

Е.А. Титенко

**ПРЕОБРАЗОВАТЕЛИ УНИТАРНЫХ КОДОВ ДЛЯ ОДНОРОДНЫХ  
ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ**

**Актуальность.** Эффективная работа вычислительных систем, в том числе, основывается на общезначимых обеспечивающих вычислениях по планированию параллельных вычислений и анализу результатов. Достаточно важными вычислительными средствами являются преобразователи (формирователи) унитарных кодов, совмещающих свойства числовой и символической информации. **Цель работы** – создание высокопроизводительных вычислительных схем для обработки унитарных кодов на единой теоретической основе. **Методы исследования.** Известные одномерные и двумерные итерационные сети являются основой для создания однородных преобразователей унитарных кодов. Такие сети имеют необходимые и достаточные условия для организации параллельных вычислений. Для синтеза преобразователей унитарных кодов были выделены следующие принципы обработки, свойственные для чисел и строк: двунаправленность обработки, разбиение на множество локальных процессов с собственными стартовыми точками, иерархия, мультифункциональность, дуализм цифра/символ. Описанные преобразователи используют известные и привносят новые схемотехнические решения. Описаны цифровой компрессор, формирователь серии логических «1», арбитр, пороговый элемент весовых и унитарных кодов. **Результаты и обсуждения.** Созданы практически значимые схемы прямых и обратных преобразователей кодов «8-4-2-1 – нормализованный код», используемые в однородных вычислительных системах – мультипроцессорах, ассоциативных процессорах и др. Количественные оценки преобразователей унитарных кодов проведены для порогового элемента весового и унитарного кодов. Данный преобразователь основан на дуальной трактовке элементов кода как цифры или символа, что позволило на завершающей фазе вычислений (против стандартного метода) исключить линейную временную зависимость для вычисления результата сравнения двух кодов. Показано, что для унитарных кодов размеров от 12 до 36 бит временной выигрыш составляет 14-16%. Данный эффект получен за счет исключения последовательных вычислений между ячейками итерационной сети. **Выводы.** Для построения эффективных по времени схем преобразования унитарных кодов использован и развит аппарат итерационных сетей, на основе которых созданы одномерные, двумерные итерационные сети с регулярными связями, а также преобразователи на основе универсальных логических модулей.

Итерационная сеть; дуализм данных; параллельная обработка.

Е.А. Titenko

**UNITARY CODE CONVERTERS FOR HOMOGENEOUS COMPUTING SYSTEMS**

**Relevance.** Effective operation of computing systems, among other things, is based on generally significant supporting calculations for planning parallel calculations and analyzing the results. Converters (formers) of unitary codes that combine the properties of numerical and symbolic information are quite important computing units. **The purpose of the work** is to create high-performance computing tools for processing unitary codes on a single theoretical basis. **Research methods.** Known one-dimensional and two-dimensional iterative networks are the basis for creating homogeneous converters of unitary codes that have the necessary and sufficient conditions for organizing parallel calculations. To synthesize unitary code converters, the following processing principles inherent in numbers and strings were identified: bidirectional processing, splitting into many local processes with their own starting points, hierarchy, multifunctionality, digit/symbol dualism. The described converters use known and introduce new circuit solutions. A digital compressor, a generator of a series of logical "1", an arbiter, a threshold element of weight and unitary codes are described. **Results and discussions.** Practically significant circuits of direct and inverse converters of "8-4-2-1 – normalized code" codes are created, used in homogeneous computing systems - multiprocessors, associative processors, etc. Quantitative assessments of unitary code converters are carried out for the created converter – a threshold element of weight and unitary codes. This converter is based on the dual interpretation of code elements as a digit and a symbol, which made it possible to exclude the linear time dependence on obtaining the result of comparing two codes at the final stage of calculations (versus the standard method). It is shown that for unitary codes of sizes from 12 to 36 bits, the time gain is 14-16%. This effect is obtained by eliminating sequential calculations between the cells of the iterative network. **Conclusions.** To construct effective time-saving schemes for converting uni-

*tary codes, the apparatus of iterative networks was used and developed, on the basis of which one-dimensional and two-dimensional iterative networks with regular connections were created, as well as converters based on universal logical modules.*

*Iterative network; data dualism; parallel processing.*

**Введение.** Создание новых информационных технологий параллельной обработки данных и знаний является одним из важнейших направлений развития высокопроизводительных вычислительных систем и устройств (ВС и ВУ) [1, 2]. Повышение их производительности возможно, как на основе технологических, структурных подходов [3], так и на основе эксплуатационно-организационных подходов [4].

По мнению ведущих ученых и специалистов в области аппаратно-программных вычислительных средств (Воеводин, В.В., Воеводин, Вл.В., Бурцев В.С., Эйсымонт Л.К., Каляев И.А., Левин И.И., Курейчик В.М., Курейчик В.В., Стемпковский Л.А., Желтов С.Ю., Огнев Б.В., Корнеев В.В. и др.) [5, 6] вновь создаваемые парадигмы интеллектуальных вычислений и гибридные методы обработки востребованы в новых проблемно-поисковых задачах, оперирующих числовой и символьной информацией [7]. Среди значимых тенденций развития ВС и ВУ выделяются подходы создания нетрадиционных архитектур ВС и ВУ, комбинирующих принципы реконфигурации операционной части ВС или ВУ [8] распараллеливания потоков команд и/или данных, конвейеризации вычислений [9], создания «умных» схем контроля параллельных процессов и обеспечения взаимодействия между ними [10].

Современная вычислительная техника, приборостроение, биомехатроника ориентируются на новый класс задач – поисково-вычислительные, слабо формализуемые задачи, задачи когнитивного моделирования и др. [11]. Также в качестве знаковых примеров ведущие ученые указывают задачи комбинаторики слов [12], вычислительной химии и физики, моделирования биологических систем [13], задачи оперативного анализа активности социальных сетей [14], NP-трудные задачи [15], поисково-переборные задачи биосинтеза объектов с заданными свойствами [16]. Общее свойств задач – переборный характер решения и недетерминированность.

