Борисенко А.А.

СИСТЕМЫ СЧИСЛЕНИЯ В ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКЕ

Аннотация

В данной работе рассмотрены общие подходы к разъяснению понятия числа и систем счисления, а также к практическому применению современных систем счисления в разных науках. Среди этих наук находится теория кодирования, теория чисел, теория цифровых автоматов. Последняя из этих наук важна для разработки более быстродействующих и надежных систем электронной вычислительной техники. Показано место, которое занимают современные позиционные системы счисления в дискретной математике при построении математических структур.

Содержание

1. Введение

В основе современной электронной вычислительной техники лежат числа и системы счисления, которые эти числа порождают. От эффективности последних зависят параметры вычислительных систем и устройств, в первую очередь показатели быстродействия и надежности. Среди систем счисления наибольшее распространение в вычислительной технике нашла двоичная система счисления. Эта система в силу своей простоты, выражающейся в нулевой сложности ее структуры, обеспечивает необходимый уровень основных параметров этой техники и пока что находится вне конкуренции. Однако сегодня обстановка в вычислительной технике вследствие значительных технологических достижений в области производства интегральных схем начинает радикально меняться, так как эти технологии позволяют в одной микросхеме размещать миллионы логических элементов, снизив тем самым в разы стоимость интегральных микросхем и повысив, при этом, их быстродействие. Отказоустойчивость этих микросхем, по крайней мере, на сегодня, даже возросла, что нельзя сказать об их устойчивости к помехам, вызываемых сбоями в работе.

Сбои – это достаточно распространенное явление в вычислительной технике и новые технологии не решают кардинально вопрос помехоустойчивости, а наоборот, иногда, в силу большой плотности логических элементов на одной подложке интегральной схемы и снижения напряжения питания для увеличения ее быстродействия, даже усугубляют его. Чтобы убедиться, что это действительно так, достаточно хотя бы вспомнить нередкие зависания компьютеров. Но компьютеры - это лишь видимая вершина айсберга в многочисленных применениях вычислительной техники и цифровых устройств. Их зависания, хотя и неприятны, но можно пережить. Значительно хуже дело обстоит с вычислительными системами и устройствами, где ошибки не допустимы, и тем более с техникой, работающей в реальном масштабе времени, где не остается времени на повторную операцию вычисления или передачу информации. Там ошибку необходимо не только обнаружить, а еще в тот же момент, когда она обнаружена, и исправить, то есть в данном случае речь идет о технике способной не только обнаруживать помехи, а и их исправлять.

Двоичная же система счисления, в силу своей предельной простоты, не обладает внутренней (естественной) избыточностью информации в своей структуре и, как следствие, не способна решать других задач, кроме задач арифметико-логических. Тем более она не может без посторонней помощи, которая проявляется во введении в нее внешней искусственной избыточной информации, обнаруживать и исправлять ошибки в своей работе. А такая внешняя избыточность, которая на сегодня широко используется в вычислительной технике, приводит к значительному росту аппаратурных затрат и соответственно к снижению надежности и быстродействия использующих ее вычислительных устройств. Поэтому этот хорошо отработанный путь повышения помехоустойчивости и отказоустойчивости вычислительной техники в какой-то степени в перспективе является тупиковым.

Однако существуют системы счисления и с более сложной структурой, чем двоичная система, – интеллектуальные (структурные), которые в силу не равной нулю сложности их структур способны за счет *внутренне* присущей им естественной избыточности информации обнаруживать и исправлять ошибки в своей работе. То есть такие системы счисления изначально, по своей природе, обладают свойствами помехоустойчивости и самоконтроля [1,2]. Кроме обычных арифметико-логических функций и защиты от помех такие системы счисления способны решать и другие более сложные задачи как, например, сжатия и защиты информации от несанкционированного доступа, порождения и перебора комбинаторных объектов, решения задач комбинаторной оптимизации и другие. Важно также и то, что эти системы счисления указанные задачи решают в аппаратном исполнении, что уже само по себе повышает надежность и быстродействие их работы, причем в разы. Но, кроме этого, быстродействие в них растет и за счет использования более простых алгоритмов работы по сравнению с алгоритмами, использующими двоичные системы счисления.

Имеется также возможность на основе интеллектуальных систем счисления строить отказоустойчивые вычислительные системы и устройства и все это за счет естественной избыточности, которая имеется в их структурах. Очевидный недостаток таких систем счисления – это повышенная сложность и соответственно избыточное по сравнению с двоичной системой счисления количество требуемых для их реализации аппаратурных затрат. Но, как уже отмечалось выше, сегодня стоимость интегральных микросхем растет значительно медленней, чем сложность реализуемых ими алгоритмов. Поэтому такой рост следует считать экономически оправданным.

Исходя из вышесказанного, можно уверенно утверждать, что при разработке новых видов вычислительной техники и новых информационных технологий нужно использовать наряду с традиционными позиционными системами счисления и уже имеющиеся наработки в области интеллектуальных систем счисления. Также необходимо проводить поиск новых таких систем счисления и использовать их преимущества по сравнению с другими системами счисления, которые на данное время уже применяются на практике и, в частности, с двоичной системой.

Но чтобы решать указанные задачи важно выяснить, что же в самом общем виде представляют собой число и система счисления. Это не такая уж и простая задача, как может показаться на первый взгляд, и, конечно, она не может быть полностью решена в данной работе, но все же некоторые аспекты ее решения, рассматриваемые ниже, будут полезны для окончательного решения поставленной задачи в дальнейшем.

2. Определение числа

В основе понятия числа лежит абстракция в виде *количества* элементов конечного множества, которое по своей сущности не имеет формы. Однако число обладает большей общностью, чем количество, поскольку, кроме того, что несет в себе информацию о нем, еще и придает ему специфическую *форму* в виде кодового *изображения*. Именно наличие этого изображения позволяет выполнять над количеством разные арифметические и логические операции.

