АНАЛИЗ СОВРЕМЕННЫХ ПРИНЦИПОВ ПОСТРОЕНИЯ СИСТЕМ УПРАВЛЕНИЯ НА ОСНОВЕ МОДЕЛЕЙ МНОГОЗНАЧНОЙ ЛОГИКИ

Есин А. А.[1]
(*ФГБУН Институт проблем управления им. В.А. Трапезникова РАН, Москва*)

Проведён анализ и представлен обзор современных приложений, в которых разработка гетерогенных вычислительных систем с небольшими вычислительными блоками на основе трёхзначной логики является математически лучшим решением по сравнению с бинарными моделями. В частности, рассматриваются приложения в телекоммуникационной отрасли, где подобные решения на базе трёхзначной логики фактически могут обеспечить 1,5-кратный рост скорости передачи данных. Для приложений необходимо реализовать схемы из чипов, работа которых основана на трёхзначной логике. Для возможности реализации таких схем должна быть решена принципиально важная задача – задача полноты классов функций трёхзначной логики. С практической стороны полнота классов таких функций гарантирует, что на базе произвольного множества чипсетов можно произвести плату с нужной функциональной схемой. В данной работе были рассмотрены операторы замыкания на множестве функций трёхзначной логики, являющиеся усилением обычного оператора подстановки, и было показано, что задача полноты для этого оператора имеет решение. Поэтому можно восстановить подрешётку замкнутых классов в общем случае замыкания функций относительно классического оператора суперпозиций. Это позволит оптимизировать возможное производство чипов для новых функциональных схем для задач передачи и обработки данных.

Ключевые слова: многозначная логика, трёхзначная логика, приложения многозначной логики, проблема полноты, оператор замыкания, функции трёхзначной логики.

1. Введение

Известно, что удельная плотность записи информации описывается следующей функцией [12]:

$$Y(a) = \frac{\ln y(a)}{a} = \frac{\ln a}{a},$$

[1] *Антон Анатольевич Есин, соискатель ИПУ РАН (ae@incarnet.ru).*

которая достигает максимума при $a = e \approx 2,718$ [12, 23, 31] (см. рис. 1), т.е. трёхзначная система счисления (из целочисленных) является самой оптимальной с точки зрения плотности записи информации [4, 5, 12, 23].

Как с практической точки зрения, так и в качестве обобщения бинарной логики, наиболее важным случаем многозначной логики является троичная логика.

Общим случаем (обобщением) для многозначной логики является троичная логика [1, 14], поэтому далее, говоря о многозначной логике, будем в первую очередь подразумевать троичную (если явно не указано другое). В троичной логике высказыванию присваивается одно из трёх значений: «истина», «ложь», «не определено» [1, 2, 12, 23]; в двоичной логике – два: либо «истина», либо «ложь». Симметричная форма представления числа на базе трёхзначной логики упрощает работу с отрицательными числами, поскольку не требует выделения дополнительного разряда для хранения знака [12].

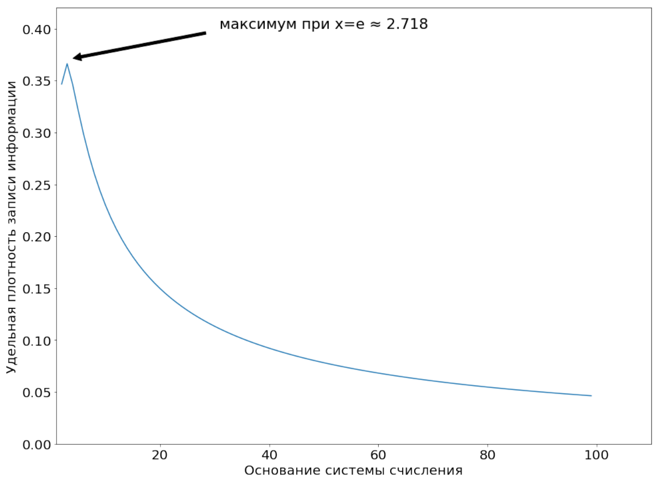

Рис. 1. Зависимость плотности записи информации от основания системы счисления

Преимущество трёхзначной логики над двузначной можно продемонстрировать на простом примере [12]. Операция сравнения двух чисел в троичной логике выполняется за один шаг: сразу выдается результат (больше, меньше или равно). В двоичной логике операция выполняется за два шага:

1) проверяется, больше ли первое число второго;

2) проверяется, не больше ли второе число первого.

Базовые троичные логические операции «Отрицание», «И», «ИЛИ» представлены ниже в таблицах 1, 2 и 3 [2, 3] («0» означает, что состояние не определено).

Таблица 1. Отрицание

a	$\mathrm{NOT}^{-}a$	$\mathrm{NOT}^{0}a$	$\mathrm{NOT}^{+}a$
$-$	$-$	$+$	0
0	$+$	0	$-$
$+$	0	$-$	$+$

Таблица 2. Конъюнкция — логическое И

$x \wedge y = \min(x, y)$	$-$	0	$+$
$-$	$-$	$-$	$-$
0	$-$	0	0
$+$	$-$	0	$+$

Таблица 3. Дизъюнкция — логическое ИЛИ

$x \vee y = \max(x, y)$	$-$	0	$+$
$-$	$-$	0	$+$
0	0	0	$+$
$+$	$+$	$+$	$+$

Слабое использование троичной логики в реальных вычислениях обусловлено исключительно тем, что технология двоичной обработки за многие десятилетия уже хорошо разработана и используется повсеместно. А стоимость производства систем (как аппаратного, так и программного обеспечения) на базе троичного кодирования более высока именно ввиду отсутствия массового производства.

Особенности работы логики такого компьютера, например, представление отрицательных чисел, создают базу для построения высокопроизводительных и надёжных систем для целого ряда современных приложений, возникших за последние несколько лет в различных предметных областях. Математически троичная логика более эффективна, чем двоичная [1, 2, 5, 12, 23].

Исследования и разработка алгоритмов на основе трёхзначной логики вновь актуальны [22], например, в телекоммуникациях [20, 43], в области искусственного интеллекта (ИИ) [18], квантовых вычислениях [20, 28, 36, 37], медицине, физике [41].

Это подтверждается и значительным ростом за последние несколько лет числа научных публикаций в ведущих научных изданиях, связанных с различными приложениями трёхзначной логики [31].

В этой статье мы представляем обзор приложений, где построение алгоритмов на основе трёхзначной логики обеспечивает большую эффективность и оказывается предпочтительнее в сравнении с двузначной. Будет дано сравнение двоичного и троичного кодирования по их эффективности для ряда приложений.

2. Обзор некоторых реальных систем, построенных на базе k-значной логики

2.1. АНАЛИЗ СТРУКТУРНЫХ ПРОЦЕССОВ И ОЦЕНКА ВЛИЯНИЯ ФАКТОРОВ НА НАДЁЖНОСТЬ ТЕХНИЧЕСКИХ СИСТЕМ

Многозначная логика позволяет вводить качественные (лингвистические) переменные вместо количественных. Количественные показатели (факторы) дискретизируются путём отображения в некоторую m-интервальную шкалу. Такой подход позволяет соединить в одной модели количественные и качественные показатели и значительно упростить достаточно сложные вероятностные схемы оценки надёжности [32, 33]. Достоверность факторов при такой дискретизации снижается минимально, что позволяет максимально полно исследовать модель. Это особенно эффективно в ситуациях, когда нет возможности количественно оценить влияние того или иного фактора на процесс; использование качественных переменных дает дополнительные возможности в оценке исследуемых факторов [9, 10].