Применение ВС с традиционными архитектурами пост-фон-неймановского типа (кластерные ВС, гибридные ВС) не позволяет решать такие проблемно-поисковые задачи с недетерминированным ходом решения. Как правило, их решение основывается на введении модельных упрощений, использовании приближенных вычислительных методов расчета или ограниченного поиска. Как следствие, время решения таких задач или получаемое качество не являются приемлемыми на практике, особенно в критических областях применения.

Таким образом, создание нестандартных технических решений для параллельной обработки информации является актуальным направлением развития однородных ВС и организации числовых и символьных вычислений.

**Постановка задачи.** Эффективная работа высокопроизводительных однородных ВС основывается не только на распараллеливании и одновременном выполнении множества рабочих вычислительных процессов. Не меньшую роль играют обеспечивающие процессы планирования и анализа полученных результатов. Так, в состав однородных мультипроцессоров входит аппаратно-программный планировщик задач, оценивающий и контролирующий ход и статус рабочих процессов, выполняющихся на вычислительных модулях. Ассоциативные процессоры машин баз данных содержат блоком выделения приоритетного решения для выдачи полученных решений из ячеек ассоциативной памяти для последующей обработки [7, 17]. Независимость рабочих процессов в однородных ВС приводит к тому, что двоичные признаки полученных результатов сами представляют новую информацию о распределении результатов, о конфигурации операционной части ВС. Двоичные признаки результатов описываются унитарными кодами (УК), а комбинационные схемы называются преобразователи (формирователи) УК [18, 19].

Унитарный код (УК) – это двоичный не весовой код, количественное значение в котором определяется числом логических «1». УК формируется как двоичный результат параллельной работы множества устройств, блоков, модулей ВС или ВУ на выполняе-

мых рабочих процессах. Основные особенности УК – вариативность представления чисел и независимость элементов кода между собой. Эти особенности позволяют дуально рассматривать УК как число в особом формате или как битовую строку. При этом трактовка УК как строки имеет признаки более общей информационной сущности, что позволяет вести ее независимую обработку при условии аппаратной поддержки элементарных микроопераций преобразования бит УК.

Наиболее важные операции над УК, известные в [19, 20], приведены в табл. 1, где ЧИ –числовая информация, СИ –символьная информация.

Для расширения функциональных возможностей и повышения производительности однородных ВС необходимы схмотехнические решения преобразователей (формирователей) УК на основе принципов однородности и параллельной обработки

Таблица 1

Типовые операции над УК

№ п/п	Операция	Тип информации
1	преобразование «УК → нормализованный УК»	ЧИ
2	подсчет количества логических «1»	ЧИ
3	поиск первого правого/левого логического «0»	СИ
4	поиск первой правой/левой логической «1»	СИ
5	поиск адреса первого правого/левого логического «0»	ЧИ
6	поиск адреса первой правой/левой логической «1»	СИ
7	поиск правой/левой серии логических «0»	СИ
8	поиск правой/левой серии логических «1»	СИ
9	преобразование «8-4-2-1 → нормализованный УК»	ЧИ
10	преобразование «нормализованный УК → 8-4-2-1»	ЧИ
11	правое дополнение серии логических «1»	СИ
12	левое дополнение серии логических «1»	СИ

**Методы решения.** Теоретической основой создания однородных преобразователей (формирователей) УК являются итерационные сети [19] и принципы синтеза типовых комбинационных схем [20]. Итерационная сеть – это однородная система вычислительных ячеек с регулярными связями близкодействия для передачи информационно-управляющих сигналов между ячейками. Итерационная сеть осуществляет прием входных операндов и выдачу выходных значений в параллельном коде, но с последовательным срабатыванием ячеек за счет связующей функции между ячейками.

Характеристиками итерационной сети являются:

- ◆ количество входных и выходного кодов;
- ◆ длины входного и выходного кодов;
- ◆ направление вычислений;
- ◆ мерность сети;
- ◆ количество соединений между ячейками.

Функциональные узлы на базе итерационных сетей разрабатываются в составе реконфигурируемых ВС, вычислительная структура которых соответствует информационному графу задачи. Известный в теории структурно-процедурных вычислений метод распараллеливания по итерациям позволяет строить конвейеризированные двумерные вычислительные структуры для распределения и потоковой обработки массивов данных (задачи сортировки, линейной алгебры, обработки разреженных матриц и др.) [21].

На рис. 1 и 2 показаны двумерная итерационная сеть с вычислительными ячейками двух типов, построенные по информационному графу задачи умножения вектора на матрицу, сортировки «пузырьком» и др. [22, 23].

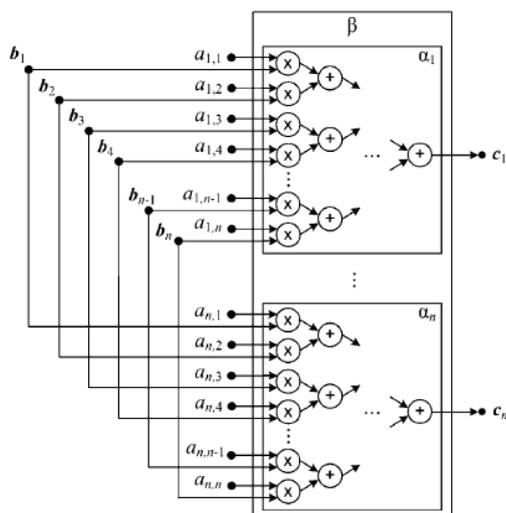


Рис. 1. Итерационная сеть умножения вектора на матрицу

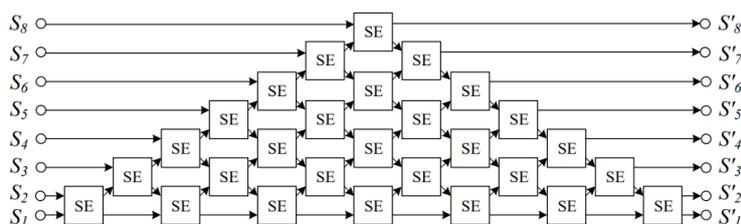


Рис. 2. Итерационная сеть сортировки «пузырьком»

Наибольшее распространение получили одномерные итерационные сети с одним или двумя входными кодами (операндами) и единственным выходным кодом с одним направлением передачи связей между ячейками. На рисунке 3 показан общий вид одномерной итерационной сети, перерабатывающей входной код  $X = x_1 x_2 \dots x_n$  в выходной код  $Y = y_1 y_2 \dots y_n$  с заданным направлением связи ячеек слева направо с помощью связующей функции  $V = v_1 v_2 \dots v_n v_{n+1}$ .