При этом следует обратить особое внимание на то, что одно и то же количество может изображаться по-разному. Например, десятичное число 123 может иметь еще изображение, в форме двоичного числа 1111011. Таких последовательностей, которые кодируют одно и то же количество, но имеют разный вид, может быть неограниченно много. Поэтому нужно отличать количество элементов, кодируемое числом, от непосредственного изображения, которое формирует его в виде числа. От изображения или формы числа в большой степени зависит эффективность выполнения арифметических и логических операций над ним.

Однако и форме числа могут отвечать разные количества. Например, пятеричное число 123 кодирует совсем другое количество, чем такое же за видом десятичное число 123. В первом случае пятеричное число 123 кодирует в десятичной системе счисления количество 38, а во втором - 123. Более того, последовательность знаков 123 может рассматриваться даже не как число, а как перестановка из трех элементов, и тогда эта последовательность элементов не кодирует никакого количества вообще .

Как выходит из сказанного выше, с одной стороны под числом можно понимать кодовое изображение количества, а из второй - само количество. Так число и воспринимается в данной работе, то есть оно рассматривается, или же, как количество, или же, как изображение этого количества в виде какой-то из его форм, или же, как это и другое понятие совместно. Понятие числа, которое используется в данной работе, будет видно из контекста. Оно зависит от акцентирования материала, который подается.

В соответствии с двумя вышеназванными функциями числа, с одной стороны как носителя количества, а с другой как ее изображения, существуют две связанные с ними теории. Первая изучает свойства чисел безотносительно к форме их представления. Этим занимается такая непростая наука как теория *чисел*. Вторая теория находит способы эффективного формирования чисел путем их кодирования, то есть придания им формы. Это уже будет другая наука – теория *кодирования* чисел.

Важную роль при кодировании чисел играет их особый вид, который называется *цифрами*. Цифры – это такие специфические числа, из которых складываются как угодно большие другие числа. Порядок размещения цифр в числах может быть разный. Чаще всего это будет линейная форма, когда цифры идут последовательно одна за другой, но могут быть и размещение цифр на плоскости в виде матриц. В тех или иных размещениях цифр проявляется способ кодирования чисел, от которого во многих случаях зависит эффективность их использования на практике.

3. Количественные и порядковые функции чисел

Число, кроме количества, также используется для определения порядка, который складывается между элементами множеств. Такое число называется еще *номером*. Как правило, это будет целое неотрицательное число. Но, как уже говорилось выше, число кодирует также и количество. То есть в общем плане за своими функциями число должно давать возможность получать ответ на вопрос, как о количестве элементов во множествах, так и о порядке их размещения в них. В первом случае ответ будет иметь вид - один, два, три ..., а во втором - первый, второй, третий … .

На ответе к первому из приведенных выше вопросов базируются *количественные* операции над элементами множеств, а на ответе ко второму – разные способы упорядочения этих элементов. Количественные операции изучает теория *кардинальных* чисел, а порядковые - теория *порядковых* чисел. Эти теории очень важны для понимания природы числа, которая до сегодняшнего дня, еще не совсем определена, и поэтому нуждается в дальнейших исследованиях.

Способы кодирования чисел не зависят от того, что определяется, – количество или порядок элементов во множествах, то есть по виду чисел можно как устанавливать количество элементов во множествах, так и номер элементов в них по порядку. Например, число 120 может определять количество - сто двадцать, а может и порядок - сто двадцатый. Другими словами свойства количества и порядка, как уже отмечалось это и ранее, отличаются между собой. Ответ на количество элементов в множестве дает *количественное* число, а на порядок размещения элементов в нем – *порядковое* число, то есть его порядковый номер.

Только числа, которые владеют указанными двумя функциями, могут считаться достаточно эффективными при их использовании на практике, потому что с помощью этих функций можно получить ответы на вопросы: сколько элементов содержит множество, и какое место в нем по порядку занимает тот или иной элемент? В последнем случае решается задача *нумерации* чисел.

При этом номерами, обычно, называются числа, которые не только решают вопросы нумерации элементов, но и требуют для своего построения *минимального* числа знаков. То есть не каждое число можно считать номером, а только то, которое несет в себе максимальное количество информации, а значит, не является избыточным. Поэтому задача нумерации относится не только к задаче определения порядка, а также и к задаче сжатия информации.

4. Натуральные числа

Количественное число характеризует некоторый класс конечных множеств, то есть множеств, в которых между их элементами можно установить взаимно однозначное соответствие [3,4]. Такими множествами, соответствующими числу 5, например, могут быть множество пальцев руки и множество букв слова „книга”, потому что каждому пальцу может быть поставлена в соответствие одна из букв слова „книга”. Очевидно, что числу пять может быть поставлено в соответствие очень большое количество разных множеств и потому оно выражает то общее, что имеют эти множества между собой, - количество элементов, которое равняется пяти. То есть число пять создает класс множеств, каждое из которых содержит ровно пять элементов.

Такие же классы создают и другие числа - 1, 2 ... . Приведенная бесконечная упорядоченная дискретная последовательность чисел носит название *натурального* ряда чисел. Если конкретное число в ряду задается количеством элементов в соответствующем ему множестве элементов, то это будет *количественное* натуральное число, а если порядковым номером элемента, то в результате образуется *порядковое* натуральное число. Эта последовательность дискретна, потому что между числами натурального ряда не существует промежуточных элементов. Например, между числами 2 и 3 нет такого элемента, как, например, 2,5. В этом ряду нет также наибольшего числа, потому что, какое бы не взяли натуральное число, за ним будет идти на единицу большее другое следующее по порядку натуральное число, которое соответствует предыдущему множеству элементов с еще присоединенным к нему одним элементом.

5. Нуль и единица в ряду натуральных чисел

Следует отдельно отметить, что натуральный ряд начинается с единицы, а не нуля. Нуль происходит от латинского слова пустой (nulus), то есть отображает пустое множество. Нуль не имеет всех свойства натуральных чисел и в то же время имеет специфические свойства, которых не имеют натуральные числа. Например, если умножить любое число на нуль, то получим нуль, то есть одно и то же число. Для других натуральных чисел - это не так.