Для установления функциональной зависимости между входными лингвистическими параметрами модели и выходным параметром (т.е. работоспособностью всей системы) необходимо построить специальную функцию, которая получила название структурной функции. Структурная функция устанавливает однозначное соответствие между возможными состояниями работоспособности элементов и работоспособностью системы [9]:

$$\varphi(x_1, \ldots, x_n) = \varphi(x) : \{0, \ldots, m-1\} \to \{0, \ldots, m-1\}.$$

Для построения структурной функции необходимо четко и ясно представлять структуру процесса и состав задействованных в нем лиц.

В качестве примера эффективной реализации системы на базе k-значной логики можно привести процедуру регистрации абитуриентов на централизованное тестирование (ЦТ) на базе автоматизированной информационной системы [9]. При построении системы важно было изучить влияние различных (внутренних и внешних) параметров на работу данной системы.

Процесс построения структурной функции надёжности процесса записи абитуриентов на тестирования состоял из следующих шагов [9]:

1) определение переменных, включаемых в математическую модель;

2) агрегирование входных переменных по степени схожести влияния на итоговый процесс и нахождение промежуточных переменных на основе экспертной базы правил;

3) расчет структурной функции на основе полученных промежуточных переменных.

По аналогии со схемой структурной функции процесса из работы [9], можно построить схему для любого процесса из любой другой предметной области. Например, на рис. 2 представлена в общем виде графическая интерпретация структурной функции процесса для оценки качества работы системы Wi-Fi-доступа к сети Internet на подвижных объектах (в автомобилях, поездах, на объектах городского пассажирского транспорта пр.). При построении данной схемы были проанализированы и отобраны следующие ключевые внутренние и внешние параметры, влияющие на рассматриваемый организационный процесс: качество связи (наличие доступа ко внешней сети по одному из трёх каналов связи), качество используемого телекоммуникационного оборудования, стабильность обеспечения электроэнергии. Для построения схемы были отобраны только основные факторы влияния. Список можно существенно расширить за счёт учёта дополнительных параметров, например, таких как: скорость движения объекта, уровень сигнала станций сотовой связи, объём нагрузки на сеть (количество пользователей), ёмкость канала, виды трафика и пр.; в этом случае будет возрастать вычислительная сложность структурной функции.

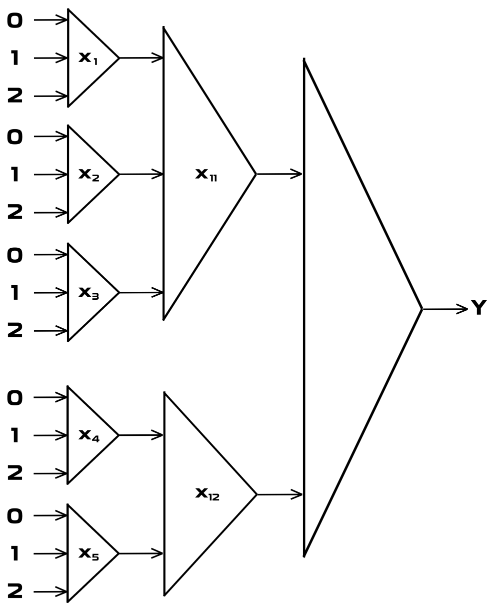

Рис. 2. Схема структурной функции процесса оценки качества работы системы доступа к сети Internet

Для уменьшения вычислительной сложности структурной функции переменные группируют по признакам принадлежности к классам: например, линии связи (x_{11}), сетевое оборудование (x_{12}); и производят расчет промежуточных переменных на основе экспертной базы правил.

В таблице 4 приведены переменные и их значения с необходимыми расшифровками для представленного выше примера. Результаты расчетов промежуточных значений представлены в таблице 5.

Таблица 4. Входные параметры для анализа качества работы системы доступа к сети Internet

Параметр	Значение параметра и его описание		
	0	1	2
Канал связи 1 (x_1)	Не доступен	Слабый сигнал	Устойчивое соединение
Канал связи 2 (x_2)	Не доступен	Слабый сигнал	Устойчивое соединение
Канал связи 3 (x_3)	Не доступен	Слабый сигнал	Устойчивое соединение
Состояние каналов связи x_{11}	Не доступен	Слабый сигнал	Устойчивое соединение
Блок питания (x_4)	Не работает	Неустойчивая работа	Работает
Состояние модема (x_5)	Не работает	Неопределено	Работает
Состояние приёмо-передающего оборудования x_{12}	Не работает	Неустойчивая работа	Работает
Качество работы системы в целом (Y)	Не работает	Нестабильный доступ к сети	Работает

На основе промежуточных переменных находятся значения выходной функции Y. Результаты расчетов представлены ниже в таблице 6.

Таким образом, результатом построения структурной функции процесса стал вектор, интерпретируемый как столбец таблицы истинности на упорядоченных наборах переменных:

$$\{00, 01, 02, 10, 11, 12, 20, 21, 22\} \to Y.$$

Таблица 5. Расчет промежуточных переменных

x_1	0	0	0	0	0	0	0	0	0
x_2	0	0	0	1	1	1	2	2	2
x_3	0	1	2	0	1	2	0	1	2
x_{11}	0	0	2	0	1	2	2	2	2
x_1	1	1	1	1	1	1	1	1	1
x_2	0	0	0	1	1	1	2	2	2
x_3	0	1	2	0	1	2	0	1	2
x_{11}	0	1	2	1	1	2	2	2	2
x_1	2	2	2	2	2	2	2	2	2
x_2	0	0	0	1	1	1	2	2	2
x_3	0	1	2	0	1	2	0	1	2
x_{11}	2	2	2	2	2	2	2	2	2
x_4	0	0	0	1	1	1	2	2	2
x_5	0	1	2	0	1	2	0	1	2
x_{12}	0	0	0	0	1	2	0	2	2

Далее полученная структурная функция может быть использована для определения эффективности и надёжности исследуемого объекта при заданных входных параметрах. Например, если известно, что при создании системы использовалось надёжное и высокопроизводительное телекоммуникационное оборудование, но нет автономных источников питания, способных обеспечить бесперебойную работу данного оборудования, или в зоне нахождения объекта нет достаточного покрытия станциями сотовой связи (или имеет место большая удаленность от них), то это может привести к неустойчивому подключению к сети и, как следствие, к потере данных при передаче по сети. Разработанная структурная функция может использоваться для принятия управленческих решений по улучшению качества работы системы, что позволит снизить риски частичного или полного отказа работоспособности исследуемых объектов в будущем. Дальнейшие исследования и дополнение данной методики позволят более полно оценить влияния различных факторов на работоспособность системы в целом.

