



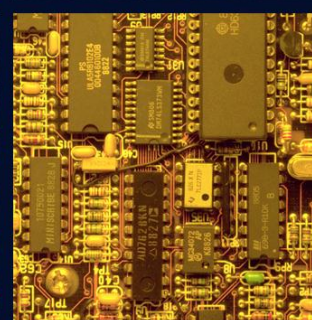
# ТЕТРАЛОГИКА, ТЕТРАВЫЧИСЛЕНИЯ И НООКОМПЬЮТИНГ



А.Я. АНОПРИЕНКО  
С.В. ИВАНИЦА



ИССЛЕДОВАНИЯ  
2010–2012



УНИТЕХ



Министерство образования и науки,  
молодежи и спорта Украины

Донецкий национальный  
технический университет

**А. Я. Аноприенко, С. В. Иваница**

# **ТЕТРАЛОГИКА, ТЕТРАВЫЧИСЛЕНИЯ И НООКОМПЬЮТИНГ**

**Исследования 2010–2012**

**М о н о г р а ф и я**

Донецк  
УНИТЕХ  
2012



УДК 004.2  
ББК 32.97  
А69

*Рецензенты:*

**Башков Е. А.**, д-р техн. наук, профессор, заведующий кафедрой прикладной математики и информатики ДонНТУ, проректор по научной работе ДонНТУ;

**Зори А. А.**, д-р техн. наук, профессор, заведующий кафедрой электронной техники ДонНТУ;

**Опанасенко В. Н.**, д-р техн. наук, ст. науч. сотр., руководитель научно-учебного центра «Микропроцессорная техника» при Институте кибернетики им. В. М. Глушкова НАН Украины.

**Анопrienко А. Я., Иваница С. В.**

**А69** Тетралогики, тетравычисления и ноокомпьютинг. Исследования 2010–2012. — Донецк: ДонНТУ, Технопарк ДонНТУ УНИТЕХ, 2012. — 308 с.

ISBN 978-966-8248-40-5

В монографии представлены результаты исследований и разработок в области теоретического обоснования и практической реализации тетралогики как наиболее эффективного варианта реализации постбинарной логики и тетравычислений и как наиболее перспективного варианта постбинарных арифметических операций. Рассмотрены также особенности перехода к ноокомпьютингу, раскрывающегося в качестве следующего этапа в развитии информационно-компьютерных технологий.

Материалы монографии предназначены для научного и образовательного использования и ориентированы на преподавателей, исследователей, специалистов и студентов, специализирующихся в области компьютерных наук и технологий.

УДК 004.2  
ББК 32.97  
А69

ISBN 978-966-8248-40-5

© Анопrienко А. Я., 2012  
© Иваница С. В., 2012  
© ДонНТУ, 2012

## СОДЕРЖАНИЕ

Введение.....	5
<b>Исследования 2010.....</b>	<b>9</b>
☑ Вызовы времени и постбинарный компьютинг .....	11
☑ Компьютерная арифметика и логика. Мифы и реальность.....	27
☑ Интервальная математика в SciLab + Int4Sci.....	33
☑ Подходы к построению логических элементов с мно- гозначной и нечеткой логикой .....	38
☑ Анализ времени выполнения простейших арифмети- ческих операций над интервалами в СКМ Mathematica 7.0.....	42
☑ Исследование интервального оценивания функций с использованием СКМ Mathematica 7.0.....	48
<b>Исследования 2011 .....</b>	<b>55</b>
☑ Компьютерные науки и технологии: следующие 50 лет .....	57
☑ Алгебра тетралогии. Реализация нульарных, унар- ных и бинарных логических операций.....	69
☑ Точное представление действительных чисел в кон- тексте расширенного кодо-логического базиса .....	79
☑ Реализация логических операций над элементами тетралогии с использованием аксиоматики теории множеств .....	84
☑ Постбинарные форматы действительных чисел. Особенности преобразования binary → pbinary .....	91
☑ Алгебра тетралогии. Реализация арифметических операций: операция сложения .....	97

☑ Представление интервальных чисел средствами постбинарного кодирования .....	105
☑ Ноографика & ноомоделирование .....	113
☑ Тетракоды в вычислениях с гибкой разрядностью .....	124
☑ Ноокомпьютинг и будущее информационно- компьютерной инфраструктуры.....	139

## **Исследования 2012..... 159**

☑ Будущее IT-индустрии в Донецке и Украине: вызовы, возможности и перспективы.....	161
☑ Постбинарный сопроцессор.....	181
☑ Алгебра тетралогии. Реализация арифметических операций: синтез постбинарного сумматора .....	187
☑ Проблемы компьютерных вычислений, вызванные использованием стандарта IEEE 754 .....	196
☑ Постбинарное округление.....	211
☑ Погрешность представления чисел в постбинарных форматах .....	220
☑ Операции с порядками постбинарных форматов чисел с плавающей запятой .....	229
☑ Введение в постбинарную арифметику.....	241
☑ Синтез постбинарных суммирующих, вычитающих и умножающих схем.....	248
☑ Алгебра тетралогии. Синтез унарных логических операций и постбинарного исключающего ИЛИ.....	259
☑ Синтез блока сокращения разрядности порядка .....	274
☑ Подготовка IT-профессионалов в украинских университетах: 50-летний опыт и следующие 50 лет .....	280

Заключение .....	291
------------------	-----

Список публикаций.....	297
------------------------	-----

## **ВВЕДЕНИЕ**

В 2010–2012 гг. на кафедре компьютерной инженерии Донецкого национального технического университета (ДонНТУ) под руководством авторов проводились регулярные научные семинары, посвященные преимущественно различным аспектам постбинарного компьютеринга. Активными участниками семинаров были аспиранты и студенты (в первую очередь магистранты) факультета компьютерных наук и технологий ДонНТУ. В общей сложности за три года было проведено более 30-ти семинаров, на которых их участниками было представлено более сотни докладов. В данное издание включены презентации 28-ми основных докладов, сделанных авторами на указанных семинарах, а также, частично, на международных научных конференциях, проходивших в указанный период. По годам представленные презентации распределяются следующим образом: 2010 год — 6, 2011 год — 10, 2012 год — 12.

Компьютерная презентация как вид научной публикации получила признание на протяжении последнего десятилетия, являясь в большинстве случаев или дополнением к тексту соответствующего доклада или достаточно информативной альтернативой ему. Высокий уровень информативности при этом обеспечивается за счет насыщенности иллюстративным материалом, а, как известно, одна хорошая иллюстрация стоит тысячи слов. В связи с этим было признано целесообразным наиболее значимые авторские презентации за 3 года исследований в области тетралогии, тетравычислений и ноокомпьютинга скомпоновать в монографию, которую можно рассматривать как весьма существенное дополнение к предыдущей работе авторов «Постбинарный компьютеринг и интервальные вычисления в контексте кодо-логической эволюции» [1]. Представленные в монографии презентации подверглись незначительному редактированию, связанному в основном с устранением повторов. Следует отметить, что впервые компьютерная презентация для сопровождения научного доклада была использована одним из авторов



в 2002 году (как альтернатива популярным в то время прозрачным пленкам — «фоли») и с тех пор стала незаменимым инструментом эффективной научной коммуникации. В связи с этим данную монографию можно рассматривать и как юбилейное издание, посвященное 10-летию активного использования компьютерных презентаций в научной работе.

Тетралогика (логика с четырьмя логическими состояниями) и тетравычисления являются наиболее перспективными и в обозримом будущем наиболее продуктивными направлениями развития постбинарного компьютеринга. Реализация тетравычислений позволяет, в частности, обеспечить эффективный контроль текущей точности и управление разрядностью вычислений в соответствии с реальной необходимостью. Концепция ноокомпьютинга (или «разумного компьютеринга»), сформировавшаяся в основном в ходе исследований 2011 и 2012 годов, имеет обобщающий характер и предполагает постбинарный компьютеринг в качестве одной из важнейших составляющих, позволяющих максимально приблизить компьютерную логику и вычисления к требованиям реального мира. Другими составляющими ноокомпьютинга являются «разумные» сетевые и интерфейсные технологии нового поколения, а также весь интенсивно развивающийся компьютерный континуум, позволяющий получить принципиально новый уровень интеллектуализации окружающей среды.

В разделе «Исследования 2010» содержатся материалы, детализирующие понятие «постбинарный компьютеринг», аргументирующие его актуальность, а также раскрывающие взаимосвязь постбинарных и интервальных вычислений.

Исследования 2011 и 2012 годов были посвящены в основном вопросам теоретического обоснования и практической реализации постбинарной логики и постбинарных арифметических операций. Анализировались также будущее IT-индустрии и перспективы развития компьютерных наук и технологий, на базе чего были сформулированы основные идеи и концепции, объединенные понятием «ноокомпьютинг».



**Участники научного семинара 2010.** Слева-направо: 1-й ряд: Е. В. Бурлака, О. О. Соловей, А. Я. Аноприенко, А. П. Коноплева, В. А. Гранковский; 2-й ряд: Е. Ф. Малеваный, Д. Ю. Плотников, И. Акулов, С. В. Иваница, А. В. Меркулов



**Участники научного семинара 2012.** Слева-направо: Р. Л. Варзар, С. В. Иваница, А. С. Щербаков, А. Я. Аноприенко, Е. В. Котов, С. В. Кулибаба, А. В. Меркулов, Е. Ф. Малеваный





**Декан факультета КНТ А. Я. Аноприенко** (второй слева) со своими учениками перед началом научного семинара

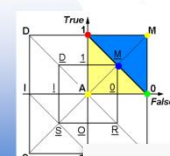


**На семинарах случается всякое...** даже прохождение «картонного» фотона сквозь «картонную» кристаллическую решетку. Картонные модели в руках участников семинара (слева-направо): Р. Л. Варзар, С. В. Иваница

# исследования 2010

## Двумерное логическое пространство

Может быть порождено базисом, состоящим из ортонормированной системы векторов «Истина» (Т – «True» или Y – «Yes»), «Ложь» (F – «False» или N – «No»):



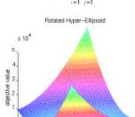
И и 0 – значения «Истина» и «Ложь» классической логики.  
А – абсолютная неопределенность,  
М – фиксированная множественность, («Истина» и «Ложь» одновременно),  
М – фиксированная равновероятность значений «Истина» и «Ложь».

Гиперлогика – логики третьего и более высоких порядков.  
Гиперкоды – системы кодирования

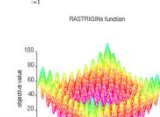
**Компьютерная арифметика и логика**

## Тестовые функции

$$f = \sum_{i=1}^n x_i^2$$



$$f = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i)) + 10n$$



**Интервальная математика в SciLab + Int4Sci**

## Нейронные процессоры

Нейрон по сути представляет собой элементный процессор, формирующий выходные сигналы в соответствии с заданной функцией.



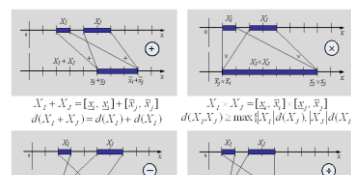
Нейронная сеть – совокупность нейронов, которые соединены друг с другом, образуя сложную структуру – «нейронный слой».

40-50 лет – первая практическая работа по созданию нейронных сетей.

Число слоев нейронной сети (НС) определяет ее сложность и способность к обучению.

**Подходы к построению логических элементов с многозначной логикой**

## ГРАФИЧЕСКАЯ ИНТЕРПРЕТАЦИЯ ИНТЕРВАЛЬНЫХ ОПЕРАЦИЙ

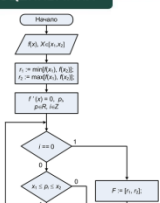


**Анализ времени выполнения операций над интервалами**

## АЛГОРИТМ ИНТЕРВАЛЬНОГО ОЦЕНИВАНИЯ

```

f[x_1] := ...
x = Interval([x1, x2]);
p[1] := NSolve[D[f(x)] = 0, x];
For [k = 0, k ≤ K, k++],
  f1[x1, x2] :=
    Interval[
      (Min[f(x1), f(x2)],
       f[f(x1) ≤ p[k] ≤ x2, p[k], x2]),
      (Max[f(x1), f(x2)],
       f[f(x1) ≤ p[k] ≤ x2, p[k], x2])
    ];
  
```



**Исследование интервального оценивания функций**



- 📖 Вызовы времени и постбинарный компьютеринг, 11
- 📖 Компьютерная арифметика и логика. Мифы и реальность, 27
- 📖 Интервальная математика в SciLab + Int4Sci, 33
- 📖 Подходы к построению логических элементов с многозначной и нечеткой логикой, 38
- 📖 Анализ времени выполнения простейших арифметических операций над интервалами в СКМ Mathematica 7.0, 42
- 📖 Исследование интервального оценивания функций с использованием СКМ Mathematica 7.0, 48



Информатика и компьютерные технологии - 2010

# **Вызовы времени и постбинарный компьютеринг**

**Аноприенко Александр Яковлевич**

Декан факультета  
компьютерных наук и технологий  
ДонНТУ

24.11.2010



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## **Основные вызовы 201X:**

(над чем предстоит работать в области компьютерных наук и технологий в 2010-2020 гг.)

1. Суперкомпьютерная гонка и экзамасштаб (Е-масштаб)
2. Энергетическое обеспечение компьютеринга
3. «Вездесущий» компьютеринг и Веб 3.0
4. Цифровой «потоп»
5. Контроль вычислений
6. Постбинарный компьютеринг



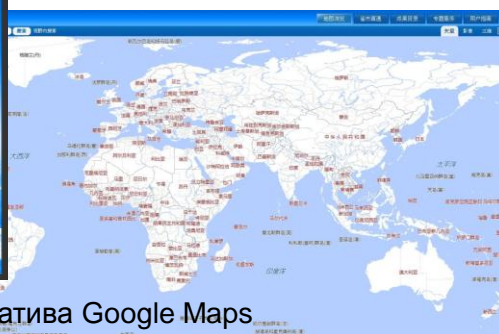
Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

Суперкомпьютерная гонка и экзамасштаб (Е-масштаб)

## 4 знаменательных события 2010 года:

1. В октябре 2010 года самый мощный в мире суперкомпьютер производительностью **2,5 петафлопс** ( $10^{15}$  операций с плавающей запятой в секунду) впервые был введен в эксплуатацию не в США или Европе, а **в Китае**



2010: «Мир по-китайски» как альтернатива Google Maps



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

Суперкомпьютерная гонка и экзамасштаб (Е-масштаб)

## 4 знаменательных события 2010 года:

1. В октябре 2010 года самый мощный в мире суперкомпьютер производительностью 2,5 петафлопс ( $10^{15}$  операций с плавающей запятой в секунду) впервые был введен в эксплуатацию не в США или Европе, а **в Китае**
2. **Впервые США, Европа и Россия начали объединять свои усилия в разработке компьютерных систем и программного обеспечения следующего поколения (следствие: ноябрь 2010, Лиссабон: NATO 3.0)**



Необходимость перемен генеральный секретарь НАТО Андерс Фог Расмуссен объяснил по-современному, с использованием компьютерной лексики:

«НАТО 1.0 – это военно-политический альянс с момента своего создания в 1949 году и до падения железного занавеса.  
НАТО 2.0 – Альянс с момента окончания холодной войны. Теперь же пришло время для НАТО 3.0».



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

### Суперкомпьютерная гонка и экзамасштаб (Е-масштаб)

## 4 знаменательных события 2010 года:

1. В октябре 2010 года самый мощный в мире суперкомпьютер производительностью 2,5 петафлопс ( $10^{15}$  операций с плавающей запятой в секунду) впервые был введен в эксплуатацию не в США или Европе, а **в Китае**
2. **Впервые США, Европа и Россия начали объединять свои усилия** в разработке компьютерных систем и программного обеспечения следующего поколения (ноябрь 2010: **NATO 3.0**)
3. Впервые основой самого мощного суперкомпьютера стали не универсальные процессоры, а графические: китайская система Тяньхэ-1А (Tianhe-1A), имеет архитектуру **GPGPU (general-purpose GPU)** и использует **7168 графических процессоров NVIDIA Tesla M2050** и 14336 CPU
4. Практически одновременно с преодолением петафлопсного рубежа началась масштабная всемирная гонка по достижению экзафлопсных рубежей, главный лозунг которой **« $10^{18}$  к 2018 году»**



Аноприенко  
Александр Яковлевич

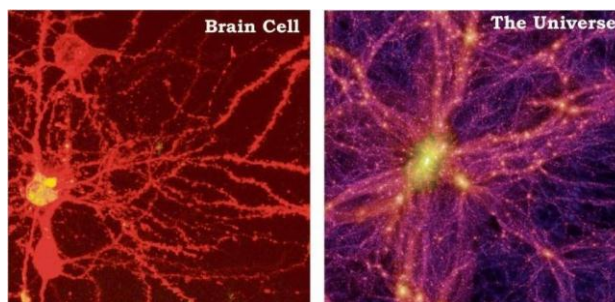
Вызовы времени и  
постбинарный компьютеринг

### Суперкомпьютерная гонка и экзамасштаб (Е-масштаб):

# $10^{18}$ к 2018 году

## Миллиард миллиардов

операций с плавающей запятой в секунду





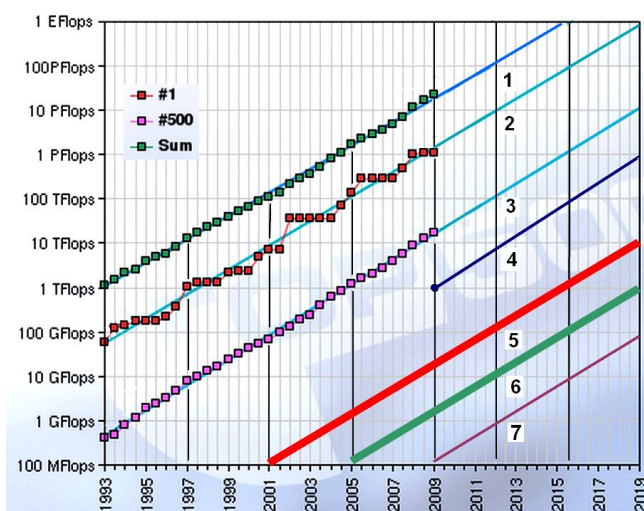


Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Суперкомпьютерная гонка и экзамасштаб (Е-масштаб):

# 10<sup>18</sup> к 2018 году



### Prefixes

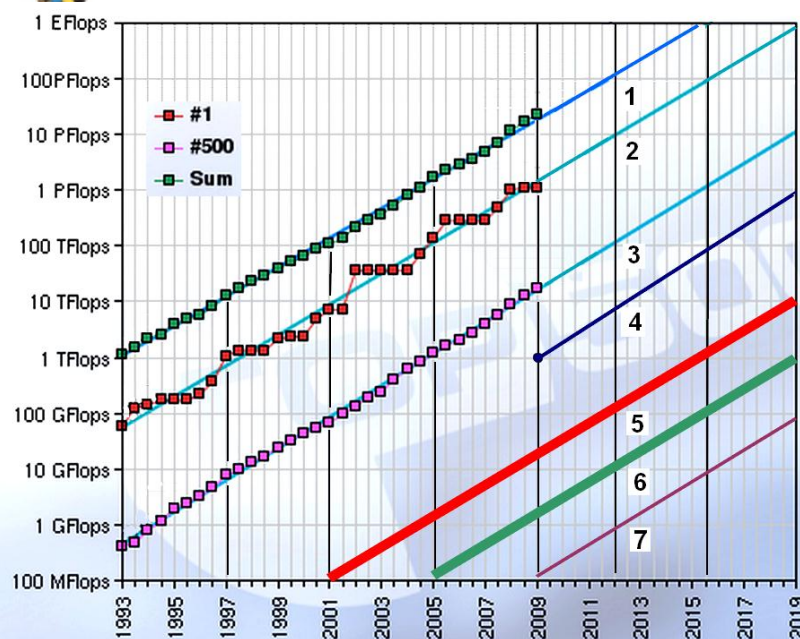
—  
Kilo – 10<sup>3</sup>  
Mega – 10<sup>6</sup>  
Giga – 10<sup>9</sup>  
Tera – 10<sup>12</sup>  
Peta – 10<sup>15</sup>  
Exa – 10<sup>18</sup>  
Zetta – 10<sup>21</sup>



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Суперкомпьютерная гонка и экзамасштаб (Е-масштаб):



### 10<sup>18</sup> к 2018 году

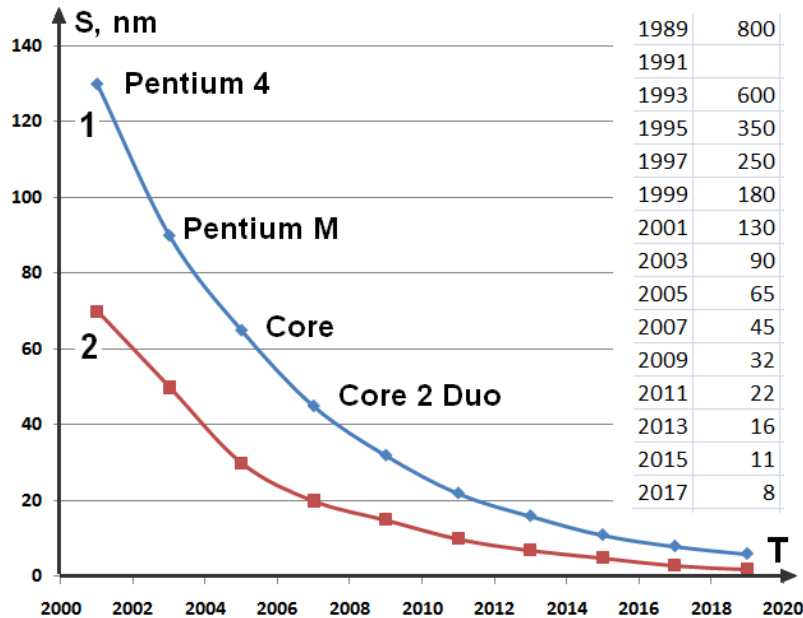
1. Сумма Top500:  
1 Ефлорс к 2015
2. №1 в Top500:  
1 Ефлорс к 2018
3. № 500 в Top500  
1 Ефлорс к 2025
4. «Персональные  
GPU-  
суперкомпьютеры»
5. «Продвинутые»  
персональные  
компьютеры
6. Усредненный  
персональный  
компьютер
7. Усредненное  
компактное  
мобильное  
устройство  
(коммуникатор)



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

### Суперкомпьютерная гонка и экзамасштаб (Е-масштаб):



Технологическое  
обеспечение:

**К 2018 году  
нанометровые  
проектные нормы  
8 нм**

(миллиардные  
доли метра)



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

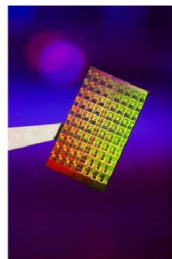
### Суперкомпьютерная гонка и экзамасштаб (Е-масштаб):

#### The Architecture & Algorithms Challenge



Oak Ridge / Cray Jaguar  
> 1.75 PFLOPS

Two Nvidia  
8800 GPUs  
> 1 TFLOPS



Intel 80-  
core chip  
> 1 TFLOPS

Главный  
архитектурный и  
алгоритмический  
вызов 2010:

Массовый  
параллелизм  
становится  
всеобщим  
явлением

В 2010 году в  
Интел начата  
разработка  
1000-ядерного  
процессора

- Parallelism is no longer optional...
- ... in every part of a computation.

UCSB



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Энергетическое обеспечение компьютеринга



**2010 год:** актуализировались почти забытые проблемы электропитания...

**Суперкомпьютер NEC Xeon Linux Cluster, введенный в эксплуатацию в ДонНТУ 1 сентября 2010 года:** 100 CPU, 200 ядер, производительность до 3-х Тфлопс.



Каждая из 3-х вычислительных стоек потребляет до 15 КВт, каждый из 3-х охлаждающих кондиционеров – 15 КВт.

**45 КВт компьютер + 45 КВт кондиционеры**

Суммарное потребление в максимальном режиме **90 КВт**.



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Энергетическое обеспечение компьютеринга



**1980 год:**

Самый мощный советский компьютер ЕС-1060, интенсивно эксплуатировавшийся в ДонНТУ с начала 80-х до начала 90-х годов

Производительность **1 Мфлопс**, потребляемая мощность 60 КВт, для охлаждения были установлены 2 шахтных вентилятора мощностью по 40 КВт (1 рабочий и 1 резервный)  
Суммарная потребляемая мощность составляла **100 КВт**.



2010: NEC Xeon Linux Cluster

Потребление **0,03 Вт на 1 Мфлопс** или производительность примерно 30 Мфлопс на 1 Вт  
**Энергоэффективность по сравнению с ЕС-1060 увеличилась в 3 миллиона раз (за 30 лет).**





Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Энергетическое обеспечение компьютеринга

Крупнейшие суперкомпьютерные центры в настоящее время потребляют **от 10-ти до 100 миллионов ватт (МВт)**, расходуемых на питание серверов, сетей, систем хранения данных и различного вспомогательного оборудования (при суммарном энергопотреблении типичного современного университета на уровне примерно 10 МВт).



При стоимости **1 МВт электроэнергии 1 миллион долларов в год** реальная цена высокопроизводительных вычислений оказывается намного большей, чем стоимость аппаратуры (средняя цена наиболее производительных суперкомпьютерных систем составляет порядка **10 миллионов долларов**).



NEC Xeon Cluster: 30 Мфлопс на 1 Вт  
IBM BlueGene/L: 200 Мфлопс на 1 Вт  
Китай, Тяньхэ-1А: 500 Мфлопс на 1 Вт

**2018: 50 000 Мфлопс на 1 Вт**

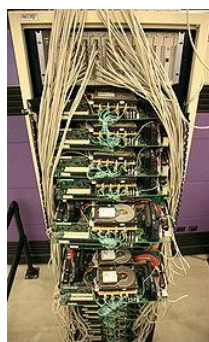


Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Энергетическое обеспечение компьютеринга

### Google: «Главный компьютер Земли»



1998: 1-3 сервера  
...  
2003: 10 000 серверов  
...  
2005: 100 000 серверов  
2006: 200 000 серверов  
2007: 1 000 000 серверов  
2008: 2 000 000 серверов  
...  
2012: 10 000 000 серверов



Как все начиналось (к концу 1998) в Станфордском университете:

1. Sun Ultra II with dual 200 MHz processors, and 256 MB of RAM (самый первый)
2. 2 × 300 MHz Dual Pentium II Servers donated by Intel, 512 MB RAM, 9×9 GB HDD
3. F50 IBM RS/6000 donated by IBM, 4 processors, 512 MB RAM, 8×9 GB HDD





Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Энергетическое обеспечение компьютеринга

Миллионы серверов Google:  
тяжелая промышленность 21 века

«Главный компьютер Земли»

...

2005: 100 000 серверов

2006: 200 000 серверов

2007: 1 000 000 серверов

2008: 2 000 000 серверов

2012: 10 000 000 серверов

В настоящее время в обслуживание  
**одного** типичного поискового  
запроса **Google** вовлекается  
**до 1000 серверов**

Всего у Google к концу 2010 года  
**около 40 серверных ферм** во всем  
мире, общее число серверов в  
которых составляет **несколько  
миллионов**.

Это крупнейший распределенный  
кластер в мире.

Только **электроэнергии** вся эта  
инфраструктура потребляет почти на  
**500 миллионов долларов в год**.



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Энергетическое обеспечение компьютеринга

Миллионы серверов Google:  
тяжелая промышленность 21 века



Новая серверная  
ферма Google в  
Орегоне, США:  
100 сотрудников,  
потребляет свыше  
**100 мегаватт**  
электроэнергии,  
(столько потребляет  
город с населением  
300 тысяч человек).

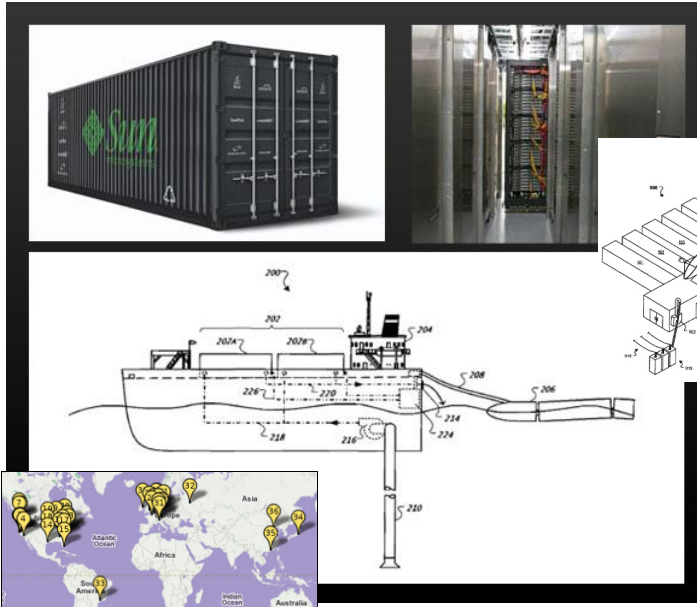
Неподалеку от  
комплекса ранее  
находился  
**алюминиевый  
завод**, который  
потреблял «всего» 85  
мегаватт, и на нем  
трудились 500  
человек.



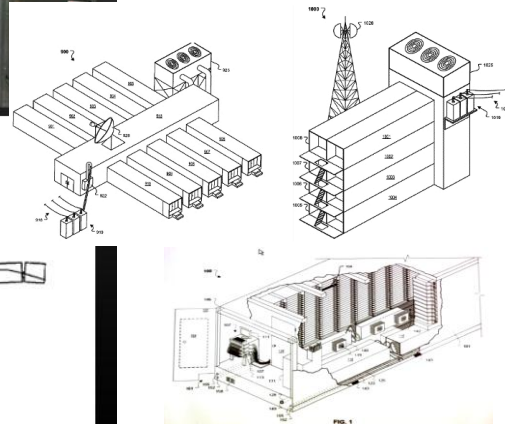
Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Энергетическое обеспечение компьютеринга



Патенты Google-2010  
Элементная база:  
контейнеры  
на 1000 серверов



"Modular Computing Environments"



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## «Вездесущий» компьютеринг и Веб 3.0



К концу  
2010 года:

Количество пользователей Интернет в мире приблизилось к **2 млрд** (еще в 2000 г. их было всего 360 млн)

Количество пользователей мобильной связи — к **5 млрд** (в 2000 г. — 720 млн),



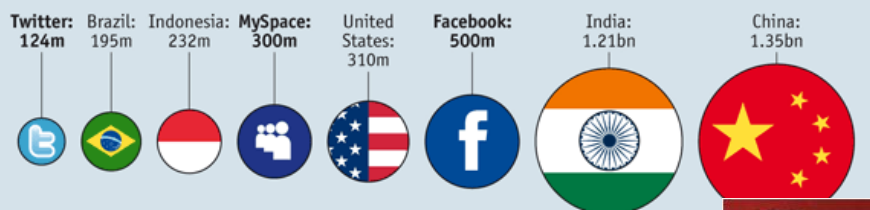
Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## «Вездесущий» компьютеринг и Веб 3.0

### Bubbling up

Population/registered users, 2010



Sources: UN; US Census Bureau; Facebook; MySpace; Twitter



2010:

Страна «Facebook» – население 500 млн

Страна «Вконтакте» – население 100 млн



Цифровая социальная революция...



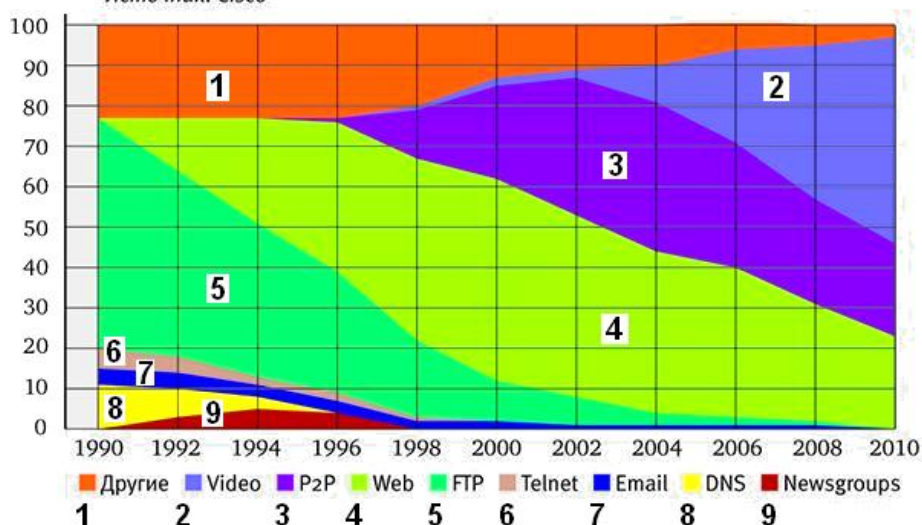
Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## «Цифровой потоп»:

Распределение трафика  
по категориям интернет-приложений, %

Источник: Cisco



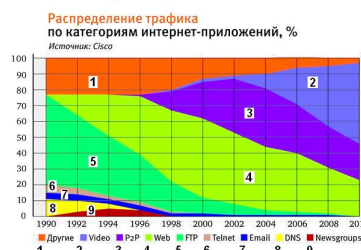


Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

### «Цифровой потоп»:

1. Популярные медиаресурсы **ежеминутно** дополняются более чем **10-ю часами** пользовательского видеоконтента
2. Поисковые системы **каждые 4 часа** обновляют данные, объем которых превосходит все содержимое хранилищ библиотеки конгресса США.
3. Число проиндексированных ресурсов достигло **1 трлн**, а количество ссылок между ними превышает 50 трлн.
4. Ежедневно осуществляется более 100 млрд переходов по веб-ссылкам
5. Электронных писем и мгновенных сообщений **ежесекундно** отправляется **2 млн и 1 млн** соответственно.