Хотя, правда, и единица, с которой начинается натуральный ряд, имеет некоторые свойства, отличающиеся от других свойств остальных чисел натурального ряда. Например, если умножить единицу на любое число, то получим, то же число. Для других чисел это не так. Поэтому не случайно выдающийся философ и математик семнадцатого века Лейбниц считал единицу и нуль особенными числами. Единицу он считал божественным началом и соответственно началом всего сущего, а нуль – небытием всего. Но все же единица входит в натуральный ряд, а нуль нет. Это связано с историей создания этого ряда, который до получения современного вида, как бесконечной последовательности целых чисел, создавался не одну тысячу лет.

При создании натурального ряда рассматривались только множества, которые не были пустыми, и поэтому не было необходимости в нуле, который бы отображал количество элементов в таком множестве. В соответствии с этим счет начинался с единицы, а не нуля. Потребность в нуле появилась только тогда, когда абстрактное мышление людей дошло до такой степени, при которой возникла потребность в пустом множестве. При его появлении необходимо было обозначить количество элементов в нем. Таким обозначением был нуль. А потребность в нуле появилась особенно тогда, когда наука счета пришла к *позиционному* счету, на что понадобились тысячи лет. Древние математики, несмотря на их могучий интеллект, не владели кажущимся таким простым для нас позиционным счетом и считали в непозиционных системах счисления, преодолевая при этом огромные трудности такого счета.

Множество, которое образуется присоединением к множеству натуральных чисел нуля, называют множеством целых *неотрицательных* чисел. Нуль в этом множестве ставится перед 1, то есть, имеем упорядоченное множество неотрицательных чисел – 0,1,2,3 ... . Множество этих чисел имеет все свойства натурального ряда чисел, то есть оно упорядочено, дискретно и бесконечно, но отличается от него тем, что начинается с нуля, а не единицы.

6. Аксиома счета

Указанные выше теории кардинальных и порядковых чисел используются не только для кодирования чисел, но еще и для *счета*, путем последовательного перебора элементов в множествах.

Счет играет особую роль при кодировании чисел, потому что с помощью счета можно одновременно определить как порядок расположения элементов во множествах, так и их количество. При этом результат счета не зависит ни от порядка, в котором он проходит, ни от системы нумерации элементов. Важно лишь то, чтобы при счете не было пропущено ни одного элемента, и чтобы каждый элемент был посчитан один раз. Только тогда может быть найден порядковый номер каждого элемента во множестве и найдено их общее количество. Это условие счета известно как *аксиома* счета [3].

7. Понятие о системах счисления

Как уже указывалось ранее, число, определенное как количество, является абстрактным понятием и не зависит от способов его изображения – кодирования, но сама техника исполнения над ним арифметических и логических операций существенно зависит от этих способов. Эти способы реализуются с помощью систем *счисления*, или, как их еще называют, систем *нумераций*.

Понятие числа неотъемлемо связанное с *системами счисление*, которые в своем развитии прошли нелегкий путь от самых простых непозиционных систем через пальцевой счет к современным позиционным системам счисления со сложной структурой.

Системой счисления или нумерацией называется совокупность правил и знаков, с помощью которых можно изобразить (кодировать) любое неотрицательное число.

К системам счисления выдвигается ряд требований, среди которых наиболее важными являются требования *однозначного* кодирования чисел 0,1 ... из некоторого их конечного множества - *диапазона P* за *конечное* число шагов, и возможность выполнения над кодами чисел *арифметических* и *логических* операций. Кроме того, системы счисления решают еще задачу *нумерации*, то есть задачу эффективного перехода от изображений чисел к их номерам. При ее решении проходит *сжатие* информации, что позволяет находить решение ряда как теоретических, так и практических задач. От удачного или неудачного выбора системы счисление зависит эффективность решений ею указанных выше задач.

При решении задачи кодирования чисел может возникнуть необходимость иметь такие их изображения, которые решают не только арифметико-логические задачи, а и некоторые вспомогательные задачи, например, решают задачу повышения помехоустойчивости обработки чисел или задачу построения дискретных математических объектов. Тогда требуется переход от номеров обычных естественных систем счисления с простейшей структурой к более сложным системам счисления с естественной избыточностью, или, при решении дискретных задач, к системам счисления, структура которых совпадает со структурой формируемых ими объектов .

При построении изображений чисел и при их нумерации системы счисления в общем плане решают задачу *кодирования,* и поэтому теорию их построения следует отнести к теории кодирования. Однако особенность систем счисления, которая позволяет выполнять ими арифметико-логические операции, приводит к необходимости исследовать их также и исходя из теории чисел.

То есть теория систем счисления граничит между теорией кодирования и теорией чисел. В этом проявляется ее особенность и специфика, хотя в большей мере эта теория все же относится к теории кодировки. Исходя из такого своего состояния, теория систем счисления должна рассматриваться как самостоятельная наука и поэтому иметь свой математический аппарат и свои только ей присущие задачи, как практического кодирования чисел, так и их теоретического анализа.

На сегодня иногда считается, что на основные практические вопросы построения систем счисления найдены ответы, потому что на практике хорошо себя зарекомендовали имеющиеся простейшие системы счисления типа двоичной. Отсюда выходит, что в настоящее время потребность в дальнейшем развитии теории систем счисления отсутствует. Но такой подход имеет свои недостатки, поскольку их развитие имеет в себе еще большой и не использованный на практике позитивный потенциал.

Это подтверждает и то, что время от времени появляются новые системы счисления, обладающие регулярной структурой, с новыми свойствами и возможностями, которые можно использовать в тех или других сферах науки и практики. Среди таких достаточно сложных систем счисления находятся системы остаточных классов, факториальные, полиадические, фибоначчиевы и много других. Да, они не настолько экономны и просты с точки зрения количества элементов в их структурах, как, например, десятичная или двоичная системы, но они имеют свои позитивные специфические свойства, которые следует использовать на практике. Это и возможность генерации различных комбинаторных объектов, таких, например, как перестановки с самыми разными ограничениями на них, так и их помехоустойчивость, и другие более специфические свойства, которые проявляются при конкретных приложениях. Кроме того, в ряде случаев с помощью таких систем счисления можно поднять быстродействие вычислительных устройств и систем к величинам, которые недосягаемые при использовании обычных традиционных позиционных систем счисления.