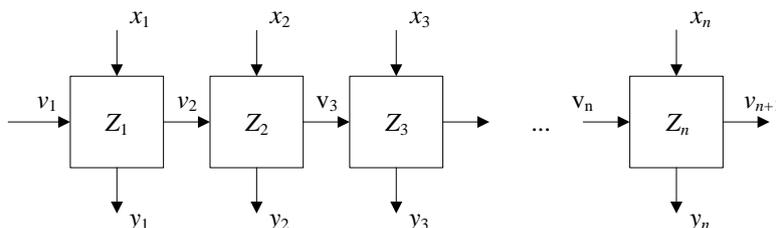


Рис. 3. Одномерная однонаправленная итерационная сеть

Время работы итерационной сети линейно зависит от длины кода  $n$  и составляет  $T = n t_{CELL}$ , где  $t_{CELL}$  – задержка одной ячейки. Динамика получения выходных разрядов в коде  $Y = y_1 y_2 \dots y_n$  описывается дискретными моментами времени  $t_{CELL}, 2t_{CELL}, \dots, n t_{CELL}$ . Следовательно, общее время работы итерационной сети определяется моментом времени срабатывания граничной ячейки  $Z_n$ .

В зависимости от решаемой задачи итерационная сеть может быть двунаправленной, задавая с двух сторон связующую функцию  $V = v_1 v_2 \dots v_n v_{n+1}$  (рис. 4).

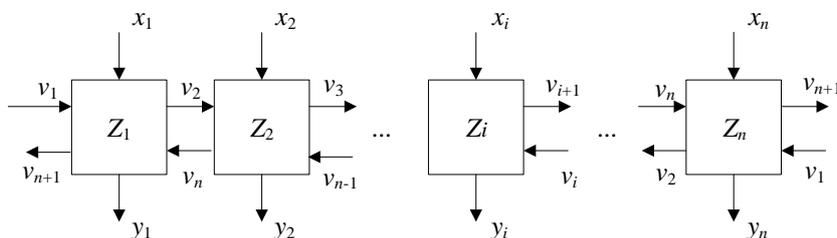


Рис. 4. Одномерная двунаправленной итерационная сеть

При такой организации итерационной сети итоговый выходной код  $Y = y_1 y_2 \dots y_n$  по всем разрядам формируется с двух направлений за время  $T = (n / 2) t_{CELL}$ . Последней вычисляется центральная ячейка итерационной сети  $Z_i$

Известным примером итерационной сети с двумя входными операндами является параллельный сумматор с последовательными переносами [20] между двоичными сумматорами.

Другими известными примерами итерационных сетей являются кодирующие или шифрующие схемы [24, 25], обработчики в конечных полях Галуа [26], тракующие входной код как строку элементов с их последовательным поэлементным преобразованием. Также известны схемотехнические решения функциональных узлов – схемных формирователей УК [27, 28] под различные прикладные задачи. Главная особенность созданных схемных формирователей – использование типовых комбинационных схем (дешифратор, мультиплексор и др.) при преобразовании УК.

Эффективная работа однородных ВС, в первую очередь, использует такие операции над УК (табл. 1) как

- ◆ нормализация унитарного кода;
- ◆ прямое преобразования стандартного весового кода 8-4-2-1 в нормализованный УК и обратное преобразование;
- ◆ формирование серии логических «1»/серии логических «0».

Под нормализованным УК (НУК) понимается двоичный код, в котором серия логических «1» выровнена по правой или левой границе кода.

Известна двумерная итерационная сеть для формирования нормализованного кода из исходного УК под названием цифровой компрессор [18]. Пример работы цифрового компрессора приведен в табл. 2, где показаны исходные УК и соответствующий им нормализованный УК.

Работа цифрового компрессора основана на последовательно-параллельных вычислительных процессах продвижения логических «1» по строкам и столбцам ячеек сети.

Таблица 2

**Пример работы цифрового компрессора**

1	0	0	1	0	0
0	1	0	1	1	0
0	0	1	0	1	1
1	1	1	0	0	1
Исходный унитарный код					Нормализованный унитарный код

Схема цифрового компрессора представляет собой двумерную итерационную сеть ячеек в форме прямоугольного треугольника. Цифровой компрессор состоит из  $(n-1)! = (n-1) \times (n-2) \times \dots \times 2$  ячеек, имеет один информационный  $n$  разрядный вход и один

информационный  $n$  разрядный выход. Каждая ячейка имеет 2 входа (первый горизонтальный и второй вертикальный входы) и 2 выхода (первый горизонтальный и второй вертикальный выходы).

Каждая текущая логическая «1» из входного кода стремится при отсутствии конфликта занять текущую строку цифрового компрессора и дойти до граничного элемента цифрового компрессора. При обнаружении конфликта между логической «1», идущей слева, и логической «1», расположенной в текущей позиции УК продвижение логической «1», идущей слева, продолжается вверх на следующую строку в цифровом компрессоре. Конфликт разрешается тем, что логическая «1», идущая слева, занимает выше расположенную строку в цифровом компрессоре с помощью двухвходового элемента И, а логическая «1», расположенная в текущей позиции УК, движется по текущей строке вправо в цифровом компрессоре с помощью двухвходового элемента ИЛИ (рис. 5).

