2-x_1x_2$	$\lvert x_1+x_2+4x_1x_2\rvert_5$
$x_1 \oplus x_2$	$x_1+x_2-2x_1x_2$	$\lvert x_1+x_2+3x_1x_2\rvert_5$
$x_1 \downarrow x_2$	$1-x_1x_2$	$\lvert 1+4x_1x_2\rvert_5$
$x_1 \uparrow x_2$	$1-x_1-x_2+x_1x_2$	$\lvert 1+4x_1+4x_2+x_1x_2\rvert_5$

Таблица 8:

$f(\mathbf{X})$	$P(\mathbf{X})$	$\mu(\mathbf{X})$ при $m=5$
$\overline{x}_i$	$2-x_i$	$\lvert 2+4x_i\rvert_5$
$\hat{x}_i$	$1+5x_i/2-3x_i^2/2$	$\lvert 1+x_i^2\rvert_5$
$x_1 \wedge x_2$	$5x_1x_2/2-x_1x_2^2-x_1^2x_2+x_1^2x_2^2/2$	$\lvert 4x_1x_2^2+4x_1^2x_2+3x_1^2x_2^2\rvert_5$
$\lvert x_1+x_2\rvert_3$	$x_1+x_2+21x_1x_2/4-15x_1x_2^2/4-$ $-15x_1^2x_2/4+9x_1^2x_2^2/4$	$\lvert x_1+x_2+4x_1x_2+x_1^2x_2^2\rvert_5$
$\lvert x_1x_2\rvert_3$	$x_1x_2/4+3x_1x_2^2/4+$ $+3x_1^2x_2/4-3x_1^2x_2^2/4$	$\lvert 4x_1x_2+2x_1x_2^2+$ $+2x_1^2x_2+3x_1^2x_2^2\rvert_5$
$x_1 \uparrow x_2$	$1+5x_2/2-3x_2^2/2+5x_1/2-$ $-7x_1x_2/4+x_1x_2^2/4-3x_1^2/2+$ $+x_1^2x_2/4+x_1^2x_2^2/4$	$\lvert 1+x_2^2+2x_1x_2+4x_1x_2^2+$ $+x_1^2+4x_1^2x_2+4x_1^2x_2^2\rvert_5$

$$
\begin{aligned}
&= \left|x_1+x_2+|21|4^{\varphi(m)-1}|_m|_m x_1x_2+\left(m-|15|4^{\varphi(m)-1}|_m|_m\right)x_1x_2^2+\right.\\
&\quad \left.+\left(m-|15|4^{\varphi(m)-1}|_m|_m\right)x_1^2x_2+|9|4^{\varphi(m)-1}|_m|_m x_1^2x_2^2\right|_m =\\
&= \left|x_1+x_2+4x_1x_2+x_1^2x_2^2\right|_5 .
\end{aligned}
$$

где $\varphi(m)$ — функция Эйлера [18].

Пример 19.

При $k=3$ получить модулярную форму арифметического полинома из примера 17 для произвольного m и $m=5$:

$$
\begin{aligned}
\mu^{(3)}(X) &= \left|2-4x_2+2x_2^2-x_1+\frac{11}{2}x_1x_2-3x_1x_2^2-\right.\\
&\quad \left.-\frac{3}{2}x_1^2x_2+x_1^2x_2^2\right|_m =\\
&= \left|2+(m-|4|_m)\,x_2+2x_2^2+(m-1)x_1+\right.\\
&\quad \left.+|11|_m 2^{\varphi(m)-1}x_1x_2+\right.
\end{aligned}
$$

Таблица 9:

$f(\mathbf{X})$	$P(\mathbf{X})$	$\mu(\mathbf{X})$ при $m=5$
$\overline{x}_i$	$3-x_i$	$\lvert 3+4x_i\rvert_5$
$\hat{x}_i$	$1-x_i/3+2x_i^2-2x_i^3/3$	$\lvert 1+3x_i+2x_i^2+x_i^3\rvert_5$
$x_1 \wedge x_2$	$29x_1x_2/6-15x_1x_2^2/4-$ $-15x_1^2x_2/4+3x_1x_2^3/4+$ $+3x_1^3x_2/4+7x_1^2x_2^2/2-$ $-3x_1^2x_2^3-3x_1^3x_2^2/4+x_1^3x_2^3/6$	$\lvert 4x_1x_2+2x_1x_2^3+$ $+2x_1^3x_2+x_1^2x_2^2+$ $+3x_1^2x_2^3+3x_1^3x_2^2+x_1^3x_2^3\rvert_5$
$\lvert x_1+x_2\rvert_4$	$x_1+x_2-121x_1x_2/9+$ $+49x_1x_2^2/3+49x_1^2x_2/3-$ $-38x_1^3x_2/9-38x_1x_2^3/9-$ $-19x_1^2x_2^2+14x_1^2x_2^3/3+$ $+14x_1^3x_2^2/3-10x_1^3x_2^3/9$	$\lvert x_1+x_2+x_1x_2+3x_1x_2^2+$ $+3x_1^2x_2+3x_1^3x_2+3x_1x_2^3+$ $+x_1^2x_2^2+3x_1^2x_2^3+3x_1^3x_2^2\rvert_5$
$x_1 \uparrow x_2$	$1-x_2/3+2x_2^2-2x_2^3/3-$ $-x_1/3-79x_1x_2/18+$ $+37x_1x_2^2/12-19x_1x_2^3/36+$ $+2x_1^2+37x_1^2x_2/12-$ $-15x_1^2x_2^2/6+5x_1^2x_2^3/12-$ $-2x_1^3/3-19x_1^3x_2/36+$ $+5x_1^3x_2^2/12-x_1^3x_2^3/18$	$\lvert 1+3x_2+2x_2^2+x_2^3+$ $+3x_1+2x_1x_2+x_1x_2^2+$ $+x_1x_2^3+2x_1^2+x_1^2x_2+$ $+x_1^3+x_1^3x_2+3x_1^3x_1^3\rvert_5$

$$
\begin{aligned}
&+(m-3)x_1x_2^2+(m-3)2^{\varphi(m)-1}x_1^2x_2+x_1^2x_2^2\big|_m = \\
= \;& \left|2+x_2+2x_2^2+4x_1+3x_1x_2+2x_1x_2^2+x_1^2x_2+x_1^2x_2^2\right|_5 .
\end{aligned}
$$

6.3. Теоретико-числовые преобразования на k-значной логике

Введем понятие модулярного базиса ЛТЧП на k-значной логике (в дальнейшем k-ЛТЧП). Обозначим как $\mathbf{\Delta}_k$ и $\mathbf{\Delta}_k^{-1}$ соответственно базовые матрицы размерности $k \times k$ прямого и обратного k-ЛТЧП. Причем, если обозначить элементы базовых матриц k-ЛДПФ $\mathbf{K}_k$ и $\mathbf{K}_k^{-1}$ соответственно как k_{ij} и $k_{ij}^{(-1)}$, то элементы δ_{ij} и $\delta_{ij}^{(-1)}$ матриц $\mathbf{\Delta}_k$ и $\mathbf{\Delta}_k^{-1}$ образуются из элементов матриц $\mathbf{K}_k$ и $\mathbf{K}_k^{-1}$ соответственно как $\delta_{ij} = |k_{ij}|_m$ и $\delta_{ij}^{(-1)} = \left|k_{ij}^{(-1)}\right|_m$. Тогда модулярный базис k-ЛТЧП — матрицы $\mathbf{\Delta}_{k^n}$ и $\mathbf{\Delta}_{k^n}^{-1}$ — определятся кронекеровским возведением в степень:

$$\mathbf{\Delta}_{k^n} = \bigotimes_{j=1}^{n} \mathbf{\Delta}_k \pmod m;$$

$$\mathbf{\Delta}_{k^n}^{-1} = \bigotimes_{j=1}^{n} \mathbf{\Delta}_k^{-1} \pmod m,$$

где запись $\bmod m$ означает, что все арифметические операции с элементами матриц выполняются по модулю m.

Пример 20.
Зададим значения модуля $m = 5$ и $m = 7$. Тогда базовые матрицы $\mathbf{\Delta}_k$ и $\mathbf{\Delta}_k^{-1}$ для различных значений k будут иметь вид, представленный в табл. ?? и ??.

Таблица 10: Расчетные $\mathbf{\Delta}_k$ и $\mathbf{\Delta}_k^{-1}$ при $m = 5$

k	$\mathbf{\Delta}_k$	$\mathbf{\Delta}_k^{-1}$
2	$\begin{bmatrix} 1 & 0 \\ 4 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$
3	$\begin{bmatrix} 2 & 0 & 0 \\ 2 & 4 & 4 \\ 1 & 3 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \end{bmatrix}$
4	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 4 & 3 & 1 & 2 \\ 1 & 0 & 2 & 2 \\ 4 & 3 & 2 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 3 \\ 1 & 3 & 4 & 2 \end{bmatrix}$
5	$\begin{bmatrix} 4 & 0 & 0 & 0 & 0 \\ 0 & 1 & 3 & 2 & 4 \\ 0 & -1 & 4 & 4 & 1 \\ 0 & 1 & 2 & 3 & 4 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 3 & 1 \\ 1 & 3 & 4 & 2 & 1 \\ 1 & 4 & 1 & 4 & 1 \end{bmatrix}$

Теорема 8.
Если для ФАЛ задана пара матричных преобразований (ЛДПФ) (??) и (??) и $m \geq k$ (m — простое), то справедливы преобразования:

$$\mathbf{\Psi}^{(k)} = R_k \mathbf{\Delta}_{k^n} \mathbf{S} \pmod m, \tag{56}$$

$$\mathbf{S} = \mathbf{\Delta}_{k^n}^{-1} \mathbf{\Psi}^{(k)} \pmod m, \tag{57}$$

где $R_k = \left|\frac{1}{N}\right|_m$ — модулярная форма нормализующего множителя; $\mathbf{\Delta}_{k^n}$ и $\mathbf{\Delta}_{k^n}^{-1}$ — соответственно модулярная форма матриц прямого и инверсного арифметического преобразования размерности $k^n \times k^n$ (базис преобразования); $\mathbf{S}$ — вектор истинности k-значной ФАЛ.

Для доказательства теоремы 8 необходимо учесть взаимооднозначность связи между матричной (??), (??) и полиномиальной (??) формами представления системы булевых функций [47]. Тогда справедливость (??) и (??) вытекает из справедливости (??).

Определение 12.
Пару преобразований (??) и (??) будем соответственно называть модулярной формой прямого и обратного матричного арифметического преобразования или ЛТЧП на k-значной логике (k-ЛТЧП).

Таблица 11: Расчетные $\mathbf{\Delta}_k$ и $\mathbf{\Delta}_k^{-1}$ при $m = 7$

k	$\mathbf{\Delta}_k$	$\mathbf{\Delta}_k^{-1}$
2	$\begin{bmatrix} 1 & 0 \\ 6 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$
3	$\begin{bmatrix} 2 & 0 & 0 \\ 4 & 4 & 6 \\ 1 & 5 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \end{bmatrix}$
4	$\begin{bmatrix} 6 & 0 & 0 & 0 \\ 3 & 4 & 5 & 2 \\ 6 & 6 & 5 & 4 \\ 6 & 3 & 4 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 1 \\ 1 & 3 & 2 & 6 \end{bmatrix}$
5	$\begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 6 & 5 & 5 & 4 & 1 \\ 0 & 1 & 2 & 0 & 4 \\ 4 & 1 & 1 & 0 & 1 \\ 1 & 3 & 1 & 3 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 1 & 2 \\ 1 & 3 & 2 & 6 & 4 \\ 1 & 4 & 2 & 1 & 4 \end{bmatrix}$
6	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 6 & 2 & 2 & 1 & 4 & 3 \\ 1 & 0 & 6 & 4 & 4 & 1 \\ 6 & 5 & 5 & 0 & 5 & 0 \\ 1 & 0 & 4 & 6 & 6 & 4 \\ 6 & 5 & 4 & 3 & 2 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 1 & 2 & 4 \\ 1 & 3 & 2 & 6 & 4 & 5 \\ 1 & 4 & 2 & 1 & 4 & 2 \\ 1 & 5 & 3 & 6 & 2 & 3 \end{bmatrix}$