Эти системы не являются, пока что, конкурентами традиционных систем счисления, как иногда это может показаться, хотя бы потому, что последние по своей универсальности и простоте выполнения арифметических и логических операций недосягаемые для любых других систем счисления с более сложной структурой. Однако более сложные позиционные системы счисления с естественной избыточностью являются эффективным дополнением к обычным универсальным системам, потому что они могут совместно с последними системами эффективно работать при решении одних и тех же задач. При этом увеличивается надежность и быстродействие устройств и систем, которые их используют. Из этого выходит, что не совсем правильным является противопоставление обычных простых и новых более сложных систем счисления. Для любой системы счисления с регулярной структурой, какой бы сложной она не была, могут найтись особенные условия, где она сможет зарекомендовать себя с лучшей стороны. Хотя в перспективе такие системы счисления как фибоначчиевые вполне могут составить достойную конкуренцию сначала при построении вычислительных систем специального назначения, где требуется высокая помехоустойчивость, а далее и для решения обычных задач. Особенностью фибоначчиевых систем счисления является удачное сочетание простоты выполнения арифметико-логических операций и наличие естественной избыточности за счет усложнения структуры, достаточной для обнаружения ошибок, а при необходимости и их оперативного исправления. Это можно сказать далеко не обо всех системах счисления со сложной структурой. Например, биномиальные системы счисления, хотя и могут обнаруживать и исправлять ошибки в своей работе, имеют слишком сложные структуры, чтобы их эффективно применить в качестве систем счисления для универсальных вычислителей, зато они могут решать сложные специальные задачи, такие, например, как сжатие и защита информации. Причем, это они могут делать с повышенным быстродействием и высокой надежностью.

8. Позиционные числа

Для позиционных систем счисления важным является принцип ***позиционности,*** который заключается в изменениях количественного значения цифр в зависимости от их позиций. Эти позиции определяются *разрядами* числа. Каждая цифра числа имеет свой разряд, в котором она находится и который отличается от других разрядов своим *номером*. Принцип позиционности позволяет легко с помощью сравнительно небольшого количества цифр и позиций, какие они занимают, изобразить любое большое натуральное число. Это важное свойство позиционных систем счисления, которое широко используется на практике.

Количество разрядов позиционного числа характеризует его *длину*. В большинстве случаев числа одной и той же системы счисления имеют одинаковую длину. Поэтому они легко различаются между собой. Такие числа называются *равномерными.* Однако эта длина и соответствующее ей количество разрядов в общем случае для разных чисел одной и той же позиционной системы счисления могут быть разными. Такие числа с неравномерной длиной называются *неравномерными.* Это приводит к тому, что возникает задача их распознавания. Эта задача решается путем изображения чисел в виде *префиксных* кодов, то есть кодов, в которых каждое кодовое слово не является началом другого кодового слова этого кода. Так, кодовые слова, представляющие собой числа, 000, 0010, 01100 создают префиксный код, а слова или числа кода 00, 001, 0101 не создают его, поскольку слово 00 является началом слова этого кода – 001 и потому не может быть выделено как самостоятельное слово.

Для равномерных чисел задача распознавания имеет значительно более простой вид, поскольку ее решение происходит путем подсчета количества цифр в числе. Если это количество будет равняться длине, которая для всех чисел одинаковая, то оно считается найденным. Далее лишь необходимо будет выделить его из других чисел такой же длины. Например, равномерные числа 00, 01, 10 и 11 легко распознаются, поскольку имеют одинаковую длину, которая состоит из двух цифр. При наличии в числе в процессе его формирования двух цифр оно считается сформированным.

9. Структурные элементы позиционных систем счисления

Любая *система* счисления, в том числе и позиционная, в общем виде в своем составе должна иметь конечное множество неотрицательных чисел – *диапазон*, который она кодирует. В него обязательно в позиционных системах счисления входит число 0 и дальше числа начала натурального ряда, которые начинаются с 1. Кроме того, в позиционную систему счисления входит еще *нумерационная* или *числовая,* функция, с помощью которой эти числа нумеруются, а также *алфавит* – конечное множество цифр, из которых состоят кодовые изображения чисел, и *ограничения* на значение цифр.

Для цифр алфавита характерным признаком является то, что они начинаются с нуля и дальше идут как ряд натуральных чисел, то есть цифры алфавита имеют порядок 0, 1 ... . Среди этих цифр не может хотя бы одна быть пропущенной. Например, для наиболее простой и распространенной на сегодня в человеческом обществе позиционной системы счисления – десятичной алфавит имеет 10 цифр – 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, среди которых обязательно будет нуль. Только в этом случае, как показывает теория обобщенных позиционных систем счисления, может быть произведено кодирование всех чисел начальной части (определяемой диапазоном) натурального ряда. Наличие нуля в алфавите и диапазоне (коде) чисел является необходимым условием построения системы счисления. Если какой-то код не содержит символа, обозначающего нуль, то можно уверенно утверждать, что этот код не является позиционной системой счисления.

Число этих десяти цифр для десятичной системы счисления, называемой также *естественной* или *степенной* системой счисления имеет название *основания.* Таким же основанием будут и числа 2, 3, 4, 5 ... , задающие число цифр в алфавитах двоичной, троичной, четверичной, пятеричной и так далее системах счисления. Такие системы счисления определены автором как *однородные* системы счисления. Их еще называют *степенными* или *естественными* системами счисления. Основание задает количество цифр с учетом нуля, которые имеет алфавит той или иной системы счисления. Поэтому максимальная цифра в алфавите десятичной системы счисления и ей подобных остальных естественных систем счисления будет на единицу меньше ее основания

Интересно, что в непозиционных системах исчисления нуль отсутствует. Чтобы понять значение нуля и соответствующего ему пустого множества, нужно было изобрести позиционные системы исчисления, а для этого должны были пройти тысячи лет. Позиционные системы счисления и счет на их основе невозможны без нуля. Поэтому нуль или ничто является одной из фундаментальных основ современного позиционного счета и осуществляющего его позиционных систем счисления. Другими словами ничто, как это не удивительно, породило современную цивилизацию и компьютерную технику.