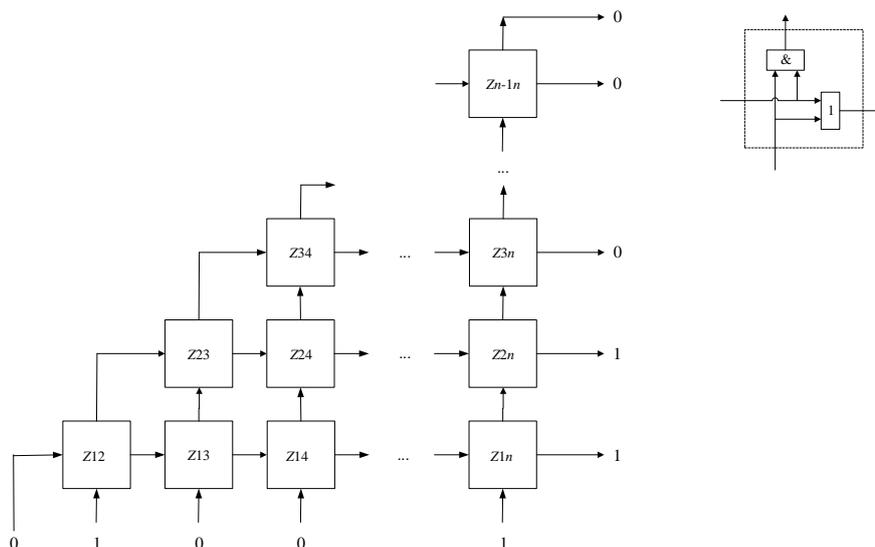


Рис. 5. Цифровой компрессор для формирования НУК

Схема прямого преобразования двоичного весового кода 8-4-2-1 в НУК имеет вид «8-4-2-1» → ЕПК → нормализованный УК» (рис. 6), где ЕПК – единичный позиционный код, формируемый дешифратором.

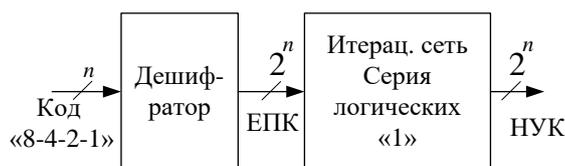


Рис. 6. Схема прямого преобразования в НУК

Одномерная итерационная сеть для формирования серии логических «1» из ЕПК позволяет получить НУК, соответствующий исходному весовому коду.

Схема обратного преобразования из УК в стандартный весовой код 8-4-2-1 имеет вид «УК → НУК → ЕПК → 8-4-2-1» (рис. 7).

Пусть задан унитарный код  $U = u_1 u_2 \dots u_n$  разрядностью  $n$  бит. Первая слева логическая «1» в составе УК является приоритетной логической «1». Арбитр как функциональный узел имеет вид согласно однонаправленной сети на рис. 3, где  $V = v_1 v_2 \dots v_n v_{n+1}$  – связующая функция для выделения приоритетной (первой слева) логической «1».

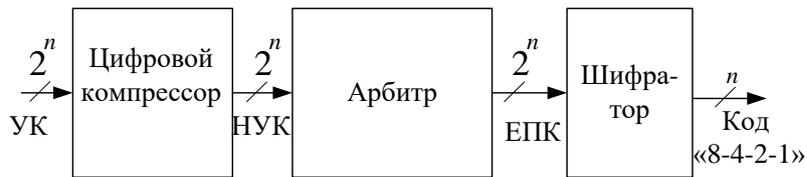


Рис. 7. Схема обратного преобразования УК

Ячейка арбитра представляет собой «черный» ящик, работа которого описывается таблицей истинности от 2 входных переменных ( $u_i, v_i$ ), вычисляющей 2 выходные переменные ( $y_i, v_{i+1}$ ). Работа арбитра состоит в получении из унитарного кода вида 0..1..1..0..1..0 выходного унитарного кода вида 0..1..0..0..0..0. Таблица истинности ячейки арбитра показана в табл. 3.

Таблица 3

Таблица истинности работы ячейки арбитра

$v_i$	$u_i$	$y_i$	$v_{i+1}$
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	1

Синтез выходных функций имеет следующий результат

$$y_i = \bar{v}_i \ \& \ u_i, \tag{1}$$

$$v_{i+1} = v_i \ \vee \ u_i. \tag{2}$$

Пусть также задан унитарный код  $U = u_1 u_2 \dots u_n$  разрядностью  $n$  бит. Требуется сформировать серию логических «1» от первой слева логической «1» в составе УК. Такая однонаправленная сеть как функциональный узел также имеет вид согласно рис. 3, где  $V = v_1 v_2 \dots v_n v_{n+1}$  – связующая функция для формирования серии логических «1».

Ячейка итерационной сети формирования серии логических «1» представляет собой «черный» ящик, работа которого описывается таблицей истинности от 2 входных переменных ( $u_i, v_i$ ), вычисляющей 2 выходные переменные ( $y_i, v_{i+1}$ ).

Работа итерационной сети формирования серии логических «1» состоит в получении из унитарного кода вида 0..1..1..0..1..0 выходного унитарного кода вида 0..1..1..1..1..1. Таблица истинности ячейки арбитра показана в табл. 4.

Таблица 4

Таблица истинности работы ячейки сети для формирования серии логических «1»

$v_i$	$u_i$	$y_i$	$v_{i+1}$
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	1

$$y_i = v_i \ \vee \ u_i, \tag{3}$$

$$v_{i+1} = v_i \ \vee \ u_i. \tag{4}$$

**Результаты и обсуждение.** На практике для интеллектуальной обработки УК достаточно важной операцией является операция приблизительного сравнения двух чисел «меньше или равно»/«больше или равно». В этом случае пороговое значение задает некий допустимый порог числа выполненных рабочих процессов, а входной УК содержит

фактическое количество выполненных рабочих процессов. Проблемная ситуация состоит в том, что пороговое значение имеет преимущественно представление в двоичном весовом коде 8-4-2-1. Традиционный подход состоит в приведении форматов данных к единому виду и последующем сравнении двух чисел в едином формате. Главная недостаток такого преобразования – избыточные затраты времени на прямое или обратное преобразования УК в весовой код 8-4-2-1. Для повышения производительности работы схемы приблизительного сравнения использованы возможности многовходового мультиплексора как универсального логического модуля [28] (рис. 8).