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

### «Цифровой потоп»:

#### Прогнозы главного футуролога Cisco:

По состоянию на 2010 год каждый житель Земли хранит в среднем порядка **100 Гбайт персональной информации**

**К 2020 году этот объем вырастет примерно в тысячу раз и составит порядка 100 Тбайт на человека.**

К 2015 г. Google проиндексирует примерно 775 млрд страниц контента

К 2020 г. персональный компьютер стоимостью \$1000 сравнится по вычислительной способности с человеческим мозгом

К 2029 г. за \$100 можно будет купить систему хранения данных емкостью 10 петабайт (условно 1 ПБ =  $10^{15}$  байт, или 1 млн ГБ)

Объем знаний будет расти огромными темпами: **сегодня мы знаем только 5% того, что нам станет известно в ближайшие 50 лет**





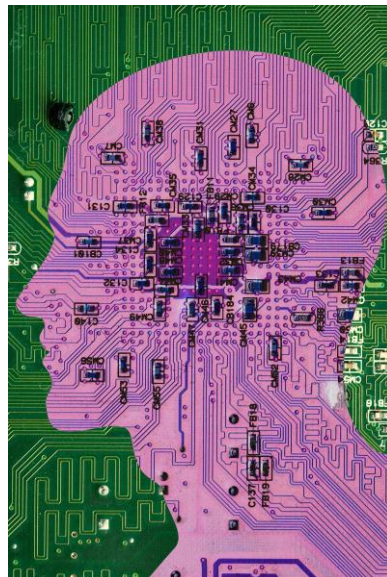
Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

### «Цифровой потоп»:

Уже сейчас очевидно, что **объем информации, который станет доступен среднему человеку в ближайшие десятилетия, намного превосходит не только его потребности, но и все мыслимые пределы возможностей его восприятия.** Другими словами, больше, чем человек физиологически способен увидеть, понять и запомнить, в него принципиально не может «поместиться», сколько бы дополнительной информации не существовало в природе.

В этой ситуации **резко начинает расти роль повышения «качества» данных**



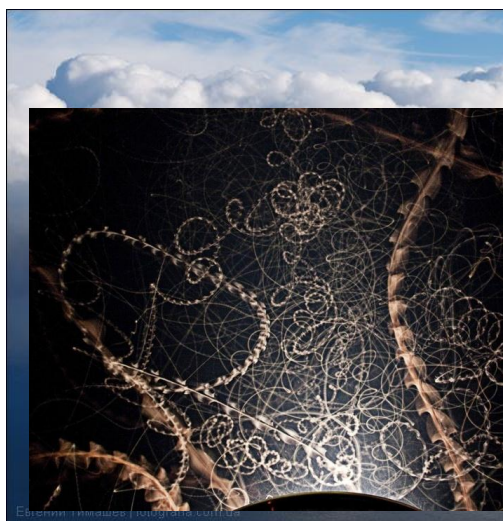
Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

### «Цифровой потоп»: лавина цифровой фотографии

Конкурс 2010 National Geographic Photography Contest:

Примерно **200 тыс** профессиональных фотографий



Отбирается только  
одна из тысячи  
фотографий!!!



Аноприенко  
Александр Яковлевич

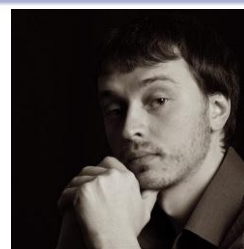
Вызовы времени и  
постбинарный компьютеринг

### Цифровой «потоп»: жесткий отбор

2010 National Geographic Photography Contest:  
Отбор 50-ти из 200 тыс лучших фотографий



Евгений Тимашев | fotografia.com.ua



#### В Топ50 2010:

Евгений  
Тимашев  
«Облако и  
корабль»  
Вид с  
Ай-Петри

Магистр  
факультета  
КНТ ДонНТУ  
2004



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

### Контроль вычислений

Чем выше производительность,  
тем быстрее накапливаются погрешности и ошибки...  
Простейший выход: наращивание разрядности:

**1985:** Стандарт вычислений с плавающей запятой IEEE 754-1985 предусматривал 2 типа чисел с плавающей запятой: **одинарной (32-разрядные) и двойной (64-разрядные) точности**. Одной из первых его аппаратных реализаций стал арифметический сопроцессор **Intel 8087**, в котором дополнительно в качестве внутреннего формата использовался **«расширенный» 80-разрядный формат с 64-разрядной мантиссой и 16-разрядным порядком**.

**2008:** Стандарт вычислений с плавающей запятой IEEE 754-2008. Основным нововведением стал **128-разрядный формат «квадро», т.е. формат «учетверенной» точности**.



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Контроль вычислений

Классический пример Румпа (известен с 1988 г)

$$f = 333.75 b^6 + a^2 (11a^2 b^2 - b^6 - 121b^4 - 2) + 5.5b^8 + a/(2b),$$

где  $a = 77617$ ,  $b = 33096$ .

При различной разрядности вычислений с плавающей запятой на современных компьютерных системах мы получаем примерно одинаковый результат:

32-bit:  $f = 1.172604$ ;

64-bit:  $f = 1.1726039400531786$ ;

128-bit:  $f = 1.1726039400531786318588349045201838$ .

Но во всех случаях этот результат весьма существенно (даже своим знаком) **отличается от правильного:**

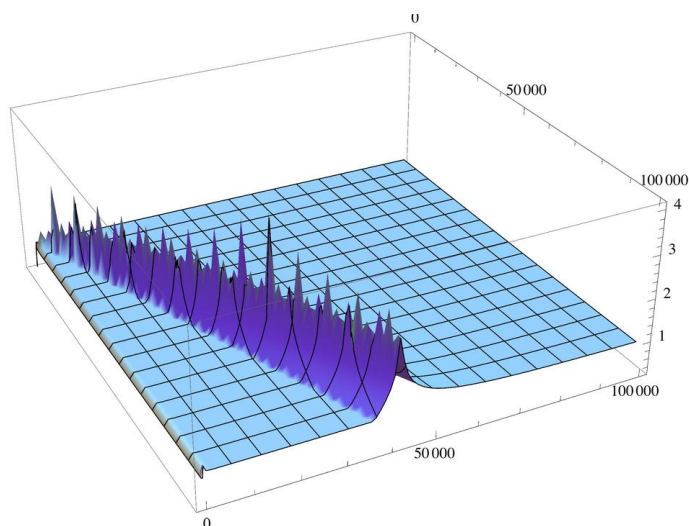
$$f = -0.827396059946821368141165095479816...$$



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Контроль вычислений



Зависимость погрешности вычислений по формуле Румпа в зависимости от значений параметров

В действительности имеется целое множество сочетаний исходных параметров, приводящих к абсолютно неверным результатам

При этом практически ни одна из использованных систем не фиксировала резкое нарастание отклонений от правильного результата...





Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Контроль вычислений

Только заведомо зная о существовании этой проблемы можно с помощью такого мощного современного пакета как Mathematica получить правильное значение одним из 3-х способов:

1. Значительным увеличением разрядности далеко за пределы стандартных значений
2. Вычислением отдельно числителя и знаменателя выражения с получением результата их деления только на самом последнем шаге
3. **Использованием интервальных вычислений**

### Выводы:

1. Необходимы средства постоянного поразрядного контроля достоверности исходных и вычисляемых значений
2. Необходимо существенное расширение базовых форматов представлений чисел с плавающей запятой
3. **Необходим постбинарный компьютеринг!**



Аноприенко  
Александр Яковлевич

Вызовы времени и  
постбинарный компьютеринг

## Составляющие постбинарного компьютеринга:

1. Постбинарная логика
2. Постбинарное представление чисел
3. Постбинарные алгоритмы
4. Постбинарные процессоры

### Преимущества постбинарного компьютеринга:

1. Естественный параллелизм
2. Естественная (нормированная) интервальность
3. Постоянный контроль исходной и реальной точности

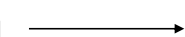


Аноприенко  
Александр Яковлевич

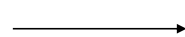
Вызовы времени и  
постбинарный компьютеринг

## Эволюция компьютеринга (основные этапы):

Прабинарный  
(добинарный)



Бинарный  
(цифровой)



Постбинарный  
(послебинарный)

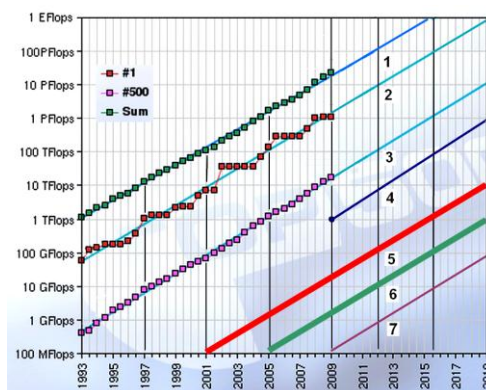
**XIII-XV вв.**

Борьба абакистов  
и алгоритмиков

**2010-2020 гг.**

Экспансия  
постбинарных  
технологий

«Настоящая цифра только 0...  
и лишь тёмная чернь  
остальные фигуры  
называет цифрами...»



## для заметок



Донецкий национальный  
технический университет

ФКНТ

# Компьютерная арифметика и логика

Аспирант  
кафедры КИ  
Иваница С.В.

Мифы и реальность

14 / 09 / 2010

## Арифметика и логика

- **Компьютер** – высокоточное вычислительное устройство, исключаящее любые ошибки в вычислениях;
- 32-, 64- и 128-битное представление действительных чисел позволяет получать достаточно точное решение практически любой прикладной задачи;
- Машинная логика, представленная всего двумя значениями «истина» и «ложь», оптимальна и способна представить весь набор известных человеку логических состояний;

*Машинный (искусственный) интеллект как совокупность компьютерной арифметики и логики ни чем не уступает человеческому интеллекту.*

**Так ли это? Это не так!**

2

## Машинная арифметика

### Вычислительные ошибки:

- Моделирование приближений:
  - Классический метод решения для задач квантовой механики;
- Ошибки разложения:
  - Из-за алгоритмических приближений, например, при разложении в ряд:
 
$$e \approx 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \dots$$
- Округление ошибок:
  - Погрешность увеличивается на  $\sim 10^{-8}$  при единичной точности, и  $\sim 10^{-16}$  при двойной точности с каждой операцией (действия).

Используя петафлопсные компьютеры, выполняющие  $10^{15}$  операций в секунду, как мы узнаем, что выполняемые нами вычисления правильны?

3

## Точность $\neq$ Точность

- Rump's example:

$$f = (333.75 - a^2)b^6 + a^2(11a^2b^2 - 121b^4 - 2) + 5.5b^8 + a / (2b)$$

$$a = 77617 \quad b = 33096$$

$$32 \text{ bit: } f = 1.172604$$

$$64 \text{ bit: } f = 1.1726039400531786$$

$$128 \text{ bit: } f = 1.1726039400531786318588349045201838$$

$$\text{correct: } f = -.827396059946821368141165095479816\dots$$

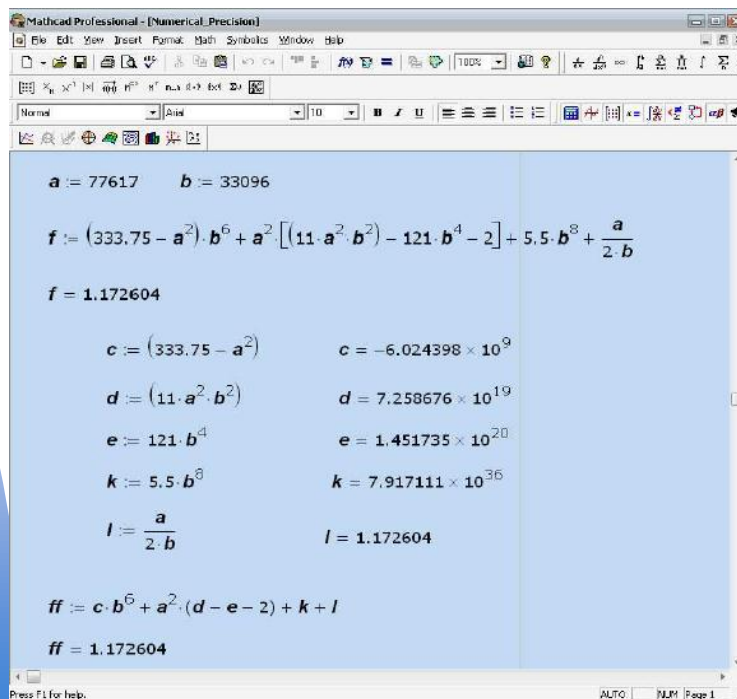
4



## Точность $\neq$ Точность

### ■ Rump's example:

Пример получения неверного результата в среде MathCad Professional



correct:  $f = -0.827396059946821368141165095479816...$

5

## Интервальный анализ

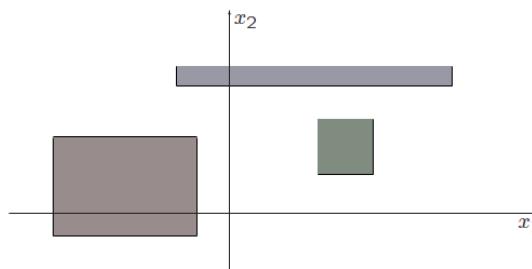
- **Интервальный анализ** — раздел вычислительной математики, посвященный учету ошибок округления при проведении расчетов на цифровых ЭВМ . . .

$[1,2], [1000,1003], \dots$

$| [1,2], [1000,1003] |$

$| [1,2] |$

$| [1000,1003] |$



6

## Интервалы

- Далеко не новая концепция выполнения арифметических действий над числами в формате с плавающей запятой:
  - Первая монография, полностью посвященная интервальному анализу, была опубликована Р.Е.Муром в 1966 г.
- Возникающие в процессе вычислений погрешности не выводят истинное значение за границы интервала;
- Интервальное число представляется двумя значениями (числами) – границами интервала:

$$[a, b] = \{x \mid a \leq x \leq b\}$$

7

## Интервальная арифметика

Классическая интервальная арифметика образована интервалами

$$\mathbf{x} = [x_1, x_2] \subset \mathbb{R} \text{ так, что}$$

$$\mathbf{x} \bullet \mathbf{y} = \{x \bullet y \mid x \in \mathbf{x}, y \in \mathbf{y}\} \text{ для } \bullet \in \{+, -, *, /\}$$

$$[a, b] + [c, d] = [a + c, b + d];$$

$$[a, b] \times [c, d] = [\min(a \times c, a \times d, b \times c, b \times d), \max(a \times c, a \times d, b \times c, b \times d)].$$

Интервальная функция  $\mathbf{F}\mathbf{x}$ ,  $\mathbf{x} \in \mathbb{IR}$  называется интервальным расширением вещественной функции  $f(x)$ ,  $x \in \mathbb{R}$ , если

- 1)  $\mathbf{F}\mathbf{x} = f(x)$  для всех  $x \in \mathbb{R}$ ;
- 2)  $\mathbf{F}\mathbf{x}$  монотонна по включению.

Внешняя оценка области значений:  $\mathbf{F}\mathbf{x} \supseteq \{f(x) \mid x \in \mathbf{x}\}$

8



## Тетракоды и интервалы

- Интервальное представление результатов вычислений по формуле Бэйли-Боруэйна-Плаффа:

$$\pi = f(n) = \sum_{k=0}^n \frac{1}{16^k} \cdot \left( \frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right);$$

$$X = [x_1, x_2] = (X \in I(\mathbb{R}), x_1 \in \mathbb{R}, x_2 \in \mathbb{R}) \Rightarrow$$

$$x_1 = f(1) = 3,14142; \quad x_2 = f(100) = 3,14159.$$

Выражение интервала  $X$  одним тетракодом

$T \in \{t \mid t \in \mathbb{H}\}$ , где  $\mathbb{H} = \{0, 1, A, M\}$ :

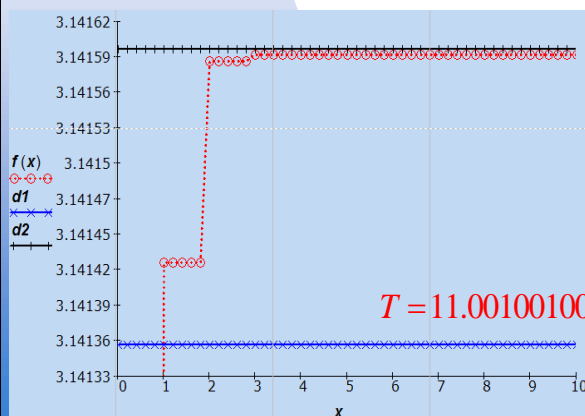
$$3,14142_{10} =$$

$$= 11.00100100001101000001100111100011_2,$$

$$3,14159_{10} =$$

$$= 11.00100100001111110011111000000011_2.$$

$$T = 11.001001000011MMMMMAAAAAAAAAAAAAA.$$



11

### для заметок





**Донецкий национальный технический университет**

## **Интервальная математика в SciLab+Int4Sci**

**Факультет ВТИ**

**Аспирант кафедры КИ  
Иваница С.В.**

**28.09.2010**

### **Постановка задачи (введение)**

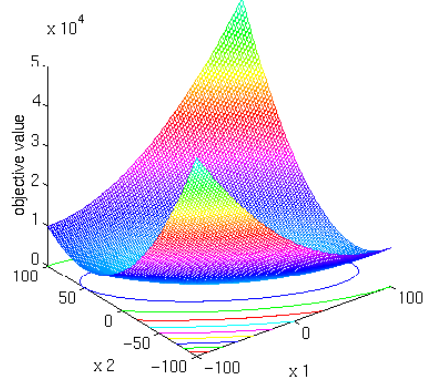
- ✦ Получение значений вычисляемых функций (Rotated Hyper-Ellipsoid, Rastrigin's function) с использованием пакета Mathematica и SciLab;
- ✦ Выполнение интервальных вычислений данных функций в среде SciLab+Int4Sci;
- ✦ Сравнительный анализ точности полученных результатов в пакетах Mathematica и SciLab;
- ✦ Получение и сравнительный анализ временной характеристики  $T$  ( $T$  – время вычисления) при вычислениях точным (обычным) и интервальным способами.

## Тестовые функции

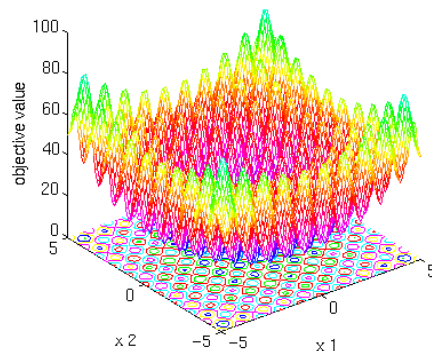
$$f = \sum_{i=1}^n \sum_{j=1}^i x_j^2$$

$$f = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i)) + 10n$$

Rotated Hyper-Ellipsoid



RASTRIGINs function



$$i \in \{1, \dots, n\}$$

## Гипер-эллипсоид в пакете Mathematica

$$f = \sum_{i=1}^n \sum_{j=1}^i x_j^2$$

```
a1 = SessionTime[]; F = Sum[Sum[x^2, {j, 1, i}], {i, 1, n}]; n = 5; x = 1.5;
```

```
F
```

```
a2 = SessionTime[]; a2 - a1
```

```
Out[3]= 33.75
```

```
Out[5]= 0.0156000
```

$n=5$

$n=25$

$n=100$

$n=500$

```
a1 = SessionTime[]; F = Sum[Sum[x^2, {j, 1, i}], {i, 1, n}]; n = 25; x = 1.5;
```

```
F
```

```
a2 = SessionTime[]; a2 - a1
```

```
Out[13]= 731.25
```

```
Out[15]= 0. x 10^-6
```

```
a1 = SessionTime[]; F = Sum[Sum[x^2, {j, 1, i}], {i, 1, n}]; n = 100; x = 1.5;
```

```
F
```

```
a2 = SessionTime[]; a2 - a1
```

```
Out[23]= 11362.5
```

```
Out[25]= 0.0156000
```

```
a1 = SessionTime[]; F = Sum[Sum[x^2, {j, 1, i}], {i, 1, n}]; n = 500; x = 1.5;
```

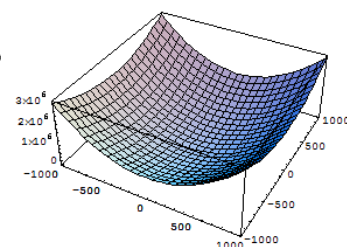
```
F
```

```
a2 = SessionTime[]; a2 - a1
```

```
Out[34]= 281813.
```

```
Out[36]= 0.2028004
```

$$x_{beg} = 1.5$$

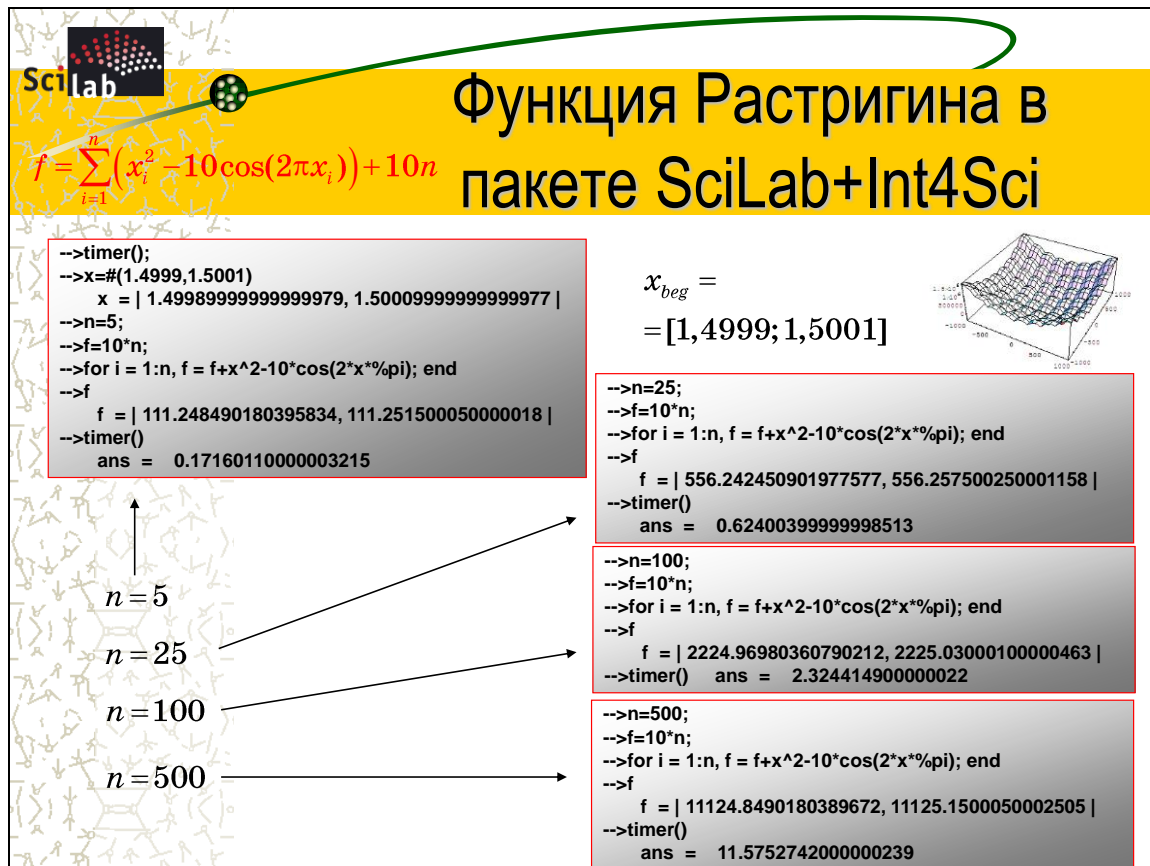


$$f = \sum_{i=1}^n \sum_{j=1}^i x_j^2$$
 $n = 500$ 

?

$$f = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i)) + 10n$$
 $n = 500$ 

```
Out[81]= 11125.  
Out[83]= 0. × 100
```



## Результаты тестирования

### Гипер-эллипсоид

$n$	$T, c$		Результат	
	SciLab	Mathematica	SciLab	Mathematica
5	0.24960 (.0312002)	0.01560	33.7455001499999696, 33.7545001500000055	33.75
25	4.1964269 (.0156001)	$0 \cdot 10^{-8}$	731.152503249993288, 731.347503250010959	731.25
100	64.132011 (.0624004)	0.0312001	11360.9850504974384, 11364.0150505016991	11362.5
500	> 20 мин (.3900025)	0.1872003	----	281813.0



## Результаты тестирования

### Функция Растригина

n	T, с		Результат	
	SciLab	Mathematica	SciLab	Mathematica
5	0.1716011 (.0312002)	$0 \cdot 10^{-8}$	111.248490180395834, 111.251500050000018	111.25
25	0.624 (.0468003)	$0 \cdot 10^{-8}$	556.242450901977577, 556.257500250001158	556.25
100	2.3244149 (.0656001)	0.01560	2224.96980360790212, 2225.03000100000463	2225.0
500	11.575274 (.0312002)	$0 \cdot 10^{-8}$	11124.8490180389672, 11125.1500050002505	11125.0

## Выводы

- ✧ В пакетах Mathematica и SciLab при одинаковых представлениях данных время выполнения вычислений практически одинаково (одного порядка);
- ✧ Интервальные вычисления в SciLab+Int4Sci обеспечивают достаточную точность и не несут ошибок округления результата;
- ✧ Интервальные вычисления ресурсоемкие и реализуемы на персональных компьютерах с большими временными затратами;
- ✧ Пакет SciLab с надстройкой Int4Sci является достаточным инструментом для выполнения интервальных вычислений.

## Аппаратная реализация гиперлогики

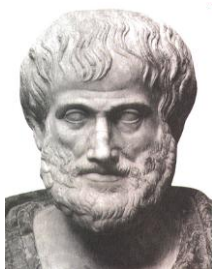
### Подходы к построению логических элементов с многозначной и нечеткой логикой

20.10.2010

Аспирант кафедры КИ  
Иваница С.В.

## Формирование идеи

Идея построения логических элементов с **нечеткой логикой (fuzzy logic)** основывается на нечеткой математике. Математическая теория нечетких множеств, предложенная проф. Л.А. Заде, являясь предметом интенсивных исследований, открывает все более широкие возможности перед системными аналитиками. Основанные на этой теории различные компьютерные системы, в свою очередь, существенно расширяют область применения нечеткой логики.



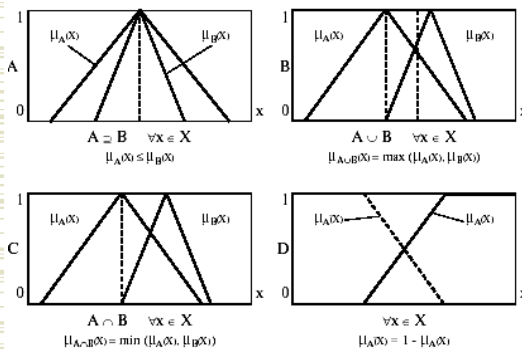
АРИСТОТЕЛЬ  
(384 - 322 гг. до н. э.)



ЛОТФИ АСКЕР ЗАДЕ  
(род. 1921 г.)

Подходы **нечеткой математики** позволяют оперировать входными данными, непрерывно меняющимися во времени, и значениями, которые невозможно задать однозначно. В отличие от традиционной формальной логики, известной со времен **Аристотеля** и оперирующей точными и четкими понятиями типа истина и ложь, нечеткая логика имеет дело со значениями, лежащими в некотором (непрерывном или дискретном) диапазоне.

## Нечеткая математика



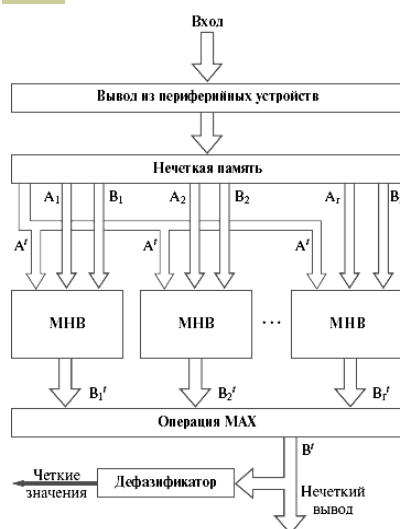
Операции включения (A), объединения (B), пересечения (C) и дополнения (D) нечеткого множества

**Функция принадлежности элементов к заданному множеству** представляет собой не жесткий порог "принадлежит – не принадлежит", а плавную сигмоиду, проходящую все значения от нуля до единицы. Теория нечеткой логики позволяет выполнять над такими величинами весь спектр логических операций – объединение, пересечение, отрицание и др.

### Теорема FAT (Fuzzy Approximation Theorem):

Любая математическая система может быть аппроксимирована системой, основанной на нечеткой логике.

## Аппаратный процессор нечеткой логики



Архитектура нечеткого компьютера (МНВ-механизм нечеткого ввода)

### Принцип решения задач с помощью нечеткой логики (принцип работы нечеткого процессора):

- 1) численные данные (показания измерительных приборов, результаты анкетирования) фаззируются (переводятся в нечеткий формат);
- 2) обрабатываются по определенным правилам;
- 3) дефаззируются и в виде привычной информации подаются на выход.

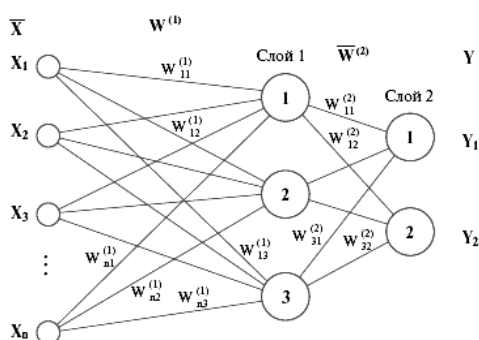
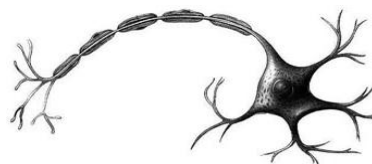
**1986** - в AT and T Bell Labs создавались процессоры с "прошитой" нечеткой логикой обработки информации. **начало 90-х** - Adaptive Logic (США) выпустила комбинированный цифровой/аналоговый кристалл, он функционирует на очень высоких скоростях (частота отсчетов входного сигнала – 10 кГц, а скорость расчета – 500 тыс. правил/с).

**наше время** - NASA рассматривает возможность применения (если еще не применяет) нечетких систем для управления процессами стыковки космических аппаратов.

## Нейронные процессоры

**Нейрон**, по сути, представляет собой элементарный процессор, характеризующийся входным и выходным состоянием, передаточной функцией (функция активации) и локальной памятью.

**Нейронная сеть** - совокупность нейронов, которые соединены друг с другом каналами обмена информации – "синаптическими связями".



40-50-е гг. - первые практические работы по искусственным нейросетям и нейрокомпьютерам.

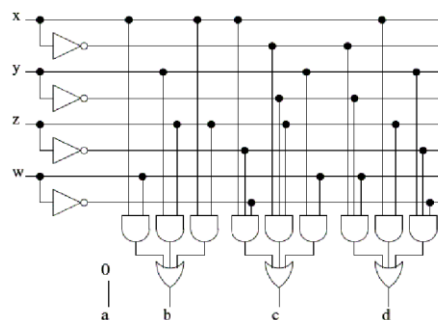
Число входов нейронной сети ( $X_1, \dots, X_n$ ) определяет размерность гиперпространства, в котором входные сигналы могут быть представлены точками или гиперобластями из близко расположенных точек. Количество нейронов в слое сети определяет число гиперплоскостей в гиперпространстве. Вычисление взвешенных сумм и выполнение нелинейного преобразования позволяют определить, с какой стороны от той или иной гиперплоскости находится точка входного сигнала в гиперпространстве.

## Аппаратная реализация логических элементов на примере троичной логики

**Первый подход** - моделирование троичных функций с помощью кодирования в двоичном виде (binary coded ternary, BCT representation).

**Общий метод синтеза BCT схем:**

- 1) составляется таблица истинности для трехзначной функции в соответствии с выбранной кодировкой (например, таблица 1);
- 2) на основе таблицы кодировки для каждого выходного разряда составляются карты Карно (Karnaugh maps) для минимизации моделирующей функции.
- 3) по результатам минимизации производится синтез бинарных схем.



Двухразрядный BCT-сумматор

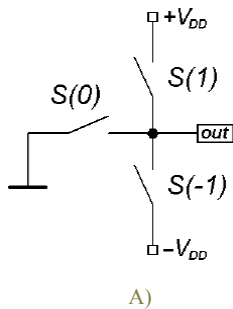
Таблица 1

3-bit binary	000	001	010	011	100	101	110	111
Decimal	0	1	2	3	4	5	6	7
2-digit ternary	00	01	02	10	11	12	20	21
BCT	0000	0001	0010	0100	0101	0110	1000	1001

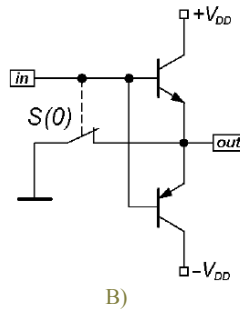
**Недостаток:** для, например, 16 битного BCT-сумматора сокращение разрядной ёмкости равно всего  $\eta = 3^8/2^{16} \approx 0,1$ , что ограничивает увеличение разрядности BCT устройств.

## Аппаратная реализация логических элементов на примере троичной логики

Второй подход к реализации трехуровневых устройств заключается в использовании управляемых электронных ключей при симметричном питании.



А)  
Простейшая трёхуровневая схема с использованием управляемых ключей



В)  
Комбинированная схема троичного повторителя на биполярных транзисторах

Данная схема (А) может работать как повторитель или же как инвертор трехуровневого сигнала. Условие её функционирования очевидно: в определённом логическом состоянии замкнут только один ключ при разомкнутых остальных. Допустим это пара ключей S(1) и S(-1), которые представляют из себя биполярные транзисторы. В результате можно получить наглядную [схему повторителя на основе двухтактного каскада \(В\)](#).

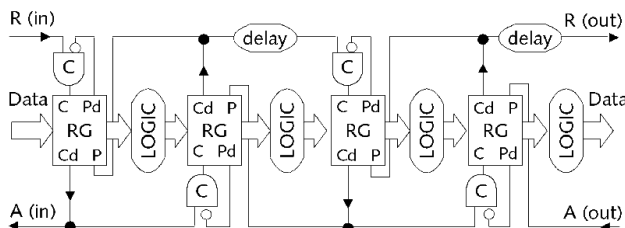
## Асинхронная логика NULL Convention Logic (NCL)

NCL-логика существует в двух вариантах, как [трёхзначная](#), где используются сигналы "true", "false" и "null", и как [четырёхзначная](#) с дополнительным сигналом "intermediate".

Избавление от синхронизации в NCL достигается путем интеграции данных и управления в одном логическом потоке, где "null" (разделитель) выполняет синхронизирующую функцию. В качестве элементной базы NCL применяются пороговые вентили с гистерезисом.

[Реализация асинхронного процессора](#) (микроконвейера) – исследовательская лаборатория Sun Microsystems под руководством Айвэна Сазерленда (Ivan E. Sutherland).

Функционирование микроконвейера основано на принципах, названных "рукопожатием" (handshaking) - реализация обратной связи путем обмена служебными сигналами.