10. Однородные системы счисления

Считается, что первыми позиционными системами счисления, которые получили распространение на практике, были пятеричная и десятичная системы, которые входят в класс однородных систем счисления, являющихся простейшими позиционными системами счисления. Позже среди однородных систем счисления появились двоичная, восьмеричная, шестнадцатеричная и другие подобные им системы счисления. Характерным свойством таких систем счисления является то, что каждая из них имеет одинаковую длину формируемых ею чисел.

В десятичной системе счисления каждое натуральное число с помощью нумерационной функции можно подать в виде суммы произведений цифр этого числа, начиная с цифры нулевого разряда, на степень числа 10 с показателями степени 0, 1 ... .

Этим показателям отвечают номера разрядов числа. Например, число 123 десятичной системы исчисления можно подать в виде выражения, которое изображает такую нумерационную функцию:

$$1 \cdot 10^2 + 2 \cdot 10^1 + 3 \cdot 10^0.$$

При этом цифры в этой функции ограничены в пределах от 0 до 9. В системах счисления с другим основанием ограничения на цифры, которые используются в них, будут другими, но они обязательно существуют. Например, в самой простой на сегодня позиционной системе счисления – двоичной, то есть в системе, основание которой равняется 2, цифры могут принимать значение только 0 и 1. Тогда число 123 в двоичной системе будет иметь вид – 1111011.

Диапазон десятичной системы счисления равняется степени числа 10, показатель которого определяется количеством разрядов десятичного числа. Диапазон чисел системы счисления, которому принадлежит десятичное число 123, очевидно, равняется 1000.

В позиционных системах счисления каждый разряд имеет свой *вес*. Этот вес для десятичной системы счисления будет равняться степени 10 с показателем степени, который равняется номеру разряда, вес которого определяется. Для вышеприведенного примера веса нулевого, первого и второго разрядов будут равняться степени 10 с показателем степени соответственно 0,1 и 2. То есть в этом десятичном числе вес нулевого разряда равняется 1, а вес каждого следующего разряда увеличивается в 10 раз, и соответственно будет равняться 10 и 100 для первого и второго разрядов.

Однако вес *разряда* в некоторых более сложных системах счисления теряет свое первоначальное содержание, но зато появляется вес *цифры* этого разряда. Этот вес для каждой цифры разряда изображает ее количественное значение, которое может отличаться от значений других цифр этого разряда. Однако для десятичной, двоичной и других подобных систем счисления он равняется весу разряда. Поэтому для этих систем счисления можно говорить о весе разряда, понимая при этом вес цифры, и наоборот, говоря о весе цифры, можно понимать при этом вес разряда.

Кроме веса цифры разряда, будем рассматривать еще и ее *суммарный* вес в этом разряде, который имеет вид суммы весов всех цифр, меньших от цифры данного разряда. Их количество с учетом 0 будет таким, которое отвечает цифре числа этого разряда.

Суммарный вес цифры 0 всегда будет равняться 0, поскольку положительные цифры, что будут меньше 0, отсутствуют. Цифра 1 показывает, что в разряде, к которому она принадлежит, может быть лишь *одна* меньшая от нее цифра – 0, вес которой в данном случае и будет суммарным весом цифры 1. Цифра 2 будет показывать, что меньшими от нее будут *две* цифры – 0 и 1, веса которых при суммировании и создают суммарный вес цифры 2. Действие данного правила распространяется и на остальные цифры числа.

Например, для цифры 2 десятичного числа 123, которая принадлежит к первому разряду, суммарный вес будет состоять из суммы весов двух меньших цифр – 0 и 1. Вес каждой из них равняется 10, то есть суммарный вес цифры 2 числа 123 будет равняться 20. Очевидно, что суммарный вес цифры 3 нулевого разряда числа 123 будет состоять из весов трех меньших цифр – 0, 1, 2, которые в этом разряде равняются 1. Поэтому она будет равняться 1+1+1=3. Суммарный вес цифры 1 второго разряда в соответствии с вышесказанным, будет состоять из веса предшествующей ей цифры 0, которая равняется 100.

Также в таком десятичном числе, как 202, вес цифры 2 нулевого разряда равняется 1, цифры 0 первого разряда – 10, цифры 2 второго разряда – 100, а суммарные веса этих цифр будут равняться суммам количественных значений меньших цифр, которые принадлежат к этим разрядам. Так, в нулевом разряде цифра 2 имеет суммарный вес, который состоит из двух количественных значений меньших цифр, – 0 и 1. Их веса, очевидно, равняются 1, и поэтому суммарный вес цифры 2 будет равняться 2.

Соответственно в первом разряде цифра 0 имеет суммарный вес, который равняется 0, цифра 2 во втором разряде имеет суммарный вес, который состоит из весов цифр 0 и 1, то есть равняется 200.

11. Унитарная система счисления

Среди систем счисления переходное место между позиционными и непозиционными системами счисления занимает система счисления с алфавитом цифр, который состоит из одной цифры – *нуля*. Это значит, что основание и диапазон чисел этой системы равняется 1. Такая система счисления, иногда, называется ***унитарной***. Наличие в ее алфавите лишь нуля, то есть одной цифры, приводит к тому, что, например, число из трех разрядов унитарной системы счисления, будет иметь вид:

$$0 \cdot 1^2 + 0 \cdot 1^1 + 0 \cdot 1^0 = 0.$$

Такая система счисления кодирует лишь одно начальное число натурального ряда – единицу. При этом вес 0 каждого разряда равняется 1, а его суммарный вес будет равняться 0.

Казалось бы, что унитарная система исчисления не может иметь прикладного значения из-за того, что кодирует лишь одно число – 1. Но это не так. Эта система использовалась ранее и используется сегодня в современной цифровой технике при построении различных надежных устройств автоматики и вычислительной техники, потому что она может быть подана с помощью так называемого *число-импульсного* кода. В этом коде каждое число изображается с помощью последовательности импульсов, количество которых равняется разрядности этого числа.