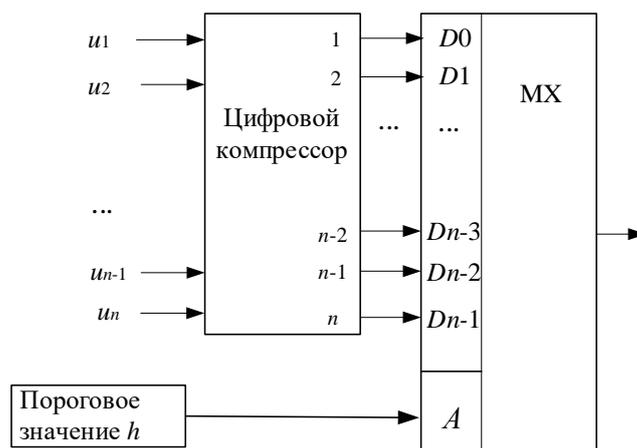


Рис. 8. Схема приблизительного сравнения УК и кода 8-4-2-1

Схема приблизительного сравнения работает следующим образом. Пороговое значение  $h$  хранит двоичный весовой код числа не совпавших элементов (если двоичный код равен нулю, то выполняется точное сравнение).

Цифровой компрессор формирует из входного  $n$ -разрядный унитарного кода нормализованный УК, который поступает по прямой схеме подключений на информационные входы многовходового мультиплексора MX. На адресный вход  $A$  многовходового мультиплексора поступает пороговое значение  $h$ . Многовходовой мультиплексор осуществляется выбор соответствующего бита из нормализованного УК. Если на выход мультиплексора в соответствии со значением порогового  $h$  значения поступает логическая «1», то на выходе схемы формируется положительный результат приблизительного поиска, так как число совпавших логических «1» в нормализованном унитарном коде больше порогового значения.

Пусть для  $n=8$  пороговое значение  $h=2$ , а нормализованный УК=0000 1111. Тогда обращение к информационному входу по адресу 2 подает на выход многовходового мультиплексора логическую «1», что соответствует положительному результату сравнения.

Пусть для  $n=8$  пороговое значение  $h=4$ , а нормализованный УК=00000111. Тогда обращение к информационному входу по адресу 4 подает на выход многовходового мультиплексора логический «0», что соответствует отрицательному результату сравнения.

На рис. 9 показаны время метода приблизительного сравнения с использованием мультиплексора (Метод 2) и стандартного метода с приведением к единому формату чисел (Метод 1) для переменной длины кода  $n=4, 8, 12, 16, 20, 24, 28, 32, 36$  бит, приняв следующие временные задержки (в условных единицах времени),  $\tau_{ЛЭ}=2, T_{compress}=2+(n-1), T_{net}=4+2(n+1)$ .

Анализ графиков на рис. 9 показывает, что эффективная область для созданного преобразователя – порогового элемента сравнения весовых и унитарных кодов начинается от 12 бит и составляет 14-16% выигрыша во времени в сравнении с традиционным преобразователем, основанным на приведении кодов к единому формату и выполнении всех вычислений на итерационных сетях.

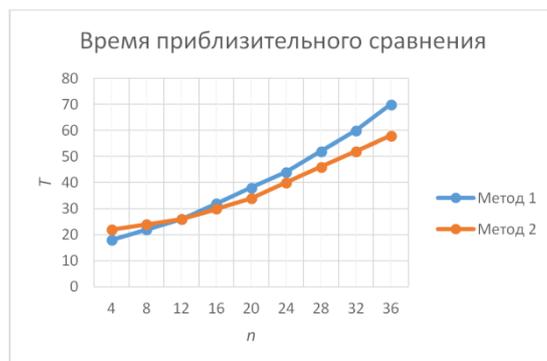


Рис. 9. Время приблизительного сравнения

Для последующего повышения эффективности работы одномерных итерационных сетей целесообразно использовать иерархический принцип обработки, состоящий в разбиении входного УК с единственной стартовой точкой на  $g$  подстрок с собственными стартовыми точками. Этот подход детально описан в [30, 31]. При формировании множества стартовых точек по входному УК возможно распараллеливание вычислений над самостоятельными подстроками, а число подстрок в составе УК определяется требованиями к временным и аппаратным затратам в однородной ВС.

**Заключение.** Преобразователи (формирователи) унитарных кодов относятся к классу комбинационных схем, принципы построения которых преимущественно проработаны для числового формата данных. Тем не менее унитарные коды достаточно часто представляю собой битовые строки, используемые в моделях обработки знаний и параллельных вычислениях, например в экспертных системах при анализе конфликтных множеств или результатов одновременного срабатывания нескольких правил, в крипто-, генетико-исследованиях и др. [32].

Для построения эффективных по времени схем преобразования унитарных кодов целесообразно использовать и развивать аппарат итерационных сетей, на основе которых возможно создание одномерных и двумерных сетей. Использование принципа иерархической обработки строки с несколькими стартовыми точками позволяет получить дополнительный временной выигрыш за счет организации множества параллельных локальных рабочих процессов, хотя не все операции на унитарным кодами допускают такое распараллеливание.