Пример 21.
Продемонстрируем применение 3-ЛТЧП к условиям, использованным в примере 16. Выберем модуль $m = 5$. Предварительно вычислим матрицы $\mathbf{\Delta}_{k^n}$ и $\mathbf{\Delta}_{k^n}^{-1}$, воспользовавшись табл. **??**:

$$\mathbf{\Delta}_{3^2} = \begin{bmatrix} 2 & 0 & 0 \\ 2 & 4 & 4 \\ 1 & 3 & 1 \end{bmatrix} \otimes \begin{bmatrix} 2 & 0 & 0 \\ 2 & 4 & 4 \\ 1 & 3 & 1 \end{bmatrix} \pmod 5 =$$

$$= \begin{bmatrix} 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 3 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 3 & 0 & 0 & 3 & 0 & 0 \\ 4 & 3 & 3 & 3 & 1 & 1 & 3 & 1 & 1 \\ 2 & 1 & 2 & 4 & 2 & 4 & 4 & 2 & 4 \\ 2 & 0 & 0 & 1 & 0 & 0 & 2 & 0 & 0 \\ 2 & 4 & 4 & 1 & 2 & 2 & 2 & 4 & 4 \\ 1 & 3 & 1 & 3 & 4 & 3 & 1 & 3 & 1 \end{bmatrix};$$

$$\mathbf{\Delta}_{3^2}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \end{bmatrix} \pmod 5 =$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 1 & 2 & 4 & 1 & 2 & 4 \\ 1 & 0 & 0 & 2 & 0 & 0 & 4 & 0 & 0 \\ 1 & 1 & 1 & 2 & 2 & 2 & 4 & 4 & 4 \\ 1 & 2 & 4 & 2 & 4 & 3 & 4 & 3 & 1 \end{bmatrix}.$$

Прямое 3-ЛТЧП (**??**), при с учетом $R_3 = 4$, примет вид:

$$\begin{aligned} \mathbf{\Psi}^{(3)} &= R_3 \cdot \mathbf{\Delta}_{3^2}\mathbf{S} \pmod 5 = \\ &= 4 \cdot \begin{bmatrix} 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 3 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 3 & 0 & 0 & 3 & 0 & 0 \\ 4 & 3 & 3 & 3 & 1 & 1 & 3 & 1 & 1 \\ 2 & 1 & 2 & 4 & 2 & 4 & 4 & 2 & 4 \\ 2 & 0 & 0 & 1 & 0 & 0 & 2 & 0 & 0 \\ 2 & 4 & 4 & 1 & 2 & 2 & 2 & 4 & 4 \\ 1 & 3 & 1 & 3 & 4 & 3 & 1 & 3 & 1 \end{bmatrix} \cdot \begin{bmatrix} 2 \\ 0 \\ 2 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 2 \end{bmatrix} \pmod 5 = \\ &= \begin{bmatrix} 2 \\ 1 \\ 2 \\ 4 \\ 3 \\ 2 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{matrix} \\ x_2 \\ x_2^2 \\ x_1 \\ x_1x_2 \\ x_1x_2^2 \\ x_1^2 \\ x_1^2x_2 \\ x_1^2x_2^2 \end{matrix}. \end{aligned}$$

Обратное преобразование (**??**) будет иметь вид:

$$\begin{aligned} \mathbf{S} &= \mathbf{\Delta}_{3^2}^{-1}\mathbf{\Psi}^{(3)} \pmod 5 = \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 1 & 2 & 4 & 1 & 2 & 4 \\ 1 & 0 & 0 & 2 & 0 & 0 & 4 & 0 & 0 \\ 1 & 1 & 1 & 2 & 2 & 2 & 4 & 4 & 4 \\ 1 & 2 & 4 & 2 & 4 & 3 & 4 & 3 & 1 \end{bmatrix} \cdot \begin{bmatrix} 2 \\ 1 \\ 2 \\ 4 \\ 3 \\ 2 \\ 0 \\ 1 \\ 1 \end{bmatrix} \begin{matrix} \\ x_2 \\ x_2^2 \\ x_1 \\ x_1x_2 \\ x_1x_2^2 \\ x_1^2 \\ x_1^2x_2 \\ x_1^2x_2^2 \end{matrix} \pmod 5 = \end{aligned}$$

$$= \begin{bmatrix} 2 \\ 0 \\ 2 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 2 \end{bmatrix}.$$

По аналогии с методикой расчета преимуществ модулярных форм арифметической логики для $k = 2$ (**??**) можно определить выигрыш от использования модулярных форм для k-значной логики. В частности для $k = 3$ выигрыш (размерность данных) от использования модулярных форм составит $\frac{3n}{\lceil \log_2 3^d \rceil}$ раз. Преимущества модулярных форм с увеличением k возрастают.

Метод вычисления многозначных ФАЛ на основе одномодульной арифметики позволяет реализовать гибкие логические вычисления с помощью аппаратных средств, функционирующих по заданному значению модуля (а не по модулю k). В качестве модулей определенные преимущества имеют простые числа Мерсенна и числа Ферма. Размерность промежуточных результатов при использовании модулярных форм уменьшается, что позволяет отказаться от использования арифметики многократной точности и сократить время логических вычислений.

6.4. Реализация систем k-значных функций

Не затрагивая вопросов представления систем k-значных ФАЛ [24] линейными арифметическими полиномами, рассмотрим некоторые принципиальные вопросы для нелинейных арифметических полиномов, решение которых могло бы дать обобщение и на линейные полиномы.

6.4.1. Арифметический полином для реализации систем k-значных функций. Будем использовать аналогию с алгоритмом 1.

Алгоритм 2.

Шаг 1. Поставим в соответствие ФАЛ

$$f_1^{(k)}(X),\ f_2^{(k)}(X),\ \ldots,\ f_d^{(k)}(X),$$

арифметические полиномы вида (**??**):

$$\begin{aligned} P_1^{(k)}(X) &= \sum_{i=0}^{k^n-1} p_{1,i}^{(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \\ P_2^{(k)}(X) &= \sum_{i=0}^{k^n-1} p_{2,i}^{(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \\ &\vdots \\ P_d^{(k)}(X) &= \sum_{i=0}^{k^n-1} p_{d,i}^{(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}. \end{aligned}$$

Шаг 2. Умножим эти полиномы на веса k^{j-1} ($j = 1, 2, \ldots, d$):

$$\begin{aligned} P_1^{*(k)}(X) &= k^0 P_1^{(k)}(X) = \sum_{i=0}^{k^n-1} p_{1,\,i}^{*(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \\ P_2^{*(k)}(X) &= k^1 P_1^{(k)}(X) = \sum_{i=0}^{k^n-1} p_{2,\,i}^{*(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \\ &\vdots \\ P_d^{*(k)}(X) &= k^{d-1} P_1^{(k)}(X) = \sum_{i=0}^{k^n-1} p_{d,\,i}^{*(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \end{aligned}$$

где $p_{j,\,i}^{*(k)} = k^{j-1} p_{j,\,i}^{(k)}$. ($j = 1, 2, \ldots, d$; $i = 0, 1, \ldots, k^n - 1$)

Шаг 3. Получение арифметического полинома

$$\begin{aligned} D^{(k)}(X) &= \sum_{i=0}^{k^n-1} \sum_{j=1}^{d} p_{j,\,i}^{*(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} = \\ &= \sum_{i=0}^{k^n-1} c_i^{(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}, \end{aligned} \tag{58}$$

где $c_i^{(k)} = \sum_{j=1}^{d} p_{j,\,i}^{*(k)}$ ($i = 0, 1, \ldots, k^n - 1$).

Если результат вычисления (**??**), получить в k-ичной системе счисления, то он будет представлять собой кортеж значений искомых ФАЛ

$$y_1^{(k)}(X) \odot y_2^{(k)}(X) \odot \cdots \odot y_d^{(k)}(X),$$

интерпретируемый как код целого неотрицательного числа:

$$(y_d^{(k)}\, y_{d-1}^{(k)}\, \ldots\, y_1^{(k)})_k = Y^{(k)} = \sum_{j=1}^{d} y_j^{(k)} k^{j-1}.$$

Однако, в силу того, что результат $Y^{(k)}$ представлен в двоичной системе счисления, для значений ФАЛ применяется оператор маскирования $\Xi^t\{Y^{(k)}\}$, служащий для определения t-го k-ичного разряда (выхода) представления

$$Y^{(k)} = (y_d^{(k)}\ \ldots\ y_t^{(k)}\ \ldots\ y_1^{(k)})_k,$$

т. е. $\Xi^t\{Y^{(k)}\} = y_t^{(k)}$.

Для вычисления оператора $\Xi^t\{Y^{(k)}\}$ используется формула:

$$\Xi^t\{Y^{(k)}\} = \left| \left\lfloor \frac{Y^{(k)}}{k^t} \right\rfloor \right|_k. \tag{59}$$

6.4.2. Модулярная форма для реализации систем k-значных функций.

Теорема 9.

Если $m > Y_{\max}^{(k)}$, где $Y_{\max}^{(k)}$ — максимальное значение, принимаемое $Y^{(k)}$, то произвольный кортеж k-значных ФАЛ может быть представлен арифметическим полиномом:

$$Y^{(k)} = \Omega^{(k)}(X) = \left| \sum_{i=0}^{k^n-1} \omega_i^{(k)} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m , \tag{60}$$

где $\omega_i^{(k)} = \left| c_i^{(k)} \right|_m$ $(i = 0, 1, \dots, k^n - 1)$.

В качестве доказательства теоремы 9 построим алгоритм.

Алгоритм 3.

Шаг 1. Поставим d ФАЛ:

$$f_1^{(k)}(X), \; f_2^{(k)}(X), \; \dots, \; f_d^{(k)}(X)$$

в соответствие арифметические полиномы вида (??):

$$\begin{aligned}
\mu_1^{(k)}(X) &= \left| \sum_{i=0}^{k^n-1} \rho_{1,\,i}^{(k)} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m , \\
\mu_2^{(k)}(X) &= \left| \sum_{i=0}^{k^n-1} \rho_{2,\,i}^{(k)} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m , \\
&\vdots \\
\mu_d^{(k)}(X) &= \left| \sum_{i=0}^{k^n-1} \rho_{d,\,i}^{(k)} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m .
\end{aligned}$$

Шаг 2. Вычисление модулярных произведений:

$$\begin{aligned}
\left| l^0 \mu_1^{(k)}(X) \right|_m &= \left| \sum_{i=0}^{k^n-1} \rho_{1,\,i}^{*(k)} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m , \\
\left| l^1 \mu_1^{(k)}(X) \right|_m &= \left| \sum_{i=0}^{k^n-1} \rho_{1,\,i}^{*(k)} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m , \\
&\vdots \\
\left| l^{d-1} \mu_1^{(k)}(X) \right|_m &= \left| \sum_{i=0}^{k^n-1} \rho_{1,\,i}^{*(k)} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m ,
\end{aligned}$$

где $\rho_{j,\,i}^{*(k)} = \left| l^{j-1} \rho_{j,\,i}^{(k)} \right|_m$; $l = l^{j-1} = |k^{j-1}|_m$; $j = 1, 2, \dots, d$; $i = 0, 1, \dots, k^n - 1$.

Шаг 3.

$$\begin{aligned}
\Omega^{(k)}(X) &= \left| \sum_{i=0}^{k^n-1} \sum_{j=1}^{d} \rho_{j,\,i}^{*(k)} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m = \\
&= \left| \sum_{i=0}^{k^n-1} \omega_i^{(k)} \, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m ,
\end{aligned}$$

где $\omega_i^{(k)} = \sum_{j=1}^{d} \rho_{j,i}^{*(k)}$ $(i = 0, 1, \ldots, k^n - 1)$.

Структура алгоритма 3 аналогична структуре **??**.

□

6.5. Выводы

- Получено обобщение модулярных форм арифметических полиномов на область k-значной логики. При этом достигаются преимущества по ограничению величины коэффициентов арифметического полинома и величины промежуточных результатов вычислений.
- Показано, что посредством модулярной формы арифметического полинома вычисления k-значных логических функций из поля рациональных чисел $\mathcal{R}$ могут быть перенесены на простое поле Галуа $PG(m)$.
- Установлена взаимосвязь между немодулярной и модулярной формами арифметического полинома на k-значной логике.
- Построены теоретико-числовые преобразования на k-значной логике.
- Модулярные формы обобщены для реализации систем k-значных ФАЛ.
- Указано, что предложенные модулярные формы позволяют повысить гибкость логических вычислений за счет обеспечения возможности применения аппаратных или программных средств, функционирующих не по «жестко» заданному значению модуля k, а и по другим значениям модуля. Или наоборот — когда необходимо выполнять вычисления для различных значностей логики — k на основе аппаратных или программных средств, функционирующих по одному, заранее заданному значению модуля.

7. Альтернативные арифметико-логические формы

Все рассмотренные выше методы построения арифметических форм основаны на принципе взвешивания — умножения полиномов, соответствующих отдельным логическим функциям, на степени значности k, или их модулярные эквиваленты. Необходимо отметить, что принцип взвешивания — не единственный прием, который может быть использован для построения (кодирования) арифметических форм и последующего различения (маскирования) логических функций. Рассмотрим два альтернативных метода кодирования [78], основанные на 1) единственности представления числа его разложением на простые множители и 2) Китайской теореме об остатках. Рассмотрим также случаи, в которых с помощью этих методов можно было бы получить некоторую пользу.

7.1. Мультипликативно-кодированные арифметико-логические формы

Теорема 10.
Произвольный кортеж булевых функций

$$y_d \odot y_{d-1} \odot \ldots \odot y_1$$

может быть единственным способом задан арифметическим выражением

$$N(X) = \prod_{i=1}^{2^n-1} N_0 N_i^{x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}}, \tag{61}$$

где $N_i > 0$, $i = 0, 1, \ldots, 2^n - 1$.