Так, в вышеприведенном примере унитарная система счисления в реальном устройстве представлена тремя цифрами, равными 0 и весами, равными 1, то есть тремя идущими друг за другом импульсами. Однако суммарный их количественный позиционный эквивалент остается равным 0. Информация же о количестве хранится в переданном числе импульсов. Это число представляет непозиционный эквивалент количества. Главный недостаток использования такой системы исчисления – это значительное снижение быстродействия устройств, которые ее применяют. Но там, где быстродействие не имеет особого значения, унитарная система счисления может быть достаточно эффективной благодаря высокой надежности аппаратуры, разработанной на ее основе.

12. Неоднородные системы счисления

В рассмотренных выше примерах однородных позиционных систем счисления вес разряда определяется степенью основания. Это наиболее простой и широко применяемый на практике способ построения таких систем счисления. Но встречаются и более сложные, и в то же время интересные для практики, *неоднородные* системы счисления. Основой этих систем, как и ранее, являются натуральные числа, однако веса разрядов в них задаются более сложными способами, чем в однородных системах.

Например, можно построить систему счисления, которая основывается на простых числах, таких как 1, 3, 5, 7 .... . Это будет так называемая полиадическая система счисления. Возьмем в этой системе счисления для нулевого разряда одно основание – 1, для первого разряда два основания – 1 и 3, для второго разряда три – 1, 3, 5 и так далее. Из этого выходит, что алфавит цифр для нулевого разряда будет состоять из одной цифры – 0, для первого разряда из трех цифр – 0, 1, 2; второго разряда из пяти цифр – 0, 1, 2, 3, 4; третьего из семи – 0, 1, 2, 3, 4, 5, 6.

Тогда вес нулевого и первого разрядов будет равняться 1, вес второго – 3, вес третьего – 15 и так далее. Запишем в этой системе счисления число 220. Соответственно число 220 будет иметь вид:

$$2 \cdot 3 \cdot 1 + 2 \cdot 1 + 0 \cdot 0.$$

В десятичной системе исчисления это число, очевидно, будет равняться 8. Наибольшее число в этой тех разрядной системе счисления получается путем подстановки во все разряды числа максимальных возможных для них цифр – 4, 2 и 0. В итоге получим десятичное число 14. Соответственно, диапазон чисел, существующих в этой системе счисления, с учетом нуля, будет равняться 15.

Полученная неоднородная система счисления хотя и близка по своей структуре к однородной десятичной системе, но все же имеет и существенные от нее отличия. Оснований в ней не одно, а несколько. Также алфавит цифр изменяется от разряда к разряду. Соответственно и их веса изменяются не по закону степени, как это имеет место в десятичной системе счисления, а за более сложными правилами. Структура этой системы счисления содержит избыточную информацию, однако использовать эффективно ее для обнаружения и исправления ошибок трудно, так как в ней отсутствуют функциональные связи между разрядами чисел.

13. Структуры позиционных систем счисления

Рассмотрим более обстоятельно структуры систем счисления, объединяющие с помощью совокупности входящих в них связей все принадлежащие им числа в единое целое. Они в аналитической форме описываются числовыми (нумерационными) функциями, алфавитами и ограничениями на количества входящих в них цифр.

На рис. 1,2,3 приведены примеры систем счисления, среди которых наиболее простой будет двоичная. Она представляет класс однородных позиционных систем счисления, в которых отсутствует естественная избыточность информации, вызванная ограничениями в ее структуре. Это будет наиболее простая из всех возможных структур позиционных систем счисления за исключением унитарной. Она изображена на рис. 1 в виде иерархического дерева, состоящего из ветвей и узлов, в которые эти ветви приходят. Ее диапазон $P = 8$ чисел. Структура системы счисления дает в наглядной форме ее полное описание, так как нумерационная или числовая функция двоичной системы счисления и ограничения на количества ее цифр в разрядах чисел – это только математическая запись ее структуры.

Дерево данной структуры образуется, путем разбиения по шагам исходного множества из 8 элементов на 2 подмножества (классы эквивалентности), кодируемых цифрами 1 и 0, до тех пор, пока на последнем шаге разбиения не будет получено подмножество, состоящее из одного элемента. Все двоичные числа при этом, как видно из рисунка 1, будут иметь одинаковую длину, а алфавит, который их порождает, будет содержать две цифры – 0 и 1.

Из рисунка 1 видно, что с каждого узла дерева выходят ровно две вершины. Это свидетельствует о том, что в данной системе счисления генерируемая на каждом шаге разбиения цифра не зависит от того, какая цифра генерировалась на предыдущем шаге, то есть в двоичной системе счисления между цифрами отсутствует функциональная связь. Ее отсутствие говорит о том, что в структуре двоичной системы счисления нет избыточности и, соответственно, она не является помехоустойчивой. Все возможные ветви в ней, а значит и числа, разрешены. Если хотя бы одна ветвь была запрещена в дереве, то обязательно появилась бы избыточность в структуре дерева и соответственно новая система счисления была бы более сложная, чем исходная.

Обратим внимание на то, что такое же именно дерево разбиения на классы эквивалентности как на рис. 1, могут иметь и другие коды, например, код Грея. Однако любое другое кодирование классов эквивалентности, которое отличается от приведенного на рис. 1, это уже не будет образовывать систему счисления, а будет представлять более сложный код, который формирует слова, а не числа. Для образования чисел необходимо лишь кодирование классов эквивалентности (подмножеств) номерами в вышеупомянутом порядке. При этом первый класс эквивалентности обязательно должен обозначаться 0, а другие классы должны кодироваться в порядке 1,2, … , соответствующем начальным числам натурального ряда. В этом требовании заключается отличие чисел, формируемых системами счисления, от комбинаций кодов не являющихся системами счисления. Эти комбинации могут формироваться в любом порядке и любыми знаками, то есть системой счисления является далеко не каждый код, но каждый код в своей основе имеет систему счисления. На ее базе можно получить неограниченное количество различных кодов, комбинации которых являются словами, а не числами.