Достаточно перспективным способом построения нетрадиционных структур преобразователей УК для однородных ВС является использование дуализма в трактовке кода, а именно как числа или как строки. Рассмотренный значимый для практики пример приблизительного сравнения на унитарных кодах показал возможность совмещения символьных и числовых характеристик в одном типовом функциональном узле (мультиплексоре). Моделирование временных затрат показало, что с увеличением длины унитарного кода применение дуализма данных дает временной выигрыш около 14-16%.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Воеводин В.В. Математические модели и методы в параллельных процессах. – М.: Наука, 1986. – 296 с.
2. Гузик В.Ф., Каляев И.А., Левин И.И. Реконфигурируемые вычислительные системы. – Ростов-на-Дону: Южный федеральный университет, 2016. – 472 с.
3. Буцнев В.С. Параллелизм вычислительных процессов и развитие архитектуры суперЭВМ: Сб. статей / сост. В.П. Торчигин, Ю.Н. Никольская, Ю.В. Никитин. – М.: ТОРУС ПРЕСС, 2006. – 416 с.
4. Корнеев В.В. Вычислительные системы. – М.: Гелиос АРВ, 2004. – 510 с.
5. Voevodin V.V., Voevodin V.I. Parallel computing. – St. Petersburg, BHV-Peterburg Publ., 2002. – 608 p.
6. Огнев И.В., Борисов В.В., Стула Н.А. Ассоциативные память, среды, системы. – М.: Горячая линия – Телеком. 2016. – 420 с.

7. *Титенко Е.А., Типикин А.П., Лапин Д.В.* Некоторые пути построения перспективных вычислительных систем для параллельной обработки массивов данных и изображений на ПЛИС // Электромагнитные волны и электронные системы. – 2016. – Т. 21, № 10. – С. 56-59.
8. *Каляев И.А., Левин И.И.* Реконфигурируемые вычислительные системы на основе ПЛИС. – Ростов-на-Дону: Южный научный центр РАН, 2022. – 475 с.
9. *Кравец О.Я., Подвальный Е.С., Титов В.С., Ястребов А.С.* Архитектура вычислительных систем с элементами конвейерной обработки: учеб. пособие. – Воронеж, Курск, Санкт-Петербург: Политехника, 2009. – 151 с.
10. *Адамов А.А. Эйсымонт Л.К.* Варианты архитектурных решений ЭКБ для систем искусственного интеллекта // Проектирование будущего. Проблемы цифровой реальности: труды 3-й Международной конференции. – М.: ИПМ им. М.В. Келдыша, 2020. – С. 112-131.
11. *Гривачев А.В., Емельянов С.Г., Титенко Е.А.* Модифицированная производственная система для решения задачи структурного распознавания образов // Научно-технические ведомости СПбГПУ. – 2014. – Т. 15, № 12. – С. 9-12.
12. *Lothaire M.* Applied Combinatorics on Words. In: Encyclopedia of Mathematics and its Applications. – Cambridge: Cambridge University Press, 2005.
13. *Титенко Е.А., Довгаль В.М.* Концептуальный подход к разработке формальной исчислительной системы как генератора ветвящихся конструктивных процессов // Системы управления и информационные технологии. – 2006. – № 1-1 (23). – С. 185-187.
14. *Titenko E.A., Degtyarev S.V.* Approximate search in the sample on the basis Manber-Wu method // Journal of Fundamental and Applied Sciences. – 2017. – Vol. 9. No. 2. – P. 914.
15. *Гэри М., Джонсон Д.* Вычислительные машины и труднорешаемые задачи. – М.: Мир, 1982. – 416 с.
16. *Гладков Л.А., Курейчик В.В., Курейчик В.М., Сороколетов П.В.* Биоинспирированные методы в оптимизации. – М.: Физматлит, 2009. – 384 с.
17. *Озкарахан Э.* Машины баз данных и управление базами данных. – М.: Мир, 1989. – 696 с.
18. *Бандман О.Л., Миренков Н.Н., Седухин С.Г.* Специализированные процессоры для высокопроизводительной обработки данных. – М.: Радио и связь, 1988. – 208 с.
19. *Фет Я.И.* Параллельные процессоры для управляющих систем. – М.: Энергоиздат, 1981. – 160 с.
20. *Потемкин И.С.* Функциональные узлы цифровой автоматики. – М.: Энергоатомиздат, 1988. – 320 с.
21. *Левин И.И., Подпригора А.В.* Модифицированный метод обработки больших разреженных неструктурированных матриц на реконфигурируемых вычислительных системах // Вычислительные методы и программирование. – 2024. – Т. 25, № 2. – С. 142-154.
22. *Левин И.И., Алексеев К.Н.* Преобразование сортирующих сетей для разной степени параллелизма // Известия ЮФУ. Технические науки. – 2023. – № 5 (235). – С. 104-118.
23. *Каляев А.В., Левин И.И.* Модульно-наращиваемые многопроцессорные системы со структурно-процедурной организацией вычислений. – М.: Изд-во "Янус-К", 2003. – 380 с.
24. *Усатюк В.С., Егоров С.И., Ватутин Э.И., Чернецкая И.Е.* Обеспечение помехоустойчивости канала связи за счет применения метода поиска слов малого веса в линейном блочном двоичном и тернарном кодах // Тр. МАИ. – 2024. – № 138.
25. *Новиков А.О., Ватутин Э.И., Егоров С.И., Титов В.С.* Исследование алгоритма Даббагьяна-Ву для построения нециклических пандиагональных латинских квадратов // Известия ЮФУ. Технические науки. – 2024. – № 3 (239). – С. 126-137.
26. *Егоров С.И., Титенко Е.А.* Математические и вычислительные схемы реализации арифметических операций в конечных полях Галуа для подвижных роботов // Информационные системы и технологии. – 2023. – № 2 (136). – С. 14-24.
27. *Ватутин Э.И., Титов В.С.* Теоретические основы и технические решения программно-аппаратного обеспечения синтеза логических мультиконтроллеров. – Курск: ЗАО «Университетская книга», 2022. – 483 с.
28. *Ватутин Э.И., Зотов И.В., Титов В.С.* Использование схемных формирователей и преобразователей двоичных последовательностей при построении комбинаторно-логических акселераторов // Известия Курского государственного технического университета. – 2008. – № 4 (25). – С. 32-39.
29. *Зельдин Е.А.* Цифровые интегральные микросхемы в информационно-измерительной аппаратуре. – Л.: Энергоатомиздат, 1986. – 280 с.
30. *Титенко Е.А., Скорняков К.С., Бусыгин К.Н.* Методы и сумматоры с параллельными групповыми процессами // Известия Юго-Западного государственного университета. Серия: Управление, вычислительная техника, информатика. Медицинское приборостроение. – 2013. – № 1. – С. 161-166.