Таблица 6. Значения выходной функции Y

x_{11}	0	0	0	1	1	1	2	2	2
x_{12}	0	1	2	0	1	2	0	1	2
Y	0	0	0	0	1	2	0	2	2

2.2. ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ПРОЦЕССОВ И СОВРЕМЕННЫЕ ЯЗЫКИ ПРОЕКТИРОВАНИЯ

Имитационное моделирование является единственным и доступным способом проверки качества и надёжности сложных и дорогостоящих технических систем на этапе их проектирования. Автоматизированные средства проектирования позволяют оценить качество с учетом реальных условий эксплуатации. Имитационное моделирование схем в системах автоматизированного моделирования часто базируется на принципах трёхзначной логики [3, 4, 13].

Функциональные элементы системы при построении имитационной модели представляются логическими блоками на входах/выходах которых формируются многозначные сигналы. Замена двузначных (промежуточных) сигналов на переходах трёхзначными и четырёхзначными позволяет более точно описать реальные параметры работы системы. Значения сигналов представляются как трёхзначные или четырёхзначные, например [4, 13]: $\{A, B, x/, x\backslash\}$, где $\{A, B\}$ – постоянные, $x/$ – положительный сигнал, $x\backslash$ – отрицательный сигнал.

Функционирование логических блоков с такими сигналами описывают таблицами, являющимися аналогами традиционных таблиц истинности в двузначной логике. Пример приведён ниже в таблице 7 [4, 13], где для простоты константы обозначены через «0» и «1», а положительный и отрицательный фронты – через «2» и «3» соответственно.

Ниже в таблице 8 приведён пример вычислений значений для следующей результирующей функции:

$$f = m_1(m(X_1; m_{01}(X_1; X_2)); A + 2(X_1)).$$

На рис. 3 приведена логическая схема построения данной функции.

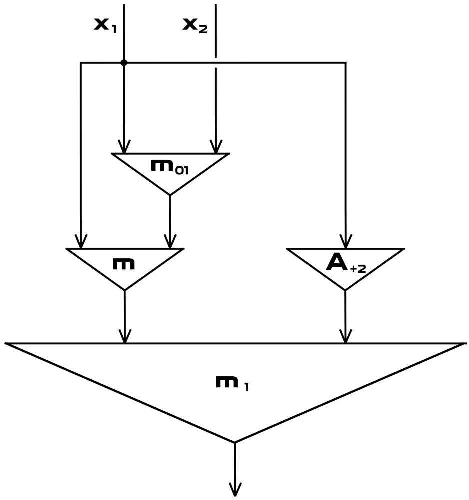

Рис. 3. Пример построения результирующей функции $f(A, B)$

Практически данный подход давно реализован и широко применяется в многих современных системах и языках автоматического проектирования функциональных схем, например, таких, как [7, 8]:

1) программируемая логическая интегральная схема (ПЛИС);

2) языки автоматического проектирования Hardware Description Language, Veri-log, VHDL.

Все эти средства проектирования поддерживают имитационное моделирование функциональных схем именно на базе k-значных $(k > 2)$ логических вычислений [7, 8].

Язык Veri-log работает с трёх- и четырёхзначными сигналами: «0», «1» – для L и H уровней, X – для неопределенного

Таблица 7. Таблицы истинности для некоторых операций четырёхзначной логики

A	$A+1$
0	1
1	2
2	3
3	0

A	$A+2$
0	2
1	3
2	0
3	1

$m(A,B)$	0	1	2	3
0	0	0	0	0
1	0	1	1	1
2	0	1	2	2
3	0	1	2	3

$m_1(A,B)$	0	1	2	3
0	0	1	2	3
1	1	1	1	1
2	2	1	3	0
3	3	1	0	1

$m_{01}(A,B)$	0	1	2	3
0	0	1	1	1
1	1	0	1	0
2	2	0	3	3
3	0	0	3	2

Таблица 8. Таблица значений результирующей функции f

$f(A,B)$	0	1	2	3
0	2	2	2	2
1	1	3	1	3
2	1	2	2	2
3	1	1	1	1

уровня. В начале моделирования логические блоки имеют состояние «не определено» [3, 11, 13]. Соответственно, связанные с ними функциональные элементы на вход получают сигнал, имеющей значение «не определено». При моделировании это указывает разработчику на необходимость начального сброса.

Ниже приведены таблицы истинности для определения логических связок с четырёх- и пятизначными сигналами (таблицы 9 и 10), где Z – плавающий уровень на входах вызывает неопределенное значение X на выходе.

Таблица 9. Логические связки с четырёхзначными сигналами

&	0	1	Z	X
0	0	0	0	0
1	0	1	X	X
Z	0	X	X	X
X	0	X	X	X

Язык проектирования VHDL поддерживает пятизначные сигналы. К перечисленным выше сигналам «0», «1» – для L и H и уровней X и Z, в VHDL к сигналам добавляется значение U – неинициализированное значение.

Таблица 10. Логические связки с пятизначными сигналами

&	0	1	Z	X	U
0	0	0	0	0	0
1	0	1	X	X	X
Z	0	1	X	X	U
X	0	X	X	X	U
U	0	U	U	0	U

2.3. НЕДВОИЧНЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

Обычно описание компьютеров на базе троичной логики ограничивается упоминанием проектов ЭНИАК (и их последователей – серии машин IBM, UNIVAC) и «Сетунь–70» [1, 2, 12]. В различных источниках данные проекты зачастую приводятся как примеры того, что многозначная логика в итоге оказалась неэффективной и окончательно «проиграла» бинарной модели компьютера.

Сейчас построение универсального компьютера на базе трёхзначной логики представляется нецелесообразным ввиду

1) более высокой стоимости трёхзначных логических элементов по сравнению с двоичными и

2) отсутствия качественного троичного программного обеспечения [12].

Несмотря на то, что троичные компьютеры общего назначения не достигли успеха в общем использовании, гетерогенные вычислительные системы с небольшими троичными вычислительными блоками являются математически лучшим решением при построении ряда узкоспециализированных вычислительных систем [22, 41].

Кроме того, с момента завершения проекта «Сетунь–70» производство электроники достигло большого прогресса, и появились новые возможности для построения троичного компьютера [23, 27, 39], в том числе на базе оптических систем [43].

Выделим несколько частных задач для применения троичной вычислительной техники.

Задача 1. *Троичная логика эффективна при построении вычислительных блоков для оборудования сетей передачи данных. Потенциально передача трёх состояний вместо двух по одному разряду может увеличить скорость передачи данных в 1,5 раза. При увеличении числа разрядов скорость может расти экспоненциально [26, 42, 43].*

На базе многозначной логики возможна реализация решений для агрегации данных и обмена данными, которые обеспечиваются единым пространством большой размерности для сетевой адресации – как для стандартных целей передачи данных [24], так и для новых задач по управлению роботизированными устройствами для Internet Of Things [20].