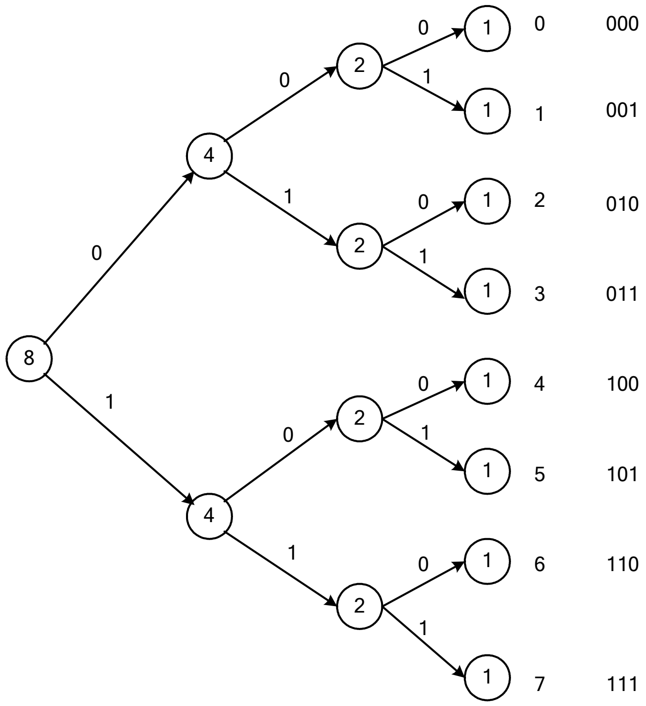

Рис. 1. Структура двоичной системы счисления

Числовая функция, которая задает количественный эквивалент двоичного числа, имеет вид:

$$A_{\langle 2\rangle} = x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \ldots + x_0 2^0 .$$

Ограничения относительно цифр, при этом, будут:

$$0 \le x_i \le 1,$$
$$i = 0,\ 1,\ldots ,n-1.$$

Структуры неоднородных систем счисления являются более сложными и поэтому содержат избыточность. Они могут строиться на основе смешанного, или функционального числового основания. Первые из них имеют числа с равной длиной, а другие с неравной.

Примером неоднородных систем счисления со смешанным числовым основанием, которые широко применяются на практике, есть факториальные системы с числовой (нумерационной) функцией

$$F = x_n n! + x_{n-1}(n-1)! + ... + x_1 1! + x_0 0!$$

и ограничениями

$$0 \le x_i \le i,$$
$$i = 0,\ 1, \ldots, n.$$

Диапазон этих систем исчисления $P = n!$.

Структура с $n = 3$ шагами разбиения факториальной системы счисления для диапазона из 6 чисел дана на рис. 2. Ее особенностью будет то, что число ветвей, выходящих с узлов на каждом уровне структуры или шаге разбиения, будет разным – на первом уровне оно равно 3, на втором - 2 и на третьем – 1. Это значит, что в данной структуре имеются запрещенные ветви и в структуре появляется избыточность. Тогда информационными разрядами этой системы счисления будут только первый и второй разряд, а нулевой разряд в данном случае является вообще избыточным.

Числовая функция и ограничение для этой конкретной системы счисления имеют вид:

$$F = x_2 2! + x_1 1! + x_0 0!,$$
$$0 \le x_i \le i,\ i = 0, 1, 2.$$

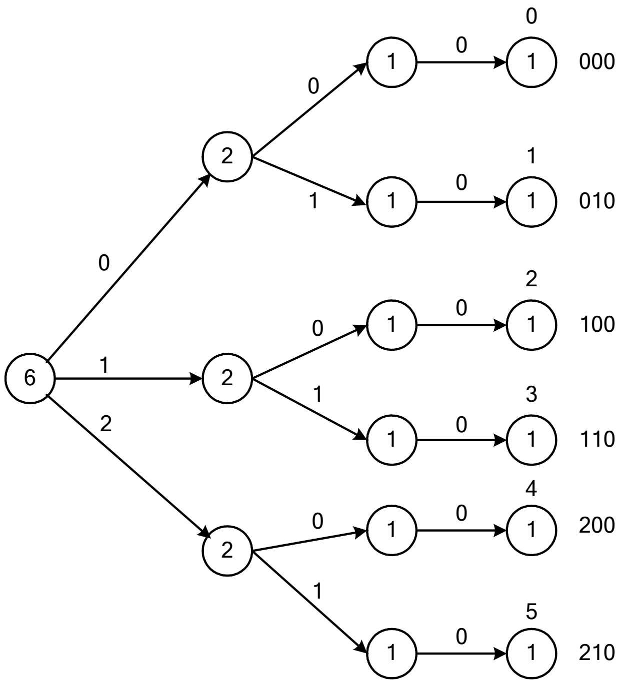

Рис2 Структура факториальной системы счисления

Полезной особенностью факториальных систем исчисления является то, что они способны генерировать перестановки. Это объясняется тем, что в качестве структуры множества перестановок выступает структура факториальной системы счисления. Кроме перестановок, эта система счисления может генерировать и другие комбинаторные объекты, так или иначе связанные с перестановками.

На рис. 3 приведена структура неоднородной системы счисления с неравной длиной чисел и двоичным алфавитом.

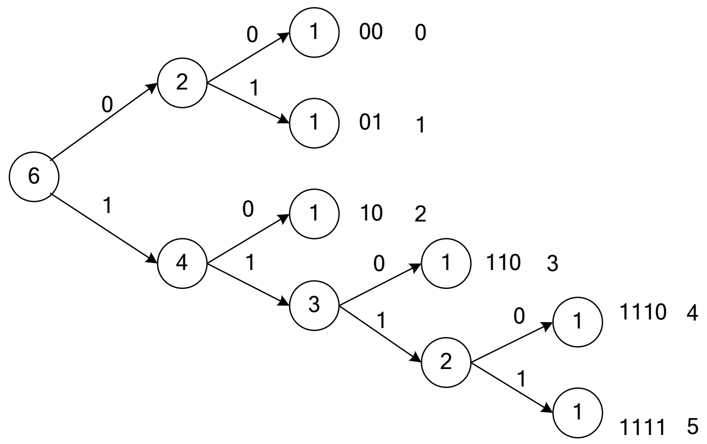

Рис. 3. Структура системы счисления с неравной длиной чисел

Множество чисел 00, 01, 10, 110, 1110, 1111 этой системы счисления образует собой префиксный код, поскольку ни одно из приведенных на рис. 3 чисел не является началом другого и в алфавите этого кода присутствует 0. При этом каждому числу отвечает десятичный номер 0, 1, 2, 3, 4, 5. Эти номера можно найти, если применить правило их вычисление с помощью числовой функции. Поскольку для этой системы счисления не заданы правила построения числовой функции, то ее следует строить с помощью таблицы, а саму систему счисления отнести к особому классу систем счисления – *табличных*. Такими системами счисления будут любые позиционные системы счисления, для которых не заданы регулярные правила их построения.