Структурная схема микроконвейера

Главное преимущество многозначной логики - простая реализация протокола рукопожатия, что значительно уменьшает количество соединений.



Донецкий национальный технический университет

# **АНАЛИЗ ВРЕМЕНИ ВЫПОЛНЕНИЯ ПРОСТЕЙШИХ АРИФМЕТИЧЕСКИХ ОПЕРАЦИЙ НАД ИНТЕРВАЛАМИ В СКМ WOLFRAM MATHEMATICA 7.0**

08/11/2010

Аспирант кафедры КИ  
Иваница С.В.

## **ПОСТАНОВКА ЗАДАЧИ**

- Сравнение времени выполнения бинарных операций (Add, Sub, Div, Mul) при традиционных и интервальных представлениях данных;
- Графическое представление зависимостей  $t(n)$ ,  $T(n)$ , и определение типа зависимостей (линейная, квадратичная);
- Определение коэффициента увеличения времени выполнения бинарных арифметических операций;
- Сравнение времени выполнения унарных операций (Exp[x], Ln[x], Sin[x], Tan[x]) при интервальных представлениях данных с различными диаметрами исходного интервала;
- Исследование зависимостей временных параметров для функции гиперболического эллипсоида и функции Растригина.

## ИНТЕРВАЛЬНЫЕ АРИФМЕТИЧЕСКИЕ ОПЕРАЦИИ

$$X = [\underline{x}, \bar{x}] = \{i \mid \underline{x} \leq i \leq \bar{x}, \underline{x}, \bar{x} \in R\};$$

$$\forall x \exists X \Leftrightarrow X = [x, x], \quad x \in R;$$

$$Z = X \bullet Y = \{z = x \bullet y \mid x \in X, y \in Y\},$$

$\bullet \in \{+, -, \times, \div\}$  - бинарная операция на множестве  $R$ ;

$$X + Y = [\underline{x} + \underline{y}, \bar{x} + \bar{y}], \quad X - Y = [\underline{x} - \bar{y}, \bar{x} - \underline{y}],$$

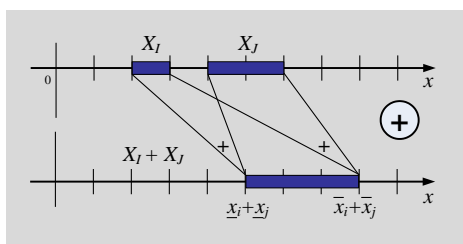
$$X \times Y = [\min\{\underline{x} \underline{y}, \underline{x} \bar{y}, \bar{x} \underline{y}, \bar{x} \bar{y}\}, \max\{\underline{x} \underline{y}, \underline{x} \bar{y}, \bar{x} \underline{y}, \bar{x} \bar{y}\}],$$

$$X \div Y = [\min\{\underline{x} / \underline{y}, \underline{x} / \bar{y}, \bar{x} / \underline{y}, \bar{x} / \bar{y}\}, \max\{\underline{x} / \underline{y}, \underline{x} / \bar{y}, \bar{x} / \underline{y}, \bar{x} / \bar{y}\}],$$

---

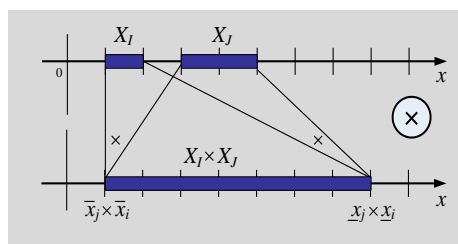

$$|X| = \max_{x \in X} |x|; \quad d(X) = \max_{x_1, x_2 \in X} |x_1 - x_2|.$$

## ГРАФИЧЕСКАЯ ИНТЕРПРЕТАЦИЯ ИНТЕРВАЛЬНЫХ ОПЕРАЦИЙ



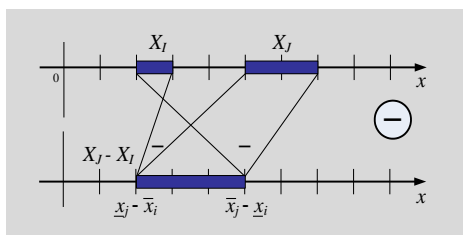
$$X_I + X_J = [\underline{x}_I, \underline{x}_J] + [\bar{x}_I, \bar{x}_J]$$

$$d(X_I + X_J) = d(X_I) + d(X_J)$$



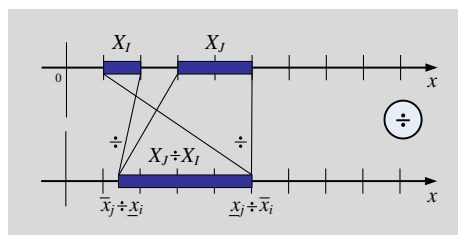
$$X_I \times X_J = [\underline{x}_I, \bar{x}_I] \times [\underline{x}_J, \bar{x}_J]$$

$$d(X_I X_J) \geq \max\{|X_I|d(X_J), |X_J|d(X_I)\}$$



$$X_I - X_J = [\underline{x}_I, \bar{x}_I] + [-1, -1] \times [\underline{x}_J, \bar{x}_J]$$

$$d(X_I - X_J) = d(X_I) + d(X_J)$$



$$X_I \div X_J = [\underline{x}_I, \bar{x}_I] \times [1/\bar{x}_J, 1/\underline{x}_J]$$

$$d(X_I / X_J) \leq |X_I|d(X_J) + |X_J|d(X_I)$$

## ВЫБОР ФУНКЦИИ ПОДСЧЕТА ВРЕМЕНИ ВЫЧИСЛЕНИЙ

- **SessionTime[ ]** возвращает количество секунд реального времени, которые прошли с начала запуска сессии Mathematica.

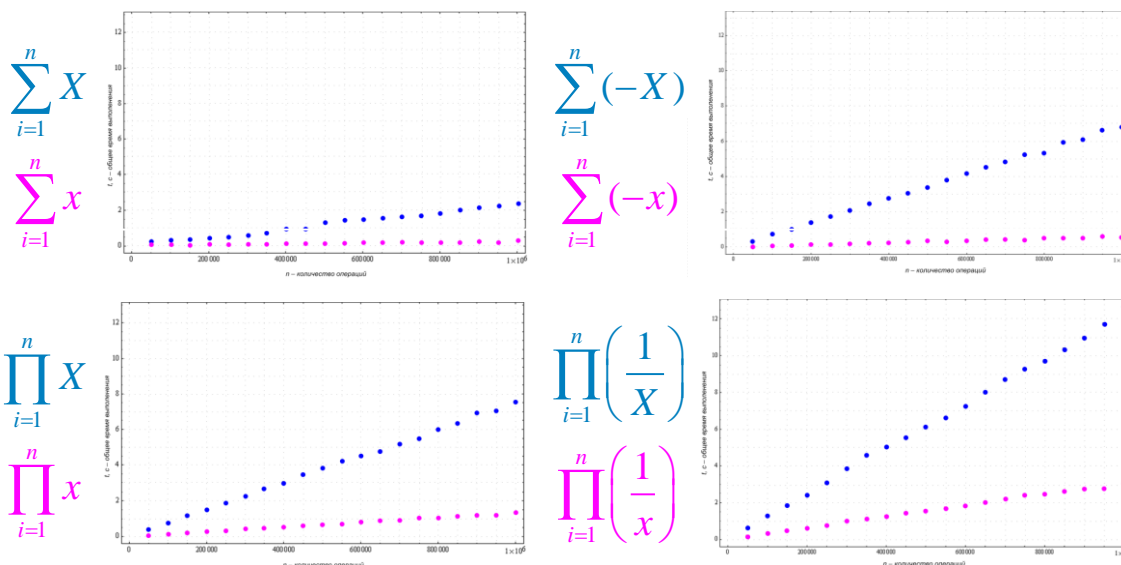
- **Timing[expr]** рассчитывает expr, и возвращает список времени в секундах, вместе с полученным результатом.

- **TimeUsed[ ]** возвращает общее количество секунд времени работы центрального процессора при вычислении текущей сессии Mathematica.

```
b1 = SessionTime[];
a1 = TimeUsed[];
x = Interval[{1.4999, 1.5001}]; n = 500 000
Timing[Sum[{(x)^2 - 10 * Cos[2 * Pi * x]} + 10 * n, {i, 1, n}]]
a2 = TimeUsed[];
b2 = SessionTime[];
a2 - a1
b2 - b1
Null
500 000
{34.616 Second, Interval[{1.11248 * 10^7, 1.11252 * 10^7}]}
34.616
34.6632609
```

## ГРАФИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ РЕЗУЛЬТАТОВ

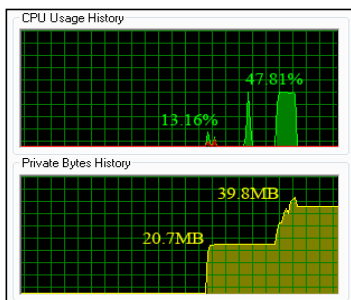
$x = 1.5$ ;  $X = [1.4999, 1.5001]$ ;  $(n, 50000, 1000000, 50000)$



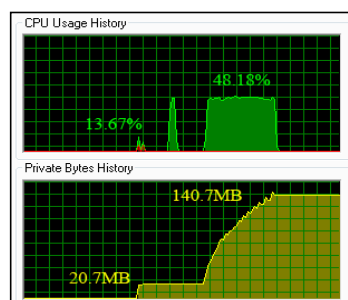
## СИСТЕМНЫЕ ПАРАМЕТРЫ ВЫЧИСЛИТЕЛЬНОГО ЯДРА

CPU: DualCore AMD Athlon 64 X2, 2800 MHz 5400+ RAM: 2048 MB

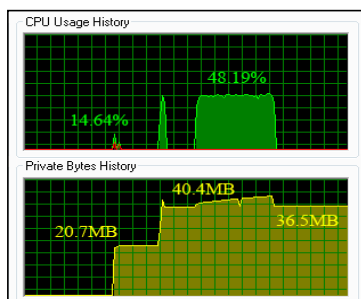
• ADD



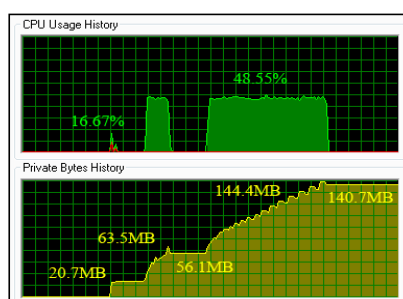
• SUB



• MUL

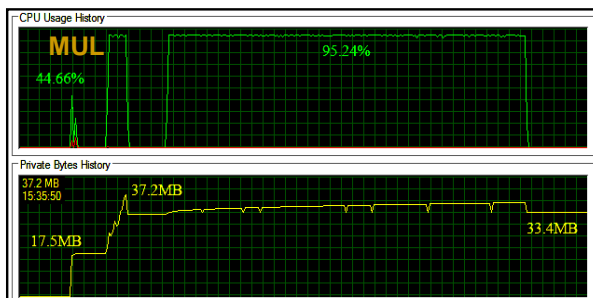
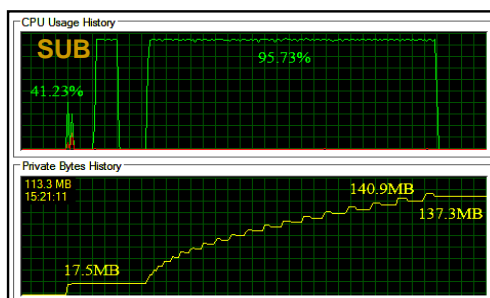
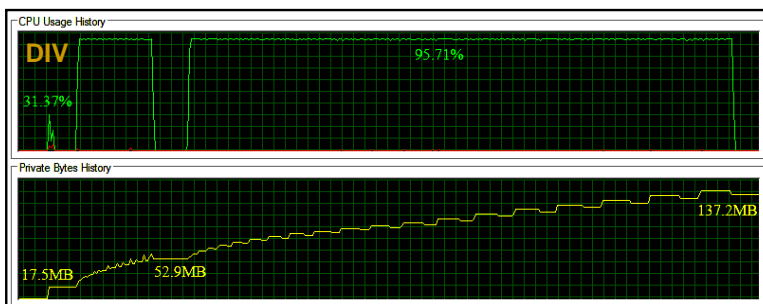
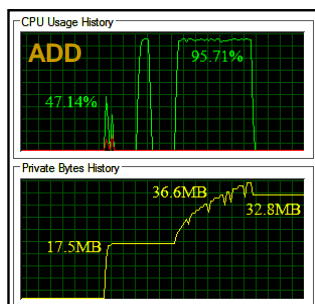


• DIV



## СИСТЕМНЫЕ ПАРАМЕТРЫ ВЫЧИСЛИТЕЛЬНОГО ЯДРА

CPU: Intel Pentium IIIIE, 1000 MHz; RAM: 512 MB



## СРАВНИТЕЛЬНЫЕ РЕЗУЛЬТАТЫ

CPU: DualCore AMD Athlon 64 X2, 2800 MHz 5400+; RAM: 2048 MB

CPU: Intel Pentium IIIЕ, 1000 MHz; RAM: 512 MB

Опера-ция	Среднее время выполнения n операций		k = T'/t'
	t'	T'	
ADD	0.1625	0.8992	5.534
SUB	0.3039	3.1774	10.455
MUL	0.2492	3.5719	14.333
DIV	1.1289	5.8828	5.211

Опера-ция	Среднее время выполнения n операций		k = T'/t'
	t'	T'	
ADD	0.6129	3.8651	6.307
SUB	1.1151	14.0442	12.595
MUL	1.053	19.0694	18.11
DIV	3.9827	28.5967	7.1802

## УНАРНЫЕ ИНТЕРВАЛЬНЫЕ ОПЕРАЦИИ

$$X = [\underline{x}, \bar{x}] = \{x \mid \underline{x} \leq x \leq \bar{x}, \underline{x}, \bar{x} \in R\} \Rightarrow x \in R, X \in I(R);$$

Если

$r(x)$  - непрерывная унарная операция на множестве  $R$ ,

$$\text{то } r(X) = [\min_{x \in X} r(x), \max_{x \in X} r(x)]$$

определяет соответствующую ей операцию на множестве  $I(R)$ .

$$X^k \ (k \in R), \ e^X, \ \ln X, \ \sin X, \ \cos X, \ \dots$$

---


$$r(x) = x^k, \ r(x) = \sqrt{x}, \ \dots \Rightarrow r(X) = [r(\underline{x}), r(\bar{x})]$$



## ФУНКЦИЯ РАСТРИГИНА

$$f = \sum_{i=1}^n (x^2 - 10 \cdot \cos(2\pi x)) + 10n;$$

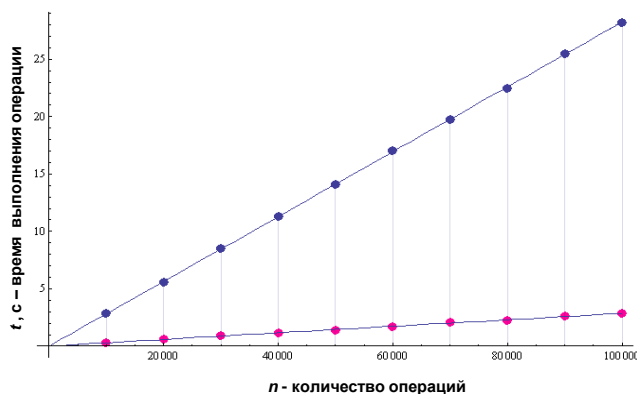
$$x = 1.5;$$

$$X = [1.4999, 1.5001];$$

$$(n, 10000, 100\ 000, 10000)$$

$$y(t) = a^{(t)} x; \quad y(T) = a^{(T)} X;$$

$$x, X \in [0, 100000].$$



$t'$	$T'$	$T'/t'$	$a(t)$	$a(T)$	$a(T)/a(t)$
0.16485	1.21795	<b>7.388</b>	0.00003378	0.00027525	<b>8.14949</b>

## ГИПЕРБОЛИЧЕСКИЙ ЭЛЛИПСОИД

$$f = \sum_{i=1}^n \sum_{j=1}^i x^2;$$

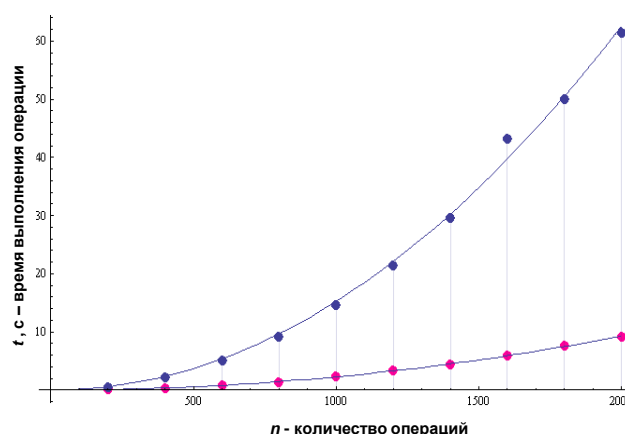
$$x = 1.5;$$

$$X = [1.4999, 1.5001];$$

$$(n, 200, 2\ 000, 200)$$

$$y(t) = a^{(t)} x^{b^{(t)}}; \quad y(T) = a^T X^{b^{(T)}};$$

$$x, X \in [0, 2\ 000].$$



$t'$	$T'$	$T'/t'$	$a(t)$	$a(T)$	$b(t)$	$b(T)$	$a(T)/a(t)$
3.542	23.7232	<b>6.698</b>	$11.664 \cdot 10^{-6}$	$1.8804 \cdot 10^{-6}$	2.028	2.039	<b>6.20306</b>



Донецкий национальный  
технический университет



## Исследование интервального оценивания функций

с использованием СКМ Mathematica 7

07 / 12 / 2010

Аспирант кафедры КИ  
Иваница Сергей Васильевич

### ПОСТАНОВКА ЗАДАЧИ

- Проблематика интервального оценивания функций;
- Представление результатов прямого вычисления интервальных функций;
- Пути оптимизации для корректного (оптимального) получения результирующего интервала;
- Представление результатов вычисления функции с предшествующим ему интервальным оцениванием;
- Сравнительный анализ полученных результатов.

## УНАРНЫЕ ИНТЕРВАЛЬНЫЕ ОПЕРАЦИИ

$$X = [\underline{x}, \bar{x}] = \{x \mid \underline{x} \leq x \leq \bar{x}, \underline{x}, \bar{x} \in R\} \Rightarrow x \in R, X \in I(R);$$

Если

$r(x)$  - непрерывная унарная операция на множестве  $R$ ,

$$\text{то } r(X) = [\min_{x \in X} r(x), \max_{x \in X} r(x)]$$

определяет соответствующую ей операцию на множестве  $I(R)$ .

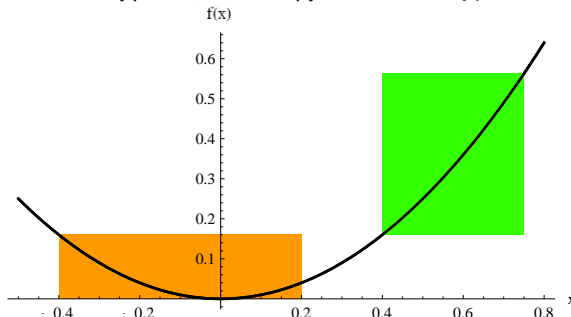
$$X^k (k \in R), e^X, \ln X, \sin X, \cos X, \dots$$

$$r(x) = x^k, r(x) = \sqrt{x}, \dots \Rightarrow r(X) = [r(\underline{x}), r(\bar{x})]$$

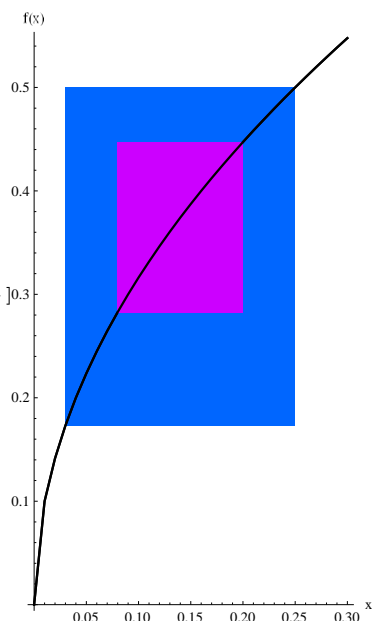
## ПРИМЕРЫ РЕАЛИЗАЦИИ УНАРНЫХ ОПЕРАЦИЙ

$$r(x) = x^k, r(x) = \sqrt{x}, \dots \Rightarrow r(X) = [r(\underline{x}), r(\bar{x})]$$

```
f[x_] := x^2;
x1 = Interval[{-0.4, .2}];
x2 = Interval[{0.4, 0.75}];
f[x1]
f[x2]
Out[91]=
Interval[{0, 0.16}]
Out[92]=
Interval[{0.16, 0.5625}]
```



```
f1[x_] := Sqrt[x];
x1 = Interval[{0.03, 0.25}];
x2 = Interval[{0.08, 0.2}];
f1[x1]
f1[x2]
Out[93]=
Interval[{0.173205, 0.5}]
Out[94]=
Interval[{0.282843, 0.447214}]
```



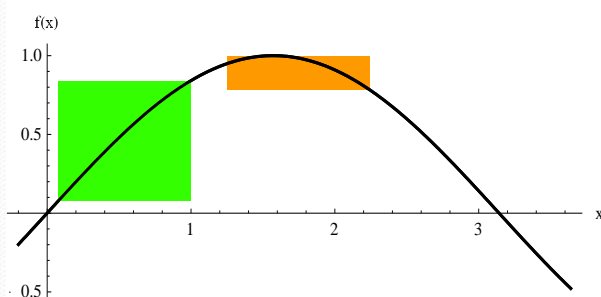
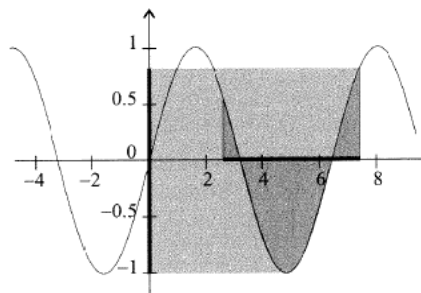
## ПРИМЕРЫ РЕАЛИЗАЦИИ УНАРНЫХ ОПЕРАЦИЙ

$$r(x) = \sin(X) \Rightarrow r(X) = \sin(X) = [\min_{x \in X} \sin(x), \max_{x \in X} \sin(x)];$$

Алгоритм:  $\sin[x] = [r]$ :

1. if  $\exists k \in \mathbb{Z} \mid 2\pi k - \frac{\pi}{2} \in [x]$  then  $\underline{r} = -1$ ;  
else  $\underline{r} = \min(\sin \underline{x}, \sin \bar{x})$ ;
2. if  $\exists k \in \mathbb{Z} \mid 2\pi k + \frac{\pi}{2} \in [x]$  then  $\bar{r} = 1$ ;  
else  $\bar{r} = \max(\sin \underline{x}, \sin \bar{x})$ .

$$\sin([2.6, 7.2]) = [-1, 0.7937];$$



```
f[x_] := Sin[x];
x1 = Interval[{1.25, 2.24}];
x2 = Interval[{0.08, 1}];
N[f[x1]] N[f[x2]]
Out[95]=
Interval[{0.784316, 1}]
Out[96]=
Interval[{0.0799147, 0.841471}]
```

## ВЫЧИСЛЕНИЕ ИНТЕРВАЛЬНЫХ ФУНКЦИЙ

Пример 1:

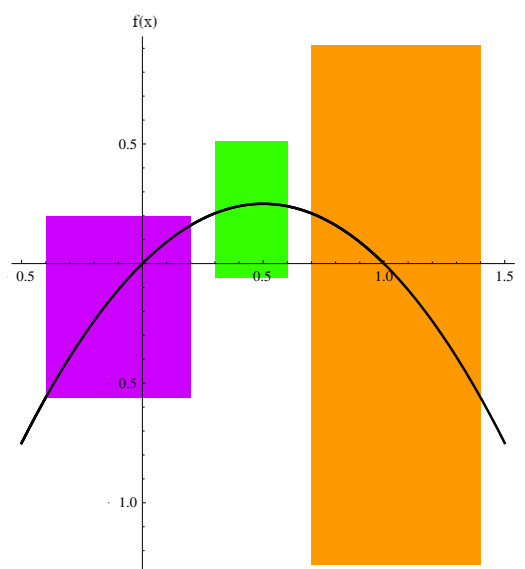
$$r(x) = x - x^2 \Rightarrow r(X) = [\min_{x \in X} (x - x^2), \max_{x \in X} (x - x^2)];$$

```
f[x_] := x - x^2;
x1 = Interval[{0.7, 1.4}];
x2 = Interval[{0.3, 0.6}];
x3 = Interval[{-0.4, 0.2}];
N[f[x1]]
N[f[x2]]
N[f[x3]]
```

```
Out[97]=
Interval[{-1.26, 0.91}]
```

```
Out[98]=
Interval[{-0.06, 0.51}]
```

```
Out[99]=
Interval[{-0.56, 0.2}]
```



## ВЫЧИСЛЕНИЕ ИНТЕРВАЛЬНЫХ ФУНКЦИЙ

**Пример 2:**  $r(x) = \sin(x) \cdot \cos(x) \Rightarrow$

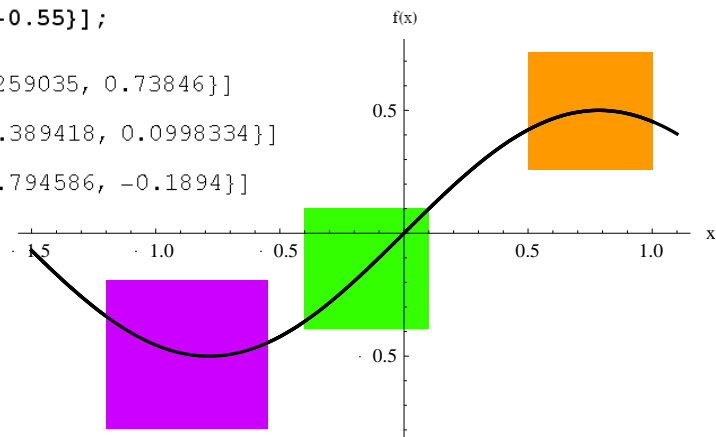
$$\Rightarrow r(X) = [\min_{x \in X} [\sin(x) \cdot \cos(x)], \max_{x \in X} [\sin(x) \cdot \cos(x)]];$$

```
f[x_] := Sin[x] * Cos[x];
x1 = Interval[{0.5, 1}];
x2 = Interval[{-0.4, 0.1}];
x3 = Interval[{-1.2, -0.55}];
```

Out[100]= Interval[{0.259035, 0.73846}]

Out[101]= Interval[{-0.389418, 0.0998334}]

Out[102]= Interval[{-0.794586, -0.1894}]



## ИНТЕРВАЛЬНОЕ ОЦЕНИВАНИЕ ФУНКЦИЙ

Диапазон изменения функции  $f(x, a)$ :

$$\begin{aligned} W(f, X; A^{(0)}, \dots, A^{(m)}) &= \\ &= \{f(x; a^{(0)}, \dots, a^{(m)}) \mid x \in X, a^{(k)} \in A^{(k)}, 0 \leq k \leq m\} = \\ &= \left[ \min_{\substack{x \in X \\ a^{(k)} \in A^{(k)}}} f(x; a^{(0)}, \dots, a^{(m)}), \max_{\substack{x \in X \\ a^{(k)} \in A^{(k)}}} f(x; a^{(0)}, \dots, a^{(m)}) \right] \end{aligned}$$

Интервальна оценка функции  $f(X, A)$ :  $f(x; a^{(0)}, \dots, a^{(m)})$

Для функции  $f(x)$ :

$$W(f, X) = \{f(x) \mid x \in X\} = \left[ \min_{x \in X} f(x), \max_{x \in X} f(x) \right]$$



## АЛГОРИТМ ИНТЕРВАЛЬНОГО ОЦЕНИВАНИЯ

```

f[x_] := ...;

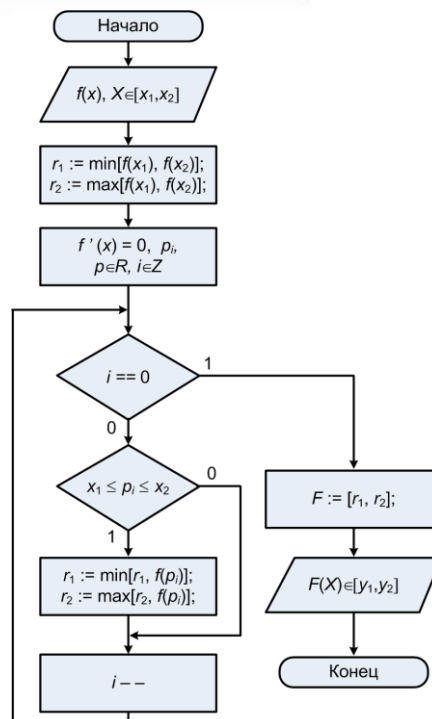
x = Interval[{x1, x2}];

p[i] := NSolve[D[f[x]] == 0, x];

For[k = 0, k ≤ i, k++,
  f1[x1_, x2_] :=
    Interval[
      {Min[f[x1], f[x2],
        f[If[x1 ≤ p[k] ≤ x2, p[k], x1]]],
      Max[f[x1], f[x2],
        f[If[x1 ≤ p[k] ≤ x2, p[k], x1]]]}
    ];
];

```

```
iopt_func[f[x], X, {A[0], A[1], ..., A[n]}]
```



## ПРИМЕРЫ ВЫЧИСЛЕНИЙ

$$r(x) = x - x^2 \Rightarrow r(X) = [\min_{x \in X} (x - x^2), \max_{x \in X} (x - x^2)];$$

```
f[x_] := x - x^2;
```

```

x1 = Interval[{0.7, 1.4}];
x2 = Interval[{0.3, 0.6}];
x3 = Interval[{-0.4, 0.2}];

```

Исходные значения:

```
Out[97]= Interval[{-1.26, 0.91}]
```

```
Out[98]= Interval[{-0.06, 0.51}]
```

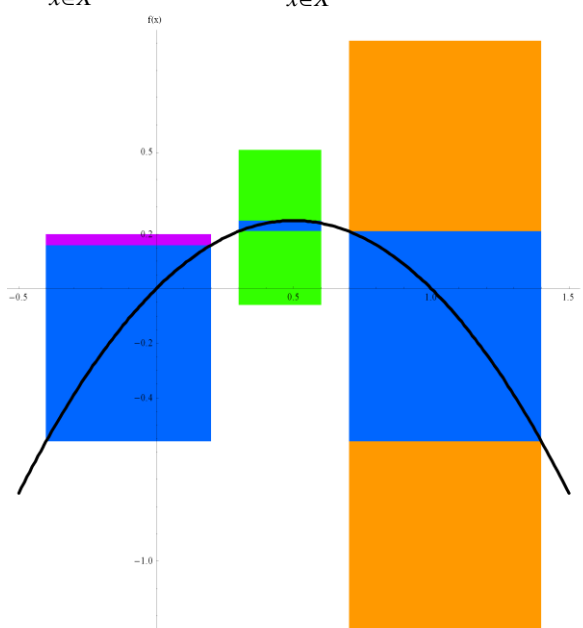
```
Out[99]= Interval[{-0.56, 0.2}]
```

Оптимизированные значения:

```
Out[110]= Interval[{-0.56, 0.21}]
```

```
Out[111]= Interval[{0.21, 0.25}]
```

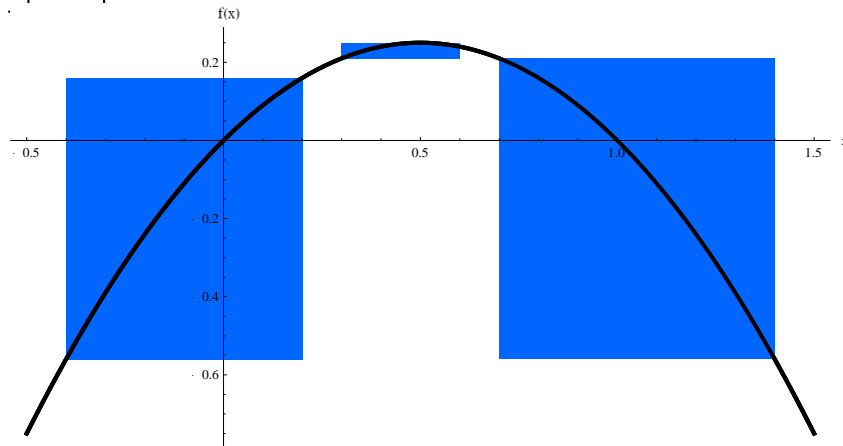
```
Out[112]= Interval[{-0.56, 0.16}]
```



## ТОЖДЕСТВЕННЫЕ ПРЕОБРАЗОВАНИЯ

$$r(x) = x - x^2 \Leftrightarrow r(x) = -\left(\frac{1}{2} - x\right)^2 + \frac{1}{4};$$

`f1[x_] := .1 - x^2`



`x1 = Interval[{0.7, 1.4}];`  
`x2 = Interval[{0.3, 0.6}];`  
`x3 = Interval[{-0.4, 0.2}];`

`Out[113]= Interval[{-0.56, 0.21}]`

`Out[114]= Interval[{0.21, 0.25}]`

`Out[115]= Interval[{-0.56, 0.16}]`

## ТОЖДЕСТВЕННЫЕ ПРЕОБРАЗОВАНИЯ

$$r(x) = \sin(x) \cdot \cos(x) \Leftrightarrow r(x) = \frac{1}{2} \sin(2x);$$

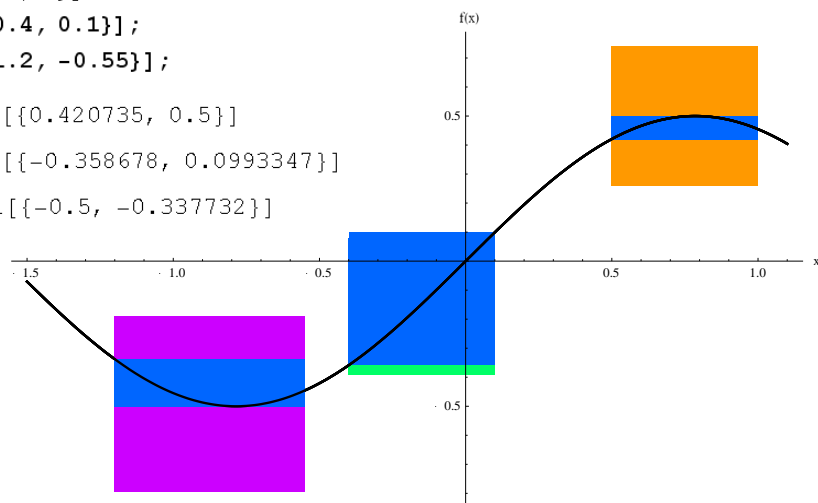
`f1[x_] := 1/2 Sin[2 x];`

`x1 = Interval[{0.5, 1}];`  
`x2 = Interval[{-0.4, 0.1}];`  
`x3 = Interval[{-1.2, -0.55}];`