В работе [43] авторы предлагают совершенно новый способ создания троичного оптического компьютера: информацию предлагается представлять двумя поляризованными состояниями с ортогональными направлениями колебаний света и одним состоянием с отсутствием интенсивности света. Реализация такой архитектуры возможна за счет удачного сочетания электрического управления и оптических вычислений: оптоволоконное кольцо предлагается использовать в качестве регистра, полупроводниковую память – в качестве тройной ячейки, жидкий кристалл – в качестве модулятора и сумматора. Данное решение на основе трёхзначной логики может оказаться эффективным и при создание специализированных вычислительных блоков для систем передачи данных

Задача 2. *Эффективной оказывается трёхзначная логика и при решении задач обработки изображений [17].*

Задача 3. *Задачи криптографии [21]. Использование только бинарных систем создаёт вычислительную однородность, неустойчивую ко многим кибератакам, ставшими особенно частыми в последние годы на всех уровнях: начиная от простых пользователей и заканчивая целыми государствами. Троичная логика может улучшить безопасность, что особенно важно в эпоху развития Интернета вещей (IoT), когда ни одно электронное устройство не может быть полностью защищенным от кибератаки. Современные схемы троичного шифрования показывают высокую эффективность при решении криптографических задач и имеют потенциал для повышения надёжности информации, передаваемой через небезопасные каналы связи. Также они могут быть использованы как вспомогательные при создании унаследованных двоичных кодов [21].*

Квантовые вычисления являются наиболее эффективным методом обеспечения криптографической защиты для мобильных роботов и Интернета вещей и безопасности распределённых приложений. Данная задача требует разработки новых схем секретного кодирования, которые также могут быть основаны на многозначной логике [14, 20].

Задача 4. *Новые алгоритмы для сжатия текстовых и графических данных в троичном представлении позволяют минимизировать длину строки битов и тем самым резко сократить длину кода [19, 34].*

Каких-либо принципиальных ограничений для создания цифровых схем на базе трёхзначной логики нет. Более того, предлагаются новые и перспективные системы на базе трёхзначной логики [30].

Эффективность алгоритмов на базе трёхзначной логики подтверждается и тем, что ведущие разработчики компьютеров, такие как Huawei, Samsung, проводят многочисленные исследования и разработки в области троичных вычислений для целого ряда задач: от обработки изображений до разработки троичных полупроводников [27, 44].

Фактически трёхзначная логика уже реализована во многих современных системах на программном уровне.

Например, в языке SQL используется трёхзначная логика: кроме значений «истина» и «ложь», результатом выполнения логических выражений также может быть ответ «неизвестно». Трёхзначная логика SQL является следствием поддержки null-значений для отметки отсутствующих данных. Если нулевое значение влияет на результат логического выражения, результат является не истинным, не ложным, а неизвестным [40]. Трёхзначная логика является неотъемлемой частью Core SQL и используется почти в каждой системе управления базами данных на базе SQL (см. таблицу 11).

Задача 5. *В связи с бурным развитием квантовых компьютеров троичные вычисления вновь стали актуальны [5, 17, 36, 37]. Ведущие IT-компании в последнее десятилетии представили свои квантовые компьютеры, работающие на нескольких десятках кубитов: квантовые процессоры IBM состоят из 65 кубитов, у Google – из 72. В планах разработчиков к 2023 году выпу-*

стить 1112-кубитный процессор под кодовым названием Condor, который должен вывести квантовые технологии на коммерческий уровень [29]. Но это пока далеко от миллионов кубитов, которые могут потребоваться для практических вычислений.

Таблица 11. Поддержка трёхзначной логики в наиболее распространённых СУБД

	Big Query	Db2 LUW	My SQL	Oracle DB	Postgre SQL	SQL Server	SQ Lite
Трёхзн. логика	+	+	+	+	+	+	+
Значения true и false	+	+	+	−	+	−	+

Ячейка для хранения единицы информации в квантовом компьютере – кубит, т.е. это аналог бита в обычном компьютере [5]. Вариантом квантовой ячейки с тремя возможными состояниями является квантовый трит или кутрит. Можно построить троичный квантовый компьютер, используя вместо кубитов кутриты. Используя в универсальных квантовых вентилях кутриты вместо кубитов, исследователи могут существенно снизить количество необходимых вентилей [5]: в обычном случае нужно 50 традиционных квантовых вентилей, а при троичном представлении – всего девять. При этом исследования последних нескольких лет показывают, что использование квантовых тритов вместо кубитов существенно упростит разработку квантовых алгоритмов и программ [19, 31, 35].

2.4. ПРИМЕНЕНИЕ В ЗАДАЧАХ АНАЛИЗА ДАННЫХ И РАЗРАБОТКИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Многозначная, в частности, троичная логика является вполне естественной в таких приложениях, как последовательный статистический анализ Вальда или принятие статистических решений при фиксированных вероятностях ошибок и первого, и второго рода. В настоящее время инструментарий многозначной логики находит широкое применение в задачах, связанных с анализом данных и построением моделей искусственного интеллекта, например, в задачах иерархической кластеризации произвольных наборов данных [18, 20].

Теория многозначной логики над полем комплексных чисел и пороговая функция многозначной логики применяются при построении модели многозначного нейрона. Для дискретнозначного многозначного нейрона отображение ввода/вывода всегда описывается некоторой пороговой функцией многозначной логики. Это применяется при рассмотрении непрерывного многозначного нейрона и построении разделяющей гиперплоскости в n-мерном пространстве, которая определяется функцией активации дискретного многозначного нейрона. Это позволяет решать задачи многоклассовой классификации [18]. Н. Айзенбергом было предложено рассматривать следующую функцию $P(z)$, которая и называется k-значным предикатом:

(1) $P(z) = \mathrm{CSIN}(z) = \varepsilon_k^j, \quad 2\pi j/k \leqslant \arg z < 2\pi(j+1)/k,$

где $z \in \mathbb{C}$. Данная функция по сути задаёт деление комплексной плоскости на k равных секторов линиями, проходящими через начало координат, и точки на единичной окружности, соответствующие корням k-й степени из единицы.

Определение 1 [18]. *Пороговой функцией k-значной логики (или многозначной пороговой функцией) называется функция*

$$f(x_1, ..., x_n) : T \to E;\ T \subseteq E_k^n,$$

если существуют $(n+1)$ *комплексные числа* $w_0, w_1, ..., w_n$*, такие, что для любого вектора* $(x_1, ..., x_n) \in T$

$$f(x_1, \ldots, x_n) = P(w_0 + w_1 x_1 + \cdots + w_n x_n),$$

где $P(z)$ *– k-значный предикат, определяемый по формуле (1).*

Вектор $W = (w_1, w_2, ..., w_n)$ – вектор весов для функции f.

Таким образом, введение многозначных функций значительно расширяет расширяет набор функций, которые могут быть представлены с использованием $(n+1)$ весов путем добавления многозначных пороговых функций к булевым пороговым функциям, а логическая пороговая функция – это частный случай многозначной пороговой функции. В качестве простого примера такой функции можно рассмотреть функцию максимума из двух аргументов [18]: $f = \max(x_1, x_2)$.