Доказательство теоремы 10.
Сопоставим булевым функциям $f_i(X)$ числа $m_i^{f_i(X)}$ $(i = 1, 2, \ldots, d)$, где m_i $(i = 1, 2, \ldots, d)$ — простые числа:

$$\begin{array}{cccc} f_1(X) & f_2(X) & \cdots & f_d(X) \\ \downarrow & \downarrow & & \downarrow \\ m_1^{f_1(X)} & m_2^{f_2(X)} & \cdots & m_d^{f_d(X)}. \end{array}$$

Построим с учетом (??) произведение

$$\begin{gathered} N(X) = m_1^{f_1(X)} \times \ldots \times m_d^{f_d(X)} = \\ = m_1^{\sum_{i=0}^{2^n-1} r_i x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}} \times \ldots \times m_d^{\sum_{i=0}^{2^n-1} r_i x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}}. \end{gathered} \tag{62}$$

Далее, (??) можем переписать в виде

$$\begin{array}{rcccccl} N(X) = & \left(m_1^{r_{1,0}}\right. & \times & m_1^{r_{1,1} x_n} & \times \ldots \times & \left. m_1^{r_{1,2^n-1} x_1 x_2 \ldots x_n}\right) & \times \\ & \times \left(m_2^{r_{2,0}}\right. & \times & m_2^{r_{2,1} x_n} & \times \ldots \times & \left. m_2^{r_{2,2^n-1} x_1 x_2 \ldots x_n}\right) & \times \\ & \vdots & & & & & \vdots \\ & \times \left(m_d^{r_{d,0}}\right. & \times & m_d^{r_{d,1} x_n} & \times \ldots \times & \left. m_d^{r_{d,2^n-1} x_1 x_2 \ldots x_n}\right) & = \end{array}$$

$$\begin{array}{rcccccl} = & \left(m_1^{r_{1,0}}\right. & & & \times \ldots \times & \left. m_d^{r_{d,0}}\right) & \times \\ \times & \left(m_1^{r_{1,1} x_n}\right. & & & \times \ldots \times & \left. m_d^{r_{d,1} x_n}\right) & \times \\ & \vdots & & & & & \vdots \\ \times & \left(m_1^{r_{d,2^n-1} x_1 x_2 \ldots x_n}\right. & & & \times \ldots \times & \left. m_d^{r_{d,2^n-1} x_1 x_2 \ldots x_n}\right) & = \end{array}$$

$$= N_0 N_1^{x_n} \ldots N_{2^n-1}^{x_1 x_2 \ldots x_n},$$

где

$$\begin{array}{rcl} N_0 & = & m_1^{r_{1,0}} \times \ldots \times m_d^{r_{d,0}}, \\ N_1 & = & m_1^{r_{1,1} x_n} \times \ldots \times m_d^{r_{d,1} x_n}, \\ & \vdots & \\ N_{2^n-1} & = & m_1^{r_{d,2^n-1} x_1 x_2 \ldots x_n} \times \ldots \times m_d^{r_{d,2^n-1} x_1 x_2 \ldots x_n}. \end{array}$$

Единственность (??) следует из 1) основной теоремы арифметики о единственности разложения числа на простые множители и 2) из единственности

представления (??) в соответствии с теоремой 1.

□

Определение 13.
Выражение (??) будем называть мультипликативно-кодированной арифметико-логической формой.

Свойство 1.
В результате вычисления (??) значение t-й булевой определяется проверкой условия делимости $N(X)$ на число m_t:

$$f_t(X) = \begin{cases} 1, & \text{если} \quad m_t | N(X), \\ 0, & \text{если} \quad m_t \not| N(X) \end{cases} \tag{63}$$

или

$$f_t(X) = \begin{cases} 1, & \text{если} \quad |N(X)|_{m_t} = 0, \\ 0, & \text{если} \quad |N(X)|_{m_t} \neq 0. \end{cases}$$

Для проверки выполнимости (??) могут быть использованы признаки делимости [18, 19].

Свойство 2.

$$\mu\left(N(X)\right) = (-1)^d, \quad \text{при} \quad \bigvee_{i=1}^{d} y_i = 1,$$

$$\mu\left(N(X)\right) = 1, \quad \text{при} \quad \bigvee_{i=1}^{d} y_i = 0,$$

где $\mu(a)$ — функция Мебиуса от a [18].

Можно предложить следующий алгоритм получения (??).

Алгоритм 4.

Шаг 1. Определение коэффициентов $r_{1,i}, r_{2,i}, \ldots, r_{d,i}$ путем получения арифметического полинома $P_j(X)$ для каждой булевой функции $y_j = f_j(X)$, $j = 1, 2, \ldots, d$:

$$\begin{cases} f_1(X) = P_1(X) = \sum\limits_{i=0}^{2^n-1} r_{1,i}\, x_1^{i_1} x_2^{i_2} \ \ldots \ x_n^{i_n}, \\ f_2(X) = P_2(X) = \sum\limits_{i=0}^{2^n-1} r_{2,i}\, x_1^{i_1} x_2^{i_2} \ \ldots \ x_n^{i_n}, \\ \vdots \\ f_d(X) = P_d(X) = \sum\limits_{i=0}^{2^n-1} r_{d,i}\, x_1^{i_1} x_2^{i_2} \ \ldots \ x_n^{i_n}. \end{cases} \tag{64}$$

Шаг 2. Выбор d простых чисел $m_1, m_2, \ldots, m_d$ и вычисление $N_0, N_1, \ldots, N_{2^n-1}$:

$$\begin{aligned} N_0 &= m_1^{r_{1,0}} \times \ldots \times m_d^{r_{d,0}}, \\ N_1 &= m_1^{r_{1,1} x_n} \times \ldots \times m_d^{r_{d,1}\, x_n}, \\ &\vdots \\ N_{2^n-1} &= m_1^{r_{d,2^n-1}\, x_1 x_2 \ldots x_n} \times \ldots \times m_d^{r_{d,2^n-1}\, x_1 x_2 \ldots x_n}. \end{aligned}$$

Пример 22.
Используем условия таблицы **??** (пример 1).

Шаг 1. Согласно примеру 2:

$$\begin{aligned} f_1(X) &= P_1(X) = \overline{x_1 \oplus x_2} = 1 - x_1 - x_2 + 2x_1x_2, \\ f_2(X) &= P_2(X) = \overline{x_1 \vee x_2} = 1 - x_1 - x_2 + x_1x_2. \end{aligned}$$

Шаг 2. Выберем первые два простых числа (порядок следования простых чисел произволен) $m_1 = 2$, $m_2 = 3$ и вычислим числа $N_0, \ldots, N_3$:

$$\begin{aligned} N_0 &= 2^1 \cdot 3^1 = 6, \\ N_1 &= 2^{-x_2} \cdot 3^{-x_2} = \left(\frac{1}{6}\right)^{x_2}, \\ N_2 &= 2^{-x_1} \cdot 3^{-x_1} = \left(\frac{1}{6}\right)^{x_1}, \\ N_3 &= 2^{2x_1x_2} \cdot 3^{x_1x_2} = 12^{x_1x_2}. \end{aligned}$$

Таким образом

$$N(X) = 6 \times \left(\frac{1}{6}\right)^{x_2} \times \left(\frac{1}{6}\right)^{x_1} \times 12^{x_1x_2}.$$

Результат реализации функций $f_1(X)$ и $f_2(X)$ на полном наборе булевых переменных x_1 и x_2 представлен в табл. **??**.

Таблица 12: Пример реализации булевых функций (двоичный полусумматор) с помощью мультипликативно-кодированной арифметико-логической формы

x_2	x_1	$N(X)$	Проверка условия (**??**)		y_2	y_1
0	0	6	$\|6\|_3 = 0$	$\|6\|_2 = 0$	1	1
0	1	1	$\|1\|_3 \neq 0$	$\|1\|_2 \neq 0$	0	0
1	0	1	$\|1\|_3 \neq 0$	$\|1\|_2 \neq 0$	0	0
1	1	2	$\|2\|_3 \neq 0$	$\|2\|_2 = 0$	0	1

7.2. Модулярно-кодированные арифметико-логические формы

Как следует из (**??**) для реализации оператора $\Xi^t\{Y^{(k)}\}$ требуется деление, округление и нахождение наименьшего неотрицательного вычета. Это обстоятельство уменьшает скорость вычислений на этапе реализации ФАЛ. Рассмотрим альтернативный принцип построения арифметического полинома, который в отличие принципа *декомпозиции*, использованного ранее, основан на *синтезе* одного арифметического полинома, обладающего свойством упрощения оператора маскирования.

Пусть дана система d ФАЛ:

$$f_1^{(k)}(X),\ f_2^{(k)}(X),\ \ldots,\ f_d^{(k)}(X),$$

где k — значность ФАЛ и, поставленная в соответствие им, система арифметических полиномов вида (??):

$$\mu_1^{(k)}(X),\ \mu_2^{(k)}(X),\ \ldots,\ \mu_d^{(k)}(X).$$

Выберем простые модули: $m_1, m_2, \ldots, m_d$ такие, что $m_i \geq k$, где $i = 1, 2, \ldots, d$. Соблюдение условия простоты модулей необходимо для обеспечения вычислений в конечном поле.

Будем отождествлять полиномы $\mu_1^{(k)}(X),\ \mu_2^{(k)}(X),\ \ldots,\ \mu_d^{(k)}(X)$ с вычетами $\phi_1, \phi_2, \ldots, \phi_d$ в формуле (??):

$$Y = |\phi_1 B_1 + \phi_2 B_2 + \ldots + \phi_d B_d|_m\,,$$

где $m = m_1 m_2 \ldots m_d$.

В связи с этим рассмотрим следующую теорему.

Теорема 11.
Если даны простые модули $m_1, m_2, \ldots, m_d$ такие, что $m_i \geq k$, где $i, j = 1, 2, \ldots, d$, то произвольный кортеж k-значных ФАЛ может быть представлен арифметическим полиномом:

$$T^{(k)} = \Theta^{(k)}(X) = \left| \sum_{i=0}^{k^n-1} \zeta_i^{(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_m, \tag{65}$$

где $0 \leq \zeta_i^{(k)} < m$, причем

$$y_1^{(k)} = \left|T^{(k)}\right|_{m_1},\ y_2^{(k)} = \left|T^{(k)}\right|_{m_2},\ \ldots,\ y_d^{(k)} = \left|T^{(k)}\right|_{m_d}.$$

Доказательство теоремы 11 вытекает из следующего алгоритма.

Алгоритм 5.

Шаг 1. Построение полиномов вида (??) для представления каждой ФАЛ:

$$\begin{aligned}
\mu_1^{(k)}(X) &= \left| \sum_{i=0}^{k^n-1} \rho_{1,i}^{(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{m_1}, \\
\mu_2^{(k)}(X) &= \left| \sum_{i=0}^{k^n-1} \rho_{2,i}^{(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{m_2}, \\
&\vdots \\
\mu_d^{(k)}(X) &= \left| \sum_{i=0}^{k^n-1} \rho_{d,i}^{(k)}\, x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \right|_{m_d};
\end{aligned}$$

и запись в виде:

$$\begin{aligned}
\mu_1^{*(k)}(X) &= \sum_{i=0}^{k^n-1} \rho_{1,\,i}^{(k)}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}, \\
\mu_2^{*(k)}(X) &= \sum_{i=0}^{k^n-1} \rho_{2,\,i}^{(k)}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}, \\
&\vdots \\
\mu_d^{*(k)}(X) &= \sum_{i=0}^{k^n-1} \rho_{d,\,i}^{(k)}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}.
\end{aligned}$$

Шаг 2. Выполнение модулярных умножений:

$$\begin{aligned}
\left| B_1 \mu_1^{*(k)}(X) \right|_m &= \sum_{i=0}^{k^n-1} \zeta_{1,\,i}^{*(k)}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}, \\
\left| B_2 \mu_1^{*(k)}(X) \right|_m &= \sum_{i=0}^{k^n-1} \zeta_{2,\,i}^{*(k)}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}, \\
&\vdots \\
\left| B_d \mu_1^{*(k)}(X) \right|_m &= \sum_{i=0}^{k^n-1} \zeta_{d,\,i}^{*(k)}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n},
\end{aligned}$$

где $\zeta_{j,\,i}^{*(k)}(X) = \left| B_j \rho_{j,\,i}^{(k)} \right|_m$; $m = m_1 m_2 \dots m_d$; $B_i = q_i m m_i^{-1}$; q_i находится из $q_i m m_k^{-1} \equiv 1 \pmod{m_i}$ $(i = 1,\, 2,\, \dots,\, d)$.

Шаг 3.

$$\begin{aligned}
\Theta^{(k)}(X) &= \left| \sum_{i=0}^{k^n-1} \sum_{j=1}^{d} \zeta_{j,\,i}^{*(k)}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m = \\
&= \left| \sum_{i=0}^{k^n-1} \zeta_i^{(k)}\, x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \right|_m,
\end{aligned}$$

где $\zeta_i = \sum_{j=1}^{d} \zeta_{j,\,i}^{*(k)}$ $(i = 0,\, 1,\, \dots,\, k^n - 1)$.