`Out[100]= Interval[{0.420735, 0.5}]`

`Out[101]= Interval[{-0.358678, 0.0993347}]`

`Out[102]= Interval[{-0.5, -0.337732}]`



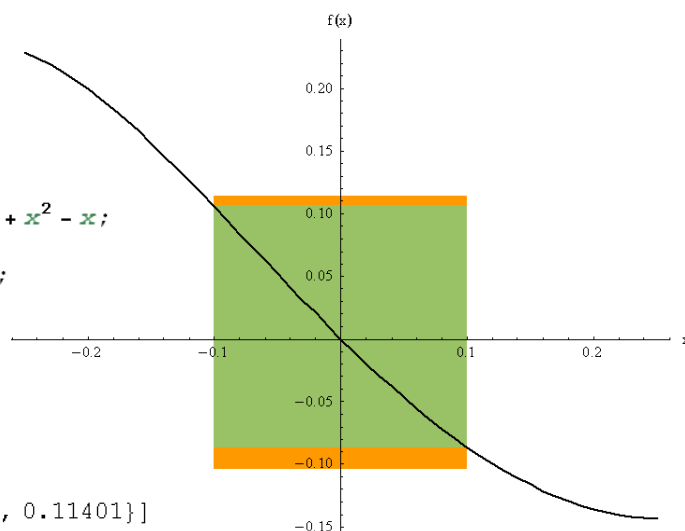
**ПРИМЕРЫ ВЫЧИСЛЕНИЙ**Пример: Полином пятой степени

```
f1[x_] := x5 - 5 * x4 + 4 * x3 + x2 - x;
```

```
x = Interval[{-0.1, 0.1}];
```

```
Out[24]= Interval[{-0.10451, 0.11401}]
```

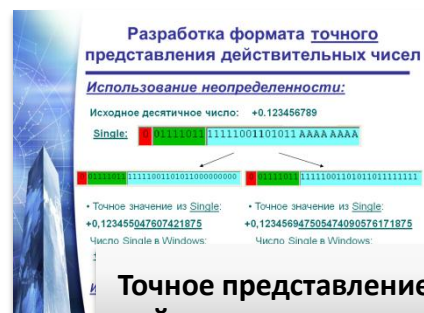
```
Out[25]= Interval[{-0.08649, 0.10549}]
```

**для заметок**

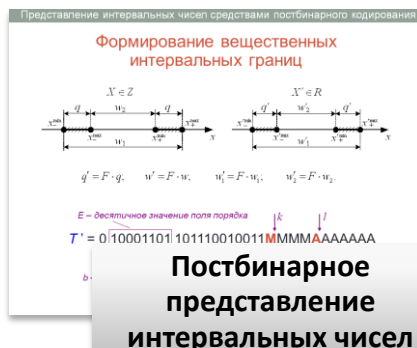
# исследования 2011



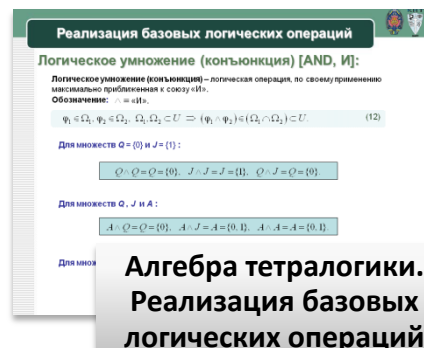
Алгебра тетралогии.  
Реализация унарных  
логических операций



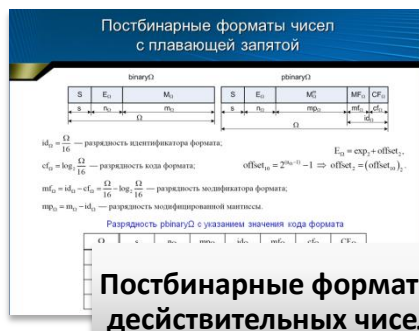
Точное представление  
действительных чисел



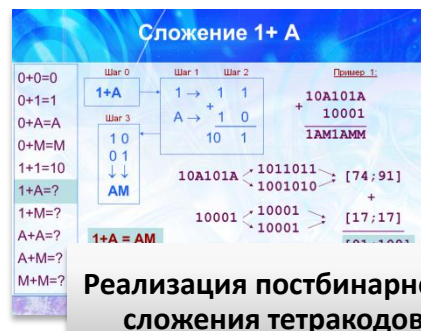
Постбинарное  
представление  
интервальных чисел



Алгебра тетралогии.  
Реализация базовых  
логических операций



Постбинарные форматы  
действительных чисел



Реализация постбинарного  
сложения тетракодов

- Р Компьютерные науки и технологии: следующие 50 лет, 57
- Р Алгебра тетралогии. Реализация нульварных, унарных и бинарных логических операций, 69
- Р Точное представление действительных чисел в контексте расширенного кодо-логического базиса, 79
- Р Реализация логических операций над элементами тетралогии с использованием аксиоматики теории множеств, 84
- Р Постбинарные форматы действительных чисел. Особенности преобразования  $\text{binary} \rightarrow \text{rbinary}$ , 91
- Р Алгебра тетралогии. Реализация арифметических операций: операция сложения, 97
- Р Представление интервальных чисел средствами постбинарного кодирования, 105
- Р Ноографика & ноомоделирование, 113
- Р Тетракоды в вычислениях с гибкой разрядностью, 124
- Р Ноокомпьютинг и будущее информационно-компьютерной инфраструктуры, 139





ИНФОРМАЦИОННЫЕ УПРАВЛЯЮЩИЕ СИСТЕМЫ – 2011

# Компьютерные науки и технологии: следующие 50 лет

Аноприенко Александр Яковлевич

Декан факультета  
компьютерных наук и технологий  
ДонНТУ

12.04.2011



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

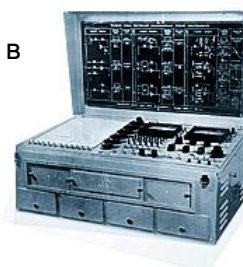


Гагарин сообщил: «Прошу доложить  
но Никите Сергеевичу Хрущеву, что  
льно, чувствую себя хорошо, травм

ека в космиче-  
грандиозные  
человечеством.



Точка отсчета истории  
компьютерных технологий в  
ДонНТУ: **12 апреля 1961**  
года в ДПИ ассистентом  
В.А. Святным проведено  
**самое первое** занятие с  
использованием  
электронной  
вычислительной машины  
(АВМ МН-7)



Советскому космонавту, впервые в мире совершившему космический полет,  
**майору ГАГАРИНУ Юрию Алексеевичу**

Дорогой Юрий Алексеевич!  
Мне доставляет большую радость горячо поздравить Вас с выдающимся героическим подвигом — первым космическим полетом на корабле-спутнике «Восток».

Совершенный Вами полет открывает новую страницу в истории человечества в покорении космоса и наполняет сердца советских людей великой радостью и гордостью за свою социалистическую Родину.

От всего сердца поздравляю Вас со счастливым возвращением из космического путешествия на родную землю. Обнимаю Вас.

До скорой встречи в Москве.

Н. ХРУЩЕВ

12 апреля 1961 года.

**12 апреля 1961 года — Т О В А Р И Щ ,  
ЗАПОМНИ ЭТОТ ДЕНЬ!**

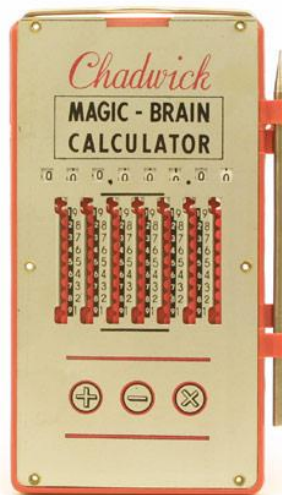


Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Что еще произошло в этом году...

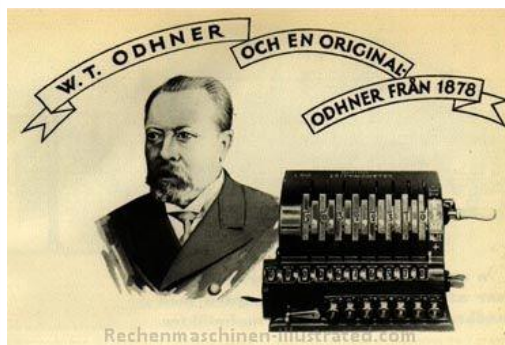
**1961**



COMPUTER  
HISTORY  
MUSEUM

Расцвет производства  
механических калькуляторов...

В СССР к концу 60-х годов выпущено  
около 300 тысяч «Феликсов» и ВК-1...



Rechenmaschinen-illustrated.com



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Что еще произошло в этом году...

**1961**



В Англии разработан  
первый в мире  
электронный  
калькулятор  
**ANITA**

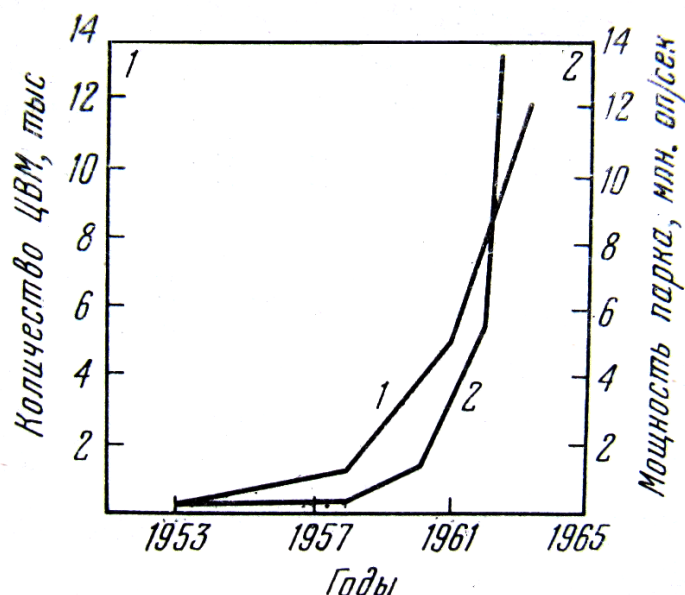
The name *ANITA*, stands variously for  
"A New Inspiration To Arithmetic" and  
"A New Inspiration To Accounting"



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

## 1961



К концу 1961 года  
в мире было  
выпущено  
**5 тыс.** цифровых  
компьютеров с  
суммарной  
производительностью  
**5 млн.** операций в  
секунду



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Что еще произошло в этом году...

## 1961



В 1961 году программист  
С. Рассел возглавил  
проект по созданию  
**первой компьютерной  
игры с графикой**  
«Spacewar», реализация  
которой на первой мини-  
ЭВМ PDP-1 потребовала  
200 человеко-часов

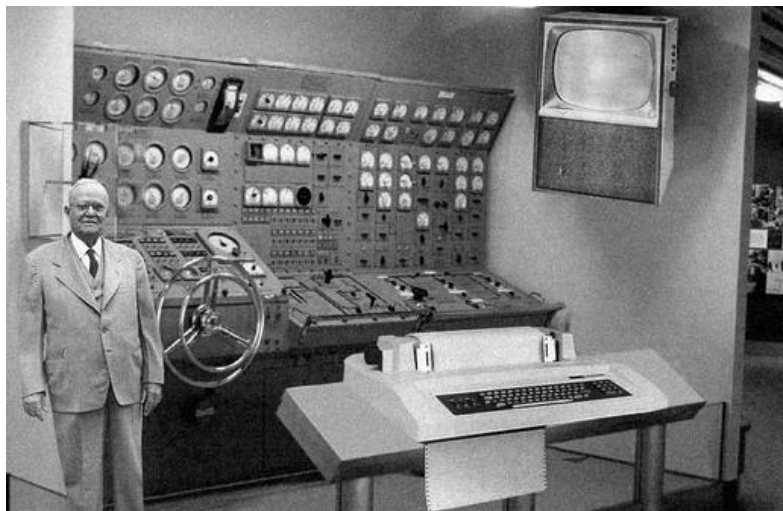


Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Что еще произошло в этом году...

1961



RAND Corp.  
реализовала  
первый прототип  
домашнего  
компьютера



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Что еще произошло в этом году...

1961



IBM IBM IBM

К 50-летию ИБМ

Первая попытка создать  
суперкомпьютер:

IBM 7030 (Stretch)

0,5 Mflops, 1 MB

10 млн. долларов

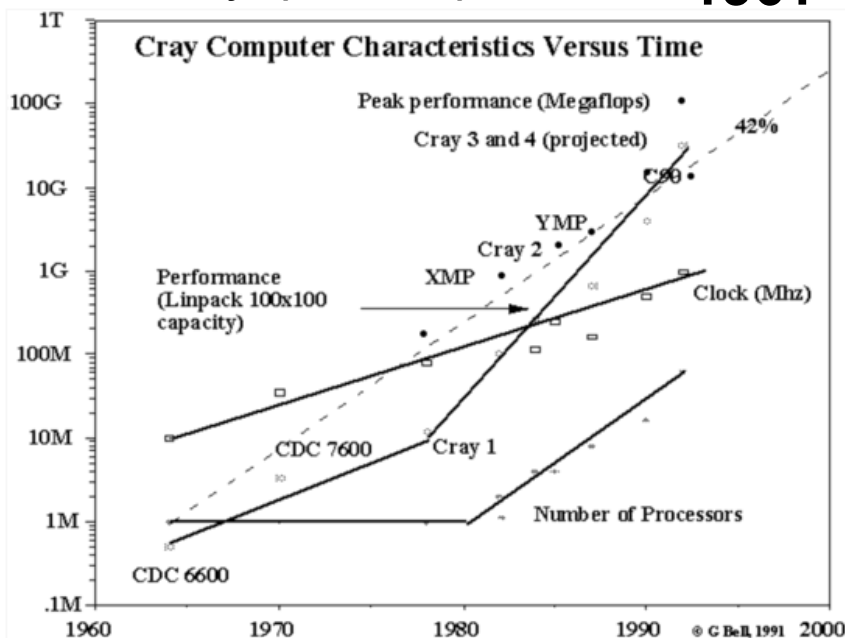




Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

## Начало суперкомпьютерной гонки... 1961



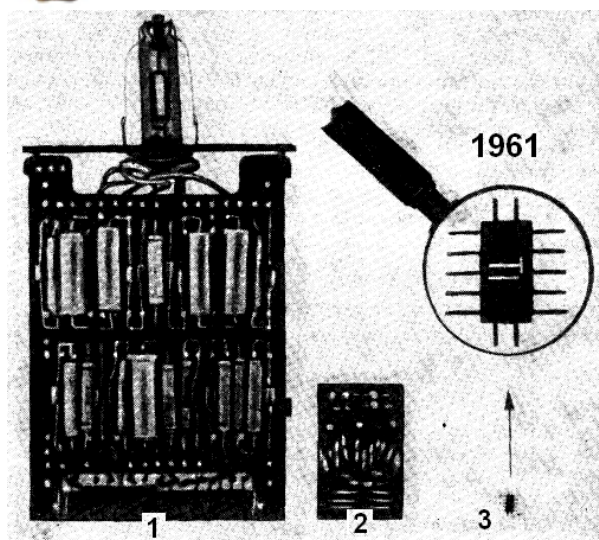
Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

## 1961

В США фирмами «Фаерчайлд» и «Тексас Инструментс» **начат серийный выпуск полупроводниковых интегральных схем.**

Фирмой «Тексас Инструментс» создана **первая экспериментальная ЦВМ на интегральных схемах** (по контракту с ВВС США)



Лампы - транзисторы - ИС





Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Что еще произошло в этом году...

**1961**



Студент **Леонард Клейнрок** (Leonard Kleinrock) из Массачусетского технологического института (Massachusetts Institute of Technology) **опубликовал свою первую статью по пакетной коммутации.**

К 1969 году эти идеи были им реализованы в первом прототипе сети Интернет

Рис. 1.25. Первый интерфейсный процессор сообщений (рядом стоит Л. Клейнрок)

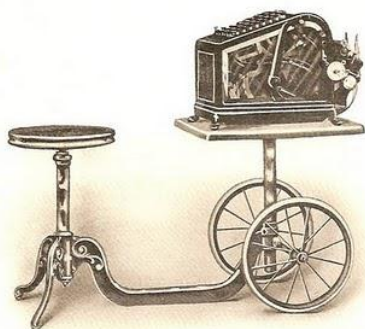


Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

**1961 и 1911:** а что началось еще на 50 лет раньше?

**Мобильный компьютеринг и параллельные вычисления**



В 1911 году фирма Барроуз выпускает 100-тысячный арифмометр

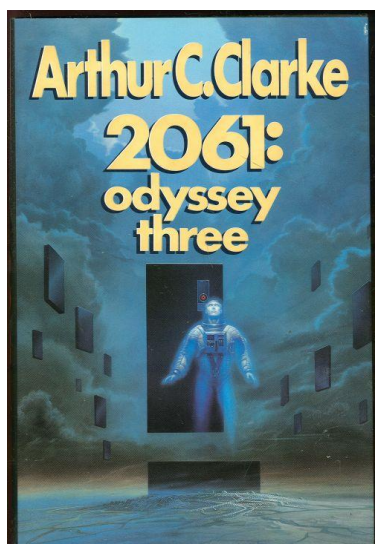




Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Заглянуть  
в будущее...



Артур Кларк:  
«фантастика будущего»

«Космическая Одиссея»

2001 – Стенли Кубрик, 1968

2010 – «Космонавт Леонов»

2061 – ... (1987)

Три закона Кларка (1961, «Профили будущего»):

**Первый Закон:** Если заслуженный, но престарелый учёный говорит, что нечто возможно, он почти наверняка прав. Если же он говорит, что нечто невозможно, он почти определённо ошибается.

**Второй Закон:** Единственный путь обнаружить пределы возможного — уйти за эти пределы, в невозможное.

**Третий Закон:** Любая достаточно ушедшая вперед технология неотличима от магии.



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

**Хронология авторских прогнозов:**

**1993:** Аноприенко А.Я. **Пределы информатики** // «Информация и рынок». Теоретический и научно-практический журнал. – 1993. – №2-3. С. 10-14.

**1997:** Аноприенко А.Я. **Новое познание в новом мире: научные исследования и высшее образование в условиях информационной супермагистрالی** // TEMPUS/TACIS Conference „Computer Networks in Higher Education“, 26-28 May 1997, National Technical University of Ukraine, Kyiv.

**2009:** Аноприенко А.Я. **Компьютерные науки и технологии в прошлом, настоящем и будущем** // Материалы V международной научно-технической конференции студентов, аспирантов и молодых ученых «Информатика и компьютерные технологии» – 24-26 ноября 2009 г., Донецк, ДонНТУ, 2009. С.15-26.

**2010:** Аноприенко А.Я. **Вызовы времени и постбинарный компьютеринг** // Информатика и компьютерные технологии / Материалы VI международной научно-технической конференции студентов, аспирантов и молодых ученых – 23-25 ноября 2010 г. Т. 1. Донецк, ДонНТУ. – 2010. С. 13-31.

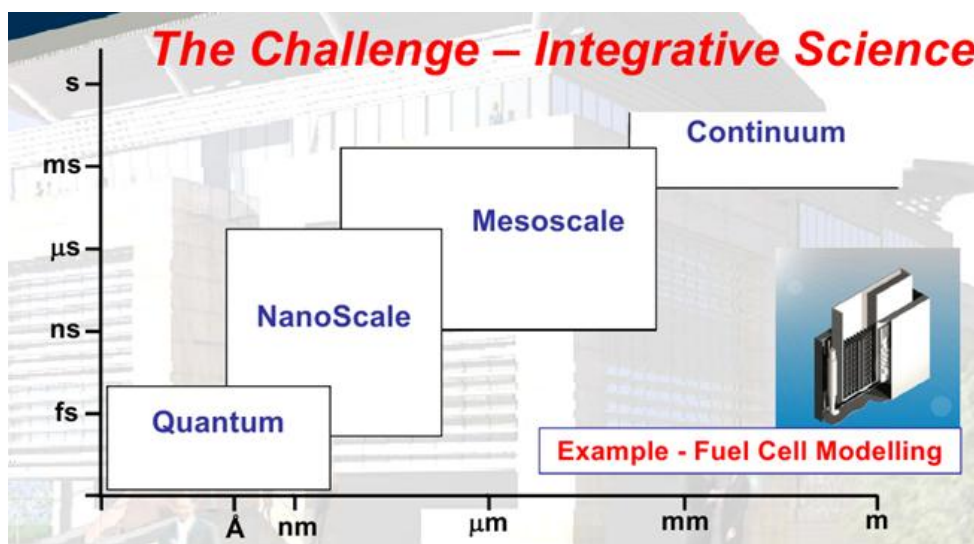
**2011: Конкретнее и «дальше за горизонт» !!!**



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

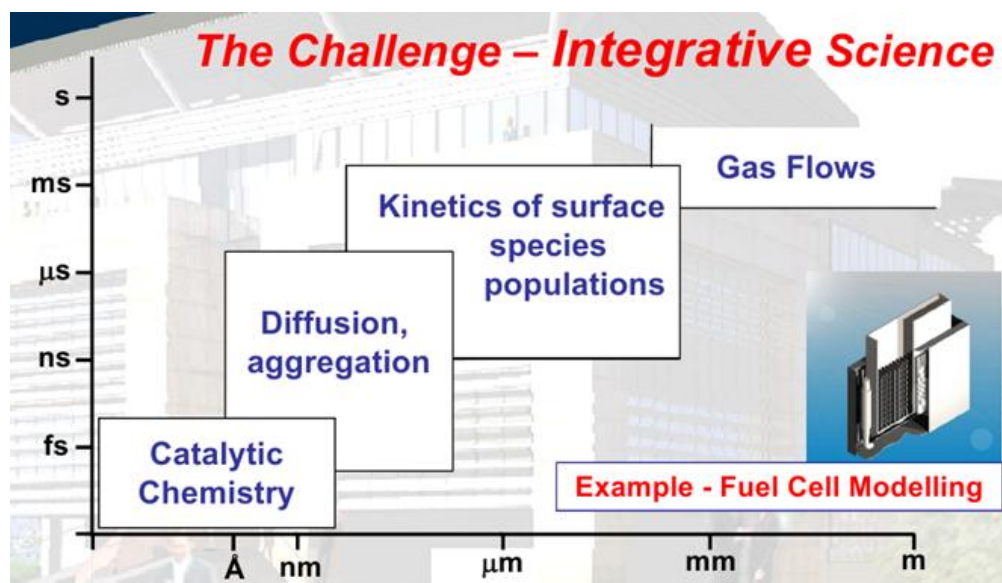
Эволюция компьютерного моделирования как главного  
потребителя вычислительных мощностей: от секунд и метров  
к фемосекундам и нанометрам



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Моделирование топливных элементов: от секунд и метров  
к фемосекундам и нанометрам



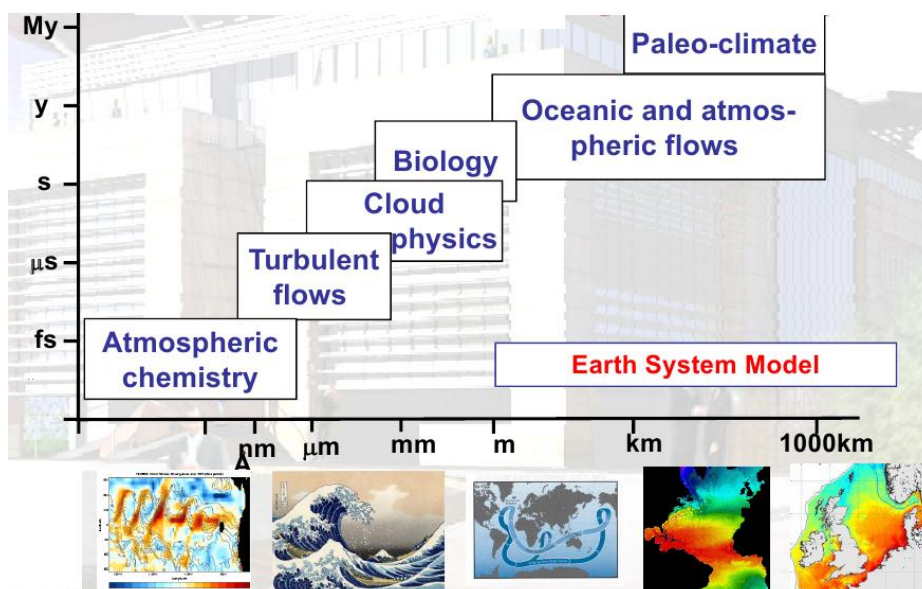




Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Моделирование климата: от миллионов лет и тысяч километров  
к фемосекундам и нанометрам



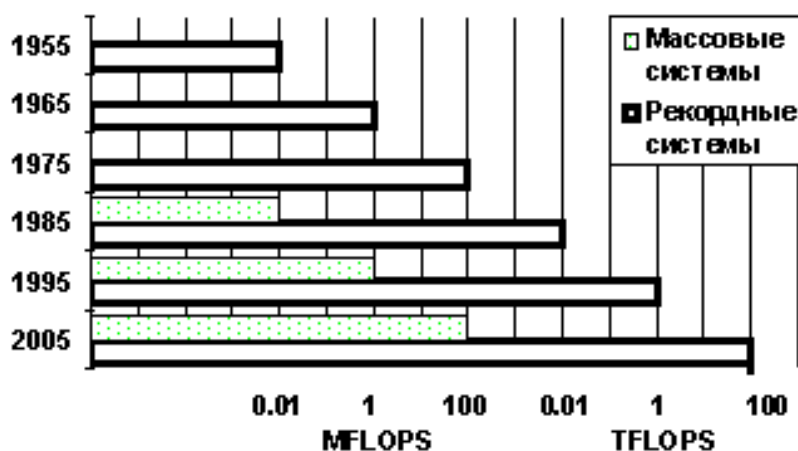
The Hartree Centre – An International Centre of Excellence in CSE



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии  
в прошлом, настоящем и будущем

Прогноз роста производительности 1996 года:  
массовые и рекордные системы



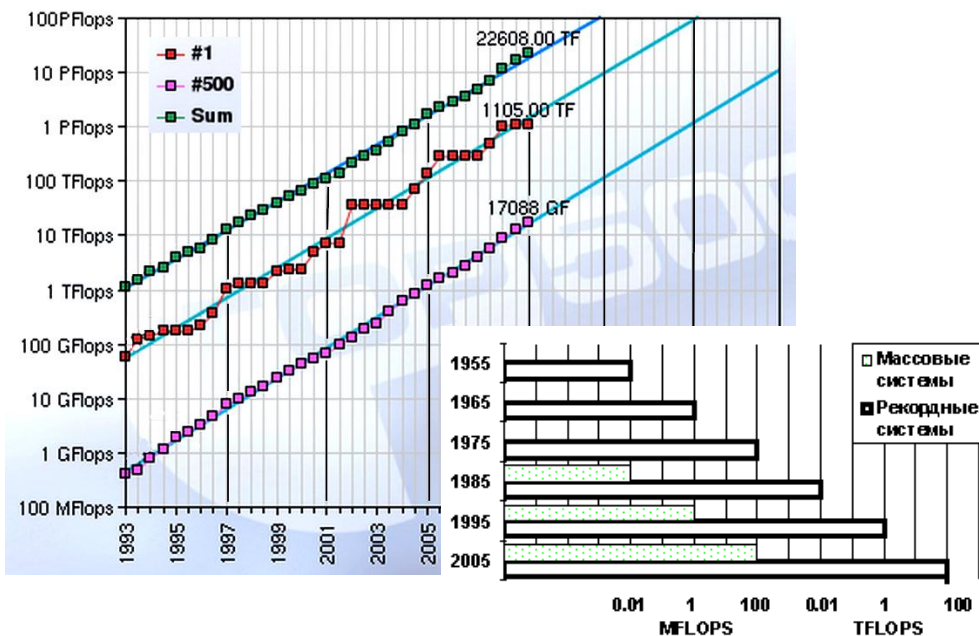


Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии  
в прошлом, настоящем и будущем

1996: 1 ТФлопс

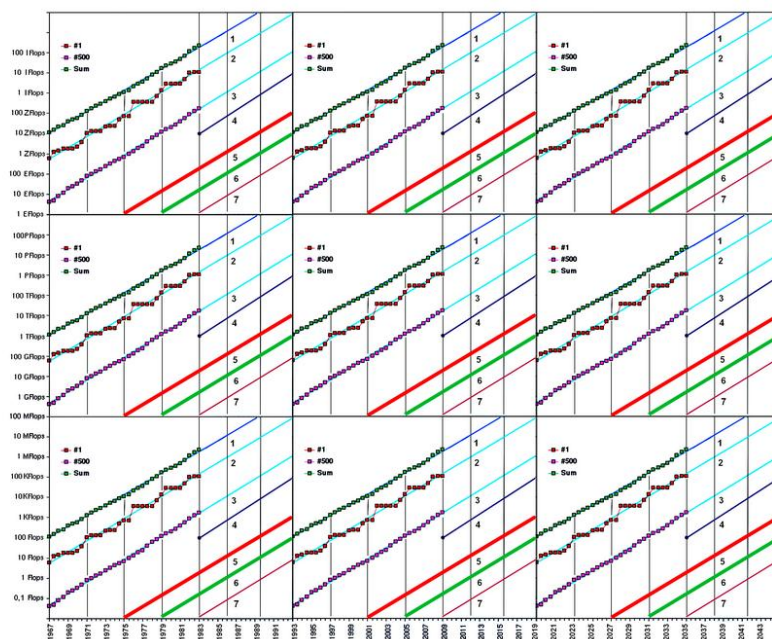
2005: 100 ТФлопс



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Что может дать экстраполяция тенденций 1993-2018 ?





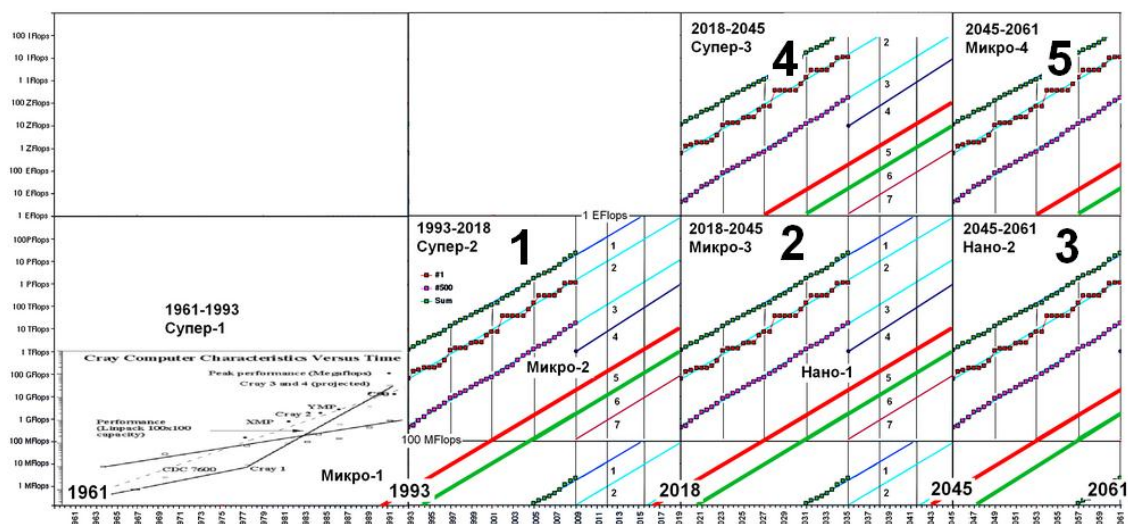


Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

1961-2011-2061 !!!

**Периодическая таблица на следующие 50 лет:**  
3 потока (супер, микро, нано), 5 основных прогнозных блоков



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет



ЭЦВМ Минск-1

**ДонНТУ**

1961 год:  
АВМ МН-7

1962 год:  
ЦВМ «Минск-12»:  
3 тыс. оп/с, 8 КВ



**2011 год:**

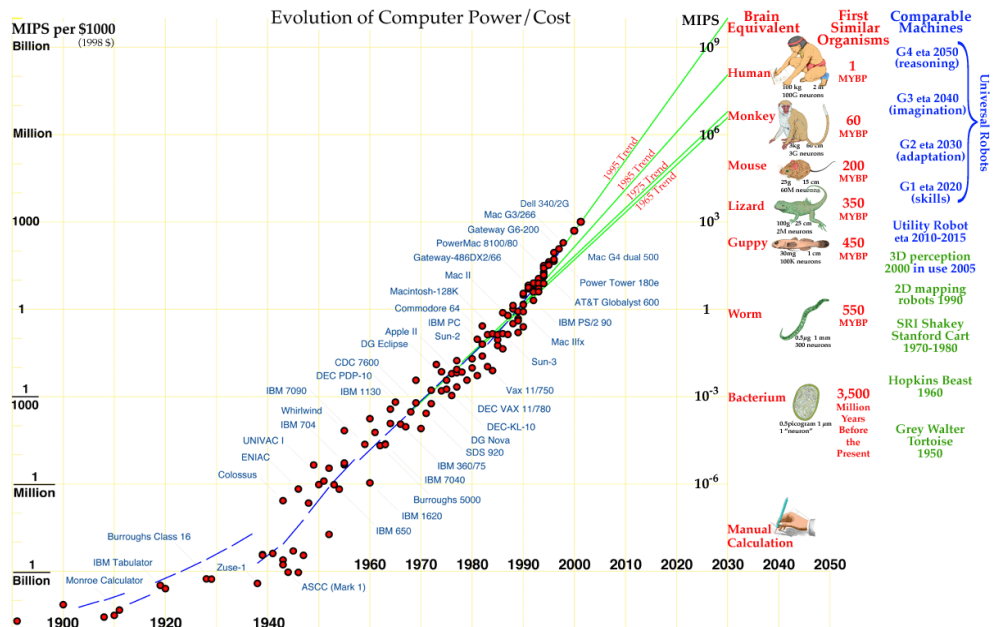
**Суперкомпьютер NEC Xeon Linux Cluster**, введенный в эксплуатацию в ДонНТУ  
1 сентября 2010 года:  
100 CPU, 200 ядер,  
производительность до **3-х Тфлопс**.