14. Структурные системы счисления

К неоднородным позиционным системам счисления относятся *структурные* системы, отличающиеся от остальных неоднородных систем счисления тем, что в них наблюдается функциональная связь между разрядами числа. К ним относится класс фибоначчиевых систем счисления, разработанный в семидесятые годы прошлого столетия советским ученым А. П. Стаховым [1]. Эти системы, как уже отмечалось выше, имеют естественную избыточность в своих структурах, что позволяет им обнаруживать ошибки в своей работе и при необходимости их исправлять. В то же время эта избыточность не настолько большая, чтобы существенно сказаться на быстродействии и аппаратурных затратах, использующих ее вычислительных устройств. Поэтому фибоначчиевы системы счисления способны стать базовыми для вычислительных систем и устройств, к которым предъявляются повышенные требования в надежности работы, в том числе и для вычислительных устройств универсального типа. Особенностью этого класса систем счисления является то, что в них между разрядами чисел наблюдаются функциональные связи, приводящие к способности обнаружения ошибок. Также эти системы счисления решают и ряд других задач, например, способны защищать информацию от несанкционированного доступа [5].

Следующими позиционными системами счисления, относящимися к классу структурных систем счисления, являются *биномиальные* системы с двоичным и многозначным алфавитом [2]. Они обладают большой избыточностью и соответственно повышенной помехоустойчивостью. В отличие от систем счисления, рассмотренных выше, их числа имеют разную длину, то есть относятся к кодам с неравномерной длиной кодовых комбинаций. Этот факт значительно усложняет работу с такими системами счисления и поэтому они не могут выступать в качестве универсальных вычислительных устройств, по крайней мере, на данном этапе развития вычислительной техники. Однако наличие повышенного интеллекта позволяет строить на основе этих систем счисления достаточно сложные специализированные устройства, например, высоконадежные системы сжатия информации или, на основе многозначной биномиальной системы счисления, системы обработки изображений. Кроме того, они позволяют строить надежную элементную базу для вычислительных систем и устройств.

Разработанная автором теория структурных систем счисления позволяет генерировать и множество других структурных систем счисления, решающих те или иные теоретические и прикладные задачи в науке и технике. Поэтому описанные выше системы счисления лишь представляют класс структурных систем счисления и поэтому могут появиться еще много интересных других подобных систем, хотя их синтез и исследование в силу большой сложности, как показывает опыт, представляет достаточно непростую задачу.

15. История систем исчисления

Исторически первыми возникли непозиционные системы исчисления. В их фундаменте лежит количественный подход к определению числа, в котором придумывались для кодирования тех или других количеств особые знаки - числа. Каждому такому знаку отвечал свой количественный эквивалент. Например, в так называемой римской нумерации знак *X* соответствовал количеству элементов во множестве, которое равнялось 10.

В дальнейшем такими знаками – числами пользовались также и для получения других чисел. Например, если перед знаком *X ставилась* вертикальная черточка, то получали знак *1X*, который значил, что от десяти нужно отнять единицу и результирующее количество будет равняться 9. Знаки подобные *X* называются *узловыми*. Они широко использовались в непозиционных системах счисления. Следует заметить еще раз, что среди этих знаков не было знака, который бы отвечал нулю. Это свидетельствует о том, что нуль в то время еще не был сформирован как число, а значит, не могло быть тогда и позиционных систем счисления.

Количество чисел, которое можно было получить с помощью непозиционного кодирования, из-за его сложности и соответственно большого количества чисел, которые нужно было запоминать, были ограничено несколькими сотнями и, кроме того, над этими числами достаточно трудно было выполнять арифметические и логические операции. Поэтому в дальнейшем при развитии науки появилась потребность в более эффективных системах счисления, которые бы имели простые правила кодирования чисел и тем самым не ограничивали бы их количество, и легко выполняли над ними арифметические и логические операции. Такими системами чисел были *позиционные* системы.

16. Выводы

Таким образом, системы счисления и на их основе решения задач кодирования чисел еще далеко не закончили свое развитие, а возможно даже те результаты, которые имеем на сегодня, это лишь начало серьезного понимания сущности систем счисления для математики и электронных систем. Сегодня системы счисления в преобладающем своем числе решают лишь задачи выполнения арифметических и логических операций в электронных вычислительных системах. Но они могут делать значительно более сложные операции, к которым относятся, и задачи сжатия информации, и задачи помехоустойчивого кодирования и задачи шифровки данных и такие необычные операции как генерирование математических объектов. Но если говорить вообще, то системы счисления в неявном виде есть то, что составляет фундамент всей дискретной математики, потому что они являются структурой любых ее объектов.

Литература

1. Стахов А.П. Введение в алгоритмическую теорию измерения. М., «Сов. Радио», 1977, 288 с.

2. Борисенко А. А. Биномиальный счет. Теория и практика. Сумы, «Университетская книга». 2004. – 167 с.

3. Повар В. М. Белый Б. М. Теоретические основы начального курса математики: Киев: Высшая школа. 1980. – 360 с.

4. Вивалюк Л. М., Григоренко В. К., Левищенко С. С. Числовые системы: Киев: Высшая школа. 1988. – 271 с..

5. Stakhov A., Massingue V, Sluchenkova A. Introduction into Fibonacci codings and cryptography. Kharkiv. 1999 - 234