Корректность вычислений в конечном поле обеспечивается соблюдением условия выбора простых модулей.

□

Определение 14.
Выражение (**??**) будем называть модулярно-кодированной арифметико-логической формой.

Пример 23.
Рассмотрим применение теоремы 11 для случая реализации двух 3-х значных ФАЛ, векторы значений которых при $n = 2$, имеют вид:

$$\begin{aligned}
\mathbf{S}_1 &= [0\ 0\ 2\ 2\ 1\ 1\ 0\ 0\ 2]^T, \\
\mathbf{S}_2 &= [2\ 0\ 2\ 1\ 1\ 1\ 0\ 1\ 2]^T.
\end{aligned}$$

Им соответствуют арифметические полиномы вида (??):

$$\begin{aligned} P_1^{(3)}(X) &= -x_2 + x_2^2 + 4x_1 - x_1x_2 - x_1x_2^2 - 2x_1^2 + \\ &\quad + \frac{1}{2}x_1^2x_2 + \frac{1}{2}x_1^2x_2^2, \\ P_2^{(3)}(X) &= 2 - 4x_2 + 2x_2^2 - x_1 + \frac{11}{2}x_1x_2 - 3x_1x_2^2 - \\ &\quad - \frac{3}{2}x_1^2x_2 + x_1^2x_2^2. \end{aligned}$$

В соответствии с теоремой 11 выберем модули: $m_1 = 3$, $m_2 = 5$. Тогда $m = m_1m_2 = 15$, значения констант B_1 и B_2 в соответствии с (??) составят:

$$\begin{aligned} B_1 &= \frac{q_1m}{m_1} = \frac{2 \cdot 15}{3} = 10, \\ B_2 &= \frac{q_2m}{m_2} = \frac{2 \cdot 15}{5} = 6. \end{aligned}$$

Применение алгоритма 5 в этом случае будет выглядеть следующим образом.

Шаг 1. Построение полиномов вида (??) (пример 19):

$$\begin{aligned} \mu_1^{(3)}(X) &= \left|2x_2 + x_2^2 + x_1 + 2x_1x_2 + 2x_1x_2^2 + x_1^2 + \right. \\ &\quad \left. + 2x_1^2x_2 + 2x_1^2x_2^2\right|_3, \\ \mu_2^{(3)}(X) &= \left|2 + x_2 + 2x_2^2 + 4x_1 + 3x_1x_2 + 2x_1x_2^2 + \right. \\ &\quad \left. + x_1^2x_2 + x_1^2x_2^2\right|_5 \end{aligned}$$

и запись их в немодулярном виде:

$$\begin{aligned} \mu_1^{*(3)}(X) &= 2x_2 + x_2^2 + x_1 + 2x_1x_2 + 2x_1x_2^2 + x_1^2 + \\ &\quad + 2x_1^2x_2 + 2x_1^2x_2^2, \\ \mu_2^{*(3)}(X) &= 2 + x_2 + 2x_2^2 + 4x_1 + 3x_1x_2 + 2x_1x_2^2 + \\ &\quad + x_1^2x_2 + x_1^2x_2^2. \end{aligned}$$

Шаг 2. Выполнение модулярных умножений:

$$\begin{aligned} \left|10\mu_1^{*(3)}(X)\right|_{15} &= |10 \cdot 2|_{15}x_2 + 10x_2^2 + 10x_1 + |10 \cdot 2|_{15}x_1x_2 + \\ &\quad + |10 \cdot 2|_{15}x_1x_2^2 + 10x_1^2 + |10 \cdot 2|_{15}x_1^2x_2 + \\ &\quad + |10 \cdot 2|_{15}x_1^2x_2^2 = \\ &= 5x_2 + 10x_2^2 + 10x_1 + 5x_1x_2 + 5x_1x_2^2 + \\ &\quad + 10x_1^2 + 5x_1^2x_2 + 5x_1^2x_2^2, \\ \left|6\mu_1^{*(3)}(X)\right|_{15} &= |6 \cdot 2|_{15} + 6x_2 + |6 \cdot 2|_{15}x_2^2 + |6 \cdot 4|_{15}x_1 + \\ &\quad + |6 \cdot 3|_{15}x_1x_2 + |6 \cdot 2|_{15}x_1x_2^2 + 6x_1^2x_2 + \\ &\quad + 6x_1^2x_2^2 = \\ &= 12 + 6x_2 + 12x_2^2 + 9x_1 + 3x_1x_2 + 12x_1x_2^2 + \\ &\quad + 6x_1^2x_2 + 6x_1^2x_2^2. \end{aligned}$$

Шаг 3.

$$\begin{aligned}\Theta^{(3)}(X) &= |0+12|_{15} + |5+6|_{15}x_2 + |10+12|_{15}x_2^2 + \\ &+|10+9|_{15}x_1 + |5+3|_{15}x_1x_2 + \\ &+|5+12|_{15}12x_1x_2^2 + |10+0|_{15}x_1^2 + \\ &+|5+6|_{15}x_1^2x_2 + |5+6|_{15}x_1^2x_2^2.\end{aligned}$$

Таким образом

$$\begin{aligned}\Theta^{(3)}(X) = 12 + 11x_2 + 7x_2^2 + 4x_1 + 8x_1x_2 + 2x_1x_2^2 + \\ +10x_1^2 + 11x_1^2x_2 + 11x_1^2x_2^2.\end{aligned}$$

Для проверки рассмотрим два случая. Первый случай — $x_1 = 0$ и $x_2 = 0$ (первая строка таблицы истинности). Второй случай — $x_1 = 2$ и $x_2 = 2$ (последняя строка таблицы истинности).
В первом случае получим:

$$\Theta^{(3)}(X) = |12|_{15} = 12.$$

Результат (см. исходные условия):

$$\begin{aligned}y_1^{(3)} &= |12|_3 = 0, \\ y_2^{(3)} &= |12|_5 = 2.\end{aligned}$$

Во втором случае:

$$\begin{aligned}\Theta^{(3)}(X) &= \big|12 + |11\cdot 2|_{15} + |7\cdot 2^2|_{15} + |4\cdot 2|_{15} + \\ &+|8\cdot 2\cdot 2|_{15} + |2\cdot 2\cdot 2^2|_{15} + \\ &+|10\cdot 2^2|_{15} + |11\cdot 2^2\cdot 2|_{15} + \\ &+ |11\cdot 2^2\cdot 2^2|_{15}\big|_{15} = \\ &= |12+7+13+8+2+1+10+13+11|_{15} = \\ &= |77|_{15} = 2.\end{aligned}$$

Результат:

$$\begin{aligned}y_1^{(3)} &= |2|_3 = 2, \\ y_2^{(3)} &= |2|_5 = 2.\end{aligned}$$

7.3. Выводы

- Предложены мультипликативно-кодированные арифметико-логические формы. Реализация систем булевых функций с помощью мультипликативно-кодированных арифметико-логических форм может быть полезной при выполнении вычислений в недвоичной системе счисления. Обобщение (??) для k-значной логики и использование для формирования (??) линейных арифметических полиномов приведет к необходимости факторизации числа $N(X)$, что существенно усложнит вычисления.

- Предложены модулярно-кодированные арифметико-логические формы k-значных ФАЛ, основанные на Китайской теореме об остатках. Эти формы являются результатом синтеза, в отличие от арифметико-логических форм (??) и (??), основанных на декомпозиции исходного полинома. Преимущество данных форм — упрощение реализации оператора маскирования.

8. Восстановление позиционной формы числа

Важное место в формах арифметической логики, основанных на многомодульной арифметике, занимают методы восстановления позиционной формы числа. Китайская теорема об остатках в своем классическом варианте далеко не всегда дает оптимальное для технической реализации решение. Это стимулирует поиск новых вариантов Китайской теоремы об остатках, удовлетворяющих тем или иным требованиям технической реализации [73]. Оригинальные материалы этой главы впервые опубликованы в [3, 6, 55, 56, 66].

8.1. Теорема о восстановлении для двух модулей

Рассмотрим классическую формулу восстановления позиционной формы числа для случая двух модулей:

$$A = |\phi_1 m_2 q_1 + \phi_2 m_1 q_2|_M \,, \tag{66}$$

где q_1 и q_2 — решения сравнений $m_2 q_1 \equiv 1 \pmod{m_1}$ и $m_1 q_2 \equiv 1 \pmod{m_2}$, $M = m_1 m_2$.

Для аппаратурной реализации алгоритма восстановления позиционной формы числа в соответствии с (??) потребуется два умножителя на константы $m_2 q_1$ и $m_1 q_2$ и один сумматор, функционирующий по модулю $M = m_1 m_2$ — числовому диапазону модулярной арифметики (рис. ??).

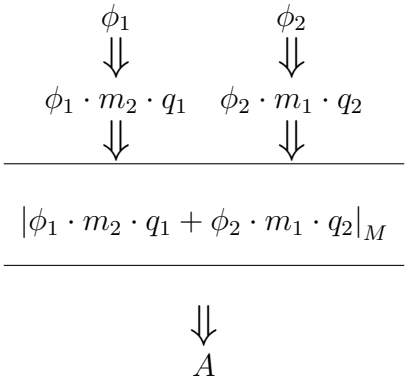

Рис. 20: Схема вычисления позиционной формы числа по двум остаткам в соответствии с (??)

Теорема 12.
Целое неотрицательное число $A < m_1 m_2$, представленное в двухмодульной арифметике вычетами ϕ_1 и ϕ_2 по системе взаимно простых модулей m_1 и m_2, таких, что $m_1 < m_2$, может быть однозначно восстановлено по формуле

$$A = m_1 \left| \delta \left| \phi_1 - \phi_2 \right|_{m_2} \right|_{m_2} + \phi_1, \tag{67}$$

где $\delta = |(m_2 - m_1)^{-1}|_{m_2}$.

Доказательство теоремы 12.

В соответствии с алгоритмом Евклида число A можно представить в виде

$$A = m_1 n_1 + \phi_1, \tag{68}$$

или

$$A = (m_2 - b)n_1 + \phi_1,$$

где $b = m_2 - m_1$.

Поскольку $A \equiv \phi_2 \pmod{m_2}$, то

$$(m_2 - b)n_1 + \phi_1 \equiv \phi_2 \pmod{m_2}.$$

Учитывая, что $m_2 n_1 \equiv 0 \pmod{m_2}$, получаем

$$bn_1 \equiv \phi_1 - \phi_2 \pmod{m_2}$$

или

$$n_1 \equiv b^{-1}(\phi_1 - \phi_2) \pmod{m_2}. \tag{69}$$

Согласно формулы Эйлера (следствие **??**):

$$b^{-1} \equiv b^{\varphi(m_2)-1} \pmod{m_2}, \tag{70}$$

где $\varphi(m_2)$ — функция Эйлера. Подставляя выражение (**??**) в (**??**), с учетом (**??**) получаем (**??**).

□

Пример 24.

Рассмотрим пару ЛТЧП КТО (**??**) и (**??**) применительно к условиям, использованным в примерах 5 и 11. При этом выберем значения модулей $m_1 = 3$ и $m_2 = 5$, что должно обеспечить числовой диапазон $M = m_1 m_2 = 15 \geq 2^3$. Тогда

$$|\mathbf{Y}|_3 = \left|\begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}\right|_3 = \left|\begin{bmatrix} 0 \\ 7 \\ 6 \\ 1 \\ 5 \\ 2 \\ 3 \\ 7 \end{bmatrix}\right|_3 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 2 \\ 2 \\ 0 \\ 1 \end{bmatrix},$$

$$|\mathbf{Y}|_5 = \left|\begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}\right|_5 = \left|\begin{bmatrix} 0 \\ 7 \\ 6 \\ 1 \\ 5 \\ 2 \\ 3 \\ 7 \end{bmatrix}\right|_5 = \begin{bmatrix} 0 \\ 2 \\ 1 \\ 1 \\ 0 \\ 2 \\ 3 \\ 2 \end{bmatrix}.$$

Преобразование (??) примет вид:

$$\begin{aligned}\mathbf{\Psi}_1 &= \mathbf{A}_{2^3}\mathbf{\Phi}_1 \pmod 3 = \mathbf{A}_{2^3}\cdot[0\ 1\ 0\ 1\ 2\ 2\ 0\ 1]^T \pmod 3 = \\ &= [0\ 1\ 0\ 0\ 2\ 2\ 1\ 1]^T, \\ \mathbf{\Psi}_2 &= \mathbf{A}_{2^3}\mathbf{\Phi}_2 \pmod 5 = \mathbf{A}_{2^3}\cdot[0\ 2\ 1\ 1\ 0\ 2\ 3\ 2]^T \pmod 5 = \\ &= [0\ 2\ 1\ 3\ 0\ 0\ 2\ 4]^T.\end{aligned}$$