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Частный случай: «интеллект» за тысячу долларов (по Курцвейлу)



Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

Сегодня  
мы знаем лишь  
5%  
того, что будем знать  
через 50 лет



ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ  
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

**АЛГЕБРА ТЕТРАЛОГИКИ**

*Реализация нульарных,  
унарных и бинарных  
логических операций*

аспирант кафедры КИ  
Иваница С.В.

09.02.2011

**ЛОГИКА**



## Логика - определения в Интернете:

- [www.ru.wikipedia.org/wiki/Логика](http://www.ru.wikipedia.org/wiki/Логика)  
Логика (λογική — «наука о рассуждении», «искусство рассуждения» от λόγος — «речь», «рассуждение», «мысль») — раздел философии, наука о формах, методах и законах интеллектуальной познавательной деятельности, формализуемых с помощью логического языка. ...
- [www.type-lab.ru/ru/articles/notion.html](http://www.type-lab.ru/ru/articles/notion.html)  
мыслительная функция психики человека, обрабатывающая информацию и принимающая решения объективно, беспристрастно ...
- [www.ru.laser.ru/authors/megeed\\_ovcharov/18.htm](http://www.ru.laser.ru/authors/megeed_ovcharov/18.htm)  
способ мышления или рассуждения, устанавливающий необходимую связь явлений объективного мира; метод оценки связей, свойств и отношений объектов, основанный не на личном восприятии, а на системе формально принятых норм и критериев.
- [pinkbeary.wordpress.com/dict/](http://pinkbeary.wordpress.com/dict/)  
точная наука о причинах и следствиях, возможности создания прямых и обратных выводов на основании имеющихся фактов.
- [www.artema.fopf.mipt.ru/lib/language.html](http://www.artema.fopf.mipt.ru/lib/language.html)  
то, чего нигде нет, но именно её использует УМ в процессе думанья. Это такая штука, которая предоставляет правила как из разных слов и значков лепить леммы, теоремы и доказательства. Как потом, из всего этого добра, проводя ЛОГИЧЕСКИЕ РАССУЖДЕНИЯ, строить научные теории...

3

## Логика - определения в Интернете:

- [www.igrology.ru/25212](http://www.igrology.ru/25212)  
умение находить связь между причинами и следствиями, определять из множества вариантов развития событий наиболее выгодный. Навык используется практически во всех интеллектуальных настольных играх.
- [soveticus5.narod.ru/755/01.htm](http://soveticus5.narod.ru/755/01.htm)  
выработанный разумом инструментарий конструирования однозначно толкуемых мысленных (умозрительных) моделей. Логика обеспечивает неискаженный обмен представлениями, в силу чего становится возможным расширенное воспроизводство знаний.
- [www.aspirantura.info/lexicon.html](http://www.aspirantura.info/lexicon.html)  
ход рассуждений, умозаключений; внутренняя закономерность вещей и явлений, разумность.
- [www.aspirantura.spb.ru/forum/showthread.php](http://www.aspirantura.spb.ru/forum/showthread.php)  
способ доказательства без применения математики.
- [www.metromir.ru/voc\\_dal/P093.HTM](http://www.metromir.ru/voc_dal/P093.HTM)  
наука здравомыслия, наука правильно рассуждать; умословие. Логик - правильный и здравый мыслитель, знающий науку правильного рассуждения.
- [www.filo-ru.ru/termines3.html](http://www.filo-ru.ru/termines3.html)  
формальная наука об общезначимых формах и средствах мысли, необходимых для рационального познания в любой отрасли знаний.
- [www.persona.mgsu.ru/courses/psih-ped\\_slov.htm](http://www.persona.mgsu.ru/courses/psih-ped_slov.htm)  
(логические конструкции, понятия) как «способ перехода от известного к неизвестному».

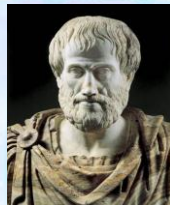
4

## Историческая справка

### 1 этап – **формальная логика**

**Аристотель (384-322гг. до н.э.)**

Ввел основные формулы абстрактного мышления



### 2 этап – **математическая логика**

**немецкий ученый и философ**

**Готфрид Вильгельм Лейбниц (1642-1716)**

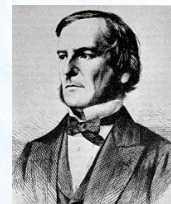
Предпринял попытку логических вычислений



### 3 этап – **алгебра высказываний (булева алгебра)**

**английский математик Джордж Буль (1815-1864)**

Ввел алфавит, орфографию и грамматику математической логики



5

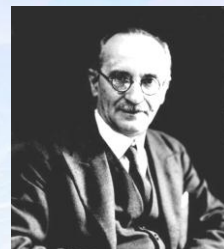
# МНОГОЗНАЧНАЯ ЛОГИКА



## Многозначная логика

**Многозначная логика** — тип формальной логики, характерный наличием более чем двух возможных истинностных значений.

**Исторически первая многозначная логика — троичная логика (трёхзначная логика) польский математик Ян Лукасевич (1898-1956)**



«Сétунь» — малая ЭВМ на основе троичной логики, разработанная в вычислительном центре МГУ в 1959 г.

Руководитель проекта — Н. П. Брусенцов.

Технические характеристики:

Тактовая частота процессора — 200 кГц.

Производительность — 4800 оп/сек

ОЗУ на ферритовых сердечниках — 162 9-разрядных ячейки.

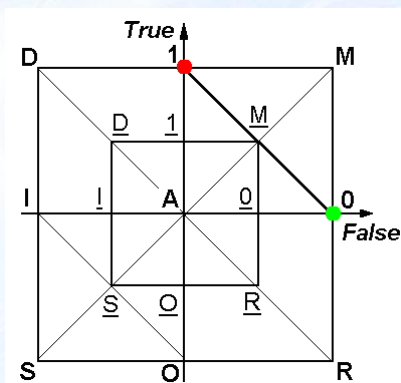
Потребляемая мощность — 2,5 кВт.

7

## Гиперлогика

### Расширенное двумерное логическое пространство

может быть порождено базисом, состоящим из ортонормированной системы векторов «Истина» (может обозначаться как Т - True или Y - «Yes») и «Ложь» (F - False или N - «No») с положительной и отрицательной полуосями.



**1 и 0** - значения «истина» и «ложь» классической логики;

**A** - абсолютная неопределенность, «непроявленность», неизвестность;

**M** - множественность, многозначность («истина» и «ложь» одновременно);

**S** - симметричность (инверсная многозначность, отражение M относительно точки A);

**I и O** - инверсные «истина» и «ложь»;

**D и R** - мнемонически соответствуют понятиям «дублирование» и «репликация», т.е. формы многозначности, по разлому комбинирующие свойства значений M и S.

8

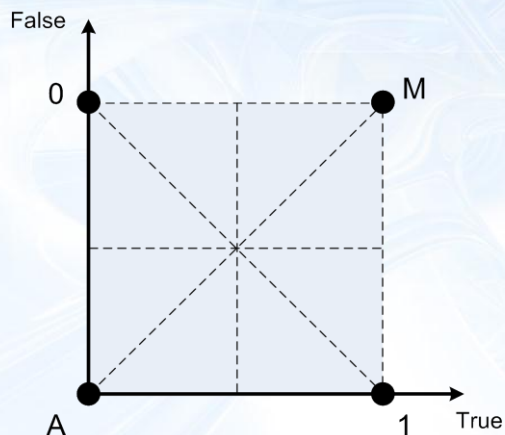


# ТЕТРАЛОГИКА

## Тетралогика и тетракоды

$$L_k = \{x_1, x_2, \dots, x_k\}$$

$k^n$  –  $n$ -разрядное пространство  
в  $k$ -значной логике



Тетралогика:

$$L_4 = \{0, 1, A, M\}$$

Тетракод:

$$C_4 = \{0, 1, A, M\}$$

$$k=4 \Rightarrow 4^n = 2^{2n}$$

0 и 1 - значения «ложь» и  
«истина» двоичной логики;  
A - на данный момент  
неизвестно, «ложь» или  
«истина»;  
M - «ложь» и «истина»  
одновременно.

## Алгебра тетралогики

**Тетрит** (англ. **tetrit**) («тетра» + «бит») — единица измерения информации источников с четырьмя равновероятными сообщениями.

**Тетрафункция:**  $\text{Tr}^n \rightarrow \text{Tr}$ ,  
где  $\text{Tr} = \{0, 1, A, M\}$  — выбранное множество тетралогики,  
 $n \geq 0$  — арность или местность функции.

Максимальное количество возможных логических операций  
тетралогики определимо как число функций в  $k$ -значной системе:

$$(k)^{k^n \cdot m},$$

где  $k$  — число знаков (основание системы счисления),  $n$  — число аргументов (входов), а  $m$  — число выходов

11

## Нульарные логические операции



$T = \{0, 1, A, M\}$  — нульарная логическая операция

*Количество нульарных логических операций:*

$$(4)^{4^0 \cdot 1} = 4 \quad (k = 4, n = 0, m = 1)$$

0 — логический тождественный ноль;

1 — логическая тождественная единица;

A — неопределенность логического состояния, принимающая значение тождественных нуля или единицы;

M — множественность логического состояния, принимающая значение тождественных нуля и единицы.

12



## Унарные логические операции



Количество унарных логических операций  
(в отличие от 4-х и 27-ми унарных функций двоичной  
и троичной логики):

$$(4)^{4 \cdot 1} = 4^4 = 2^8 = 256 \quad (k=4, n=1, m=1)$$

Таблица (начало) - Простейшие унарные функции

0000  
0001  
000A  
000M  
...  
MMM0  
MMM1  
MMAA  
MMMM

№	x				Название функции	Обозначение
	A	I	M	O		
1	0	0	0	0	тождественный ноль	O(x)=0
29	0	1	M	0	минимизация неопределенности	MIN_A(x)
40	0	A	1	M	циклический сдвиг (поворот, вращение) назад на 1 (1/4 оборота) или вперед на 3 (3/4 оборота)	SHIFT_1B(x), SHIFT_3F(x)
55	0	M	1	A	вероятностная инверсия по оси «FALSE»	SWAP_FL(x)
86	1	1	1	1	тождественная единица	I(x)=1
93	1	1	M	0	максимизация неопределенности	MAX_A(x)
100	1	A	0	M	вероятностная инверсия по оси «TRUE»	SWAP_TR(x)
115	1	M	0	A	циклический сдвиг (поворот, вращение) вперед на 1 (1/4 оборота) или назад на 3 (3/4 оборота)	SHIFT_1F(x), SHIFT_3B(x)

13

## Унарные логические операции

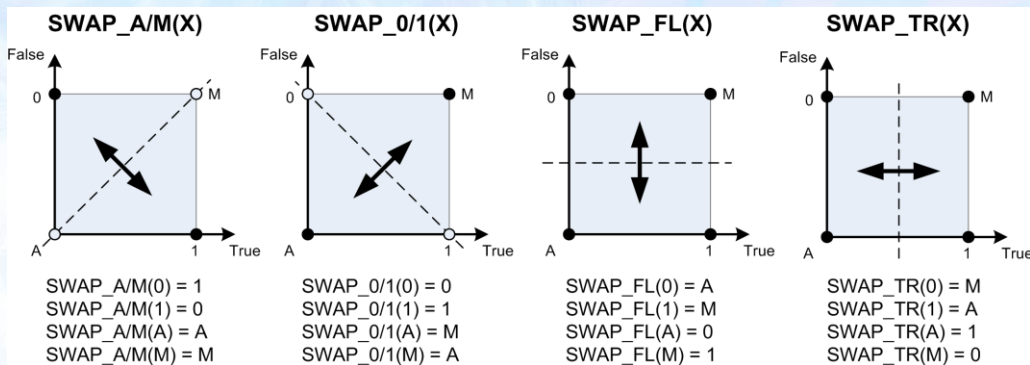
Таблица (продолжение) - Простейшие унарные функции тетралогии

№	x				Название функции	Обозначение
	A	I	M	O		
115	1	M	0	A	циклический сдвиг (поворот, вращение) вперед на 1 (1/4 оборота) или назад на 3 (3/4 оборота)	SHIFT_1F(x), SHIFT_3B(x)
142	A	0	M	1	инверсия около неопределенности и множественности (традиционная инверсия)	$\bar{x}$ , #x, SWAP_A/M(x)
145	A	1	0	0	минимизация множественности	MIN_M(x)
149	A	1	1	0	максимизация множественности	MAX_M(x)
157	A	1	M	0	тождественная функция, повторитель	$x$ , SHIFT_0B(x), SHIFT_0F(x)
171	A	A	A	A	абсолютная неопределенность	A(x)=A
202	M	0	A	1	циклический сдвиг (поворот, вращение) вперед на 2 (1/2 оборота) или назад на 2 (1/2 оборота)	SHIFT_2F(x), SHIFT_2B(x)
217	M	1	A	0	инверсия около нуля и единицы	SWAP_0/1(x)
256	M	M	M	M	абсолютная множественность	M(x)=M

14

## Унарные логические операции

### Унарные логические операции инверсной группы (swap group)



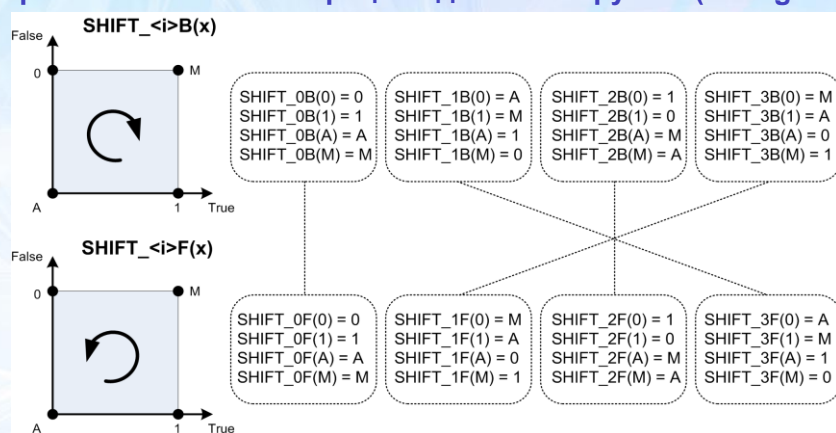
Закон «двойное отрицание» высказывания X:

$$\text{SWAP}_\Psi(\text{SWAP}_\Psi(X)) \leftrightarrow X \quad \text{при } \Psi = \{A/M, 0/1, FL, TR\}.$$

15

## Унарные логические операции

### Унарные логические операции сдвиговой группы (shift group)



Закон  $k$ -ного сдвига в  $k$ -ичной логике, согласно которому  $k$  сдвигов вперед/назад равносильны повторению (утверждению)

$$\text{SHIFT}_{<i>\vartheta} \leftrightarrow X \quad \text{при } \vartheta = \{B, F\}, i = (\omega)_{\text{mod}4} = 0$$

16

## Бинарные логические операции



Количество бинарных логических операций  
(в отличие от 16 и 19 683 бинарных функций двоичной  
и троичной логики):

$$(4)^{4^2 \cdot 1} = 4^{16} = 2^{32} = 4\,294\,967\,296 \quad (k=4, n=2, m=1)$$

Логическое умножение (конъюнкция):  $\wedge \equiv \text{"AND"}$

Логическое сложение (дизъюнкция):  $\vee \equiv \text{"OR"}$

$$X \wedge Y = \min(X, Y), \quad X \vee Y = \max(X, Y) \quad X, Y \in \{0, 1, A, M\}$$

$$BMIN\_A(A, M) = A, \quad BMAX\_A(A, M) = M$$

$$BMIN\_M(A, M) = M, \quad BMAX\_M(A, M) = A$$

17

## Бинарные логические операции

Таблица - Реализация BMIN

BMIN_A BMIN_M	0	1	A	M
0	0	0	0	0
1	0	1	A	M
A	0	A	A	A M
M	0	M	A M	M

Таблица - Реализация BMAX

BMAX_A BMAX_M	0	1	A	M
0	0	1	A	M
1	1	1	1	1
A	A	1	A	M A
M	M	1	M A	M

$$\wedge^{<A>} \equiv BMIN\_A, \quad \wedge^{<M>} \equiv BMIN\_M.$$

$$\underline{\wedge}$$

$$\overline{\wedge}$$

$$\vee^{<A>} \equiv BMAX\_A, \quad \vee^{<M>} \equiv BMAX\_M.$$

$$\underline{\vee}$$

$$\overline{\vee}$$

18



## Бинарные логические операции

Законы логических операций конъюнкции и дизъюнкции тетралогии:

### 1) Законы коммутативности (переместительности):

$$X \wedge Y \equiv Y \wedge X, \quad X \vee Y \equiv Y \vee X.$$

Например,  $A \bar{\wedge} M \equiv M \bar{\wedge} A \Leftrightarrow M \equiv M$ ;  $A \bar{\vee} M \equiv M \bar{\vee} A \Leftrightarrow A \equiv A$ ;  
 $A \bar{\vee} M \equiv M \bar{\vee} A \Leftrightarrow A \equiv A$ ,  $A \bar{\wedge} M \equiv M \bar{\wedge} A \Leftrightarrow M \equiv M$ ;

### 2) Законы ассоциативности (сочетательности):

$$(X \wedge Y) \wedge Z \equiv X \wedge (Y \wedge Z), \quad (X \vee Y) \vee Z \equiv X \vee (Y \vee Z).$$

Например,  $(A \bar{\wedge} M) \bar{\wedge} M \equiv A \bar{\wedge} (M \bar{\wedge} M) \Leftrightarrow M \equiv M$ ;  
 $(M \bar{\vee} A) \bar{\vee} A \equiv M \bar{\vee} (A \bar{\vee} A) \Leftrightarrow A \equiv A$ .

### 3) Законы дистрибутивности (распределительности):

$$X \wedge (Y \vee Z) \equiv (X \wedge Y) \vee (X \wedge Z), \quad X \vee (Y \wedge Z) \equiv (X \vee Y) \wedge (X \vee Z).$$

Например,  $A \bar{\wedge} (M \bar{\vee} A) \equiv (A \bar{\wedge} M) \bar{\vee} (A \bar{\wedge} A) \Leftrightarrow A \equiv A$ ;  
 $A \bar{\vee} (M \bar{\wedge} A) \equiv (A \bar{\vee} M) \bar{\wedge} (A \bar{\vee} A) \Leftrightarrow A \equiv A$ .

19

*для заметок*







## **Разработка формата точного представления действительных чисел**

### **Теоретический инструментарий:**

- Тетралогика;
- СОК – система остаточных классов (несократимые дроби Фарея);
- Аппроксиметика Юровицкого;
- Интервальный анализ.

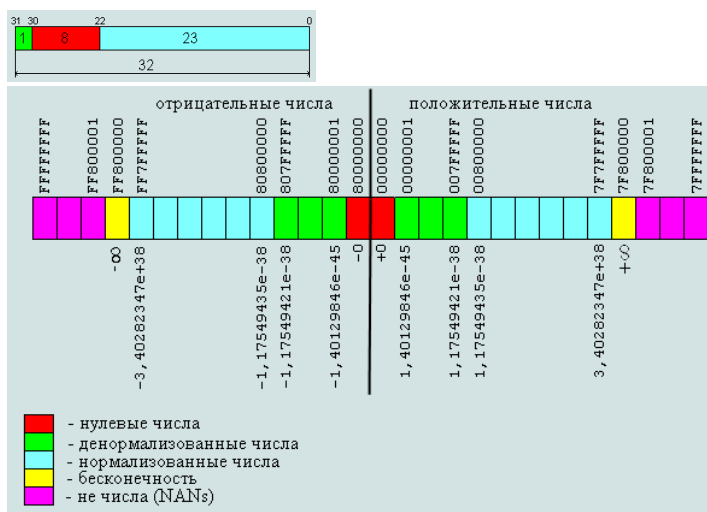
## Разработка формата точного представления действительных чисел

### Формат исходных данных с учетом неопределенности:

- Основа: стандарт IEEE754;
- Основные форматы: 32 бита и 64 бита?
- Нормализованные ( $1 \leq M < 2$ ) и денормализованные ( $0.1 \leq M < 1$ ) числа?
- Положительный и отрицательный нули?
- Положительная и отрицательная бесконечность?
- Не числа (NaNs)?

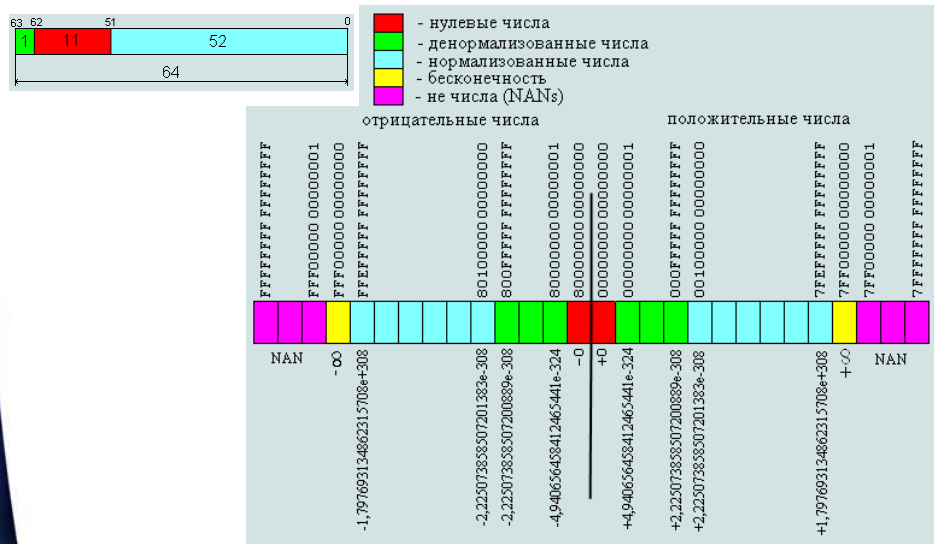
## Разработка формата точного представления действительных чисел

### Полный диапазон чисел одинарной точности (32 бит) по стандарту IEEE754:



## Разработка формата точного представления действительных чисел

### Полный диапазон чисел двойной точности (64 бит) по стандарту IEEE754:



## Разработка формата точного представления действительных чисел

### Предварительный анализ:

Исходное десятичное число: **+0.123456789**

Single (32bit IEEE754): = 3DFCD6EA hex

0 01111011 11111001101011011101010

• Точное значение полученное из Single:

**+0,12345679104328155517578125**

Число Single отображаемое в Windows: +0.1234568

Double (64bit IEEE754): = 3FBF9ADD3739635F hex

0 011111011 111110011010110111010011011100110001101011111

• Точное значение полученное из Double :

**+0,1234567889999999733605449137030518613755702972412109375**

Число Double отображаемое в Windows: +0.123456789

## Разработка формата точного представления действительных чисел

### Погрешность:

Double – Single =

**-2,04328155783972675862969481386244297027587890625 e-9**

Исходное – Double =

**+2,66394550862969481386244297027587890625 e-18**

Исходное – Single =

**-2,04328155517578125 e-9**

## Разработка формата точного представления действительных чисел

### Использование неопределенности:

Исходное десятичное число: **+0.123456789**

Single: 0 01111011 11111001101011 AAAA AAAA

0 01111011 111110011010110000000000 0 01111011 1111100110101101111111

• Точное значение из Single:

**+0,123455047607421875**

Число Single в Windows:

+0.123455

• Точное значение из Single:

**+0,123456947505474090576171875**

Число Single в Windows:

+0.1234569

Использование только значений неопределенности приводит к еще большим погрешностям!



## Разработка формата точного представления действительных чисел

### Чередование A и M:

Исходное десятичное число: +0.123456789

0 01111011 111110011010MMAAAA AAAA

0 01111011 1111100110101001111111 M = 0 A = 1

• Точное значение из Single: +0,123453132808208465576171875

Число Single в Windows: +0.1234531

0 01111011 11111001101011100000000 M = 1 A = 0

• Точное значение из Single: +0,1234569549560546875

Число Single в Windows: +0.1234569

**+0.123456789 = [ +0.1234531; +0.1234569 ]**

**для заметок**







# Донецкий национальный технический университет

Факультет компьютерных наук и технологий



II международная  
научно-техническая  
конференция  
студентов и молодых  
учёных

Информационные  
управляющие системы и  
компьютерный мониторинг  
2011



## Реализация логических операций над элементами тетралогии

с использованием аксиоматического аппарата  
теории множеств

12.04.2011

докладчик: аспирант Иваница С.В.  
руководитель: к.т.н, доц. Аноприенко А.Я.

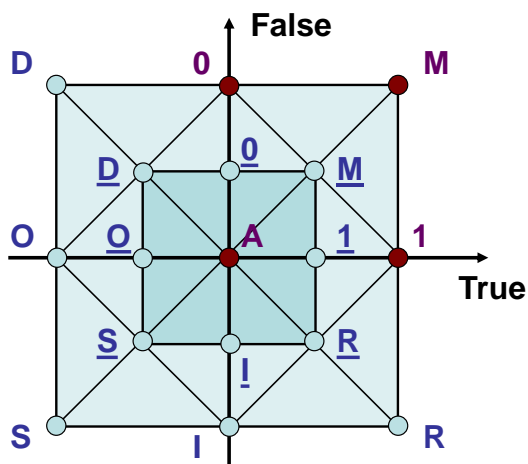
### Введение



- Рассмотрение постбинарного логического пространства с выделением основных состояний тетралогии;
- Представление выделенных состояний с помощью аксиоматического аппарата теории множеств;
- Построение логических постбинарных операций, в котором в качестве операндов выступают множества, включающие в себя различные сочетания высказываний тетралогии;
- На основании полученных операций отрицания, конъюнкции и дизъюнкции представление полной группы базовых логических операций тетралогии.

## Расширенное кодо-логическое пространство

### Двумерное логическое пространство:



1 и 0 - значения «истина» и «ложь» классической логики;

A - абсолютная неопределенность, «непроявленность»;

M - множественность, многозначность;

S - симметричность (инверсная многозначность);

I и O - инверсные «истина» и «ложь»;

D и R - мнемонически соответствуют понятиям «дублирование» и «репликация».

Символы с подчеркиванием

мнемонически могут ассоциироваться с дробностью, половинчатостью, принимающие вероятностный характер, т.е. равновероятны равноудаленные значения.

Например, значение M предполагает равновероятность 0 и 1.

Тетралогия:  $L_4 = \{0, 1, A, M\}$ .

## Почему тетралогия?

**Тетралогия** – наука о формах, методах и законах интеллектуальной познавательной деятельности, которые формализуются с помощью алфавита логики, отражающего четыре состояния.

- «Очеловечивание» машинной двужанной логики;
- Возможность обработки противоречивой информации (сохранение чистоты информации);
- Возможность создания алгебры в расширенной шкале логики, которая позволяют объективно описывать свойства природных и информационных явлений;
- Начало создания вычислительной техники нового поколения – «постбинарный компьютеринг».

## Представление элементов тетралогии

Аксиоматический аппарат теории множеств:  
система аксиом Цермело-Френкеля (ZF – Zermelo-Fraenkel)

$$\forall q \exists Q (q \subseteq Q \rightarrow q = 0) \quad (1)$$

$$\forall j \exists J (j \subseteq J \rightarrow j = 1) \quad (2)$$

$$\forall a \exists A (a \subset A \rightarrow a \in Q \vee a \in J) \quad (3)$$

$$\forall m \exists M (m \subset M \rightarrow m \in Q \wedge m \in J) \quad (4)$$

$$\forall q \forall j \forall a \forall m \exists U = \{u : u \in q \vee u \in j \vee u \in a \vee u \in m\} \quad (5)$$

$Q = \{0\}$  – множество «ложь», представляющее значение «ложь» (логический 0) классической логики ( $q \in Q \rightarrow q = \text{const } 0$ );

$J = \{1\}$  – множество «истина», представляющее значение «истина» (логическая 1) классической логики ( $j \in J \rightarrow j = \text{const } 1$ );

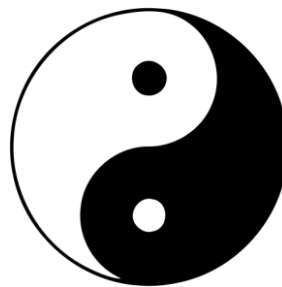
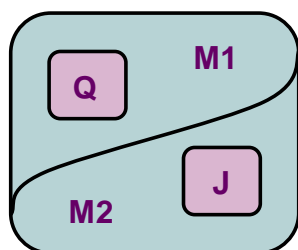
$A$  – множество абсолютной неопределенности (неизвестно, или «истина» или «ложь»), которая выражена объединением множеств «истина» и «ложь» ( $a \in (Q \cup J)$ );

$M$  – множественность, многозначность (и «истина» и «ложь» одновременно, т.е. невозможна однозначная фиксация высказывания), которую нужно выразить пересечением множеств «истина» и «ложь» ( $m \in (Q \cap J)$ ).

## Тетралогика $\leftrightarrow$ сущность бытия

### Закономерности:

- $0 = Q$  (в частности,  $q \in \{0\}$ );
- $1 = J$  (в частности,  $j \in \{1\}$ );
- $(A \cap M) \subset (A \cup M) \subset U$  – для множеств  $A$  и  $M$  на универсуме  $U$  определены операции объединения и пересечения;
- $a \in (Q \cup J)$ ,  $j \in A$ ,  $q \in A$ ,  $(Q \cup J) \neq \emptyset \Rightarrow a \notin \emptyset \vee a \in \{0, 1\}$  – формирование множества неоднозначности  $A$ ;
- $m \in (Q \cap J)$ ,  $j \in M$ ,  $q \in M$ ,  $(Q \cap J) \neq \emptyset \Rightarrow m \notin \emptyset \vee m \in \{0\} \wedge m \in \{1\}$  – выделение противоречия при формировании множественности  $M$ : множество  $M$  не может быть пустым, однако для множеств  $Q$  и  $J$  не определена операция пересечения.





## Реализация базовых логических операций

### Логическое отрицание [NOT, НЕ]:

**Отрицание в логике** – унарная (одноместная) операция над высказываниями, результатом которой является высказывание (в известном смысле) «противоположное» исходному.

$$\varphi \in \Omega, \Omega \subset U \Rightarrow \bar{\varphi} \in \bar{\Omega}, \bar{\Omega} \subset U \Rightarrow \bar{\varphi} \in \bar{\Omega} \neq \varphi \in \Omega, \varphi, \bar{\varphi} \in U. \quad (6)$$

Для множеств  $Q = \{0\}$  и  $J = \{1\}$  :

$$\bar{Q} = J, \bar{J} = Q \Leftrightarrow \bar{0} = 1, \bar{1} = 0.$$

$$\bar{Q} \cup \bar{J} = J \cup Q = Q \cup J = U, \quad (7)$$

в силу коммутативности операции объединения множеств.

Для множества  $A$  :

$$\forall a \in A, U \exists \bar{A} \left( \bar{A} = \{ \bar{a} : \bar{a} \notin A \vee \bar{a} \in U \rightarrow \bar{a} \notin (Q \cup J) \vee \bar{a} \in (\bar{Q} \cup \bar{J}) \rightarrow \bar{a} \in A \} \right). \quad (8)$$

Если  $\bar{a} \in (\bar{Q} \cup \bar{J})$ , то  $a \in (Q \cup J)$ , значит  $\bar{\bar{A}} = A.$

## Реализация базовых логических операций

### Логическое отрицание [NOT, НЕ]:

Для множества  $M$  :

$$\forall m \in M, U \exists \bar{M} \left( \bar{M} = \{ \bar{m} : \bar{m} \notin M \vee \bar{m} \in U \rightarrow \bar{m} \notin (Q \cap J) \vee \bar{m} \in (\bar{Q} \cap \bar{J}) \rightarrow \bar{m} \in M \} \right). \quad (9)$$

Если  $\bar{m} \in (\bar{Q} \cap \bar{J})$ , то  $m \in (Q \cap J)$ , значит  $\bar{\bar{M}} = M.$

**Множество  $M_d$  (доминирующее над неопределенностью):**

Отрицание доминирования:  $\bar{M}_d = M, \bar{\bar{M}} = M_d.$

Таблица 1 – Операция отрицания значений множества  $u \in U$

$u$	0	1	$A$	$M$	$M_d$
$\bar{u}$	$1^{tt}$	$0^{tt}$	$A^{st}$	$M_d^{st}$	$M^{st}$

Для любого элемента гиперкодового множества возможна операция двойного отрицания, приводящая к исходному значению элемента:

$$\bar{\bar{0}} = 0, \bar{\bar{1}} = 1, \bar{\bar{A}} = A, \bar{\bar{M}} = M.$$





## Реализация базовых логических операций



### Логическое сложение (дизъюнкция) [OR, ИЛИ]:

**Логическое сложение (дизъюнкция)** – логическая операция, по своему применению максимально приближенная к союзу «или» в смысле «или то, или это, или оба сразу».