Данная функция является пороговой функцией (так же, как, например, $f = \min(x_1, x_2)$. Многозначные функции над полем комплексных чисел существенно расширяют набор пороговых функций и перспективы в пороговой логике, в нейронных сетях и в решении задач мультиклассовой классификации. Ниже в таблице 12 приведены значения функции Поста $f = \max(x_1, x_2)$ (пороговая функция трёхзначной логики) с вектором весов $W = (-2 - 4\varepsilon_3,\ 4 + 5\varepsilon_3,\ 4 + 5\varepsilon_3)$ [18].

Таблица 12. Многозначная функция Поста $f = \max(x_1, x_2)$

#	x_1	x_2	$z = w_0 + w_1x_1 + w_2x_2$	$\arg(z)$	$P(z)$	$f_{\max}(x_1, x_2)$
1	ε_3^0	ε_3^0	$6 + 6\varepsilon_3$	1,0471	ε_3^0	ε_3^0
2	ε_3^0	ε_3^1	$2 + 5\varepsilon_3 + 5\varepsilon_3^2$	π	ε_3^1	ε_3^1
3	ε_3^0	ε_3^2	$7 + \varepsilon_3 + 4\varepsilon_3^2$	5,7596	ε_3^2	ε_3^2
4	ε_3^1	ε_3^0	$2 + 5\varepsilon_3 + 5\varepsilon_3^2$	π	ε_3^1	ε_3^1
5	ε_3^1	ε_3^1	$2 + 4\varepsilon_3 + 10\varepsilon_3^2$	3,6652	ε_3^1	ε_3^1
6	ε_3^1	ε_3^2	$-2 + 9\varepsilon_3^2$	4,5223	ε_3^2	ε_3^2
7	ε_3^2	ε_3^0	$7 + \varepsilon_3 + 4\varepsilon_3^2$	5,7596	ε_3^2	ε_3^2
8	ε_3^2	ε_3^1	$3 + 9\varepsilon_3^2$	4,4223	ε_3^2	ε_3^2
9	ε_3^2	ε_3^2	$8 - 4\varepsilon_3 + 8\varepsilon_3^2$	5,2359	ε_3^2	ε_3^2

2.5. ПРИМЕНЕНИЕ В ЭКОНОМИЧЕСКИХ ИССЛЕДОВАНИЯХ

В завершение обзора приложений многозначной логики отметим ещё одно реальное применение в экономических исследованиях и в ситуациях, связанных с проблемой коллективно выбора, где возникает «цикличная логика» как частный случай k-значной логики [6].

Один из наиболее характерных примеров связан с проблемой выбора. Рассмотрим выбор из трёх альтернатив (например, выборы с тремя кандидатами) A, B, C. При этом, голосующий имеет некую систему предпочтений, устроенную циклическим образом: $A \prec B \prec C \prec A$. Тогда задание отношений между элементами определяет аналоги операций «ИЛИ» и «И» для трёхзначной логики (таблицы 13 и 14).

Ещё одно применение многозначная логика находит в экономических исследованиях при оценке инвестиционных рисков и прогнозировании стоимости страховых убытков. Постановки задач в этой области приложений аналогичны постановке задачи по оценке эффективности и надёжности технических систем, рассмотренной выше: сначала определяется полный спектр возможных сценариев развития инвестиционного процесса и факторы, оказывающие влияние на этот процесс, а затем находятся оценки (как значения функции от вектора факторов) для принятия инвестиционного решения.

Таблица 13. Операция «ИЛИ»

ИЛИ	B	C	A
B	B	C	B
C	C	C	A
A	B	A	A

Таблица 14. Операция «И»

И	B	C	A
B	B	B	A
C	B	C	C
A	A	C	A

В отличие от технических систем, факторы задаются не лингвистическими переменными (оцениваются не точечно), а полем интервальных значений с некоторым распределением вероятностей.

3. Теоретические аспекты построения реальных систем на базе трёхзначной логики

Любая из выше рассмотренных задач сводится к определению факторов, влияющих на процесс, и рассмотрению счетного множества P_3 состояний этих факторов (любое счётное число состояний можно аппроксимировать до принципиально трех состояний) [16]: 0, 1, 2.

Для принятия решения необходимо найти значение выходной функции Y. Соответственно, мы можем представить выходную функцию Y как комбинацию предикатов на множестве P_3 [15]: для этого мы будем рассматривать сложные предикаты и суперпозиции этих предикатов на P_3.

Данные предикаты практически могут быть реализованы как схемы из чипов, работа которых основана на трёхзначной логике.

Для возможности реализации таких схем должна быть решена принципиально важная задача – *задача полноты классов функций трёхзначной логики* [15]. С практической стороны полнота классов таких функций гарантирует, что на базе произвольного множества чипсетов можно создать плату с нужной функциональной схемой. Для двузначной логики эта задача была также решена Постом, что привело к бурному росту электроники [38].

Классическая теорема Поста описывает пять предполных классов в множестве булевых функций[38].

Для случая трёхзначной логики задача была решена С.В. Яблонским в 1958 г. [15, 16]. Он показал, что для функций трёхзначной логики существует 18 предполных классов. В работах [15, 16] рассматривалось замыкание множества функций относительно оператора подстановки.

К сожалению, для трёхзначной логики было доказано, что в общем случае эта задача не может быть решена [16]. Если в случае двузначной логики решётка замкнутых классов счетная, то в случае трёхзначной логики она экспоненциальная.

Однако, можно рассматривать *операторы замыкания* на множестве функций трёхзначной логики, являющиеся усилением обычного оператора подстановки. Решение для этого нового оператора замыкания задач полноты и нахождение структуры решётки замкнутых классов поможет не только восстановить подрешётку замкнутых классов в общем случае замыкания функций относительно классического оператора суперпозиций, но и позволит оптимизировать возможное производство чипов для функциональных схем для решения задач, описанных выше в разделе 2.

3.1. ИССЛЕДОВАНИЕ ОПЕРАТОРА ЗАМЫКАНИЯ И ОПИСАНИЕ ПРЕДПОЛНЫХ КЛАССОВ В P_3

Пусть дано множество M функций из P_3. Через $\Re_2(M)$ будем обозначать результат замыкания множества функций M относительно операции подстановки и перехода от функции g к эквивалентной функции $f \sim g$, где

$$f \sim g \Leftrightarrow (\forall \vec{x}\,[\,(f(\vec{x}) = g(\vec{x})) \vee (f(\vec{x}), g(\vec{x}) \in \{0,1\})\,])\&(|X_f| = |X_g|).$$

В дальнейшем все рассуждения ведутся для $k = 3$.

Пусть $\alpha^n \in E_3^n$, $\alpha^n = (a_1, \ldots, a_n)$, обозначим $O_2(\alpha^n) = |\{a_i | a_i = 2, i \in \{1, n\}\}|$ – *порядок двойки* набора α^n.

Определение 2. Для $f \in P_3$, такой что $|X_f| = n$, считаем $f \in \nabla_2$, если для каждого α^n такого, что $\alpha^n \in E_3^n$ и $O_2(\alpha^n) > 0$, выполняется $f(\alpha^n) = 2$. Также считаем, что все константы содержатся в ∇_2.