Для получения Y используем (??):

$$Y^{(i)} = m_1\left|\delta_1\left|\phi_{i,1}-\phi_{i,2}\right|_{m_2}\right|_{m_2} + \phi_{i,1} = 2\left|2\left|\phi_{i,1}-\phi_{i,2}\right|_5\right|_5 + \phi_{i,1},$$

где $i = 0,\ 1,\ \ldots,\ 2^n-1$. В результате получим:

$$Y^{(0)} = 2\left|2\left|0-0\right|_5\right|_5 + 0 = 0, \quad Y^{(4)} = 2\left|2\left|2-0\right|_5\right|_5 + 2 = 5,$$

$$Y^{(1)} = 2\left|2\left|1-2\right|_5\right|_5 + 1 = 7, \quad Y^{(5)} = 2\left|2\left|2-2\right|_5\right|_5 + 2 = 2,$$

$$Y^{(2)} = 2\left|2\left|0-1\right|_5\right|_5 + 0 = 6, \quad Y^{(6)} = 2\left|2\left|0-3\right|_5\right|_5 + 0 = 3,$$

$$Y^{(3)} = 2\left|2\left|1-1\right|_5\right|_5 + 1 = 1, \quad Y^{(7)} = 2\left|2\left|1-2\right|_5\right|_5 + 1 = 7.$$

Таким образом:

$$\begin{aligned}\mathbf{\Psi} &= \mathbf{CRT}^2_{k=1}\left[\psi_{0,k},\ \psi_{1,k},\ \ldots,\ \psi_{2^3-1,k}\right]^T \pmod{m_k} = \\ &= [0\ 7\ 6\ 8\ 5\ 0\ 2\ 9]^T, \\ \mathbf{Y} &= \mathbf{CRT}^2_{k=1}\left[\phi_{0,k},\ \phi_{1,k},\ \ldots,\ \phi_{2^3-1,k}\right]^T \pmod{m_k} = \\ &= [0\ 7\ 6\ 1\ 5\ 2\ 3\ 7]^T.\end{aligned}$$

Аппаратурная реализация формулы (??) требует использования вычитателя, функционирующего по модулю m_2, для вычисления разности $|\phi_1-\phi_2|_{m_2}$, умножителя полученной разности на константу δ по модулю m_2, позиционного умножителя на модуль m_1 и позиционного сумматора для суммирования ϕ_1 [3].

8.2. Применение специальных модулей

Как правило для представления вычетов в модулярной арифметике используется двоичная система счисления. Это обстоятельство может быть использовано для упрощения как программной, так и аппаратной реализаций алгоритмов восстановления позиционной формы числа. В качестве модулей удобно использование чисел $2^i \pm 1$, $D_i = 2^i$ и др. [52, 91, 92, 104].

Поскольку модуль вида 2^i имеет в своем каноническом разложении единственный простой множитель 2, он не может быть использован в качестве числового диапазона модулярной арифметики. Пусть $m_1 = 2^i-1$ и $m_2 = 2^i+1$. Тогда $M = m_1m_2 = 2^{2i}-1$, а операцию суммирования по модулю $M = 2^{2i}-1$ — числовому диапазону модулярной арифметики — можно свести к суммированию дополнительного кода. Это дает наиболее простое решение задачи восстановления позиционной формы числа в двухмодульной арифметике в рамках классической теоремы об остатках.

Теперь рассмотрим формулу (??). Сократим δ за счет применения модулей, обладающих свойством $m_2 - m_1 = 1$. Далее, используя модули D_i и 2^i+1, получим

$$A = D_i \left|\phi_1 - \phi_2\right|_{2^i+1} + \phi_1. \tag{71}$$

Вычеты ϕ_1, ϕ_2 и результат восстановления числа A представлены в системе счисления с основанием 2:

$$\begin{aligned}
\phi_1 &= \left(a^{(1)}_{\lceil \log_2 m_1 \rceil}\, a^{(1)}_{\lceil \log_2 m_1 \rceil - 1} \ldots a^{(1)}_1\right)_2; \\
\phi_2 &= \left(a^{(2)}_{\lceil \log_2 m_2 \rceil}\, a^{(2)}_{\lceil \log_2 m_2 \rceil - 1} \ldots a^{(2)}_1\right)_2; \\
A &= \left(a^{(3)}_{\lceil \log_2 M \rceil}\, a^{(3)}_{\lceil \log_2 M \rceil - 1} \ldots a^{(3)}_1\right)_2,
\end{aligned}$$

где $a^{(i)}_j$ — двоичные цифры.

Аппаратурная реализация формулы (??) содержит один вычитатель $\sum^-$ разрядности $\lceil \log_2 m_2 \rceil$, функционирующий по модулю $2^i + 1$ (??). Поскольку умножение числа, представленного в двоичной системе счисления, на D_i реализуется путем сдвига представления числа на i разрядов в сторону старших разрядов, умножение разности $|\phi_1 - \phi_2|_{2^i+1}$, получаемой на выходе $\sum^-$, на модуль D_i и последующее суммирование результата умножения с ϕ_1 выполняется соответствующим распределением старших и младших разрядов на выходе устройства.

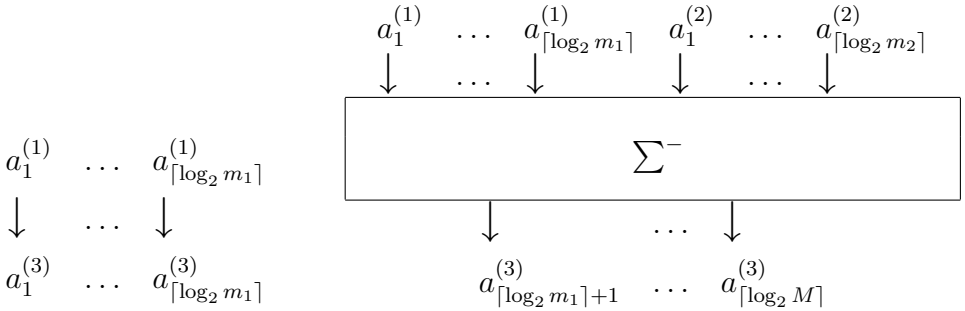

Рис. 21: Схема вычисления позиционной формы числа по двум остаткам в соответствии с теоремой 12

Если вычитатель $\sum^-$ имеет табличную реализацию, то формулу (??) можно усилить не усложняя устройства:

$$A = D_i \left|\delta \left|\phi_1 - \phi_2\right|_m\right|_m + \phi_1, \tag{72}$$

где

$$\delta = \left|(m - D_i)^{-1}\right|_m;$$

m — нечетный модуль, взаимно простой с D_i и больший D_i.

Таблица вычитателя $\sum^-$ строится в соответствии с выражением

$$\left|\delta \left|\phi_1 - \phi_2\right|_m\right|_m.$$

Пример 25.
Пусть арифметика задана модулями D_4 и $m = 21$. Тогда

$$\delta = \left|(21-16)^{-1}\right|_{21} = \left|5^5\right|_{21} = 17.$$

Даны вычеты $\phi_1 = 13$ и $\phi_2 = 18$, тогда в соответствии с (**??**) получим

$$A = 16\,|17\,|13-18|_{21}|_{21} + 13 = 333.$$

Действительно, $|333|_{16} = 13$, $|333|_{21} = 18$.

8.3. Обобщения для произвольного количества модулей

Теорема 13.
Целое неотрицательное число $Y < m = \prod_{k=1}^{v} m_k$, представленное вычетами $\phi_1, \phi_2, \ldots, \phi_v$ по системе попарно простых модулей $m_1 < m_2 < \ldots < m_v$ может быть однозначно восстановлено посредством рекурсии

$$\left\{\begin{array}{l} h_1 = \phi_2, \\ h_2 = m_1 \left|\delta_1\,|\phi_1 - h_1|_{M_1}\right|_{M_1} + \phi_1, \\ h_3 = m_3 \left|\delta_2\,|\phi_3 - h_2|_{M_2}\right|_{M_2} + \phi_3, \\ h_4 = m_4 \left|\delta_3\,|\phi_4 - h_3|_{M_3}\right|_{M_3} + \phi_4, \\ \vdots \\ h_v = Y = m_v \left|\delta_{v-1}\,|\phi_v - h_{v-1}|_{M_{v-1}}\right|_{M_{v-1}} + \phi_v, \end{array}\right. \tag{73}$$

где

$$\begin{aligned} M_1 &= m_2; \\ M_i &= \prod_{j=1}^{i} m_j; \\ \delta_1 &= \left|(M_1 - m_1)^{-1}\right|_{M_1}; \\ \delta_i &= \left|(M_i - m_{i+1})^{-1}\right|_{M_i}. \end{aligned}$$

*Доказательство (**??**)* основано на обобщении теоремы 13 путем поэтапного перехода к составным модулям и укрупнения числового диапазона.

Для получения Y с помощью (**??**) потребуется v шагов.

Преимуществом (**??**) является ограничение промежуточных результатов вычислений верхней границей M_v.

При обеспечении возможности распараллеливания вычислений возможен вариант построения алгоритма восстановления позиционной формы числа, основанный на теореме 13, требующий $\lceil \log_2 v \rceil$ шагов преобразования [66]. При этом используется принцип рекурсивного сдваивания.

Например, число A представлено вычетами ϕ_1, ϕ_2, ϕ_3, ϕ_4 по попарно взаимно простым и упорядочинным модулям $m_1 < m_2 < m_3 < m_4$.

Тогда алгоритм восстановления числа A состоит из двух шагов (рис. **??**, здесь $\phi_{11} = |A|_{m_1 m_2}$ и $\phi_{12} = |A|_{m_3 m_4}$).

Алгоритм 6.

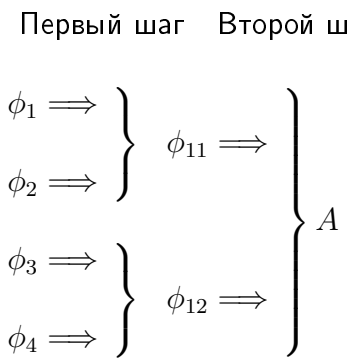

Рис. 22: Схема вычисления позиционной формы числа по четырем остаткам с упорядочинными номерами модулей

Шаг 1. В соответствии с (**??**) имеем

$$\begin{aligned} \phi_{11} &= m_1 \left|\delta_{11} \left|\phi_1 - \phi_2\right|_{m_2}\right|_{m_2} + \phi_1; \\ \phi_{12} &= m_3 \left|\delta_{12} \left|\phi_3 - \phi_4\right|_{m_4}\right|_{m_4} + \phi_3, \end{aligned}$$

где

$$\begin{aligned} \delta_{11} &= \left|(m_2 - m_1)^{-1}\right|_{m_2}; \\ \delta_{12} &= \left|(m_4 - m_3)^{-1}\right|_{m_4}. \end{aligned}$$

Шаг 2.

$$A = m_1 m_2 \left|\delta \left|\phi_{11} - \phi_{12}\right|_{m_3 m_4}\right|_{m_3 m_4} + \phi_{11},$$

где

$$\delta = \left|(m_3 m_4 - m_1 m_2)^{-1}\right|_{m_3 m_4}.$$

Номера модулей в рассмотренной схеме и алгоритме восстановления позиционной формы числа могут быть изменены. Для понимания сказанного рассмотрим частный случай, демонстрируемый с помощью рис. **??**. При этом полагаем, что выполняется условия: $m_1 m_4 > m_2 m_3$.

Такой способ вычисления полезен для выравнивания величины и получения необходимых свойств промежуточных укрупненных модулей, образованных произведениями модулей предыдущих ступеней. При этом алгоритм восстановления будет выглядеть следующим образом.

Алгоритм 7.

Шаг 1. В соответствии с (**??**) имеем

$$\begin{aligned} \phi_{11} &= m_1 \left|\delta_{11} \left|\phi_1 - \phi_4\right|_{m_4}\right|_{m_4} + \phi_1; \\ \phi_{12} &= m_2 \left|\delta_{12} \left|\phi_2 - \phi_3\right|_{m_3}\right|_{m_3} + \phi_2, \end{aligned}$$

где

$$\begin{aligned} \delta_{11} &= \left|(m_4 - m_1)^{-1}\right|_{m_4}; \\ \delta_{12} &= \left|(m_3 - m_2)^{-1}\right|_{m_3}. \end{aligned}$$

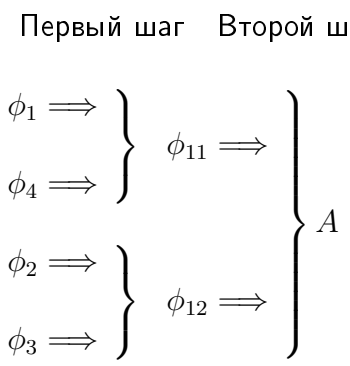

Рис. 23: Схема вычисления позиционной формы числа по четырем остаткам с переупорядочиванием номеров модулей

Шаг 2.

$$A = m_1 m_4 \left|\delta \left|\phi_{11} - \phi_{12}\right|_{m_2 m_3}\right|_{m_2 m_3} + \phi_{11},$$

где

$$\delta = \left|(m_2 m_3 - m_1 m_4)^{-1}\right|_{m_2 m_3}.$$

Вычеты ϕ_{11} и ϕ_{12} могут быть получены и любым другим методом, например в соответствии с Китайской теоремой об остатках или прямой дешифрацией.