31. Титенко Е.А., Семенухин Е.А., Петрик Е.А., Воронин Д.А. Структурно-функциональная организация арбитра параллельной обработки запросов // Информационно-измерительные и управляющие системы. – 2010. – Т. 8, № 11.
32. Ханис А.Л., Беспалько С.В., Титенко Е.А. [и др.]. Обзор исследований киберстрахования // Интеллектуальные информационные системы: тенденции, проблемы, перспективы: Матер. докладов VII всероссийской очной научно-практической конференции «ИИС-2019», Курск, 25 ноября 2019 года. – Курск: Юго-Западный государственный университет, 2019. – С. 108-118.

## REFERENCES

1. Voevodin V.V. Matematicheskie modeli i metody v parallel'nykh protsessakh [Mathematical models and methods in parallel processes]. Moscow: Nauka, 1986, 296 p.
2. Guzik V.F., Kalyaev I.A., Levin I.I. Rekonfiguriruemye vychislitel'nye [Reconfigurable computing systems]. Rostov-on-Don: Yuzhnyy federal'nyy universitet, 2016, 472 p.
3. Burtsev V.S. Parallelizm vychislitel'nykh protsessov i razvitiye arkhitektury superEVM: Sb. statey [Parallelism of computing processes and development of supercomputer architectures: collection of articles], compilers V.P. Torchigin, Yu.N. Nikol'skaya, Yu.V. Nikitin. Moscow: TORUS PRESS, 2006, 416 p.
4. Korneev V.V. Vychislitel'nye sistemy [Computing systems]. Moscow: Gelios ARV, 2004, 510 p.
5. Voevodin V.V., Voevodin V.I. Parallel computing. Saint Petersburg, BHV-Peterburg Publ., 2002, 608 p.
6. Ognev I.V., Borisov V.V., Sutula N.A. Assotsiativnye pamyat', sredy, sistemy [Associative memory, environments, systems]. Moscow: Goryachaya liniya – Telekom. 2016, 420 p.
7. Titenko E.A., Tipikin A.P., Lapin D.V. Nekotorye puti postroeniya perspektivnykh vychislitel'nykh sistem dlya parallel'noy obrabotki massivov dannykh i izobrazheniy na PLIS [Some ways of promising computing systems for parallel processing of data arrays and images on FPGA], *Elektromagnitnye volny i elektronnye sistemy* [Electromagnetic waves and electronic systems], 2016, Vol. 21, No. 10, pp. 56-59.
8. Kalyaev I.A., Levin I.I. Rekonfiguriruemye vychislitel'nye sistemy na osnove PLIS [Reconfigurable computing systems based on FPGAs]. Rostov-on-Don: Yuzhnyy nauchnyy tsentr RAN, 2022, 475 p.
9. Kravets O.YA., Podval'nyy E.S., Titov V.S., Yastrebov A.S. Arkhitektura vychislitel'nykh sistem s elementami konveyernoy obrabotki: ucheb. posobie [Architecture of computing systems with elements of pipeline processing: textbook]. Voronezh, Kursk, Sankt-Peterburg: Politekhnik, 2009, 151 p.
10. Adamov A.A., Eysymont L.K. Varianty arkhitekturnykh resheniy EKB dlya sistem iskusstvennogo intellekta [Variants of architectural solutions for an electronic component base for artificial intelligence systems], *Proektirovaniye budushchego. Problemy tsifrovoy real'nosti: trudy 3-y Mezhdunarodnoy konferentsii* [Designing the Future. Problems of Digital Reality: Proceedings of the 3rd International Conference]. Moscow: IPM im. M.V. Keldysha, 2020, pp. 112-131.
11. Grivachev A.V., Emel'yanov S.G., Titenko E.A. Modifitsirovannaya produktsionnaya sistema dlya resheniya zadachi strukturnogo raspoznavaniya obrazov [Modified production system for solving the problem of structural pattern recognition], *Naukoemkie tekhnologii* [Science-intensive technologies], 2014, Vol. 15, No. № 12, pp. 9-12.
12. Lothaire M. Applied Combinatorics on Words. In: Encyclopedia of Mathematics and its Applications. Cambridge: Cambridge University Press, 2005.
13. Titenko E.A., Dovgal' V.M. Kontseptual'nyy podkhod k razrabotke formal'noy vychislitel'noy sistemy kak generatora vetyashchikhsya konstruktivnykh protsessov [Conceptual approach to the development of a formal enumeration system as a generator of branching constructive processes], *Sistemy upravleniya i informatsionnye tekhnologii* [Control systems and information technologies], 2006, No. 1-1 (23), pp. 185-187.
14. Titenko E.A., Degtyarev S.V. Approximate search in the sample on the basis Manber-Wu method, *Journal of Fundamental and Applied Sciences*, 2017, Vol. 9. No. 2, pp. 914.
15. Geri M., Dzhonson D. Vychislitel'nye mashiny i trudnoreshaemye zadachi [Computing machines and intractable problems]. Moscow: Mir, 1982, 416 p.
16. Gladkov L.A., Kureychik V.V., Kureychik V.M., Sorokoletov P.V. Bioinspirirovannyye metody v optimizatsii [Bioinspired methods in optimization]. Moscow: Fizmatlit, 2009, 384 p.
17. Ozkarakhan E. Mashiny baz dannykh i upravlenie bazami dannykh [Database machines and database management]. Moscow: Mir, 1989, 696 p.
18. Bandman O.L., Mirenkov N.N., Sedukhin S.G. Spetsializirovannyye protsessory dlya vysokoproizvoditel'noy obrabotki dannykh [Specialized processors for high-performance data processing]. Moscow: Radio i svyaz', 1988, 208 p.
19. Fet Ya.I. Parallelnyye protsessory dlya upravlyayushchikh system [Parallel processors for control systems]. Moscow: Energoizdat, 1981, 160 p.