**Обозначение:**  $\vee \equiv$  «ИЛИ».

$$\varphi_1 \in \Omega_1, \varphi_2 \in \Omega_2, \Omega_1, \Omega_2 \subset U \Rightarrow (\varphi_1 \vee \varphi_2) \in (\Omega_1 \cup \Omega_2) \subset U. \quad (10)$$

Для множеств  $Q = \{0\}$  и  $J = \{1\}$ :

$$Q \vee Q = Q = \{0\}, \quad J \vee J = J = \{1\}, \quad Q \vee J = J = \{1\}.$$

Для множеств  $Q, J$  и  $A$ :

$$A \vee Q = A = \{0, 1\}, \quad A \vee J = J = \{1\}, \quad A \vee A = A = \{0, 1\}.$$

Для множеств  $Q, J$  и  $M$ :

$$M \vee Q = M, \quad M \vee J = J = \{1\}, \quad M \vee M = M.$$

## Реализация базовых логических операций



### Логическое сложение (дизъюнкция) [OR, ИЛИ]:

Для множеств  $A$  и  $M$ :

$$\varphi_1 \vee \varphi_2 = \max(\varphi_1, \varphi_2), \quad \varphi_1, \varphi_2 \in \{0, 1, A, M\}. \quad (11)$$

$$A \vee M = A = \{0, 1\}, \quad M \vee M = M.$$

$$A \vee M_d = M_d, \quad M \vee M_d = M_d, \quad M_d \vee M_d = M_d.$$

Таблица 2 – Операция дизъюнкции значений множества  $u \in U$

$\vee$	0	1	$A$	$M$	$M_d$
0	$0^{tt}$	$1^{tt}$	$A^{st}$	$M^{st}$	$M_d^{st}$
1	$1^{tt}$	$1^{tt}$	$1^{st}$	$1^{st}$	$1^{st}$
$A$	$A^{st}$	$1^{st}$	$A^{st}$	$A^{st}$	$M_d^{st}$
$M$	$M^{st}$	$1^{st}$	$A^{st}$	$M^{st}$	$M_d^{st}$
$M_d$	$M_d^{st}$	$1^{st}$	$M_d^{st}$	$M_d^{st}$	$M_d^{st}$

## Реализация базовых логических операций



### Логическое умножение (конъюнкция) [AND, И]:

**Логическое умножение (конъюнкция)** – логическая операция, по своему применению максимально приближенная к союзу «И».

**Обозначение:**  $\wedge \equiv \text{«И»}$ .

$$\varphi_1 \in \Omega_1, \varphi_2 \in \Omega_2, \Omega_1, \Omega_2 \subset U \Rightarrow (\varphi_1 \wedge \varphi_2) \in (\Omega_1 \cap \Omega_2) \subset U. \quad (12)$$

Для множеств  $Q = \{0\}$  и  $J = \{1\}$ :

$$Q \wedge Q = Q = \{0\}, \quad J \wedge J = J = \{1\}, \quad Q \wedge J = Q = \{0\}.$$

Для множеств  $Q, J$  и  $A$ :

$$A \wedge Q = Q = \{0\}, \quad A \wedge J = A = \{0, 1\}, \quad A \wedge A = A = \{0, 1\}.$$

Для множеств  $Q, J$  и  $M$ :

$$M \wedge Q = Q = \{0\}, \quad M \wedge J = M, \quad M \wedge M = M.$$

## Реализация базовых логических операций



### Логическое умножение (конъюнкция) [AND, И]:

Для множеств  $A$  и  $M$ :

$$\varphi_1 \wedge \varphi_2 = \min(\varphi_1, \varphi_2), \quad \varphi_1, \varphi_2 \in \{0, 1, A, M\}. \quad (13)$$

$$A \wedge M = M, \quad M \wedge M = M.$$

$$A \wedge M_d = A, \quad M \wedge M_d = M, \quad M_d \wedge M_d = M_d.$$

Таблица 3 – Операция конъюнкции значений множества  $u \in U$

$\wedge$	<b>0</b>	<b>1</b>	$A$	$M$	$M_d$
<b>0</b>	$0^{tt}$	$0^{tt}$	$0^{st}$	$0^{st}$	$0^{st}$
<b>1</b>	$0^{tt}$	$1^{tt}$	$A^{st}$	$M^{st}$	$M_d^{st}$
$A$	$0^{st}$	$A^{st}$	$A^{st}$	$M^{st}$	$A^{st}$
$M$	$0^{st}$	$M^{st}$	$M^{st}$	$M^{st}$	$M^{st}$
$M_d$	$0^{st}$	$M_d^{st}$	$A^{st}$	$M^{st}$	$M_d^{st}$

## Реализация базовых логических операций



### Важнейшие законы полученной алгебры тетралогии:

Для значений:  $x, y, z \in \{0, 1, A, M\}$

$$\overline{\overline{x}} = x. \quad (14)$$

$$x \vee y = y \vee x; \quad (15)$$

$$x \wedge y = y \wedge x; \quad (16)$$

$$(x \vee y) \vee z = x \vee (y \vee z); \quad (17)$$

$$(x \wedge y) \wedge z = x \wedge (y \wedge z); \quad (18)$$

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z); \quad (19)$$

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z); \quad (20)$$

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z); \quad (21)$$

$$\overline{x \vee y} = \overline{x} \wedge \overline{y}; \quad (22)$$

$$\overline{x \wedge y} = \overline{x} \vee \overline{y}; \quad (23)$$

$$x \vee x = x; \quad (24)$$

$$x \wedge x = x; \quad (25)$$

$$1 \wedge x = x \quad (1 \vee x = 1); \quad (26)$$

$$0 \vee x = x \quad (0 \wedge x = 0). \quad (27)$$

$$0 \vee x = x \quad (0 \wedge x = 0). \quad (28)$$

(15) и (17), (16) и (18) – коммутативность и ассоциативность дизъюнкции и конъюнкции;  
(26) и (27) – дистрибутивные (распределительные) законы дизъюнкции и конъюнкции;  
(28) и (29) – законы де Моргана.

## Выводы



- Рассмотрены основные операции тетралогии как одного из важнейших вариантов постбинарной логики;
- Показано одно из направлений в формировании математического аппарата постбинарной логики;
- Выявлено противоречие неявных состояний неопределенности и множественности;
- Предложен подход, в котором данное противоречие преодолевается путем приведения множественности к доминирующему над неопределенностью состоянию;
- На основании полученных операций отрицания, конъюнкции и дизъюнкции представлена полная группа базовых логических операций тетралогии:

$x \oplus y$  – исключающее «ИЛИ» («XOR»);  
 $x \downarrow y$  – отрицание дизъюнкции (стрелка Пирса – функция «ИЛИ-НЕ»);  
 $x \leftrightarrow y$  – эквиваленция;  
 $x \rightarrow y, x \leftarrow y$  – импликация (следование);  
 $x \mid y$  – отрицание конъюнкции (штрих Шеффера – функция «И-НЕ»).



# Постбинарные форматы действительных чисел

Особенности преобразования  
 $\text{binary}_{\Omega} \rightarrow \text{pbinary}_{\Omega}$

сентябрь 2011

Аспирант кафедры КИ  
Иваница С.В.

## Краткие сведения о формате IEEE 754-1985(2008)

### IEEE (Institute of Electrical and Electronics Engineers)

IEEE 754 используется для представления действительных чисел (чисел с плавающей точкой) в двоичном коде.

#### Стандарт IEEE 754-1985 определяют:

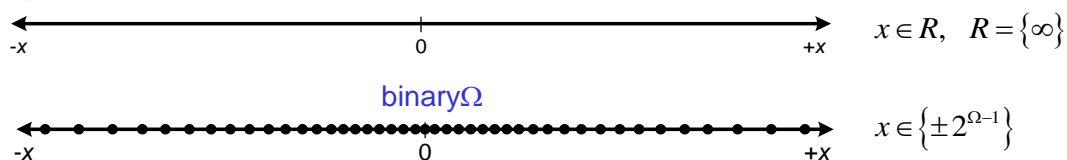
- положительные и отрицательные числа;
- денормализованные положительные и отрицательные числа;
- нулевые числа
- бесконечность (Infiniti);
- «Не число» (NaN или NaNs);
- четыре режима округления.

#### IEEE 754-1985 определяет четыре формата представления чисел с плавающей запятой:

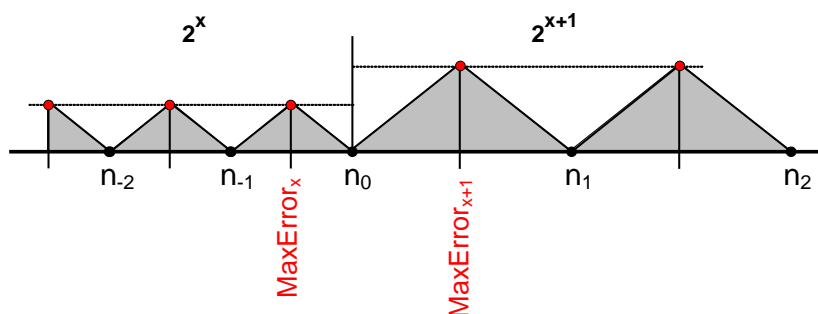
- (single-precision) 32 бита;
- (double-precision) 64 бита;
- (single-extended precision)  $\geq 43$  бит (редко используемый);
- (double-extended precision)  $\geq 79$  бит (обычно используют 80 бит).

В 2008г. ассоциация IEEE выпустила стандарт IEEE 754-2008, который включил в себя стандарт IEEE 754-1985 + формат числа четверичной точности.

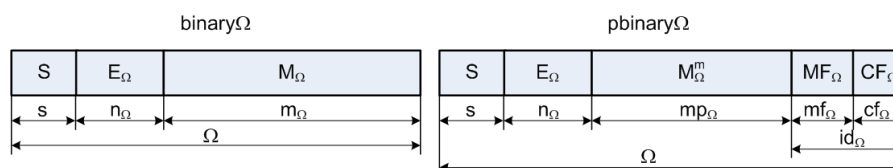
## Недостаток форматов IEEE 754



Числа представленные в формате IEEE754 представляют конечное множество, на которое отображается бесконечное множество вещественных чисел.



3

Постбинарные форматы чисел  
с плавающей запятой

$id_{\Omega} = \frac{\Omega}{16}$  — разрядность идентификатора формата;

$cf_{\Omega} = \log_2 \frac{\Omega}{16}$  — разрядность кода формата;

$mf_{\Omega} = id_{\Omega} - cf_{\Omega} = \frac{\Omega}{16} - \log_2 \frac{\Omega}{16}$  — разрядность модификатора формата;

$mp_{\Omega} = m_{\Omega} - id_{\Omega}$  — разрядность модифицированной мантиисы.

$$E_{\Omega} = \exp_2 + \text{offset}_2,$$

$$\text{offset}_{10} = 2^{(n_{\Omega}-1)} - 1 \Rightarrow \text{offset}_2 = (\text{offset}_{10})_2.$$

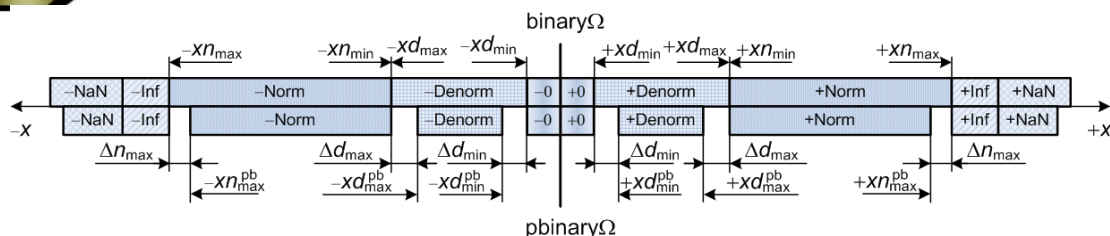
Разрядность pbinaryΩ с указанием значения кода формата

$\Omega$	s	$n_{\Omega}$	$mp_{\Omega}$	$id_{\Omega}$	$mf_{\Omega}$	$cf_{\Omega}$	$CF_{\Omega}$
32	1	8	21	2	1	1	0
64	1	11	48	4	2	2	01
128	1	15	104	8	5	3	011
256	1	20	219	16	12	4	0111

4



## Диапазоны представления чисел в формате с плавающей запятой согласно IEEE754



$$[-xn_{\max}^{pb}; -xd_{\min}^{pb}] \cup [xd_{\min}^{pb}; +xn_{\max}^{pb}] \quad (1)$$

$$\Delta d_{\min} = |xd_{\min} - xd_{\min}^{pb}| = 2^{(1-\text{offset}_{10}-\text{mp}_{\Omega})} \cdot (1 - 2^{-\text{id}_{\Omega}}); \quad \Delta d_{\max} = \Delta d_{\min} \quad (2)$$

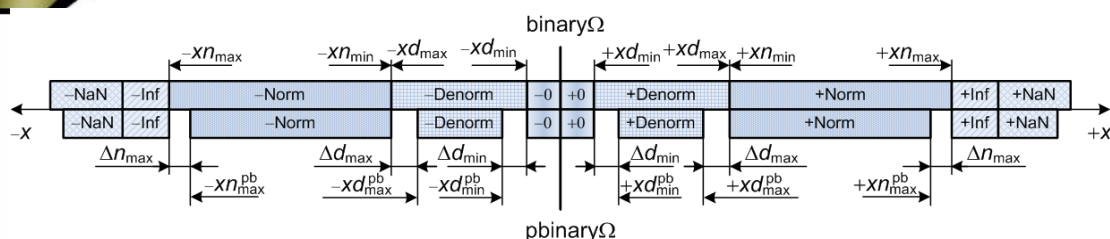
$$\Delta n_{\max} = |xn_{\max} - xn_{\max}^{pb}| = 2^{(\text{offset}_{10}-\text{mp}_{\Omega})} \cdot (1 - 2^{-\text{id}_{\Omega}}). \quad (3)$$

$$xd = (-1)^S \cdot 2^{(1-\text{offset}_{10})} \cdot \frac{(M_{\Omega}^m)_{10}}{2^{\text{mp}_{\Omega}}}; \quad xn = (-1)^S \cdot 2^{((E_{\Omega})_{10}-\text{offset}_{10})} \cdot \left(1 + \frac{(M_{\Omega}^m)_{10}}{2^{\text{mp}_{\Omega}}}\right); \quad (4)$$

$$dd = (-1)^S \cdot 0, M_{\Omega}^m \cdot \exp_2^{-\text{offset}_2}; \quad dn = (-1)^S \cdot 1, M_{\Omega}^m \cdot \exp_2^{(E_{\Omega}-\text{offset}_2)}; \quad (5)$$

5

## Диапазоны представления чисел в формате с плавающей запятой согласно IEEE754



$$xd_{\min}^{pb} = k_1 \cdot xd_{\min}; \quad xd_{\max} = k_1 \cdot xd_{\max}^{pb}; \quad k_1 = 2^{\text{id}_{\Omega}}.$$

$$[xd_{\min}^{pb}; xd_{\max}^{pb}] = [xd_{\min} + \Delta d_{\min}; xd_{\max} - \Delta d_{\max}] = \left[ k_1 \cdot xd_{\min}; \frac{1}{k_1} \cdot xd_{\max} \right]. \quad (7)$$

$$xn_{\min}^{pb} = k_2 \cdot xn_{\max} \quad k_2 = \frac{2^{-(\text{mp}_{\Omega}+1)-\text{id}_{\Omega}} - 1}{2^{-(\text{mp}_{\Omega}+1)} - 1} \lim_{a \rightarrow \infty} \frac{2^{-(a+1)-b} - 1}{2^{-(a+1)} - 1} = 1 \quad k_2 \approx 1.$$

$$[xn_{\min}^{pb}; xn_{\max}^{pb}] = [xn_{\min}; xn_{\max} - \Delta n_{\max}] = [xn_{\min}; k_2 \cdot xn_{\max}] \approx [xn_{\min}; xn_{\max}]. \quad (8)$$

6

## Диапазоны представления чисел в формате с плавающей запятой согласно IEEE754

Значения максимальных и минимальных границ диапазонов Norm и Denorm положительных и отрицательных чисел формата pbinary<sub>Ω</sub> (в скобках указаны значения абсолютной погрешности представления чисел в сопоставлении с форматом binary<sub>Ω</sub>)

Ω	$xd_{\min}^{\text{pb}} (\Delta d_{\min})$	$xd_{\max}^{\text{pb}}$	$xn_{\min}^{\text{pb}}$	$xn_{\max}^{\text{pb}} (\Delta n_{\max})$
32	$\pm 2^{-147} \approx$ $\approx \pm 5,60519386 \cdot 10^{-45}$ ( $\approx 4,204 \cdot e^{-45}$ )	$\pm 2^{-126} \cdot (1 - 2^{-21}) \approx$ $\approx \pm 1,17549379 \cdot 10^{-38}$	$\pm 2^{-126} \approx$ $\approx \pm 1,17549435 \cdot 10^{-38}$	$\pm 2^{127} \cdot (2 - 2^{-21}) \approx$ $\approx \pm 3,40282286 \cdot 10^{38}$ ( $\approx 6,085 \cdot e^{31}$ )
64	$\pm 2^{-1070} \approx$ $\approx \pm 7,90505033 \cdot 10^{-323}$ ( $\approx 7,411 \cdot e^{-323}$ )	$\pm 2^{-1022} \cdot (1 - 2^{-48}) \approx$ $\approx \pm 2,22507386 \cdot 10^{-308}$	$\pm 2^{-1022} \approx$ $\approx \pm 2,22507386 \cdot 10^{-308}$	$\pm 2^{1023} \cdot (2 - 2^{-48}) \approx$ $\approx \pm 1,79769313 \cdot 10^{308}$ ( $\approx 2,994 \cdot e^{293}$ )
128	$\pm 2^{-16486} \approx$ $\approx \pm 1,65764483 \cdot 10^{-4963}$ ( $\approx 1,651 \cdot e^{-4963}$ )	$\pm 2^{-16382} \cdot (1 - 2^{-104}) \approx$ $\approx \pm 3,36210314 \cdot 10^{-4932}$	$\pm 2^{-16382} \approx$ $\approx \pm 3,36210314 \cdot 10^{-4932}$	$\pm 2^{16383} \cdot (2 - 2^{-104}) \approx$ $\approx \pm 1,18973150 \cdot 10^{4932}$ ( $\approx 2,921 \cdot e^{4900}$ )
256	$\pm 2^{-524505} \approx$ $\approx \pm 1,8286336 \cdot 10^{-157892}$ ( $\approx 1,829 \cdot e^{-157892}$ )	$\pm 2^{-524286} \cdot (1 - 2^{-219}) \approx$ $\approx \pm 1,54061213 \cdot 10^{-157826}$	$\pm 2^{-524286} \approx$ $\approx \pm 1,54061213 \cdot 10^{-157826}$	$\pm 2^{524287} \cdot (2 - 2^{-219}) \approx$ $\approx \pm 2,59637057 \cdot 10^{157826}$ ( $\approx 1,541 \cdot e^{157760}$ )

7

## Погрешности представления чисел

binary<sub>Ω</sub>pbinary<sub>Ω</sub>

абсолютная максимально возможная погрешность

$$\Delta_{\max} = 2^{((E_{\Omega})_{10} - \text{offset}_{10} - m_{\Omega} + 1)}$$

$$\Delta_{\max} = 2^{((E_{\Omega})_{10} - \text{offset}_{10} - mp_{\Omega} + 1)}$$

относительная максимально возможная погрешность  $\delta_{N_{\max}}$  нормализованного числа

$$\delta_{N_{\max}} = \frac{1}{2^{m_{\Omega}} + (M_{\Omega})_{10}} \cdot 100\%$$

$$\delta_{N_{\max}} = \frac{1}{2^{mp_{\Omega}} + (M_{\Omega}^m)_{10}} \cdot 100\%$$

относительная максимально возможная погрешность  $\delta_{D_{\max}}$  денормализованного числа

$$\delta_{D_{\max}} = \frac{1}{2 \cdot (M_{\Omega})_{10}} \cdot 100\%$$

$$\delta_{D_{\max}} = \frac{1}{2 \cdot (M_{\Omega}^m)_{10}} \cdot 100\%$$

8

## Погрешности представления чисел

Максимальная возможная ошибка для чисел Single ( $\Omega = 32$ )

IEEE754, hex число, dec	абсолютная ошибка, dec	относительная, %
00000001 $2^{-149} \approx 1,401298e-45$	$2^{-150} \approx 0,700649e-45$	$\approx 50$
00000002 $2^{-148} \approx 2,802597e-45$	$2^{-150} \approx 0,700649e-45$	$\approx 25$
00000032 $\approx 7,00649e-44$	$2^{-150} \approx 0,700649e-45$	$\approx 1$
007FFFFF $\approx 1,175494e-38$	$2^{-150} \approx 0,700649e-45$	$\approx 5,96e-6$
00800001 $\approx 1,175494e-38$	$2^{-149} \approx 1,401298e-45$	$\approx 11,9209e-6$
0DA24260 $\approx 1,0e-30$	$2^{-123} \approx 9,4039e-38$	$\approx 9,4039e-6$
1E3CE508 $\approx 1,0e-20$	$2^{-90} \approx 8,0779e-28$	$\approx 8,0779e-6$
2EDBE6FF $\approx 1,0e-10$	$2^{-57} \approx 6,9389e-18$	$\approx 6,9389e-6$
3F800000 $\approx 1,0$	$2^{-23} \approx 1,192e-7$	$\approx 11,9209e-6$
41200000 $\approx 10,0$	$2^{-20} \approx 9,5367e-7$	$\approx 9,5367e-6$
42C80000 $\approx 1,0e+2$	$2^{-17} \approx 7,6294e-6$	$\approx 7,62939e-6$
501502F9 $\approx 1,0e+10$	$2^{10} \approx 1,024e+3$	$\approx 10,24e-6$
60AD78EC $\approx 1,0e+20$	$2^{43} \approx 8,7961e+12$	$\approx 8,7961e-6$
7149F2CA $\approx 1,0e+30$	$2^{76} \approx 7,5558e+22$	$\approx 7,5558e-6$
7FFFFFFF $\approx 3,40282e+38$	$2^{104} \approx 2,02824e+31$	$\approx 5,96e-6$

$2^{-23} \times 100\% = 11,920928955078125 \text{ e-6 } \%$

$2^{-21} \times 100\% = 4,768371582031312 \text{ e-5 } \%$

9

## Погрешности представления чисел

Максимальная возможная ошибка для чисел Double ( $\Omega = 64$ )

IEEE754, hex	число, dec	абсолютная ошибка, dec	относительная, %
00000000 00000001	$2^{-1074} \approx 4,940656e-324$	$2^{-1075} \approx 2,470328e-324$	$\approx 50$
00000000 00000002	$2^{-1073} \approx 9,881313e-324$	$2^{-1075} \approx 2,470328e-324$	$\approx 25$
00000000 00000032	$\approx 2,470328e-322$	$2^{-1075} \approx 2,470328e-324$	$\approx 1$
000FFFFFF FFFFFFFF	$\approx 2,225073e-308$	$2^{-1075} \approx 2,470328e-324$	$\approx 1,110223e-14$
00100000 00000001	$\approx 2,225074e-308$	$2^{-1074} \approx 4,940656e-324$	$\approx 2,220446e-14$
2B2BFF2E E48E0530	$\approx 1,0e-100$	$2^{-385} \approx 1,268971e-116$	$\approx 1,268971e-14$
3FF00000 00000000	$\approx 1,0$	$2^{-52} \approx 2,220446e-16$	$\approx 2,220446e-14$
54B249AD 2594C37D	$\approx 1,0e+100$	$2^{280} \approx 1,942669e+84$	$\approx 1,942669e-14$
6974E718 D7D7625A	$\approx 1,0e+200$	$2^{612} \approx 1,699641e+184$	$\approx 1,699641e-14$
7FEFFFFFF FFFFFFFF	$\approx 1,79769e+308$	$2^{971} \approx 1,99584e+292$	$\approx 1,110223e-14$

$2^{-52} \times 100\% = 2,2204460492503130808472633361816 \text{ e-14 } \%$

$2^{-48} \times 100\% = 3,5527136788005009293556213378906 \text{ e-13 } \%$

10

## Предлагаемая модификация форматов

Предлагаемые модификации формата pbinary $\Omega$ 

Модификатор MF[11:0]	<b>pbinary32</b> MF[0]	<b>pbinary64</b> MF[1:0]	<b>pbinary128</b> MF[4:0]	<b>pbinary256</b> MF[11:0]
0000 ... 0000	pbinary32	pbinary64	pbinary128	pbinary256
0000 ... 0001	pbinary32/16p	pbinary64/32f	pbinary128/64f	pbinary256/128f
0000 ... 0010		pbinary64/32i	pbinary128/64i	pbinary256/128i
0000 ... 0011		pbinary64/32p	pbinary128/64p	pbinary256/128p
0000 ... 0100			pbinary128/32fp	pbinary256/64fp
0000 ... 0101			pbinary128/32ip	pbinary256/64ip
...			резерв	резерв

f – дробь;

i – интервал;

p – постбинарное (тетракодовое) число;

fp – постбинарное дробное число;

ip – постбинарное интервальное число.

11

## Использование модифицированных форматов

- Арифметические операции (бит);
- Гибкая разрядность;
- Отложенное деление (арифметические операции с дробями);
- Интервальная математика (вычисления);
- Арифметические операции (тетрит);

12





ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ  
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

## АЛГЕБРА ТЕТРАЛОГИКИ

Реализация  
арифметических операций:  
*операция сложения*

аспирант кафедры КИ  
Иваница С.В.

18.10.2011

## Постановка задачи

1. Реализация операции сложения над операндами  $T_1$  и  $T_2$ , где  $T_1, T_2 \in \{0, 1, A, M\}$ :  
$$T_1 + T_2 = ?$$
2. Проверка (доказательство) законов операции сложения:
  - коммутативности:  $T_1 + T_2 == T_2 + T_1$ ;
  - ассоциативности:  $(T_1 + T_2) + T_3 == T_1 + (T_2 + T_3)$ ;
3. Формирование таблицы сложения тетракодов:
  - а) Полный набор сочетаний из  $n=4$ :  $2^n = 2^4 = 16$ ;
  - б) Число сочетаний с повторениями из  $n=4$  по  $k=2$  равно биномиальному коэффициенту:
$$\binom{n+k-1}{k} = \binom{5}{2} = \frac{5!}{2! \cdot 3!} = 10.$$



## Начальный этап формирования таблицы сложения

$0 + 0 = 0$	$1 + A = ?$
$0 + 1 = 1$	$1 + M = ?$
$0 + A = A$	$A + A = ?$
$0 + M = M$	$A + M = ?$
$1 + 1 = 10$	$M + M = ?$

3

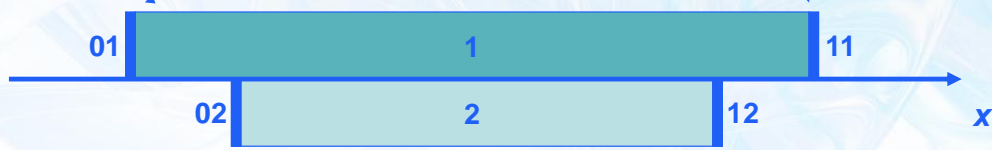
# ТЕТРАКОДЫ & ИНТЕРВАЛЫ

## Интервальное представление тетрадовых значений

$$T \in \{0, 1, M, A\} \quad x \in \mathbb{Z}$$

$$T_{01} \in \{0, 1, M \rightarrow 0, A \rightarrow 0\}$$

$$T_{11} \in \{0, 1, M \rightarrow 1, A \rightarrow 1\}$$



$$T_{02} \in \{0, 1, M \rightarrow 0, A \rightarrow 1\}$$

$$T_{12} \in \{0, 1, M \rightarrow 1, A \rightarrow 0\}$$

$$[a_1; a_2] + [b_1; b_2] = [a_1 + b_1; a_2 + b_2]$$

5

## Алгоритм выполнения операции сложения для двух операндов

**Шаг 1:** Преобразование тетрадовых операндов к интервальному типу данных с формированием соответствующих границ интервала:

$$\min = \{0, 1, M \leftarrow 0, A \leftarrow 1\}, \max = \{0, 1, M \leftarrow 1, A \leftarrow 0\};$$

**Шаг 2:** Получение границ результирующего интервала, используя формулу сложения интервалов

$$[a_1; a_2] + [b_1; b_2] = [a_1 + b_1; a_2 + b_2]$$

**Шаг 3:** Путем побитового сопоставляя min и max границ результирующего интервала, сформировать результирующий тетракод, придерживаясь следующих правил:

- если значение текущего бита не изменилось, бит переносится в результат:  $(0 \rightarrow 0) \rightarrow 0; (1 \rightarrow 1) \rightarrow 1.$

- если значение текущего бита изменилось, то в текущем бите результата формируется значение множественности (M) или неопределенности (A):  $(0 \rightarrow 1) \rightarrow M; (1 \rightarrow 0) \rightarrow A.$

6



# Сложение $1 + A$

0+0=0

0+1=1

0+A=A

0+M=M

1+1=10

1+A=?

1+M=?

A+A=?

A+M=?

M+M=?

Шар 0

1+A

Шар 3

1	0
0	1
↓	↓
A	M

Шар 1

Шар 2

1 →	1	1
A →	+	1
	0	0
	10	1

Пример 1:

10A101A	+	10001
1AM1AMM		

10A101A	↗	1011011	↘	[74; 91]
	↘	1001010	↗	
			+	
10001	↗	10001	↘	[17; 17]
	↘	10001	↗	
			+	
				[91; 108]
1AM1AMM	↗	1101100	↘	[91; 108]
	↘	1011011	↗	

7

# Сложение $1 + M$

0+0=0

0+1=1

0+A=A

0+M=M

1+1=10

1+A=AM

1+M=?

A+A=?

A+M=?

M+M=?

Шаг 0
Шаг 1
Шаг 2

**1+M**

1 → 1 1

M →  $\begin{array}{r} 0 \ 1 \\ \hline 1 \ 10 \end{array}$

Шаг 3

0 1

1 0

↓ ↓

**MA**

Пример 2:

**1M0110A**

**+ 100M01**

---

**MAAMAAAM**

**1M0110A**  $\begin{cases} \nearrow 1001101 \\ \searrow 1101100 \end{cases}$

**100M01**  $\begin{cases} \nearrow 100001 \\ \searrow 100101 \end{cases}$

$\begin{matrix} \nearrow & \searrow \\ [77;108] & \end{matrix}$

**+**

$\begin{matrix} \nearrow & \searrow \\ [33;37] & \end{matrix}$

---

**[110;175]**

**MAAMAAAM**  $\begin{cases} \nearrow 01101110 \\ \searrow 10010001 \end{cases}$

$\begin{matrix} \nearrow & \searrow \\ [110;175] & \end{matrix}$

**1+M = MA**

**M+1 = MA**

8

## Сложение $A + A$ и $M + M$

$0+0=0$   
 $0+1=1$   
 $0+A=A$   
 $0+M=M$   
 $1+1=10$   
 $1+A=AM$   
 $1+M=MA$   
 $A+A=?$   
 $A+M=?$   
 $M+M=?$

Шаг 0

**$A+A$**

Шаг 1 Шаг 2

$$\begin{array}{r} A \rightarrow 1 \ 0 \\ + \\ A \rightarrow 1 \ 0 \\ \hline 10 \ 0 \end{array}$$

Шаг 3

1 0  
0 0  
↓ ↓  
 **$A0$**

**$A+A = A0$**

**$M+M = M0$**

**$M+M$**

Шаг 0

Шаг 1 Шаг 2

$$\begin{array}{r} M \rightarrow 0 \ 1 \\ + \\ M \rightarrow 0 \ 1 \\ \hline 0 \ 10 \end{array}$$

Шаг 3

0 0  
1 0  
↓ ↓  
 **$M0$**

9

## Сложение $A + M$

$0+0=0$   
 $0+1=1$   
 $0+A=A$   
 $0+M=M$   
 $1+1=10$   
 $1+A=AM$   
 $1+M=MA$   
 $A+A=A0$   
 $A+M=?$   
 $M+M=M0$

Шаг 0

**$A+M$**

Шаг 1 Шаг 2

$$\begin{array}{r} A \rightarrow 1 \ 0 \\ + \\ M \rightarrow 0 \ 1 \\ \hline 1 \ 1 \end{array}$$

Шаг 3

1  
1  
↓ ↓  
**1**

**$A+M = 1$**

**$M+A = 1$**

Пример 3:

$$\begin{array}{r} 1MAAM01 \\ + AMMA01M1 \\ \hline 10AM011A0 \end{array}$$

$$\begin{array}{l} 1MAAM01 \rightarrow \begin{array}{l} 10111001 \\ 11000101 \end{array} \rightarrow \begin{array}{l} [185;197] \\ + \end{array} \\ AMMA01M1 \rightarrow \begin{array}{l} 10010101 \\ 01100111 \end{array} \rightarrow [149;103] \end{array}$$

$$\begin{array}{l} 10AM011A0 \rightarrow \begin{array}{l} 101001110 \\ 100101100 \end{array} \rightarrow \begin{array}{l} [334;300] \\ [334;300] \end{array} \end{array}$$

$10AM011A0 \in [300;334]$

10



## Сформированная таблица сложения

$0 + 0 = 0$	$1 + A = AM$
$0 + 1 = 1$	$1 + M = MA$
$0 + A = A$	$A + A = A0$
$0 + M = M$	$A + M = 1$
$1 + 1 = 10$	$M + M = M0$

11

## ПРОВЕРКА АССОЦИАТИВНОСТИ СЛОЖЕНИЯ



## Примеры ассоциативности сложения

1.  $1+1+A =$   $\left\{ \begin{array}{l} (1+1)+A = 10+A=1A \\ 1+(A+1) = 1+AM=1A \end{array} \right.$
2.  $1+1+M =$   $\left\{ \begin{array}{l} (1+1)+M = 10+M=1M \\ 1+(1+M) = 1+MA=1M \end{array} \right.$
3.  $A+M+1 =$   $\left\{ \begin{array}{l} (A+M)+1 = 1+1=10 \\ A+(M+1) = A+MA=10 \end{array} \right.$
4.  $A+M+M =$   $\left\{ \begin{array}{l} (A+M)+M = 1+M=MA \\ A+(M+M) = A+M0=MA \end{array} \right.$
5.  $A+A+M =$   $\left\{ \begin{array}{l} (A+A)+M = A0+M=AM \\ A+(A+M) = A+1=AM \end{array} \right.$

13

## СЛОЖЕНИЕ ГРУПП ЗНАЧЕНИЙ A и M

## Примеры сложения двух полей неопределенности

### 1. Удвоение:

```

+ .....AAAA
+ .....AAAA
+ .....AAAA0 (шаг1)
+ .....AAAA0
+ .....AAAA00 (шаг2)
+ .....AAAA00
+ .....AAAA000 (шаг3)
+ .....AAAA000
+ .....AAAA0000 (шаг4)
+ .....AAAA0000
+ .....AAAA00000 (шаг5)
+ .....AAAA00000
+ .....AAAA000000 (шаг6)

```

### 2. Инкремент:

```

+ .....AAAA
+ .....A
+ .....A0000 (шаг1)
+ .....A
+ .....A000A (шаг2)
+ .....A
+ .....A00A0 (шаг3)
+ .....A
+ .....A00AA (шаг4)
+ .....A
+ .....A0AA0 (шаг5)
+ .....A
+ .....A0AAA (шаг6)

```

### 3. Сумма с const:

```

+ .....AAAA
+ .....AAAA
+ .....AAAA0 (шаг1)
+ .....AAAA
+ .....A0AA0A (шаг2)
+ .....AAAA
+ .....AAAA00 (шаг3)
+ .....AAAA
+ .....A00A0AA (шаг4)
+ .....AAAA
+ .....A0AA0A0 (шаг5)
+ .....AAAA
+ .....AA0A00A (шаг6)

```

Аналогично для полей множественности...