Формула (**??**) впервые была рассмотрена в [3]. Ее обобщение (**??**) для произвольного количества модулей было получено в [6]. Метод рекурсивного сдваивания и метод применения специальных модулей опубликован в [66]. Случаи применения модулей, обладающих свойствами $m_2 - m_1 = 1$ и $m_2 > m_1 = 2^i$ рассмотрены соответственно в [55] и [56].

Похожие на приведенные выше результаты были позднее опубликованы в [50]. Так, для восстановления позиционной формы числа по двум модулям предлагается формула [50]

$$A = m_1 \zeta \left|\phi_2 - \phi_1\right|_{m_2} + \phi_1, \tag{74}$$

где $\zeta = \left|m_1^{-1}\right|_{m_2}$.

Обобщение (**??**) для произвольного количества модулей дает, согласно [50], рекурсию:

$$\begin{cases} h_1 = \phi_1, \\ h_2 = M_2 \zeta_2 \left|\phi_2 - h_1\right|_{m_2} + \phi_1, \\ h_3 = M_3 \zeta_3 \left|\phi_3 - h_2\right|_{m_3} + \phi_2, \\ h_4 = M_4 \zeta_4 \left|\phi_4 - h_3\right|_{m_4} + \phi_3, \\ \vdots \\ h_v = Y = M_v \zeta_v \left|\phi_v - h_{v-1}\right|_{m_v} + \phi_{v-1}, \end{cases} \tag{75}$$

где $M_i = \prod_{j=1}^{i} m_j$; $\zeta_i = \left|M_i^{-1}\right|_{m_i}$.

Использование схемы в соответствии с рис. **??** и модификация алгоритма 7 с учетом (**??**) даст следующий результат.

Алгоритм 8.

Шаг 1. В соответствии с (**??**) имеем

$$\begin{aligned}\phi_{11} &= m_1\zeta_{11}\left|\phi_2-\phi_1\right|_{m_2}+\phi_2;\\ \phi_{12} &= m_3\zeta_{12}\left|\phi_4-\phi_3\right|_{m_4}+\phi_4,\end{aligned}$$

где

$$\begin{aligned}\zeta_{11} &= \left|m_1^{-1}\right|_{m_2}\\ \zeta_{12} &= \left|m_3^{-1}\right|_{m_4}.\end{aligned}$$

Шаг 2.

$$A = m_1m_2\zeta\left|\phi_{12}-\phi_{11}\right|_{m_3m_4}+\phi_{12},$$

где

$$\zeta = \left|(m_1m_2)^{-1}\right|_{m_3m_4}.$$

Для схемы, представленной на рис. **??** алгоритм 8 примет вид.

Алгоритм 9.

Шаг 1. В соответствии с (**??**) имеем

$$\begin{aligned}\phi_{11} &= m_1\zeta_{11}\left|\phi_4-\phi_1\right|_{m_4}+\phi_4;\\ \phi_{12} &= m_2\zeta_{12}\left|\phi_3-\phi_2\right|_{m_3}+\phi_3,\end{aligned}$$

где

$$\begin{aligned}\zeta_{11} &= \left|m_1^{-1}\right|_{m_4}\\ \zeta_{12} &= \left|m_2^{-1}\right|_{m_3}.\end{aligned}$$

Шаг 2.

$$A = m_1m_4\zeta\left|\phi_{12}-\phi_{11}\right|_{m_2m_3}+\phi_{12},$$

где

$$\zeta = \left|(m_1m_4)^{-1}\right|_{m_2m_3}.$$

Рассмотренные алгоритмы 7 и 9 могут быть обобщены для произвольного количества модулей и ступеней преобразования. При этом сортировка исходных и промежуточных модулей может выполняться не только на первой, но и на любой последующей ступени преобразования.

8.4. Использование полиадической системы счисления

Представление числа A с помощью полиадической системы счисления имеет вид:

$$A = z_v q_{v-1} \dots q_2 q_1 + \dots + z_2 q_1 + z_1, \tag{76}$$

где $0 \le z_i < q_i$ $(i = 1, 2, \dots, v)$; $q_1, q_2, \dots, q_v$ — основания системы счисления (поэтому ее еще называют системой счисления со смешанными основаниями).

Если в (??) выбрать такие основания, что $m_i = q_i$, где $i = 1, 2, \dots, v$, то представление

$$A = z_v m_{v-1} \dots m_2 m_1 + \dots + z_2 m_1 + z_1, \tag{77}$$

где $0 \le z_i < m_i$ $(i = 1, 2, \dots, v)$, может быть использовано для восстановления позиционной формы числа [12]. При этом используется схема вычислений, представленная на рис ??.

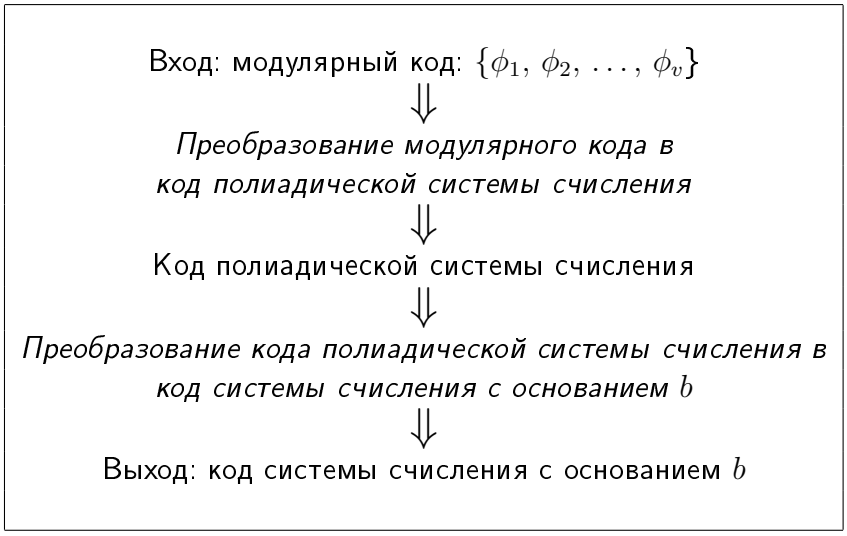

Рис. 24: Принцип использования полиадической системы счисления для восстановления позиционной формы числа в заданной системе счисления

Вычислить разрядные цифры $z_1, z_2, \dots, z_v$ полиадической системы счисления можно, например, с помощью рекурсии [50]:

$$\begin{cases} z_1 = \phi_1, \\ z_2 = \zeta_2 \left|\phi_2 - z_1\right|_{m_2}, \\ z_3 = \zeta_3 \left|\phi_3 - (M_2 z_2 + z_1)\right|_{m_3}, \\ \vdots \\ z_v = \zeta_v \left|\phi_v - (M_{v-1} z_{v-1} + \dots + M_2 z_2 + z_1)\right|_{m_v}, \end{cases}$$

где числа ζ_i и M_i $(i = 1, 2, \dots, v)$ взяты из формулы ??.

Так как представления весов полиадической системы счисления с помощью b-ичной системы счисления — известные константы (например $b = 2$):

$$\begin{aligned} m_1 &= (a_{g-1}^{(1)} \dots a_1^{(1)} a_0^{(1)})_2, \\ m_1 m_2 &= (a_{g-1}^{(2)} \dots a_1^{(2)} a_0^{(2)})_2, \\ &\vdots \\ m_1 m_2 \dots m_{v-1} &= (a_{g-1}^{(v-1)} \dots a_1^{(v-1)} a_0^{(v-1)})_2, \end{aligned}$$

то вычисление (??) с представлением цифр z_i $(i = 1, 2, \ldots, v)$ и весов m_1, m_1m_2, $\ldots$, $m_1m_2 \ldots m_{v-1}$ в итоговой системе счисления с основанием b даст результат в системе счисления с основанием b. Здесь следует указать, что разрядные цифры итогового представления в системе счисления с основанием b, могут формироваться синхронно (с запаздыванием на одну цифру) с формированием разрядных цифр полиадической системы счисления.

Помимо сказанного, полиадическая система счисления весьма удобна для выполнения немодульных операций в собственном (полиадическом) представлении. Однако эти возможности выходят за рамки обсуждаемой темы.

8.5. Выводы

- Предложены варианты Китайской теоремы об остатках для случаев применения двух и произвольного количества модулей, позволяющие по сравнению с классической формулой существенно ограничить величину результатов промежуточных вычислений.

- Получено два варианта построения алгоритма восстановления позиционной формы числа для произвольного количества модулей, позволяющие получить соответственно v (количество модулей) и $\lceil \log_2 v \rceil$ шагов преобразования.

- Предложен метод упрощения реализации формулы для восстановления позиционной формы числа, основанный на применении специальных модулей.

- Равномерность величины промежуточных модулей алгоритма восстановления позиционной формы числа может быть обеспечена переупорядочиванием номеров исходных и промежуточных модулей.

9. Заключение

Основы вычислительных методов алгебры логики, используемые в настоящее время, были созданы в «докомпьютерную» эпоху и *плохо согласуются* с методами организации вычислений в современной компьютерной технике. Напротив, арифметическая логика полностью *соответствует* принципам построения современных и перспективных ЭВМ и позволяет раскрыть неиспользуемый в настоящее время потенциал вычислительной техники по реализации высокопроизводительных, гибких, параллельных логических вычислений.

Модулярные арифметико-логические формы (общая классификация представлена на рис. ??) обладают рядом новых полезных свойств и ориентированы на *воплощение в современную и перспективную практику цифровой обработки информации идей арифметической логики на основе высокоразвитого и прогрессивного научно-методического аппарата модулярной арифметики.*

Достоинствами модулярных арифметико-логических форм являются:

- высокая степень *параллелизма* логических вычислений, которая может быть классифицирована как сверхпараллелизм;

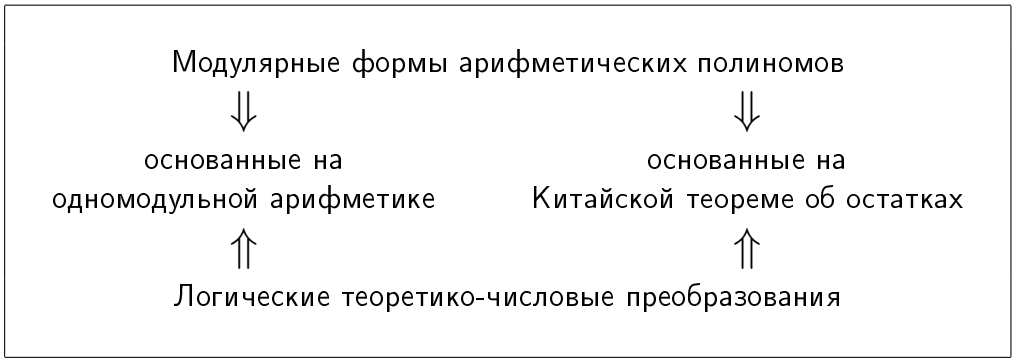

Рис. 25: Классификация модулярных форм

- уникальные возможности по обеспечению *отказоустойчивости* и *живучести* средств логических вычислений;
- обеспечение *контроля* и *коррекции* ошибок на всех стадиях обработки, хранения, а также передачи информации;
- создание благоприятных условий для приоритетного использования быстродействующих *табличных* операционных устройств (в том числе на базе программируемой логики) за счет существенного уменьшения (по сравнению с двоичной системой счисления — на порядки [9]) объема таблиц;
- уменьшение *сложности* представления логических функций;
- возможность *многоцелевого* использования средств логических вычислений, которая, в свою очередь, может быть использована для обеспечения отказоустойчивости и живучести вычислительной системы [16, 21, 25, 54, 64] или для сокращения аппаратурных затрат за счет разделения решения задач во времени;
- повышение *защищенности* средств обработки от методов инженерной разведки за счет существенного усложнения побочных электромагнитных излучений и наводок.

В настоящее время модулярная арифметика широко применяется в методах и средствах цифровой обработки сигнальной информации. Модулярные арифметико-логические формы позволяют *задействовать* высокоразвитый математический аппарат и совершенные технические средства цифровой обработки сигналов, базирующиеся на методах модулярной арифметики, для высококачественной реализации параллельных логических вычислений.

Таким образом, модулярные арифметико-логические формы, по-видимости, позволяют преодолеть главное *противоречие* двух основных способов реализации логических алгоритмов: *программного* (гибкого) и *аппаратного* (жесткого). *Логические вычисления, обладающие достоинствами программной реализации, становится возможным реализовать специализированнми вычислительными средствами*, характеризующимися требуемым комплексом технических характеристик.