20. *Potemkin I.S.* Funktsional'nye uzly tsifrovoy avtomatiki [Functional units of digital automation]. Moscow: Energoatomizdat, 1988, 320 p.
21. *Levin I.I., Podoprigora A.V.* Modifitsirovanny metod obrabotki bol'shikh razrezhennykh nestrukturirovannykh matrits na rekonfiguriruemykh vychislitel'nykh sistemakh [Modified method for processing large sparse unstructured matrices on reconfigurable computing systems], *Vychislitel'nye metody i programmirovaniye* [Computational methods and programming], 2024, Vol. 25, No. 2, pp. 142-154.
22. *Levin I.I., Alekseev K.N.* Preobrazovanie sortiruyushchikh setey dlya raznoy stepeni parallelizma [Transformation of sorting networks for different degrees of parallelism], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2023, No. 5 (235), pp. 104-118.
23. *Kalyaev A.V., Levin I.I.* Modul'no-narashchivaemye mnogoprotsessornye sistemy so strukturno-protsedurnoy organizatsiyey vychisleniy [Modularly scalable multiprocessor systems with structural-procedural organization of computations]. Moscow: Izd-vo "Yanus-K", 2003, 380 p.
24. *Usatyuk V.S., Egorov S.I., Vatutin E.I., Chernetskaya I.E.* Obespechenie pomekhoustoychivosti kanala svyazi za schet primeneniya metoda poiska slov malogo vesa v lineynom blochnom dvoichnom i ternarnom kodakh [Ensuring noise immunity of the communication channel due to the use of the method of searching for small-weight words in linear block binary and ternary codes], *Tr. MAI* [Proceedings of MAI], 2024, No. 138.
25. *Novikov A.O., Vatutin E.I., Egorov S.I., Titov V.S.* Issledovanie algoritma Dabbaghyana-Vu dlya postroeniya netsiklicheskiykh pandiagonal'nykh latinskikh kvadratov [Study of the Dabbaghyan-Wu algorithm for constructing non-cyclic pandiagonal Latin squares], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2024, No. 3 (239), pp. 126-137.
26. *Egorov S.I., Titenko E.A.* Matematicheskie i vychislitel'nye skhemy realizatsii arifmeticheskikh operatsiy v konechnykh polyakh Galua dlya podvizhnykh robotov [Mathematical and computational schemes for implementing arithmetic operations in finite Galois fields for mobile robots], *Informatsionnye sistemy i tekhnologii* [Information systems and technologies], 2023, No. 2 (136), pp. 14-24.
27. *Vatutin E.I., Titov V.S.* Teoreticheskie osnovy i tekhnicheskie resheniya programmno-apparatnogo obespecheniya sinteza logicheskikh mul'tikontrollerov [Theoretical foundations and technical solutions for software and hardware support for the synthesis of logical multicontrollers]. Kursk: ZAO «Universitetskaya kniga», 2022, 483 p.
28. *Vatutin E.I., Zotov I.V., Titov V.S.* Ispol'zovanie skhemnykh formirovateley i preobrazovateley dvoichnykh posledovatel'nostey pri postroenii kombinatorno-logicheskikh akseleratorov [Using circuit shapers and converters of binary sequences in the construction of combinatorial logic accelerators], *Izvestiya Kurskogo gosudarstvennogo tekhnicheskogo universiteta* [Bulletin of the Kursk State Technical University], 2008, No. 4 (25), pp. 32-39.
29. *Zel'din E.A.* Tsifrovye integral'nye mikroskhemy v informatsionno-izmeritel'noy apparature [Digital integrated circuits in information and measuring equipment]. Leningrad: Energoatomizdat, 1986, 280 p.
30. *Titenko E.A., Skornyakov K.S., Busygin K.N.* Metody i summatory s parallelnymi gruppovymi protsessami [Methods and adders with parallel group processes], *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta. Seriya: Upravlenie, vychislitel'naya tekhnika, informatika. Meditsinskoe priborostroenie* [Bulletin of the South-West State University. Series: Management, computing, informatics. Medical instrument making], 2013, No. 1, pp. 161-166.
31. *Titenko E.A., Semenikhin E.A., Petrik E.A., Voronin D.A.* Strukturno-funktsional'naya organizatsiya arbitra parallelnoy obrabotki zaprosov [Structural and functional organization of the arbitrator of parallel query processing], *Informatsionno-izmeritel'nye i upravlyayushchie sistemy* [Information, measuring and control systems], 2010, Vol. 8, No. 11.
32. *Khanis A.L., Bepal'ko S.V., Titenko E.A. [i dr.].* Obzor issledovaniy kiberstrakhovaniya [Review of cyber insurance research], *Intellektual'nye informatsionnye sistemy: tendentsii, problemy, perspektivy: Mater. dokladov VII vserossiyskoy ochnoy nauchno-prakticheskoy konferentsii «IIS-2019»*, Kursk, 25 noyabrya 2019 goda [// Intelligent information systems: trends, problems, prospects: materials of reports of the VII All-Russian face-to-face scientific and practical conference "IIS-2019", Kursk, November 25, 2019]. Kursk: Yugo-Zapadnyy gosudarstvennyy universitet, 2019, pp. 108-118.

**Титенко Евгений Анатольевич** – Юго-Западный государственный университет; e-mail: johntit@mail.ru; г. Курск, Россия; тел.: +79051588904; к.т.н., доцент; доцент кафедры программной инженерии.

**Titenko Evgeny Anatolievich** – South-West State University; e-mail: johntit@mail.ru; Kursk, Russia; phone: +79051588904; cand. of eng. sc.; associate professor; associate professor of the Software Engineering Department.