Можно показать [25], что будет иметь место

Теорема 1 (Полноты)**.** *Существует пять $\Re_2$-предполных классов $P_3 : T_2, T_{01}, T_{\sim}, \Delta_2, P_3(x)$.*

Доказательство. Доказательство следует из лемм 1–3, которые формулируются и доказываются ниже.

Лемма 1. *Множество ∇_2 является $\Re_2$-замкнутым.*

Доказательство. $\Re_2$ – многократная суперпозиция расширения $R_2(\cdot)$ и стандартного замыкания $[\cdot]$. Очевидно, в силу определения замыкания, что расширение не выходит за пределы ∇_2.

Осталось доказать, что ∇_2 замкнуто относительно $[\cdot]$. Пусть $f \in \nabla_2$. Очевидно, что после подстановки констант вместо одной (или нескольких) переменных функции f она не потеряет свойство принадлежности классу ∇_2. Если же мы вместо всех существенных переменных подставим константы, то получим константу, которая, в свою очередь, также лежит в ∇_2. Пусть $f \in \nabla_2$, а $g_i, i = \overline{1,n}$, – либо переменные, либо $g_i \in \nabla_2$. Возьмем $h = f(g_1, \ldots, g_n)$. Без ограничения общности считаем, что h существенно зависит от всех своих переменных, так как если это не так, то берем $h' = h$, полученную путем отождествления фиктивных переменных. Пусть $|X_h| = k$. Если на вход h подается набор $\gamma^k = (c_1, \ldots, c_k)$ и для некоторого i выполняется $c_i = 2$, тогда эта 2-ка попадет на вход к одной из g_i. Но если g_i – переменная, то получаем, что на вход к f подошла двойка, а $f \in \nabla_2$, значит, $h(\gamma^k) = 2$. Если же $g_i \in \nabla_2$, то на выходе g_i будем иметь 2, которая также идет на вход f, но $f \in \nabla_2$, значит, $h(\gamma^n) = 2$, и получаем $h \in \nabla_2$.

Лемма 2. *Класс $P_3(x)$ является $\Re_2$-предполным.*

Доказательство. $\Re_2$-замкнутость класса $P_3(x)$ следует из того, что $[P_3(x)] = P_3(x)$, и по определению из fR_2g вытекает, что $|X_f| = |X_g|$, т.е. расширение не выйдет за пределы $P_3(x)$.

Докажем ,что $\Re_2(P_3(x) \cup f^{P_3(x)}) = P_3$. Построим вектор принадлежности. Пусть $g_1(x) = x + 2$, $g_2(x) = x^2$. Прямая проверка определений классов из $\mathcal{K}_{[\cdot]}$ показывает, что $g_1 \notin M_1^3 \cup M_2^3 \cup M_3^3 \cup T_{01} \cup T_{02} \cup T_{12} \cup T_0 \cup T_1 \cup T_2 \cup H_0 \cup H_1 \cup H_2 \cup T_\sim \cup T_{\{0,2\},1} \cup T_{\{1,2\},0}$, $g_2 \notin L \cup S_{x+1}$.

Из определения Sl выполняется $P_3(x) \subset Sl$.

Докажем, что $\Re_2(P_3(x) \cup f^{P_3(x)}) = P_3$. Если $f^{P_3(x)} \notin Sl$, то, в силу теоремы Слупецкого, все доказано. Пусть $f = f^{P_3(x)} \in Sl$, т.е. $f \notin P_3(x), f \in Sl$. Пусть $|X_f| = n$. Рассмотрим вектор значений $f^n = (f_1, \ldots, f_{3^n})^\top$. По определению f множество $\{0, 1, 2\}$ не содержится в $\{f_1, \ldots, f_{3^n}\}$, следовательно, существует единственное $a \in \{0, 1\}$ такое, что $f_i \in \{a, 2\}$, $i = \overline{1, 3^n}$. Пусть без ограничения общности $a = 0$. Возможны три случая.

а. Пусть $O_2(f^n) \leqslant 3^n - 2$, т.е. имеются α_1^n, α_2^n такие, что $f(\alpha_1^n) = f(\alpha_2^n) = 0$. Перейдем к функции g такой, что fR_2g и g совпадает с f на всех наборах, кроме α_1^n, на котором $g(\alpha_1^n) = 1$.

В силу того, что $1 \notin \{f_1, \ldots, f_{3^n}\}$, расширение корректно (т.е. $|X_f| = |X_g|$). Заметим, что $g \notin Sl$.

б. Пусть теперь $O_2(f^n) = 3^n - 1$, т.е. имеется единственное α_1^n такое, что $f(\alpha_1^n) = 0$. Заметим, что существует $g' \in P_3(x)$, $g'^1 = (2, 2, 0)^\top$. Рассмотрим функцию $h = g'(f)$. Очевидно, что $|X_f| = |X_h|$. Также заметим, что если $h^n = (h_1, \ldots, h_{3^n})$, то $h_i \in \{0, 2\}$, $i = \overline{1, 3^n}$, и $0_2(h^n) = 1 \leqslant 3^n - 2$. Таким образом пункт б сводится к пункту а.

в. Если $O_2(f^n) = 0$, то с помощью функции $x + 1$ можно построить $f'(\tilde{x}) = f(\tilde{x}) + 1$. При этом $f' \in \Re_2(P_3(x) \cup f^{P_3(x)})$, и для f' выполнено либо условие а, либо условие б, а, значит, найдется $f^{Sl} \in \Re_2(P_3(x) \cup f_3^P(x))$, следовательно, $P_3(x)$–$\Re_2$ – предполный класс.

Лемма 3. *Если после замены в векторе функции $f, |X_f| \geqslant 1$ нуля на единицу или единицу на ноль получился вектор функции $g, |X_g| < |X_f|$, то с помощью подстановки констант в функцию f можно получить функцию $h \in \tilde{P}_2(x)$.*

Доказательство. Пусть после перехода от f к g переменная x_i стала несущественной, тогда существуют наборы $\alpha_1^n = (a_1, \ldots, a_{i-1}, a_i^1, a_{i+1}, \ldots, a_n)$, $\alpha_2^n = (a_1, \ldots, a_{i-1}, a_i^2, a_{i+1}, \ldots, a_n)$, $\alpha_3^n = (a_1, \ldots, a_{i-1}, a_i^3, a_{i+1}, \ldots, a_n)$, что существует перестановка $\sigma \in \mathcal{S}$, $\sigma(a_i^j) = j$, при $j \in E_3$. Заметим, что $f(\alpha_1^n) R f(\alpha_2^n) R f(\alpha_3^n)$, и $f(\alpha_1^n)$, $f(\alpha_2^n)$, $f(\alpha_3^n) \in E_2$, причем существуют $l, m \in E_3$, что $f(\alpha_l^n) \neq f(\alpha_m^n)$. Получаем, что функция $h(x) = f(a_1, \ldots, a_{i-1}, x, a_{i+1}, \ldots, a_n)$ будет удовлетворять всем требуемым условиям.

Таким образом для оператора $\Re_2$ доказано, что при $k = 3$ существует 5 предполных классов.