Наглядный пример сочетания ряда противоречивых требований — специализированные вычислители, устанавливаемые на малых космических аппаратах с длительными сроками активного существования (7–10 лет и более) [62]. Проблемы, возникающие при проектировании бортовых средств цифровой обработки заключаются в обеспечении заданной производительности, определяемой высокой скоростью и большими объемами передаваемой информации (например, оптической и радиоголографической), а также надежности функционирования в условиях размещения приборов на стенках негерметизированных сотовых панелей в открытом космосе с общей защитой до 0, 3 г/см^2, при жестких ограничениях на энергопотребление и массогабаритные показатели.

Методы арифметической логики в этих условиях позволяют существенно упростить поиск оптимального решения. Методы модулярной арифметики были применены для повышения надежности бортовой вычислительной машины аэрокосмической ракеты STAR [34].

Известной трудностью современной практики программирования криптографических алгоритмов, реализуемых на основе сигнальных процессоров, является максимизация использования возможностей процессора для реализации специфических криптографических функций. Труд программиста, в этом случае, требует высокой (дорогостоящей) квалификации и приближается в большей степени к *искусству*, нежели к *технологии*. Использование методов арифметической логики позволяет свести разнообразие используемых логических функций к реализации однотипных арифметических полиномов, что делает процесс программирования более *технологичным* (повышает качество, сокращает сроки и стоимость программирования).

Структура технического средства (даже при использовании средств программируемой логики), основанного на методах арифметической логики, *не отражает* реализуемый алгоритм, что может быть использовано в качестве дополнительного средства повышения стойкости шифрования.

Известно, что *проговая функция* может быть реализована линейным арифметическим полиномом и условным оператором [17]. В [84] этот результат развит на область реализации логических функций линейными арифметическими полиномами с *маскированиями*. Таким образом и *система* пороговых функций может быть задана *одним* линейным арифметическим полиномом с маскированиями. Этот фундаментальный результат, в частности, может быть использован для моделирования нейронных сетей методами арифметической логики, где каждому слою нейронной сети поставлен в соответствие линейный арифметический полином.

Модулярные арифметико-логические формы могут составить модулярную арифметико-логическую модель нейронной сети. Таким образом, преимущества модулярных арифметико-логических форм могут быть распространены и на область нейровычислений.

Методы модулярной арифметики продолжают развиваться и находят все более широкое применение, прежде всего, в алгоритмах и средствах цифровой обработки сигнальной информации [33, 35, 38, 41, 52, 57, 81, 98], а также в задачах ассиметричной криптографии [65, 67, 69, 77, 93, 95–97].

Исследуются вопросы построения высокопроизводительных модулярных

структур на перспективной оптической и электрооптической элементной базе [7, 53, 68, 94].

Развиваются методы и средства ввода и вывода информации, повышающие эффективность специализированных вычислителей за счет уменьшения этапов промежуточных преобразований. Так, в [2, 4, 5] предложен ряд технических решений, обеспечивающих аналого-цифровое преобразование непосредственно в код модулярной арифметики. В [68, 71] предложены аналого-цифровые преобразователи в код модулярной арифметики, основанные на электрооптической и сверхпроводниковой (впервые) элементной базе.

Отмечается интерес к *нейроподобным* модулярным вычислительным структурам [20, 82, 83, 100, 101, 105–108], которые помимо рассмотренных в книге средств могут быть использованы в целях аппаратной поддержки методов арифметической логики (в широком смысле), в том числе, и для аппаратной поддержки нейровычислений (в узком смысле), основанных на реализации модулярных арифметико-логических форм. Здесь имеются принципиальные отличия от указанного выше способа построения модулярных арифметических моделей нейронных сетей, которые *инвариантны* по отношению к способу их технической реализации. Напротив, *нейроподобные* (приближающиеся к структуре нейронных сетей) модулярные структуры — вполне определенный способ *технической реализации* модулярных вычислений, имеющий преимущества по показателю отказоустойчивости, которые достигаются органическим сочетанием свойств неронных сетей и свойств модулярной арифметики.

Поэтому, хотя модулярная арифметика является в известном смысле проблемно-ориентированным математическим средством, расширение решаемых на ее основе задач, принадлежащих, как ранее считалось, к диаметрально противоположным классам, позволяет сделать вывод о *становлении единой математической и технической базы, основанной на методах модулярной арифметики*, предназначенной для решения широкого спектра задач интенсивной, достоверной, гибкой обработки информации в различных сферах практического применения.

Список литературы

1. Авсаркисян Г. С., Брайловский Г. С. Представление логических функций в виде полиномов Жегалкина // Автоматика и вычисл. техника. 1975. № 6.
2. А. с. 1259487 СССР, МКИ 4 Н 03 М 1/28. Преобразователь перемещения в код системы остаточных классов / С. Н. Хлевной, О. А. Финько // Открытия. Изобрет. 1986. № 35.
3. А. с. 1343553 СССР, МКИ 4 Н 03 М 7/18. Преобразователь кода системы остаточных классов в позиционный код / Н. И. Червяков, О. Е. Коршунов, О. А. Финько // Открытия. Изобрет. 1987. № 37.
4. А. с. 1368989 СССР, МКИ 4 Н 03 М 1/28. Аналого-цифровой преобразователь в код системы остаточных классов /О. А. Финько и др. // Открытия. Изобрет. 1988. № 3.
5. А. с. 1372620 СССР, МКИ 4 Н 03 М 1/28. Аналого-цифровой преобразователь в системе остаточных классов / О. А. Финько и др. // Открытия. Изобрет. 1988. № 5.
6. А. с. 1388996 СССР, МКИ 4 Н 03 М 7/18. Преобразователь кода из системы остаточных классов в позиционный код / Н. И. Червяков, О. Е. Коршунов, О. А. Финько // Открытия. Изобрет. 1988. № 14.
7. Акаев А. А., Майоров С. А. Когерентные оптические вычислительные машины. Л.: Машиностроение, 1977. 440 с.
8. Акритас А. Основы компьютерной алгебры с приложениями: Пер. с англ. М.: Мир, 1994. 544 с.

9. Акушский И. Я., Юдицкий Д. И. Машинная арифметика в остаточных классах. М.: Сов. радио, 1968. 440 с.
10. Акушский И. Я., Амербаев В. М., Пак И. Т. Основы машинной арифметики комплексных чисел. Алма-Ата: Наука, 1970. 248 с.
11. Алферов А. П., Зубов А. П., Кузьмин А. С., Черемушкин А. В. Основы криптографии. М.: Гелиос АРВ, 2001. 480 с.
12. Амербаев В. М. Теоретические основы машинной арифметики. Алма-Ата: Наука, 1976. 324 с.
13. Артюхов В. Л., Кондратьев В. Н., Шалыто А. А. Реализация булевых функций арифметическими полиномами // Автоматика и телемеханика. 1988. № 4. С. 138–147.
14. Бабаш А. В., Шанкин Г. П. Криптография. М.: СОЛОН-Р, 2002. 512 с.
15. Байоми М. А. Заказные матрицы СБИС для структур, основанных на системе счисления в остаточных классах // ТИИЭР. 1987. Т. 38. № 12. С. 134–139.
16. Березюк И. Т. Живучесть микропроцессорных систем. Киев: Техника, 1989. 143 с.
17. Бутаков Е. А. Методы синтеза релейных устройств из пороговых элементов. М.: Энергия,1970.
18. Бухштаб А. А. Теория чисел. М.: Просвещение, 1966. 384 с.
19. Воробьев Н. Н. Признаки делимости. 4-е изд., испр. М.: Наука. Гл. ред. физ.-мат. лит., 1988. 96 с.
20. Галушкин А. И. Нейрокомпьютеры в разработках военной техники США (обзор по материалам открытой печати). Часть I // Зарубежная радиоэлектроника. 1995. № 5. С. 3–48.
21. Гуляев А. В. Организация живучих вычислительных систем // Управляющие системы и машины. 1987. № 5. С. 26–29.
22. Даджион Д., Мерсеро Р. Цифровая обработка многомерных сигналов: Пер. с англ. М.: Мир, 1988. 488 с.
23. Дзегеленок И. И., Оцоков Ш. А. О распараллеливании безошибочных вычислений на ПМК-сети «Курс-2000» // Вычислительные сети. 2003. № 1.
24. Дзюжаньски П., Малюгин В., Шмерко В., Янушкевич С. Линейные модели схем на многозначных элементах // Автоматика и телемеханика. 2002. № 6. С. 99–119.
25. Додонов А. И. и др. Введение в теорию живучести вычислительных систем. Киев: Наукова думка, 1990. 184 с.
26. Долгов А. И. Диагностика устройств, функционирующих в системе остаточных классов. М.: Радио и связь, 1982. 64 с.
27. Евстигнеев В. Г. Недвоичная машинная арифметика и специализированные процессоры. М.: МИФИ СЕРВИС, 1992. 266 с.
28. Жегалкин И. И. О технике вычисления предложений в символической логике // Математический сборник Московского математического общества. 1927. Т. 34. С. 9–28.
29. Жегалкин И. И. Арифметизация символической логики // Математический сборник Московского математического общества. 1928. Т. 35. С. 311–374, 1929. Т. 36. С. 205–338.
30. Инютин С. А. Параллельные вычисления в сверхбольших компьютерных диапазонах // I Междунар. конф. «Параллельные вычисления и задачи управления» (PACO-2001). Москва, 2931 января 2001. Сборник трудов. М.: Ин-т проблем управления им. В. А. Трапезникова РАН, 2001. С. 76–87.
31. Инютин С. А. Теория и методы моделирования вычислительных структур с параллелизмом машинных операций: Автореф. дис. ... д-ра техн. наук. М.: МГИЭТ (ТУ), 2001. 33 с.
32. Кнут Д. Э. Искусство программирования, том 2. Получисленные алгоритмы, 3-е из. М.: Издательский дом «Вильямс», 2000.
33. Коляда А. А., Пак И. Т. Модулярные структуры конвейерной обработки цифровой информации. Мн.: Университетское, 1992. 256 с.
34. Краснобаев В. А. и др. Методы повышения надежности специализированных ЭВМ систем и средств связи. Харьков: ХВВКМУ РВ, 1990. 172 с.
35. Кравченко В. Ф., Крот А. М. Методы и микроэлектронные средства цифровой фильтрации сигналов и изображений на основе теоретико-числовых преобразований // Зарубежная радиоэлектроника. Успехи современной радиоэлектроники. 1997. № 6. С. 3–31.
36. Кухарев Г. А., Шмерко В. П., Зайцева Е. Н. Алгоритмы и систолические процессоры для обработки многозначных данных. Минск: Наука и техника, 1990.
37. Лебедев Е. К. Цифровая фильтрация в системе остаточных классов // Изв. вузов. Радиотехника. 1985. Т. 28, № 8. С. 58–62.
38. Лебедев Е. К. Синтез нелинейных непозиционных устройств обработки марковских сигналов // Изв. вузов. Радиотехника. 1987. Т. 30, № 12. С. 69–72.