## Примеры сложения полей неопределенности и множественности

### 1. Соприкосновение:

```

+ .....MMAA
+ .....AAAA
+ .....AMMA0 (шаг1)
+ .....AAAA
+ .....A0MM0A (шаг2)
+ .....AAAA
+ .....AAMM00 (шаг3)
+ .....AAAA
+ .....AA11AA (шаг4)
+ .....AAAA
+ .....A0011A0 (шаг5)
+ .....AAAA
+ .....A0A110A (шаг6)

```

### 2. Наложение:

```

+ .....AAAA
+ .....MMMM
+ .....1111 (шаг1)
+ .....MMMM
+ .....M111A (шаг2)
+ .....MMMM
+ .....MA11A1 (шаг3)
+ .....MMMM
+ .....M111AA (шаг4)
+ .....MMMM
+ .....M0A1A11 (шаг5)
+ .....MMMM
+ .....M011A1A (шаг6)

```

### 3. Чередование:

```

+ .....AAAA
+ .....MMMM
+ .....1111 (шаг1)
+ .....AAAA
+ .....A111M (шаг2)
+ .....MMMM
+ .....11110 (шаг3)
+ .....AAAA
+ .....AM11MA (шаг4)
+ .....MMMM
+ .....101101 (шаг5)
+ .....AAAA
+ .....1A110M (шаг6)

```

Донецкий национальный технический университет

# ПРЕДСТАВЛЕНИЕ ИНТЕРВАЛЬНЫХ ЧИСЕЛ СРЕДСТВАМИ ПОСТБИНАРНОГО КОДИРОВАНИЯ

декабрь, 2011

Аспирант кафедры КИ  
Иваница Сергей Васильевич

Представление интервальных чисел средствами постбинарного кодирования

## Введение

(мотивация и постановка задачи)

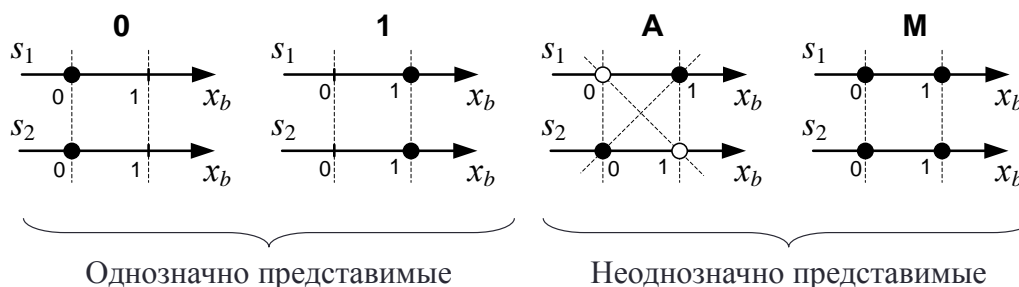
- ☐ Постбинарный компьютеринг;
- ☐ Тетралогики и тетракоды;
- ☐ Тетракод с набором тетритов (четверичных разрядов)  $t \in \{0, 1, A, M\}$ ;
- ☐ Наличие однозначно и неоднозначно представимых тетритов на числовой оси;
- ☐ Вещественный интервал, выраженный на числовой оси крайними значениями (границами);
- ☐ Интервальное оценивание в рамках представления тетракод  $\rightarrow$  интервал.

1

## Представление интервальных чисел средствами постбинарного кодирования

Позиционирование тетритов на  
числовой оси

**Тетракод** — набор тетритов  $t \in \{0, 1, A, M\}$ ;



$s_i$  — событие сведения тетрита к биту.

$x_b$  — ось бинарных значений (бинарная числовая ось).

2

## Представление интервальных чисел средствами постбинарного кодирования

## Критерии нормированности тетракода

- ❑ Тетриты 0 и 1 являются основной информационной составляющей и фактически определяют позицию числа на числовой оси.
- ❑ Наличие «группы М» — наличие тетрита или группы тетритов М обязательно для представления границ интервала: группа М неразрывна и располагается в младших разрядах тетракода.
- ❑ Наличие «группы А» — наличие тетрита или группы тетритов А необязательно, однако носит уточняющий характер для представления границ интервала: группа А неразрывна и занимает более младшие разряды тетракода, чем группа М.
- ❑ Между группами М и А **не допускается** появления однозначно представимых тетритов 0 и 1.

Все представленные далее результаты справедливы только для нормированных тетракодов, причем тетракод, содержащий в себе группу М обладает **нормированностью первого рода**, а содержащий в себе группы М и А — **нормированностью второго рода**.

3



Представление интервальных чисел средствами постбинарного кодирования

## Примеры нормированных тетракодов

$$T_1 = 10110110110111001 \underbrace{\text{MMMM}}_{\text{Группа М}}$$

$k = 4$   
4 3 2 1 0

$$T_2 = 100011010011 \underbrace{\text{MMMM}}_{\text{Группа М}} \underbrace{\text{AAAAAA}}_{\text{Группа А}}$$

$k = 10$     $l = 6$   
10 9 8 7 6 5 4 3 2 1 0

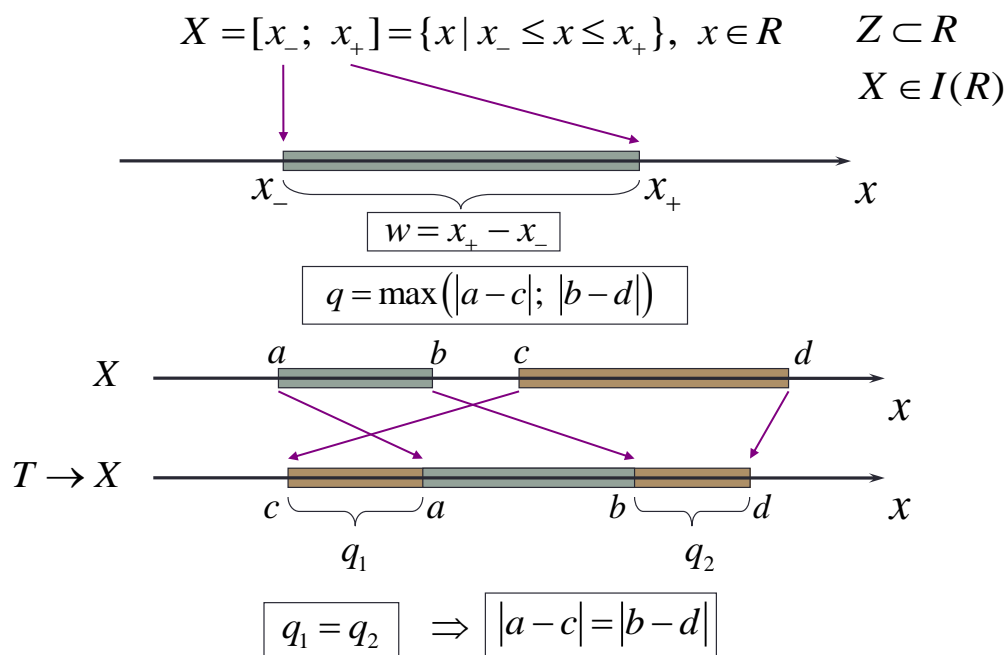
$T_1$  – номированный тетракод первого рода.

$T_2$  – номированный тетракод второго рода.

4

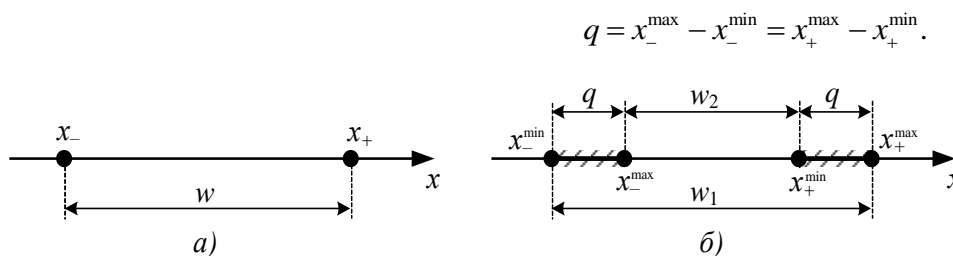
Представление интервальных чисел средствами постбинарного кодирования

## Интервал



5

## Представление интервальных чисел средствами постбинарного кодирования

Формирование целочисленных  
интервальных границ

$$w = 2^{k+1} - 1.$$

Декодирование  
нормированного тетракода  
первого (а) и второго (б) рода.

$$w_1 = w = 2^{k+1} - 1.$$

$$q = 2^{l+1} - 1,$$

$$w_2 = w_1 - 2q = 2^{k+1} - 1 - 2 \cdot (2^{l+1} - 1) = 2^{k+1} - 2^{l+2} + 1.$$

6

## Представление интервальных чисел средствами постбинарного кодирования

## Пример

$$T = 101\text{MMAAA}$$

$k = 4$        $l = 2$   
 7 6 5 4 3 2 1 0

$$\text{Возможный разброс границ: } q = 2^{l+1} - 1 = 2^3 - 1 = 7.$$

$$\text{Максимально возможная ширина: } w_1 = 2^{k+1} - 1 = 2^5 - 1 = 31.$$

$$\text{Минимально возможная ширина: } w_2 = w_1 - 2q = 31 - 2 \cdot 7 = 17.$$

$$w_2 = 2^{k+1} - 2^{l+2} + 1 = 2^5 - 2^3 + 1 = 17.$$

Ширина интервала будет не больше  $w_1 = 31$  и не меньше  $w_2 = 17$ .

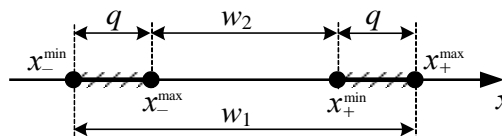
7

Представление интервальных чисел средствами постбинарного кодирования

## Проверка

$$T = \overset{7}{1}\overset{6}{0}\overset{5}{1}\overset{4}{M}\overset{3}{M}\overset{2}{A}\overset{1}{A}\overset{0}{A}$$

$q = 7, w_1 = 31, w_2 = 17.$



↓ — сведение к двоичному 0, ↑ — сведение к двоичной 1:

$$X^{\max} = [x_{-}^{\min}; x_{+}^{\max}] = [101\ 00\ 000; 101\ 11\ 111]_2 = [160; 191]_{10}.$$

$\text{all } M \downarrow \text{all } A \downarrow \quad \text{all } M \uparrow \text{all } A \uparrow$

$$X^{\min} = [x_{-}^{\max}; x_{+}^{\min}] = [101\ 00\ 111; 101\ 11\ 000]_2 = [167; 184]_{10}.$$

$\text{all } M \downarrow \text{all } A \uparrow \quad \text{all } M \uparrow \text{all } A \downarrow$

$$q = 160 - 167 = 191 - 184 = 7, \quad w_1 = 191 - 160 = 31, \quad w_2 = 184 - 167 = 17.$$

Диапазон «плавания» границ интервала  $X$  закодированного в  $T$ :

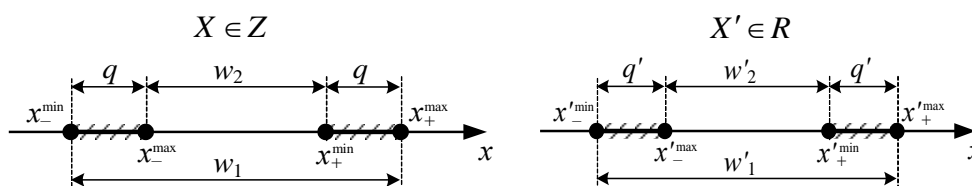
[160; 167] — для левой границы;

[184; 191] — для правой границы.

8

Представление интервальных чисел средствами постбинарного кодирования

## Формирование вещественных интервальных границ



$$q' = F \cdot q; \quad w' = F \cdot w; \quad w'_1 = F \cdot w_1; \quad w'_2 = F \cdot w_2.$$

$E$  — десятичное значение поля порядка

$$T' = 0 \overset{\text{Знак}}{1} \overset{\text{Порядок}}{\boxed{0001101}} \overset{\text{Мантисса}}{101110010011 \text{ M M M M A A A A A A}}$$

$b$  — кол-во разрядов порядка

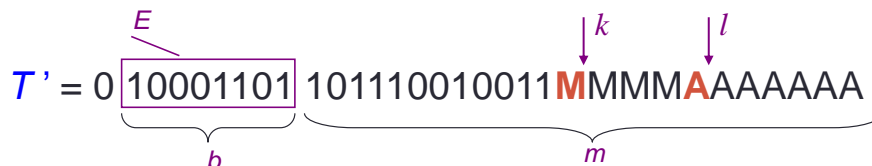
$t$  — кол-во разрядов мантиссы

$k$  — кол-во разрядов порядка

$l$  — кол-во разрядов мантиссы

9

## Представление интервальных чисел средствами постбинарного кодирования

Формирование вещественных  
интервальных границ

Коэффициент для **нормализованных** ( $F_1$ )  
и **денормализованных** ( $F_2$ ) чисел:

$$F_1 = 2^{E-\text{exp}-m};$$

$$F_2 = 2^{1-\text{exp}-m}.$$

$\text{exp} = 2^{b-1} - 1$  — заданное смещение экспоненты имеющей  $b$  двоичных разрядов (в 32-битном IEEE-754 оно равно +127, в 64-битном — +1023).

Для **нормализованных** чисел:

$$w'_1 = w' = 2^{E-\text{exp}-m} \cdot (2^{k+1} - 1);$$

$$w'_2 = 2^{E-\text{exp}-m} \cdot (2^{k+1} - 2^{l+2} + 1);$$

$$q' = \frac{1}{2} \cdot (w'_1 - w'_2) = 2^{E-\text{exp}-m} \cdot (2^{l+1} - 1),$$

Для **денормализованных** чисел:

$$w'_1 = w' = 2^{1-\text{exp}-m} \cdot (2^{k+1} - 1);$$

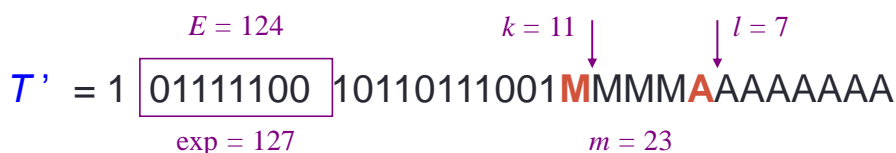
$$w'_2 = 2^{1-\text{exp}-m} \cdot (2^{k+1} - 2^{l+2} + 1);$$

$$q' = 2^{1-\text{exp}-m} \cdot (2^{l+1} - 1),$$

10

## Представление интервальных чисел средствами постбинарного кодирования

## Пример



Возможный разброс границ:

$$q' = 2^{124-127-23} \cdot (2^8 - 1) = 2^{-26} \cdot 255 = \frac{255}{67108864} = 3,7997961... \cdot 10^{-6} \approx 3,8 \cdot 10^{-6};$$

Максимально возможная ширина:

$$w'_1 = 2^{-26} \cdot (2^{12} - 1) = \frac{4095}{67108864} = 6,10202550... \cdot 10^{-5};$$

Минимально возможная ширина:

$$w'_2 = 2^{-26} \cdot (2^{12} - 2^9 + 1) = \frac{3585}{67108864} = 5,3420663... \cdot 10^{-5}.$$

Ширина интервала будет не больше  $6,102 \cdot 10^{-5}$  и не меньше  $5,534 \cdot 10^{-5}$ .

11



Представление интервальных чисел средствами постбинарного кодирования

## Проверка

$$T' = 1 \boxed{01111100} \boxed{10110111001} \text{MMMMAAAAAA}$$

↓ — сведение к двоичному 0, ↑ — сведение к двоичной 1:

$$\begin{aligned} X'^{\max} &= [x'_{\min}; x'_{\max}] = \\ &= [1 \ 01111100 \ 101101110011111111111111]; \\ &\quad \text{all M} \uparrow \quad \text{all A} \uparrow \\ &\quad 1 \ 01111100 \ 101101110010000000000000]_2 = \\ &\quad \text{all M} \downarrow \quad \text{all A} \downarrow \\ &= [-0,21447752; -0,21441650]_{10}. \end{aligned}$$

$$\begin{aligned} X'^{\min} &= [x'_{\max}; x'_{\min}] = \\ &= [1 \ 01111100 \ 101101110011111100000000]; \\ &\quad \text{all M} \uparrow \quad \text{all A} \downarrow \\ &\quad 1 \ 01111100 \ 101101110010000111111111]_2 = \\ &\quad \text{all M} \downarrow \quad \text{all A} \uparrow \\ &= [-0,21447372; -0,21442030]_{10}. \end{aligned}$$

$$q' = -0,21447372 + 0,21447752 = -0,21441650 + 0,21442030 = 3,8 \cdot 10^{-6}$$

Диапазон «плавания» границ интервала  $X'$  закодированного в  $T'$ :

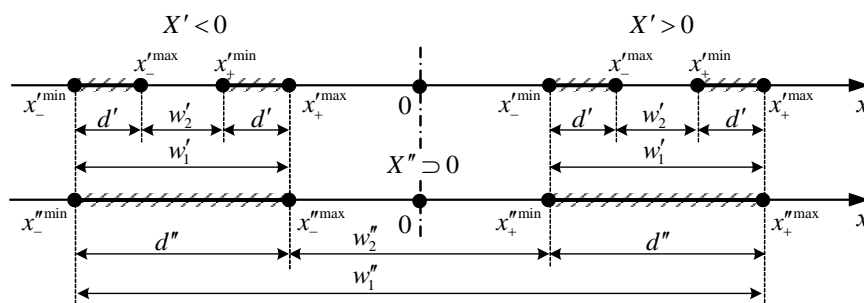
$[-0,21447752; -0,21447372]$  — для левой границы;

$[-0,21442030; -0,21441650]$  — для правой границы.

12

Представление интервальных чисел средствами постбинарного кодирования

## Формирование вещественных интервальных границ для интервала содержащего нулевое значение



$$q'' = w'_1 \Rightarrow \begin{aligned} d'' &= 2^{E-\exp-m} \cdot (2^{k+1} - 1); \quad \text{— для нормализованных чисел;} \\ d'' &= 2^{1-\exp-m} \cdot (2^{k+1} - 1). \quad \text{— для денормализованных чисел.} \end{aligned}$$

$E$  — десятичное значение поля порядка

$$T' = \text{Знак} \boxed{10001101} \boxed{101110010011} \text{MMMMAAAAAA}$$

$b$   $m$

13

## Представление интервальных чисел средствами постбинарного кодирования

## Пример

$$T' = \underset{\text{exp} = 127}{\text{M}} \overset{E = 124}{\boxed{01111100}} \underset{m = 23}{\overset{k = 11 \downarrow}{10110111001\text{MMMMAAAAAAA}}}$$

Возможный разброс границ:

$$q'' = 2^{124-127-23} \cdot (2^{12} - 1) = \frac{4095}{2^{-26}} = 6,10202550... \cdot 10^{-5} \approx 6,102 \cdot 10^{-5};$$

14

## Представление интервальных чисел средствами постбинарного кодирования

## Проверка

$$T' = \text{M} \boxed{01111100} \boxed{10110111001\text{MMMMAAAAAAA}}$$

↓ — сведение к двоичному 0, ↑ — сведение к двоичной 1:

$$X''^{\max} = [x_{-}''^{\min}; x_{+}''^{\max}] =$$

$$= \underset{\text{M}\uparrow}{[ \underset{\text{M}\downarrow}{1} \quad \underbrace{01111100}_{\text{all M}\uparrow} \quad \underbrace{10110111001111111111}_{\text{all A}\uparrow} ]}_2 =$$

$$= [-0,21447752; 0,21447752]_{10}.$$

$$X''^{\min} = [x_{-}''^{\max}; x_{+}''^{\min}] =$$

$$= \underset{\text{M}\uparrow}{[ \underset{\text{M}\downarrow}{1} \quad \underbrace{01111100}_{\text{all M}\downarrow} \quad \underbrace{101101110010000000000000}_{\text{all A}\downarrow} ]}_2 =$$

$$= [-0,21441650; 0,21441650]_{10}.$$

$$q'' = -0,21441650 + 0,21447752 = 0,21447752 - 0,21441650 = 6,102 \cdot 10^{-5}$$

Диапазон «плавания» границ интервала  $X'$  закодированного в  $T'$ :

$[-0,21447752; -0,21441650]$  — для левой границы;

$[-0,21441650; -0,21447752]$  — для правой границы.

15

**МКГ-2011**  
4-я международная  
научно-техническая конференция



**Ноографика  
&  
ноомоделирование**

**А.Я. Аноприенко**

Декан факультета компьютерных наук и технологий  
Кафедра компьютерной инженерии ДонНТУ  
7 октября 2011 года

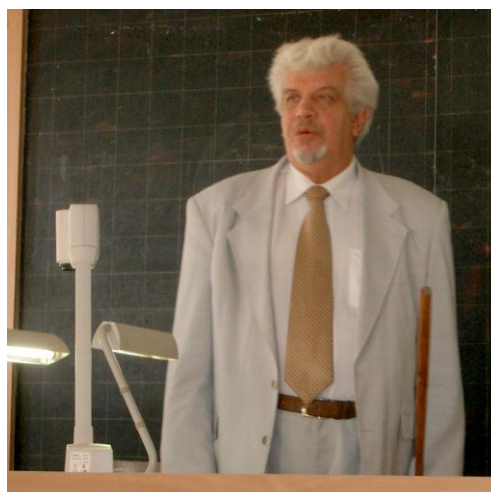
**МКГ-2011**  
Ноографика & ноомоделирование



**2005 год:**  
**1-я конференция «МКГ»**  
Пленарный доклад

**Валькман Юрий Роландович**

**«О моделировании  
визуального мышления  
в компьютерных  
технологиях»**



МКГ-2011

## Ноографика &amp; ноомоделирование



«**Ноологизмы**» - новые слова, отражающие суть тех качественных изменений, которые с необходимостью происходят в условиях ноосферной реальности.

В данном докладе предлагается и обосновывается целесообразность использования таких ноологизмов как **ноографика, ноограммы и ноомоделирование**



МКГ-2009

## Археомоделирование и неогеография

в контексте эволюции моделей и образов мира



WIKIPEDIA

<b>English</b> <i>The Free Encyclopedia</i> 2 683 000+ articles	<b>日本語</b> フリー百科事典 550 000+ 記事
<b>Deutsch</b> <i>Die freie Enzyklopädie</i> 847 000+ Artikel	<b>Español</b> <i>La enciclopedia libre</i> 431 000+ artículos
<b>Français</b> <i>L'encyclopédie libre</i> 746 000+ articles	<b>Italiano</b> <i>L'enciclopedia libera</i> 527 000+ voci
<b>Polski</b> <i>Wolna encyklopedia</i> 566 000+ haseł	<b>Português</b> <i>A enciclopédia livre</i> 449 000+ artigos
<b>Русский</b> <i>Свободная энциклопедия</i> 344 000+ статей	<b>Nederlands</b> <i>De vrije encyclopedie</i> 508 000+ artikelen



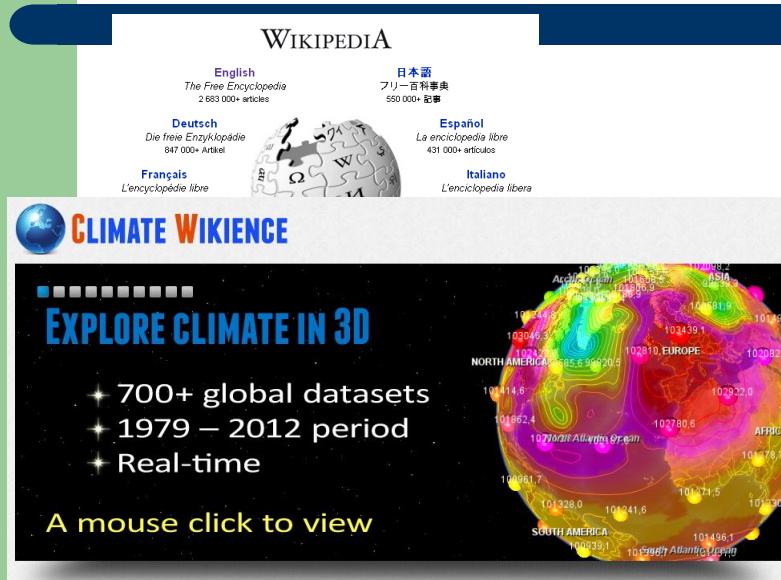
Самый  
известный  
НООЛОГИЗМ:  
«**Википедия**»

Современный символ ноосферы



## МКГ-2011

### Ноографика & ноомоделирование



Самый  
известный  
ноологизм:  
«Википедия»

Ноологизм-2011  
от аспиранта  
ДонНТУ  
Родригеса  
Залипинеса:  
«Викиенс»

## МКГ-2009

### Археомоделирование и неогеография

в контексте эволюции моделей и образов мира



2009:

«Ноогеография» (от гр. ноос – разум) –  
география эпохи ноосферы,  
**ноологизм**, отражающий стремительную  
интеллектуализацию современных моделей  
мира, что можно считать одним из ярких  
проявлений общего процесса «цефализации  
техносферы» и формирования ноосферы

МКГ-2011

## Ноографика &amp; ноомоделирование



«**Ноограммы**», т.е. «разумные диграммы» или «диаграммы эпохи ноосферы», представляющие из себя

такое визуальное представление систематизированной информации,



которое позволяет наглядно увидеть и понять основные закономерности развития, поведения и/или взаимосвязи визуализируемых объектов, процессов и явлений.

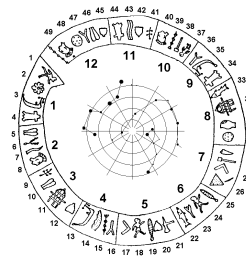
МКГ-2011

## Ноографика &amp; ноомоделирование

**Ноографика**

(различные разновидности ноограмм)

- такое визуальное представление систематизированной информации, в том числе динамическое,



которое позволяет **наглядно увидеть** и **понять основные закономерности** развития, поведения и/или взаимосвязи визуализируемых объектов, процессов и явлений.

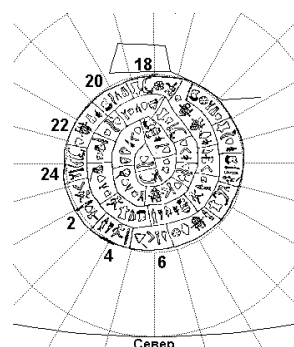
МКГ-2011

Ноографика & ноомоделирование



## Ноомоделирование

– такое вычислительное моделирование, которое сопровождается **наглядно-образной визуализацией** хода и/или результатов моделирования в виде различных ноограмм, **стимулирующих процессы осознания и понимания** сути, смысла и основных свойств моделируемых процессов и явлений.



[Веб](#)
[Картинки](#)
[Карты](#)
[Новости](#)
[Переводчик](#)
[Вопросы и ответы](#)
[Gmail](#)
[ещё](#)

[Настройки поиска](#)
[Войти](#)

Google

Расширенный поиск

**Поиск**      Результаты: 8 (0,22 сек.)

Все результаты

**Картинки**

Карты

Видео

Новости

Ещё

**По релевантности**

По темам

**Любого размера**

Большие

Средние

Маленькие

Больше чем...

Точный размер

**Любого цвета**

Цветные

Черно-белые

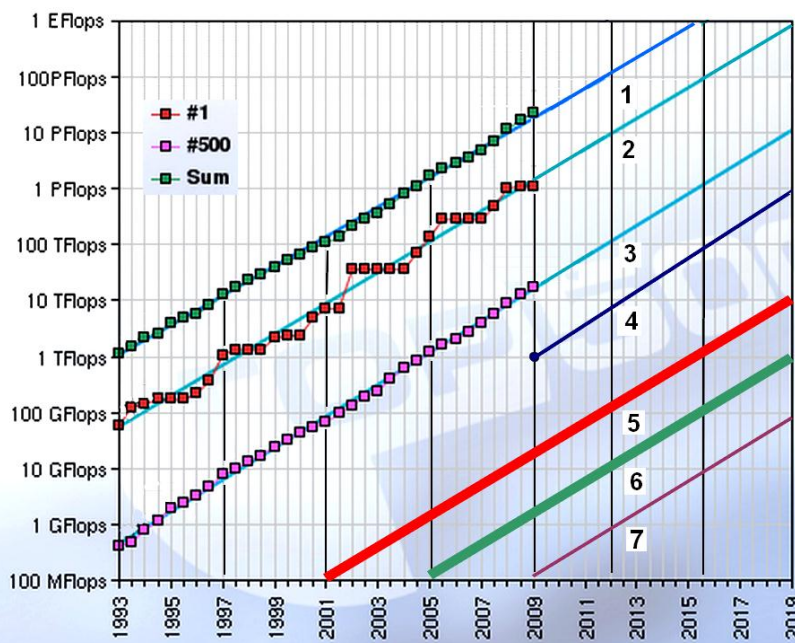
**1997:**

**«когнитогаммы»**

Результаты поиска изображений по запросу «когнитогаммы» (сентябрь 2011 г.):

## МКГ-2011

### Ноографика & ноомоделирование



Между  
ноограммой  
и  
ноограммой:

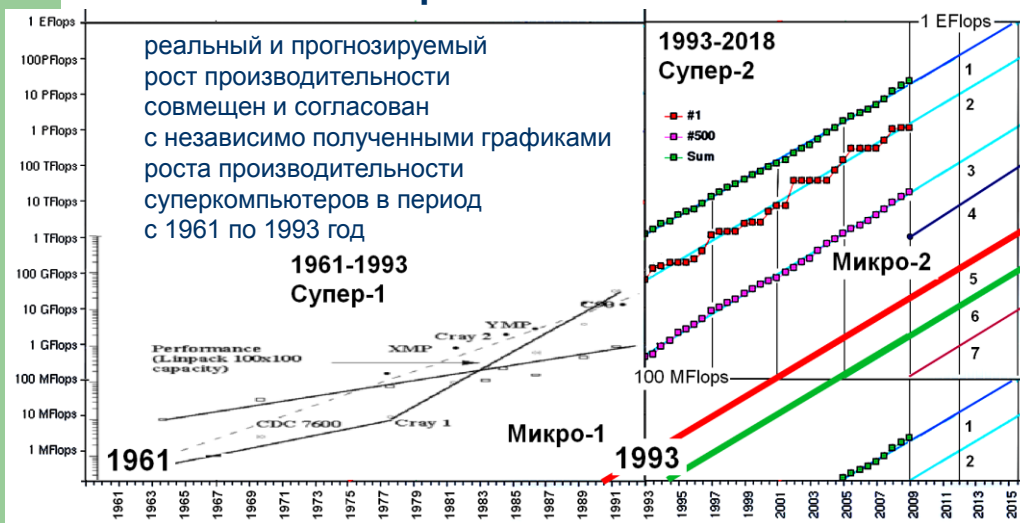
2010: реальный и  
прогнозируемый  
рост  
производительности  
различных типов  
компьютерных  
систем

## МКГ-2011

### Ноографика & ноомоделирование



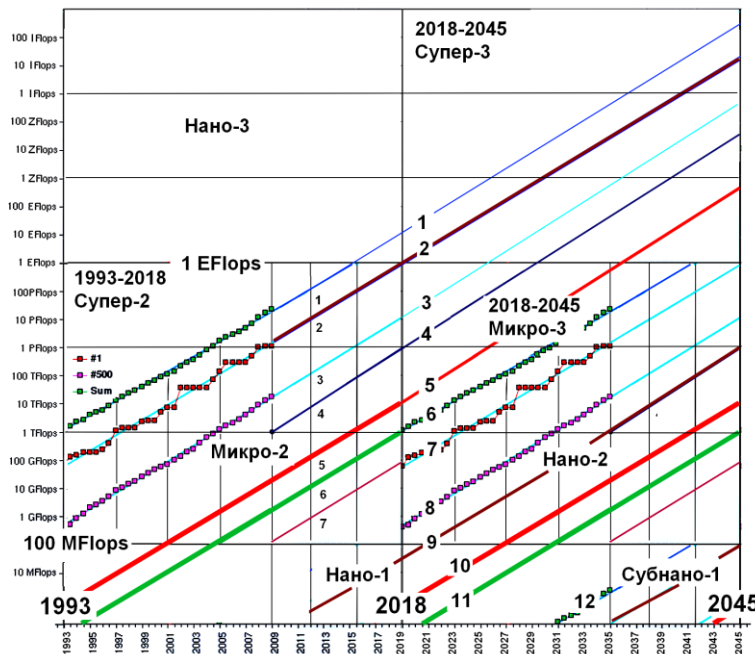
#### 2011: Уже ноограмма...





## МКГ-2011

### Ноографика & ноомоделирование



Ноограмма,  
позволяющая  
прогнозировать  
появление  
новых классов  
компьютеров

## МКГ-2011

### Ноографика & ноомоделирование



Авторский пример: почему 2011? 2012? ...

Ноомоделирование с целью найти и понять  
**рациональные основы фиксации**  
**начальной точки отсчета нашей эры,**  
которая была загадочно «вычислена»  
в Риме в остготской среде  
«скифским монахом»  
в VI веке нашей эры,  
т.е. спустя более 500-т лет  
от легендарных событий в Иерусалиме.