4. Выводы

В статье был проведён анализ и представлен обзор современных приложений, где построение алгоритмов на основе трёхзначной логики обеспечивает большую эффективность и оказывается предпочтительнее в сравнении с двузначной.

Создание гетерогенных вычислительных систем с небольшими троичными вычислительными блоками является математически лучшим решением при построении узкоспециализированных вычислительных систем, особенно для задач передачи, обработки и защиты данных, распознавания изображений, квантовых вычислений, в том числе для проведения научных исследований в современных областях физики.

В области телекоммуникаций решения на базе трёхзначной логики фактически могут обеспечить 1,5-кратный рост скорости передачи данных (а в перспективе и экспоненциальный). Для практического решения данной задачи необходимо реализовать схемы из чипов, работа которых основана на трёхзначной логике. Для возможности реализации таких схем должна быть решена принципиально важная задача – *задача полноты классов функций трёхзначной логики*. С практической стороны полнота классов таких функций гарантирует, что на базе произвольного множества чипсетов можно создать плату с нужной функциональной схемой.

В данной работе были рассмотрены *операторы замыкания* на множестве функций трёхзначной логики, являющиеся усилением обычного оператора подстановки. Было показано, что задача полноты для этого оператора имеет решение, то есть можно восстановить подрешетку замкнутых классов в общем случае замыкания функций относительно классического оператора суперпозиций. Это позволит оптимизировать возможное производство чипов для новых функциональных схем для задач передачи и обработки данных.

Автор статьи выражает благодарность Г.А. Зверкиной и М.П. Фархадову за полезное обсуждение статьи и ценные советы.

Литература

1. БРУСЕНЦОВ Н.П. *Об использовании троичного кода и троичной логики в цифровых машинах* // Вычислительная техника и вопросы кибернетики. – 1970. – Вып. 7. – С. 3–33.

2. БРУСЕНЦОВ Н.П. *Трехзначное обобщение алгебры логики. Преодоление несовершенности ДНФ трехзначным обобщением логики* // Историко-математические исследования. Вторая серия. – 2014. – Выпуск 15(50). – С. 241–242.
3. ДМИТРИЕНКО В.Д., ЛЕОНОВ С.Ю. *Моделирование цифровых устройств на основе многозначных алфавитов и k-значного дифференциального исчисления* // Вестник Национального технического университета Харьковский политехнический институт. Серия: Информатика и моделирование. – 2008. – Т. 57, №4. – С. 42–51.
4. ЗЫКОВ А.Г., ПОЛЯКОВ В.И., СКОРУБСКИЙ В.И. *Математическая логика и теория алгоритмов: Учебное пособие.* – СПб: НИУ ИТМО, 2013. – 131 с.
5. *Замена двоичной логики – увеличит ли это производительность?* [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/166679/ (дата обращения: 10.10.2020).
6. *«Иная» логика и обратимые вычисления* [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/274645/ (дата обращения: 10.10.2020).
7. КАЛИННИКОВ В.А. *Применение многозначной логики в цифровой технике. (Обзор)* // Приборы и техника эксперимента. – 2006. – №6. – С. 5–17.
8. КАЛИННИКОВ В.А. *Применение многозначной логики в цифровой технике.* – Дубна, Объед. ин-та ядер. исслед., 2005.
9. КОВАЛЕНКО И.В., ПОТТОСИНА С.А. *Об использовании многозначной логики в исследовании социально-экономических систем* // Доклады Белорусского государственного университета информатики и радиоэлектроники. – 2015. – №1(62). – 7 с.
10. ЛЕВАШЕНКО В.Г. , КОЗЛОВА И.К., ПОТТОСИНА С.А. *Анализ чувствительности структурных функций многозначной логики в системах поддержки принятия решений* // Доклады Белорусского государственного университета ин-

форматики и радиоэлектроники. – 2008. – №1(31). – 8 с.
11. МАРЧЕНКОВ С.С. *S-классификация функций трёхзначно логики.* – М., Физматлит, 2001.
12. *Недвоичная логика* [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/160595/ (дата обращения: 10.10.2020).
13. ПОЛЯКОВ В.И., СКОРУБСКИЙ В.И. *Использование многозначной логики при проектировании функциональных схем* // Известия высших учебных заведений. Приборостроение. – 2014. – Т. 57, №4. – С. 57–60.
14. ФАТХИ Д.В., ДУДЕНКОВ А.В. *Оценка угроз в сфере высоких технологий на основе многозначной логики с переменными весами операций* // Информационная безопасность регионов. – 2012. – №1. – URL: https://cyberleninka.ru/article/n/otsenka-ugroz-v-sfere-vysokih-tehnologiy-na-osnove-mnogoznachnoy-logiki-s-peremennymi-vesami-operatsiy (дата обращения: 10.10.2020).
15. ЯБЛОНСКИЙ С.В., ГАВРИЛОВ Г.П., КУДРЯВЦЕВ В.Б. *Функции алгебры логики и классы Поста.* – М.: Изд-во «Наука», 1966.
16. ЯБЛОНСКИЙ С.В. *Функциональные построения в k-значной логике* // Тр. Матем. инст. им. В.А. Стеклова. – 1958. – Том 51.
17. ABIRI E., DARABI A., SALEM S. *Design of multiple-valued logic gates using gate-diffusion input for image processing applications* // Computers Electrical Engineering. – 2018. – Vol. 69. – P. 142–157. – DOI: 0.1016/j.compeleceng.2018.05.019.
18. AIZENBERG I. *Complex-Valued Neural Networks with Multi-Valued Neurons.* Studies in Computational Intelligence. – 2011. – Vol. 353. – 273 p. – DOI: 10.1007/978-3-642-20353-4.
19. ASADI M., MOSLEH M., HAGHPARAST M. *Toward novel designs of reversible ternary 6:2 Compressor using efficient reversible ternary full-adders* // The Journal of Supercomputing. – 2020. – DOI:

https://doi.org/10.1007/s11227-020-03485-7.

20. BYKOVSKY A.YU. *Heterogeneous Network Architecture for Integration of AI and Quantum Optics by Means of Multiple-Valued Logic* // Quantum Rep. – 2020. – No. 2. – P. 126–165. – DOI:10.3390/quantum2010010.
21. CAMBOU B., FLIKKEMA P.G., PALMER J., TELESCA D., PHILABAUM C. *Can Ternary Computing Improve Information Assurance?* // Cryptography. – 2018. – Vol.2, Iss.1. – DOI: 10.3390/cryptography2010006.
22. COBREROS P., EGRE P., RIPLEY D., VAN ROOIJ R. *Three-valued logics and their applications* // Journal of Applied Non-Classical Logics. – 2014. – Vol. 24, Iss. 1-2. – P. 1–11. – DOI: 10.1080/11663081.2014.909631.
23. CONNELLY J. *Ternary Computing Testbed 3-Trit Computer Architecture.* PhD thesis. Computer Engineering Department. California Polytechnic State University. – 2008. – 192 p. – URL: http://xyzzy.freeshell.org/trinary/CPE%20Report%20-%20Ternary%20Computing%20Testbed%20-%20RC6a.pdf (дата обращения: 10.10.2020).
24. ESIN A., YAVORSKIY R., ZEMTSOV N. *Brief Announcement Monitoring of Linear Distributed Computations* // In: Dolev S. (eds) Distributed Computing. DISC 2006. Lecture Notes in Computer Science. – Vol. 4167. – Springer, Berlin, Heidelberg, 2006. – DOI: https://doi.org/10.1007/11864219-47.
25. ESIN A.A. *On function classes in P3 precomplete with respect to a strengthened closure operator* // Math Notes. – 2008. – Vol. 83, No. 5. – P. 594–603. – DOI: 10.1134/S0001434608050027.
26. GAUDET V. *A survey and tutorial on contemporary aspects of multiple-valued logic and its application to microelectronic circuits* // IEEE Journal on Emerging and Selected Topics in Circuits and Systems. – 2016. – Vol. 6, March, No. 1. – P. 5–12. – DOI: 10.1109/JETCAS.2016.2528041.