39. Лебедев Е. К. Быстрые алгоритмы цифровой обработки сигналов: Монография. Красноярск, 1989.
40. Лебедев Е. К. Микропроцессорные корреляторы: Монография. Йошкар-Ола, 1996.
41. Макклеллан Дж., Рейдер Ч. М. Применение теории чисел в цифровой обработке сигналов: Пер. с англ. / Под ред. Ю. И. Манина. М.: Радио и связь, 1983.
42. Малюгин В. Д. Надежность переключательных схем // Автоматика и телемеханика. 1964. № 3. С. 1375–1383.
43. Малюгин В. Д. О полиномиальной реализации кортежа булевых функций // ДАН СССР. 1982. Т. 265, № 6. С. 1338–1341.
44. Малюгин В. Д. Реализация булевых функций арифметическими полиномами // Автоматика и телемеханика. 1982. № 4. С. 84–93.
45. Малюгин В. Д. Реализация кортежей булевых функций посредством линейных арифметических полиномов // Автоматика и телемеханика. 1984. № 2. С. 114–121.
46. Малюгин В. Д., Кухарев Г. А., Шмерко В. П. Преобразования полиномиальных форм булевых функций: Препринт. М.: Ин-т проблем управления им. В. А. Трапезникова РАН, 1986.
47. Малюгин В. Д. Параллельные логические вычисления посредством арифметических полиномов. М.: Наука. Физматлит, 1997.
48. Молдовян А. А., Молдовян Н. А. Метод скоростного преобразования для защиты информации в АСУ // Автоматика и телемеханика. 2000. № 4. С. 151–165.
49. Молдовян А. А., Молдовян Н. А., Гуц Н. Д., Изотов Б. В. Криптография: скоростные шифры. СПб.: БВХ-Петербург, 2002. 496 с.
50. Ноден П., Китте К. Алгебраическая алгоритмика: Пер. с франц. М.: Мир, 1999. 720 с.
51. Носов В. А. Специальные главы дискретной математики. Учебное пособие. М.: 1990. 156 с.
52. Нуссбаумер Г. Быстрое преобразование Фурье и алгоритмы вычисления сверток: Пер. с англ. М.: Радио и связь, 1985. 248 с.
53. Оптическая передача и обработка информации: Пер. с фр./ А. Козанне, Ж. Флере, Г. Руссо. М.: Мир, 1984. 504 с.
54. Отказоустойчивые цифровые системы // ТИИЭР. 1978. Т. 66, № 10. 239 с.
55. Пол. решение ВНИИГПЭ РФ о выдаче патента на изобрет. по заявке № 97104283/09, МКИ 6 H 03 M 7/18. Преобразователь кода системы остаточных классов, заданной модулями $p_2 - p_1 = 1$, в позиционный код / О. А. Финько, С. Б. Елесин, В. В. Корниенко // Открытия. Изобрет. 1998. № 7.
56. Пол. решение ВНИИГПЭ РФ о выдаче патента на изобрет. по заявке № 97104346/09, МКИ 6 H 03 M 7/18. Преобразователь кода системы остаточных классов, заданной модулями $2^i < p$, в позиционный код / О. А. Финько, С. Б. Елесин, В. В. Корниенко // Открытия. Изобрет. 1998. № 5.
57. Рабинер Л., Гоулд Б. Теория и применение цифровой обработки сигналов. М.: Мир, 1978. 848 с.
58. Садыхов Р. Х., Татур М. М. Технический сервис однородных вычислительных устройств. Мн.: Университетское, 2001. 279 с.
59. Сачков В. Н., Солодовников В. И., Федюкин М. В. Дискретные функции, используемые в криптографии. М., 1998.
60. Свобода А. Развитие вычислительной техники в Чехословакии. Система счисления остаточных классов // Кибернетический сборник. М.: Мир, 1964. Вып. 8. С. 115-148. / Svoboda A. Computer progress in Czechoslovakia. II. The numerical system of residual classes (SRE) in Digital Informations Wandles, 1962.
61. Синьков М. В., Губарени Н. М. Непозиционные представления многомерных числовых систем. Киев: Наукова думка, 1977. 149 с.
62. Сиренко В. Г., Гришин В. Ю., Еремеев П. М., Зубов Н.,Н. Проблемы и стратегия проектирования высокопроизводительных бортовых комплексов управления малыми космическими аппаратами с длительными сроками активного существования // Третья научно-техническая конф. «Перспективы использования новых технологий и научно-технических решений в изделиях ракетно-космической техники разработки ГКНПЦ им. М. В. Хруничева». Москва, 1618 декабря 2003. Пленарные доклады. М.: Ин-т проблем управления им. В. А. Трапезникова РАН, 2003. С. 51–72.
63. Торгашев В. А. Система остаточных классов и надежность ЦВМ. М.: Сов. радио, 1973. 120 с.
64. Турута Е. П. Обеспечение отказоустойчивости управления микропроцессорных систем путем перераспределения задач отказавших модулей/ Системы управления информационных сетей. М.: Наука, 1983. С. 187–189.

65. Финько О. А. и др. Непозиционное представление данных в криптосистемах с открытым ключом // V научно-техническая конференция Ракетных Войск. Краснодар, 1719 сентября 1997. Тезисы докладов. Часть I. Краснодар: КВВКИУ РВ, 1997. С. 13.
66. Финько О. А. Восстановление числа в системе остаточных классов с минимальным количеством оснований // Электронное моделирование. 1998. Т. 20, № 3. С. 56–61.
67. Финько О. А., Елесин С. Б. Принципы построения средств аппаратной поддержки криптосистем с открытым ключом // 4-й сессия Междунар. научно-технической школы-семинара «Передача, обработка и отображение информации». Сочи-Теберда, март-апрель 1997, Материалы. Ставрополь: Ставропольский воен. авиац. ин-т, 1998.
68. Финько О. А. Синтез параллельных электрооптических аналого-цифровых преобразователей для вычислителей, функционирующих в модулярной арифметике // Изв. ВУЗов. Приборостроение. 1999. Т. 42, № 3–4. С. 30–32.
69. Финько О. А. и др. Методика защиты модулярных криптопроцессоров от аппаратных ошибок // Межвузовский сборник научных трудов. Краснодар: Краснодарский военный ин-т, 2001. С. 171–175.
70. Финько О. А. Сверхпараллельные логические вычисления методами модулярной арифметики // Междунар. конф. «Искусственные интеллектуальные системы» (IEEE AIS'02) и «Интеллектуальные САПР» (CAD-2002). Геленджик, 5–10 сентября 2002. Сборник трудов. М.: Наука. Физматлит, 2002. С. 448–455.
71. Финько О. А. Сверхпроводниковый аналого-цифровой преобразователь для устройств цифровой обработки сигналов, функционирующих в остаточных классах // 5-я Междунар. конф. «Цифровая обработка сигналов и ее применение» (DSPA-2003). Москва, 12–14 марта 2003. Сборник трудов. М.: Радиотехника, 2003. С. 575–579.
72. Финько О. А. Логические вычисления на основе теоретико-числовых преобразований // Вторая Междунар. конф. по проблемам управ. (МКПУ II). Москва, 16–20 июня 2003. Сборник трудов. М.: Ин-т проблем управ. им. В. А. Трапезникова РАН, 2003. Т. 2. С. 159–166.
73. Финько О. А. Варианты Китайской теоремы об остатках, ориентированные на техническую реализацию // Междунар. конгресс «Математика в XXI в. Роль механико-математического факультета Новосибирского гос. ун-та в науке, образовании и бизнесе». Новосибирск. Академгородок, 2528 июня 2003. Тез. докл. http://www.sbras.ru/ws/MMF-21/.
74. Финько О. А. Модулярные формы арифметических полиномов для реализации систем булевых функций // Междунар. конф. «Искусственные интеллектуальные системы» (IEEE AIS'03) и «Интеллектуальные САПР» (CAD-2003). Геленджик, 310 сентября 2003. Сборник трудов. М.: Наука. Физматлит, 2002. С. 548–560.
75. Финько О. А. Синтез арифметических форм булевых функций посредством теоретико-числовых преобразований // Перспективные информационные технологии и интеллектуальные системы. Таганрог: Таганрогский гос. радиотехнический ун-т, 2003. № 15. С. 45–53.
76. Финько О. А. Вариант классификации арифметических форм представления логических функций // XIV Междунар. школа-семинар «Синтез и сложность управляющих систем». Н. Новгород, 27 октября 1 ноября 2003. Сборник трудов / Под ред. академика РАН О. Б. Лупанова. Н. Новгород: Изд-во Нижегородского педагогического ун-та, 2003. С. 83–84.
77. Финько О. А. Групповой контроль ассиметричных криптосистем методами модулярной арифметики // XIV Междунар. школа-семинар «Синтез и сложность управляющих систем». Н. Новгород, 27 октября 1 ноября 2003. Сборник трудов/ Под ред. академика РАН О. Б. Лупанова. Н. Новгород: Изд-во Нижегородского педагогического ун-та, 2003. С. 85–86.
78. Финько О. А. Модулярная арифметика параллельных логических вычислений: Монография М.: ИПУ РАН; Краснодар: КВИ, 2003. 224 с.
79. Финько О. А. Применение цифровой обработки сигналов для реализации интенсивных логических вычислений // 6-я Междунар. конф. «Цифровая обработка сигналов и ее применение» (DSPA-2004). Москва, 31 марта 2 апреля 2004. Сборник трудов. М.: Радиотехника, 2004. Т. 1. С. 265–268.
80. Финько О. А. Реализация систем булевых функций большой размерности методами модулярной арифметики // Автоматика и телемеханика. 2004. № 6. С. 37–60.
81. Червяков Н. И., Велигороша А. и др. Цифровые фильтры в системе остаточных классов // Теоретическая радиотехника. 1995. Вып. 38, № 8. С. 11–20.

82. Червяков Н. И., Сахнюк П. А. и др. Модулярные параллельные вычислительные структуры нейропроцессорных систем. М.: Физматлит, 2003. 288 с.
83. Червяков Н. И., Сахнюк П. А. и др. Нейрокомпьютеры в остаточных классах. Кн. 11. М.: Радиотехника, 2003. 272 с.
84. Шалыто А. А. Логическое управление. Методы аппаратной и программной реализации алгоритмов. СПб.: Наука, 2000. 780 с.
85. Шеннон К. Теория связи в секретных системах // В кн.: Работы по теории информации и кибернетике. М.: ИЛ, 1963. 480 с.
86. Шмерко В. П. Синтез арифметических форм булевых функций посредством преобразования Фурье // Автоматика и телемеханика. 1989. № 5. С. 134–142.
87. Шмерко В. П. Теоремы Малюгина: новое понимание в логическом управлении, проектировании СБИС и структурах данных для новых технологий // Автоматика и телемеханика. 2004. № 6.
88. Aiken H. H. The Annals of the Computation Laboratory of Harvard University // Synthesis of electronic Computing and Control Circuits. Cambridge. Massachusetts. Harvard University. 1951. Vol. XXVII.
89. Garner H. L. The residue number system // Ire transactions on electronic computers. 1959. Vol. 8, No 6. P. 140–147.
90. Good I. J. The relationship between two fast Fourier transforms // IEEE Trans. on Computers. 1971. No 3.
91. Hiasat A. A., Abdel-Aty-Zohdy S. H. Residue-to-binary arithmetic converter for the moduli set $(2k;\ 2^{k-1};\ 2^{k-1}-1)$ // IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing. 1998. Vol. 45, No 2. P 204–209.
92. Ibrahim Khalid M., Saloum Salat N. An afficient residue-to-binary converter design // At the same place. 1988. Cas.-35, No 9. P. 1156–1158.
93. Kawamura S., Koike M., Sano F. and Shimbo A. Cox-Rower Architecture for Fast Parallel Montgomery Multiplication // Proceedings EUROCRYPT 2000, LNCS 1807. Springer Verlag, 2000. P. 523–538.
94. Mirsalehi Mir M., Shmir Joseph, Caulfield H. John. Residue arithmetic processing utiliring optical Fredkin gate arrays // Appl. Opt. 1987. Vol. 26, No 19. P. 3940–3946.
95. Paillier P. Low-cost double-size modular exponentiation or how to stretch your cryptoprocessor. In H. Imai and Y. Zheng, editors, Second International Workshop on Practice and Theory in Public Key Cryptography, PKC'99, LNCS-1560, Springer Verlag, 1999. P. 223–234.
96. Posch K. C., Posch R. Residue number systems, a key to parallelism in public key cryptography // Proc. of Fourth IEEE SymP. on Parallel and Distributed Processing. 1992. P. 432–435.
97. Posch K. C., Posch R. RNS-Modulo Reduction in Residue Number Systems // IEEE Trans. on Parallel and Distributed Systems. 1995. Vol. 6, No 5. P. 449454.
98. Soderstrand M. A., Jenkins W. K., Jullien G. A. and Tailor F. J. Residue Number System Arithmetic: Modern Applications in Digital Signal Processing. N. Y.: IEEE Press, 1986.
99. Szabo N. S., Tanaka R. I. Residue arithmetic and its applications to computer technology. N. Y.: McGraw-Hill, 1967.
100. Wigley N. M., Jullien G. A. On modulus replication for residue arithmetic computations of complex inner products. IEEE Trans. on Computers. 1990. Vol. 39, No 8. P. 1065–1076.
101. Wigley N. M., Jullien G. A., Reaume D. Large Dynamic Range Computations over Small Finite Rings // IEEE Trans. on Computers. 1994. Vol. 43, No 1. P. 78–86.
102. Yanushkevich S., Shmerko V., Dziurzanski P. Word Level Decision Diagrams Upon the Conditions of Linearity // IEEE Trans. Comput. Design of Integrated Syst. 2001. Vol. XX, month. P. 1–40.
103. Yanushkevich S., Dziurzanski P., Shmerko V. Word-Level Models for Efficient Computation of Multiple-Valued Functions, Part 1: LAR // IEEE 32th Int'l SymP. on Multiple-Valued Logic. Boston. USA. 2002. P. 202–208.
104. Zhang C. N., Shirazi B., Jun D., Y. Residue number conversion // Explor. Technol.: Today and tomorrow. Fall Joint Comput. Conf., Dallas. Tex., 2529 Oct., 1987. Washington: D. C., 1987. P. 390–396.
105. Zhang D., Jullien G. A., Miller W. C. Recursive reduction in finite ring computations // 23rd Asilomar Conf. Signals, Syst. and Comput., Pasific Grove, Calif., Oct. 30 Nov. 1, 1989: Conf. Rec. Vol. 2, San Jose (Calif.). 1989. P. 854–857.
106. Zhang D., Jullien G. A., Miller W. C. A neural-like approach to finit ring computation // IEEE Trans. Circuits and Syst. 1990. Vol. 37, No 8. P. 1048–1052.

108. Zhang D., Jullien G. A., Miller W. C. VLSI implementation of neural- like networks for finite ring computations // Proc. 32nd Midwest SymP. Circuits and SYst., Champaign, III, Aug. 14–16, 1989. Vol. 1, N. Y., 1990. P. 485–488.
109. Zhang D. Parallel VLSI neural system designs. // Springer, 1998. P. 257.