27. HRUSKA J. *Samsung-Backed Researchers Debut Ternary Semiconductor Design* // ExtremeTech. – 2019, July 22. – URL: https://www.extremetech.com/computing/295424-back-off-binary-samsung-backed-researchers-debut-ternary-semiconductor (дата обращения: 10.10.2020).
28. HU ZH., DEIBUK V. *Design of ternary reversible/quantum sequential elements* // Journal of Thermoelectricity. – 2018. – No. 1. – P. 5–16.
29. *IBM Quantum Summit 2020: Exploring the Promise of Quantum Computing for Industry* [Электронный ресурс]. – Режим доступа: https://www.ibm.com/blogs/research/2020/09/quantum-industry/ (дата обращения: 30.10.2020).
30. *Implementation of a Simple Ternary System* [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/431726/ (дата обращения: 10.10.2020).
31. KAK SUBHASH *On Ternary Coding and Three-Valued Logic.* – 2018. – URL: https://arxiv.org/abs/1807.06419 (дата обращения: 10.10.2020).
32. KALIMULINA E.YU. *Analysis of system reliability with control, dependent failures, and arbitrary repair times* // Int. Journal of System Assurance Engineering. – 2017. – Vol. 8. – P. 180–188. – DOI: https://doi.org/10.1007/s13198-016-0520-5.
33. KALIMULINA E.YU. *Rate of Convergence to Stationary Distribution for Unreliable Jackson-Type Queueing Network with Dynamic Routing* // Communications in Computer and Information Science. – 2017. – Vol. 678. – DOI: https://doi.org/10.1007/978-3-319-51917-3_23.
34. KATUGAMPOLA U. *A New Technique for Text Data Compression.* – 2012. – URL: https://arxiv.org/abs/1012.4241 (дата обращения: 10.10.2020).
35. MOHAGHEGH S.M., SABBAGHI-NADOOSHAN R., MOHAMMADI M. *Designing ternary quantum-dot cellular automata logic circuits based upon an alternative model* // Computers Electrical Engineering. – 2018. – Vol. 71. – P. 43–59. – URL: https://doi.org/10.1016/j.compeleceng.2018.07.001.

36. MUTHUKRISHNAN A., STROUD C.R.JR. *Multivalued logic gates for quantum computation* // Phys. Rev. A. – 2000. – Vol. 62, Iss. 5. – DOI: https://doi.org/10.1103/PhysRevA.62.052309.
37. MUTHUKRISHNAN A. *Classical and Quantum Logic Gates: An Introduction to Quantum Computing.* – Rochester Center for Quantum Information (RCQI). Quantum Information Seminar, 1999. – 22 p. [Электронный ресурс]. – Режим доступа: http://www2.optics.rochester.edu/ stroud/presentations/ muthukrishnan991/LogicGates.pdf (дата обращения: 10.10.2020).
38. POST E.L. *Two-valued iterative systems of mathematical logic* // Annals of Math. Studies. – 1941. – Vol. 5.
39. SIMONETTA A., PAOLETTI M.C. *Designing Digital Circuits in Multi-Valued Logic* // Int. Journal on Advanced Science Engineering Information Technology. – 2018. – Vol. 8, No. 4. – URL: https://core.ac.uk/download/pdf/325990569.pdf.
40. *The Three-Valued Logic of SQL* [Электронный ресурс]. – Режим доступа: https://modern-sql.com/concept/three-valued-logic (дата обращения: 10.10.2020).
41. WARZECHA M., OSZAJCA M., PILARCZYK K., SZACILOWSKI K. *A three-valued photoelectrochemical logic device realising accept anything and consensus operations* // Chemical Communications. – 2015. – Vol.51, Iss.17. – P. 3559–3561. – DOI: 10.1039/C4CC09980J.
42. WU HAIXIA, BAI YILONG, LI XIAORAN, WANG YIMING *Design of High-Speed Quaternary D Flip-Flop Based on Multiple-valued Current-mode* // Journal of Physics: Conference Series. – 2020, October. – Vol. 1626. – DOI: 10.1088/1742-6596/1626/1/012067.
43. YI JIN, HUACAN HE, YANGTIAN L. *Ternary Optical Computer Architecture* // Physica Scripta. – 2005. – T118. – DOI: 10.1238/physica.topical.118a00098.
44. ZHANG WEI, HOU LU, YIN YICHUN, LIFENG SHANG, XIAO CHEN, XIN JIANG, QUN LIU. *TernaryBERT:*

Distillation-aware Ultra-low Bit BERT. – 2020. – URL: https://arxiv.org/abs/2009.12812 (дата обращения: 10.10.2020).

ANALYSIS AND DESIGN PRINCIPLES OF MODERN CONTROL SYSTEMS BASED ON MULTI-VALUED LOGIC MODELS

Anton Esin, V.A. Trapeznikov Institute of Control Sciences of RAS, Moscow, PhD student (ae@incarnet.ru).

Abstract: An analysis and an overview of modern applications based on three-valued logic have been presented in this paper. Small computing units based on three-valued logic is a better solution (in comparison with binary models) for several applications such as the telecommunications industry, where three-valued logic units may increase the data transmission rate by one and a half times. It is important to have a possibility to assembly any circuits from three-valued logic chips. An important fundamental problem of class completeness for three-valued logic functions must be solved to make such implementation possible. The class completeness for three-valued logic functions guarantees that any digital circuit may be assembled from the finite number of ternary chipsets. The closure operator on the set of three-valued functions has been considered in this paper. It is a strength of the substitution operator. The completeness problem for this operator has been proved. This fact allows to restore in the general case the sublattice of closed classes with respect to the classical operator of superpositions. It's a principal theoretical result that can optimize the assembly process for new digital circuits for transmission and data processing problems.

Keywords: multivalued logic, three-valued logic, applications of multivalued logic, completeness problem, closure operator, functions of three-valued logic.

УДК 510.644, 519.716.32
ББК 22.12
DOI: 10.25728/ubs.2020.88.4

Статья представлена к публикации членом редакционной коллегии Э.Ю. Калимулиной.

Поступила в редакцию 13.11.2020.
Дата опубликования 30.11.2020.