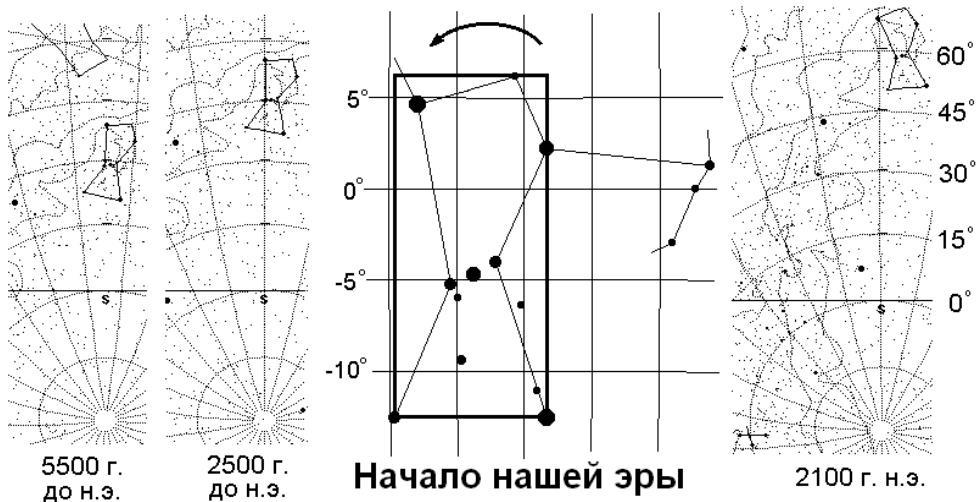




МКГ-2011

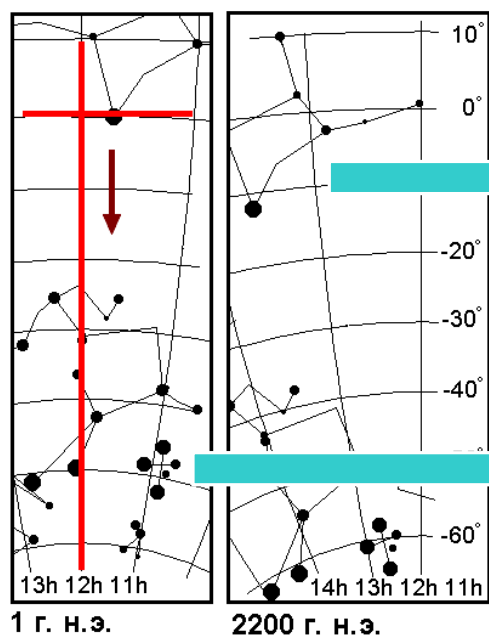
## Ноографика &amp; ноомоделирование

Прецессионное изменение положения Ориона



МКГ-2011

## Ноографика &amp; ноомоделирование



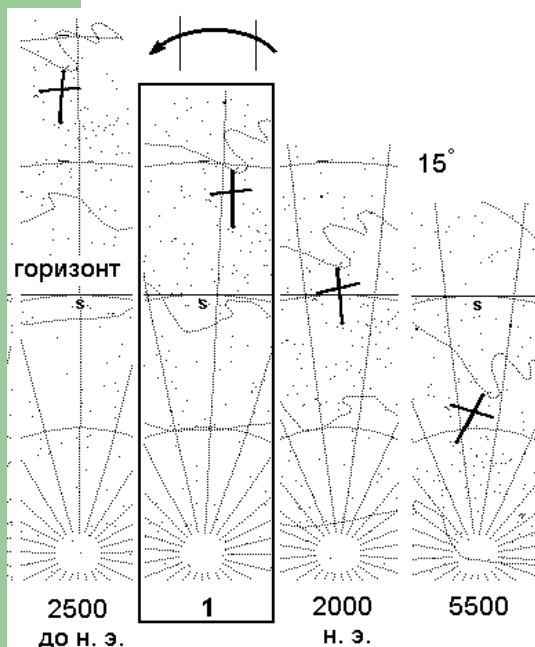
Прецессионное изменение положения созвездия Девы относительно Южного Креста (слева), отраженное в типичной алтарной композиции (справа)

## МКГ-2011

## Ноографика &amp; ноомоделирование



Прецессионное  
изменение  
положения  
созвездия  
Южного Креста



## МКГ-2011

## Ноографика &amp; ноомоделирование



В данном случае  
ноомоделирование позволяет  
сделать следующий вывод:

рациональной основой начала  
нового летоисчисления явилось  
**уникальное сочетание звездных событий**  
**(в частности, положение созвездий Ориона,**  
**Девы и Южного Креста), связанных с прецессией**  
**звездного неба.**



## МКГ-2009

### Археомоделирование и неогеография в контексте эволюции моделей и образов мира



**Ноомоделирование:**

древние  
истоки...



Портал археомоделирования ДонНТУ

**MALTA PLATE**

Year: 2009  
Month: Сентябрь  
Day: 27

Amount of days, passing from the beginning of year: 270

Amount of days, passing from the beginning of design: 0

Start Stop

Save date

Whip off in a current date

Whip off in the stored date

Solar year 0  
Moon year 0  
1/6 of the solar year 0  
1/6 of the solar year 0  
Double sidereal month 0  
Double sidereal month 0  
Sidereal form of the saros 0  
Synodic form of the saros 0

**Cycle of reproduction of life**

Synodic cycle time for Mercury (7.14)  
Synodic cycle time for Venus (5.01)  
Synodic cycle time for Mars (4.51)  
Synodic cycle time for Jupiter (8.00)  
Synodic cycle time for Saturn (9.53)



## МКГ-2011

### Ноографика & ноомоделирование



**Основная идея  
ноомоделирования:**

**ВЫЧИСЛИТЕЛЬНАЯ**  
модель первична;

**образ вторичен,  
но чрезвычайно важен,  
в качестве активатора  
визуального мышления**

**Так было, есть и будет...**

**MALTA PLATE**

Year: 2009  
Month: Сентябрь  
Day: 27

Amount of days, passing from the beginning of year: 270

Amount of days, passing from the beginning of design: 0

Start Stop

Save date

Whip off in a current date

Whip off in the stored date

Solar year 0  
Moon year 0  
1/6 of the solar year 0  
1/6 of the solar year 0  
Double sidereal month 0  
Double sidereal month 0  
Sidereal form of the saros 0  
Synodic form of the saros 0

**Cycle of reproduction of life**

Synodic cycle time for Mercury (7.14)  
Synodic cycle time for Venus (5.01)  
Synodic cycle time for Mars (4.51)  
Synodic cycle time for Jupiter (8.00)  
Synodic cycle time for Saturn (9.53)





122

МКГ-2011

## Ноографика & ноомоделирование



Развитие когнитивного «ноокомплекса»:

1. ноограммы... ноографика...
2. ноомоделирование...
3. **НООКОМПЬЮТИНГ** (постбинарный)



Максимальная наглядность,  
информативность и **понимание**  
на базе эффективного синтеза компьютерной  
графики и компьютерного моделирования

*для заметок*





ДонНТУ Семинар в деканате КНТ

## Тетракоды в вычислениях с гибкой разрядностью

**Аноприенко Александр Яковлевич**

Декан факультета  
компьютерных наук и технологий  
ДонНТУ

27.10.2011

1



Аноприенко  
Александр Яковлевич

Тетракоды в вычислениях с гибкой  
разрядностью

**Основная проблема:**

Современное компьютерное число не содержит никакой информации об исходной или текущей точности – по мере роста разрядности это превращается в большую проблему

Проблемы «интервальных вычислений»:

1. Сложность аппаратной поддержки
2. Существенное противоречие, связанное с излишней (максимальной) точностью задания границ интервалов, отражающих зачастую как раз неточность исходных данных

**Вариант решения: переход к нормированной («естественной») интервальности на базе тетракодов, позволяющей учитывать и хранить информацию об исходной точности операндов**

2





Аноприенко  
Александр Яковлевич

## Тетракоды в вычислениях с гибкой разрядностью

### Примеры частных проблем фиксированной разрядности:

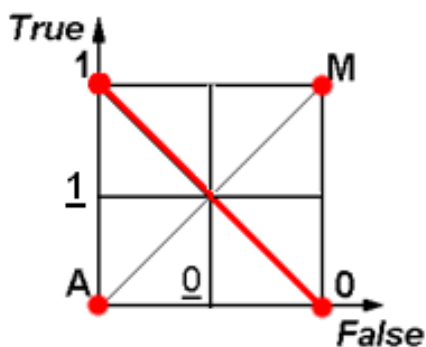
1. **Феномен Румпа:** катастрофическое искажение результатов по причине округлений в случае операций с очень большими или очень малыми операндами, близкими по значению
2. **Феномен Юровицкого:** накопление суммы несоизмеримо малых величин приводит к изменению больших значений, но в современной компьютерной математике малые величины просто игнорируются  
Выход: «переход к эпохе **метрологических чисел** и метрологического компьютеринга»
3. **Феномен мультимасштабного моделирования** (от макро- до наномасштаба).  
при приведении всех значений к одной размерности и масштабу получаем несоизмеримые величины...

3



Аноприенко  
Александр Яковлевич

## Тетракоды в вычислениях с гибкой разрядностью



Традиционный тетракод,  
включающий состояния  
**1, 0, A, M**

### Преимущества:

Расширенные возможности  
компактного кодирования  
относительно сложных структур

### Недостатки:

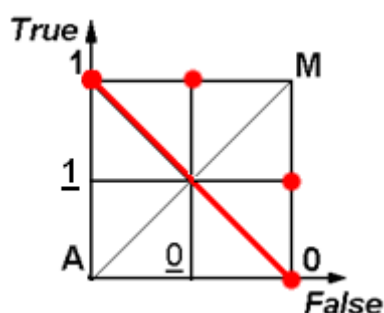
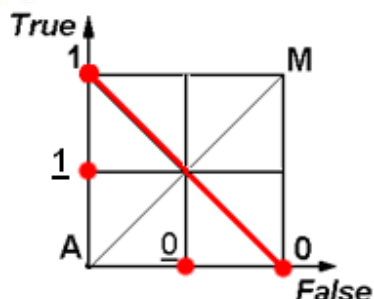
Несовместимость с традиционной  
компьютерной арифметикой

4



Аноприенко  
Александр Яковлевич

### Тетракоды в вычислениях с гибкой разрядностью



Тетракод для вычислений  
с гибкой разрядностью,  
обеспечивающий совместимость  
со стандартной арифметикой:

1, 0, 1, 0 (1 и 0 – недоопределенные  
значения только в младших разрядах)

$$\underline{0} + \underline{0} = \underline{0}$$

$$\underline{1} + \underline{0} = \underline{1}$$

«Заражающее» действие:

$$\underline{0} + \underline{0} = \underline{0}$$

$$\underline{1} + \underline{0} = \underline{1}$$

Блокирование «заражения»  
при переносе в старшие разряды:

$$\underline{1} + \underline{1} = \underline{10}$$

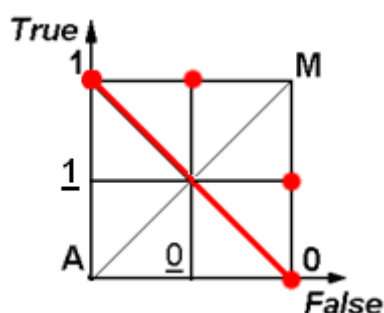
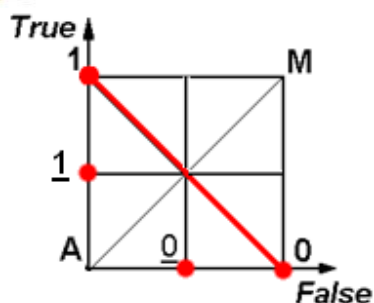
$$\underline{1} + \underline{1} = \underline{10} \quad ?$$

5



Аноприенко  
Александр Яковлевич

### Тетракоды в вычислениях с гибкой разрядностью



Тетракод для вычислений  
с гибкой разрядностью,  
обеспечивающий совместимость  
со стандартной арифметикой:

1, 0, 1, 0

Число вида  $x \dots x \underline{x} \dots \underline{x}$   
является **интервальным числом**,  
в котором нижняя граница –  
само это число ( $x \dots x \underline{x} \dots \underline{x}$ ),  
а верхняя вычисляется как

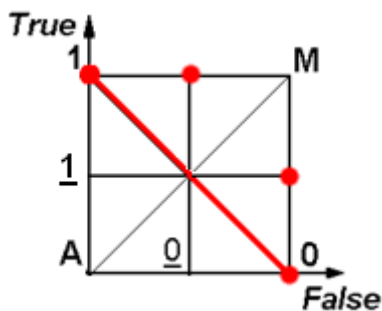
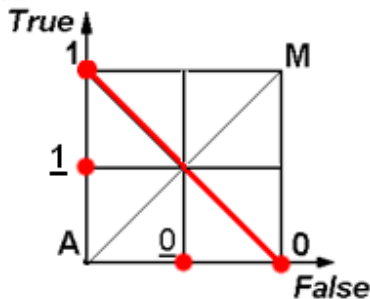
$$\begin{array}{r} x \dots x \underline{x} \dots \underline{x} \\ + 0 \dots 01 \dots 1 \\ \hline \end{array}$$

6



Аноприенко  
Александр Яковлевич

### Тетракоды в вычислениях с гибкой разрядностью



Тетракод для вычислений с гибкой разрядностью, обеспечивающий совместимость со стандартной арифметикой: **1, 0, 1, 0**

**2 основных правила** работы с недоопределенными значениями: (положительные числа)

1. При расширении разрядности число дополняется справа значениями 0
2. При уменьшении разрядности справа могут отбрасываться все разряды, содержащие 0 вплоть до крайнего справа значения 1 или 1<sub>7</sub>



Аноприенко  
Александр Яковлевич

### Тетракоды в вычислениях с гибкой разрядностью

Пример:

```

10100000
+00000101
=10100101
+00000101
=10101010
+00000110
=10110000
+00101010
=11011010
+11011010
=110110100    p+1
+110110100
=1101101000    p+1
  
```

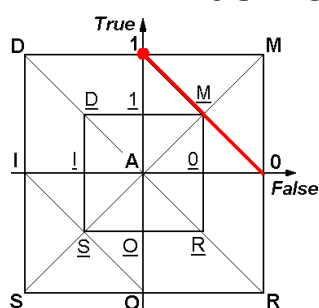


**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

Какой вариант компьютерного кодирования с основанием 4 будет качественно лучше традиционного бинарного кода и сможет быть соизмеримым по информационной эффективности с генетическим кодированием???

**Квазигенетический код**  
**+ квазигенетическая логика**  
**= квазигенетический компьютер !!!**



**Выход логических основ компьютеринга за пределы «тонкой красной линии» в рамках двумерного логического пространства**  
Основной вариант: **тетралогика**  
Кроме 0 и 1 также  
А – Абсолютная неопределенность  
М - Множественность

9



**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

Какой вариант компьютерного кодирования с основанием 4 будет качественно лучше традиционного бинарного кода и сможет быть соизмеримым по информационной эффективности с генетическим кодированием???

Еще об истоках идеи:

1990 год, профессор Андреас Ройтер, основатель  
Института параллельных и суперкомпьютерных систем  
Штуттгартского университета:

**Необходимость пост-неймановской (и пост-тьюринговской, и пост-бинарной) парадигмы для современного компьютеринга, ориентированного на постоянный рост степени параллелизма**

10



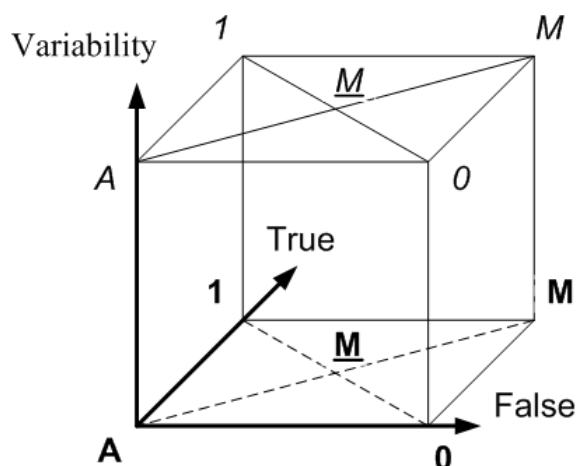




**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

И далее:  
еще ближе к квазигенетической логике:



**Трёхмерное логическое пространство** может быть порождено базисом, состоящим из ортонормированной системы векторов «Истина» (может обозначаться как Т – «True» или Y – «Yes»), «Ложь» (F – «False» или N – «No») и «Вариабельность» (V – «Variability»).

13



## Эволюция кодо-логического базиса

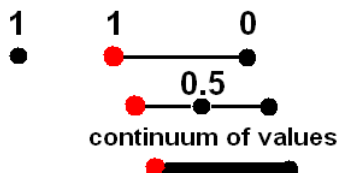
0D

1D

2D

3D

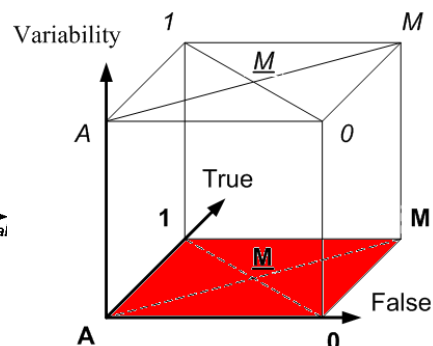
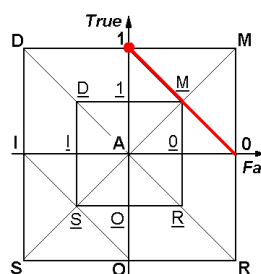
... 0 AD ... 2000 AD ...



Моно-эпоха

Ди-эпоха

Гипер-эпоха



14

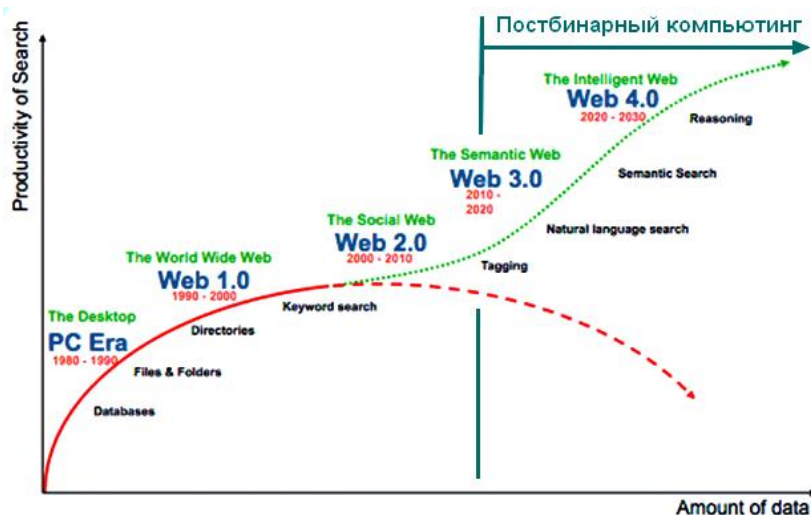




Аноприенко  
Александр Яковлевич

Постбинарный компьютеринг... в  
контексте кодо-логической эволюции

## Постбинарный компьютеринг 2010-2030: «Семантическая» революция:



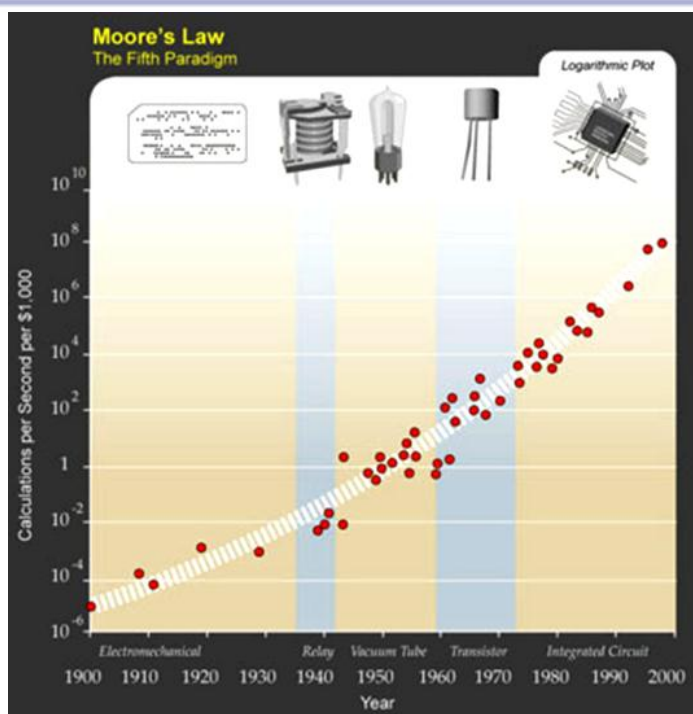
1. Web-технологии 3.0 и 4.0
2. Grid-технологии
3. Глобальное сетевое программирование на базе сетевых API
4. Постбинарная логика и постбинарные вычисления!!!

17



Аноприенко  
Александр Яковлевич

Постбинарный компьютеринг... в  
контексте кодо-логической эволюции



Закон Мура  
реально  
действует  
уже более  
100 лет...

И, скорее  
всего,  
будет  
действовать  
минимум  
еще столько  
же

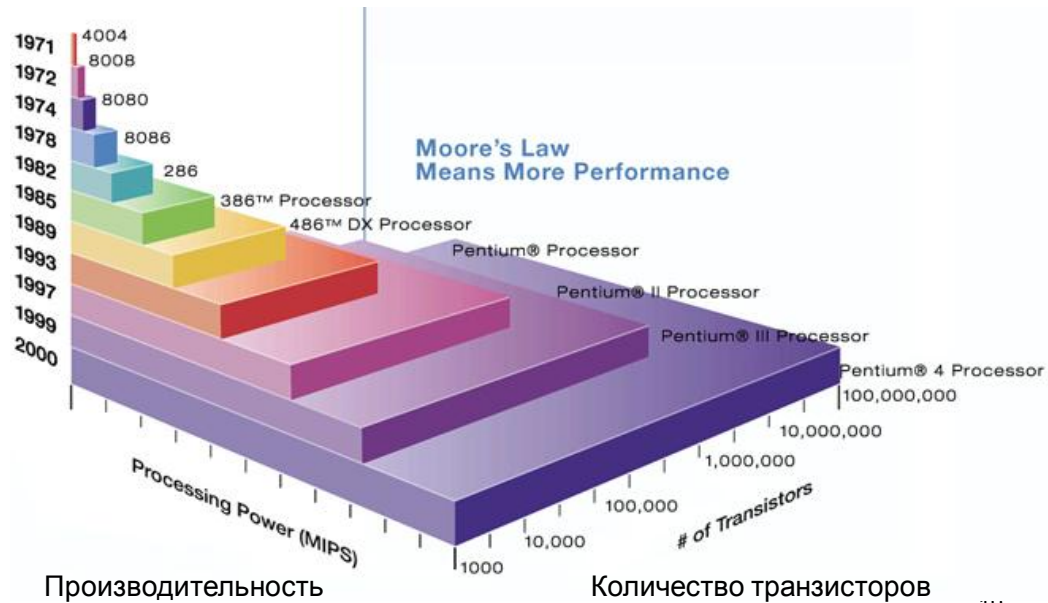
18



Аноприенко  
Александр Яковлевич

Постбинарный компьютеринг... в  
контексте кодо-логической эволюции

### Действие закона Мура в 1970-2000 гг. (на примере микропроцессоров Интел)



19



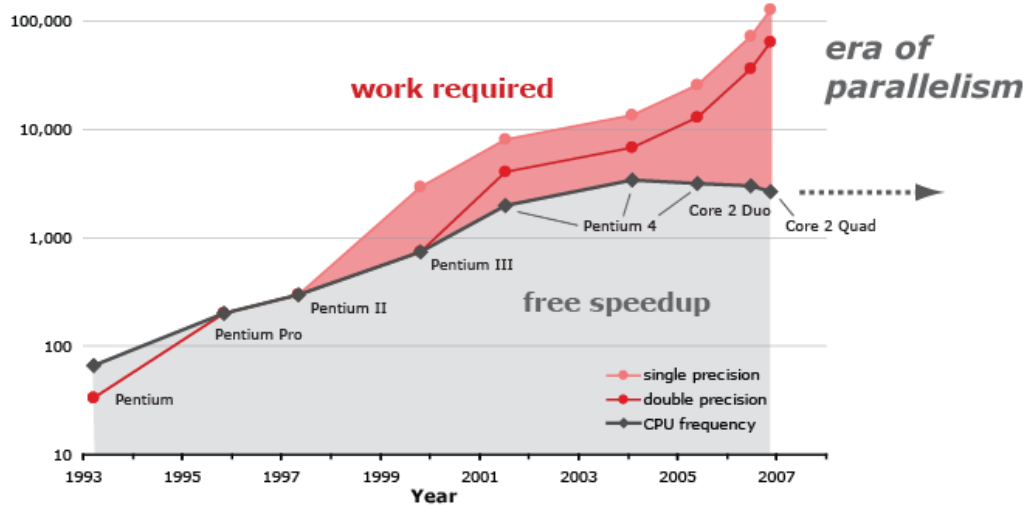
Аноприенко  
Александр Яковлевич

Постбинарный компьютеринг... в  
контексте кодо-логической эволюции

### Закон Мура с 2000 года: начало эры параллелизма

#### Evolution of Intel Platforms

Floating point peak performance [Mflop/s]  
CPU frequency [MHz]



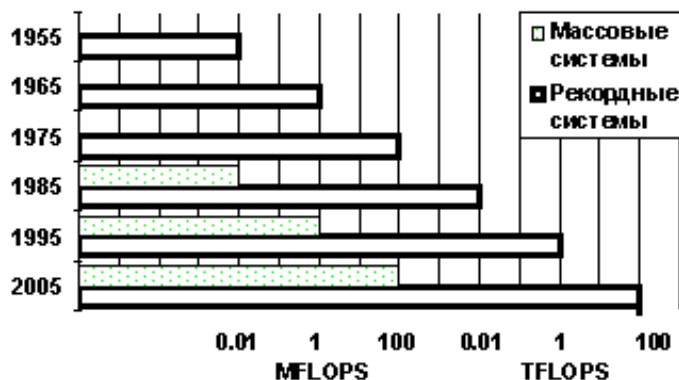
20



Аноприенко  
Александр Яковлевич

Постбинарный компьютеринг... в  
контексте кодо-логической эволюции

Гипотеза-1995: Рекордные и массовые системы –  
30-летний «лаг» в 6 порядков  
(на порядок за 5 лет?)



Рекордные/  
массовые  
1985: 0.01 ТФлопс/  
0.01 МФлопс  
1995: 1 ТФлопс/  
1 МФлопс  
2005: 100 ТФлопс/  
100 ТФлопс

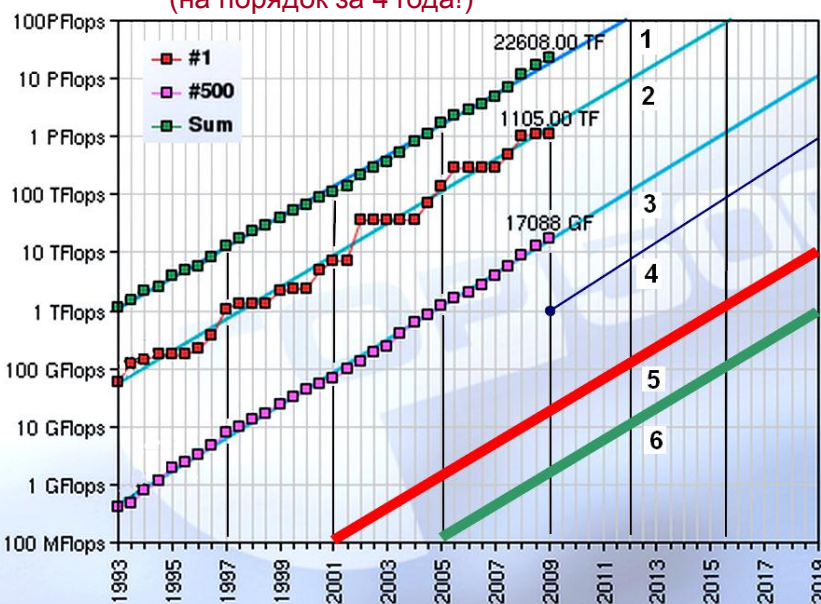
21



Аноприенко  
Александр Яковлевич

Постбинарный компьютеринг... в  
контексте кодо-логической эволюции

Реальные закономерности роста  
производительности систем разного уровня:  
(на порядок за 4 года!)



1 - Сумма «Топ 500»  
2 - № 1 «Топ 500»  
3 - № 500 «Топ 500»

4 - «Персональные  
суперкомпьютеры»  
5 - Сервера /  
персональные  
«Топ-компьютеры»  
6 - Массовые  
персональные  
компьютеры

1 TFlops: «лаги»

1 – 1993 (X-4)  
2 – 1997 (X)  
3 – 2005 (X+8)  
4 – 2009 (X+12)  
5 – 2015 (X+18)  
6 – 2019 (X+22)<sub>22</sub>





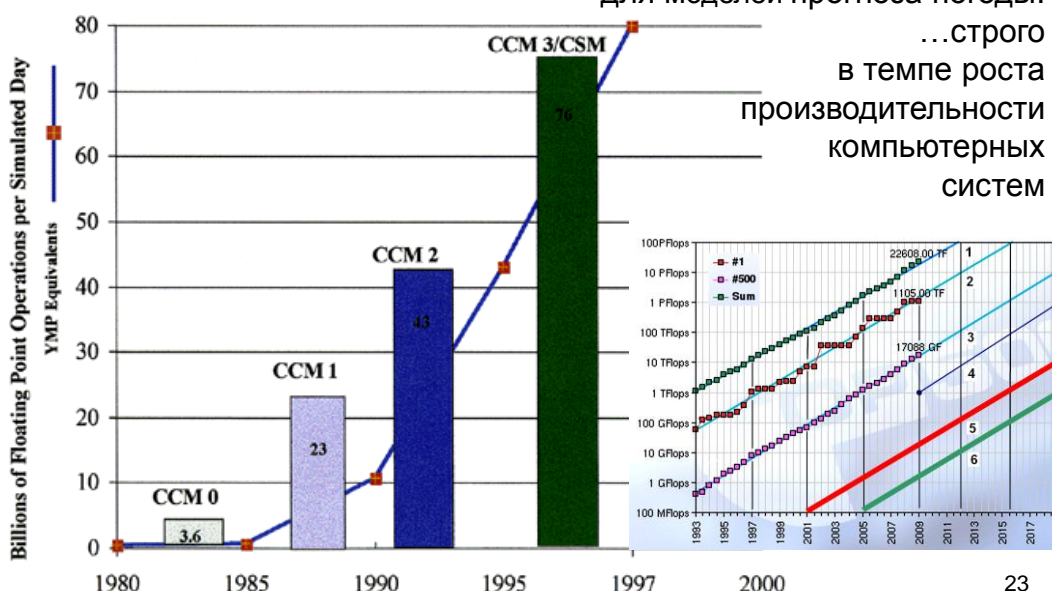
**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

Гфлопс  
на один день  
прогноза погоды

Динамика роста  
требуемой производительности  
для моделей прогноза погоды:

... строго  
в темпе роста  
производительности  
компьютерных  
систем



**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

### Основные форматы стандарта IEEE 754-2008 (вместо IEEE 754-1985)

	Название	Разрядность мантиссы со знаком (десятичных знаков)	Разрядность порядка (со знаком)	Макс. значение порядка
binary32	Single (одинарная точность)	24 (>6)	8	+127
binary64	Double (двойная точность)	53 (>12)	11	+1023
binary128	Quad (квадро точность)	113 (>28)	15	+16383
$\pi$	3,14...	3,1416...	3,1415926535897932384626433832795...	24



**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

## Основные форматы стандарта IEEE 754-2008 (вместо IEEE 754-1985)

binary32	Single (одинарная точность)	24 (>6)	8	+127
binary64	Double (двойная точность)	53 (>12)	11	+1023
binary128	Quad (квадро точность)	113 (>28)	15	+16383

$\pi$  3,14 3,1416 3,1415926535897932384626433832795...

Максимально доступная на сегодня точность: до стомиллионного знака: 42 МБ

25



**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

3,14 3,1416 3,1415926535897932384626433832795...

Но большинство исходных данных  
представлены с существенно меньшей точностью:

От 5% «инженерной погрешности» эпохи логарифмических линеек  
до примерно 0,005 %, получаемых  
от современных электронных датчиков

Кроме этого, потеря точности во время вычислений и пр.

**Современное компьютерное число не содержит никакой информации  
об исходной или текущей точности – по мере роста разрядности  
это превращается в большую проблему**

Все это породило «интервальные вычисления»,  
помогающие решить многие проблемы вычислительного  
моделирования сложных систем





**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

3,14    3,1416    3,14159**2653589793238462643383**2795...

**Современное компьютерное число не содержит никакой информации об исходной или текущей точности – по мере роста разрядности это превращается в большую проблему**

Проблемы «интервальных вычислений»:

1. Сложность аппаратной поддержки
2. Существенное противоречие, связанное с излишней (максимальной) точностью задания границ интервалов, отражающих зачастую как раз неточность исходных данных

**Вариант решения: переход к нормированной («естественной») интервальности на базе тетракодов**

27



**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

Современное компьютерное число не содержит никакой информации об исходной или текущей точности – по мере роста разрядности это превращается в большую проблему

**Правильное (постбинарное) представление количественных значений на базе тетракода:**

**0101101111101001MMMMMMMMAAAAAAAAAAAAAAAAAAAA**

«точечная»	«интервальная»	«незначимая»
часть (бинарная)	часть	часть

**Естественный «параллелизм» такого числа определяется количеством разрядов множественности М**

28



**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

### Основные перспективные направления постбинарных исследований в ДонНТУ:

1. Использование в теории и практике **моделирования клеточных автоматов** (вначале это было просто средство наглядной визуализации двумерных тетракодов)
2. Реализация «естественных» интервальных вычислений программным путем (в ближайшее время **на базе SciLab**) на системах различного масштаба (от мобильных до суперкомпьютерных)
3. Аппаратная реализация постбинарных процессоров (на базе лаборатории интегральной схемотехники фирмы Сименс)

29



**Аноприенко**  
Александр Яковлевич

**Постбинарный компьютеринг...** в  
контексте кодо-логической эволюции

### Одно из направлений постбинарных исследований в ДонНТУ:



Реализация «естественных» интервальных вычислений программным путем **на базе SciLab** на системах различного масштаба (от мобильных до суперкомпьютерных)

Доступная с 2011 года техническая база:  
переданный ДонНТУ  
Штуттгартским университетом  
суперкомпьютер  
**NEC Xeon Linux Cluster**  
100 2-х процессорных узлов  
**2,5 ТФлопс**  
(В 2005-2006 в Top 500)

30



Міжнародний науковий конгрес  
з розвитку інформаційно-комунікаційних технологій  
та розбудови інформаційного суспільства в Україні

Київ, 17-18 листопада 2011 р.

# Ноокомпьютинг и будущее информационно-компьютерной инфраструктуры

**Аноприенко Александр Яковлевич**

Декан факультета  
компьютерных наук и технологий  
Донецкого национального технического университета (ДонНТУ)  
Директор Технопарка ДонНТУ УНИТЕХ

1



Міжнародний науковий конгрес  
з розвитку інформаційно-комунікаційних технологій  
та розбудови інформаційного суспільства в Україні

Київ, 17-18 листопада 2011 р.

# Университетская составляющая в формировании информационного общества в Украине и мире

**Аноприенко Александр Яковлевич**

Декан факультета  
компьютерных наук и технологий  
Донецкого национального технического университета (ДонНТУ)  
Директор Технопарка ДонНТУ УНИТЕХ

2



Аноприенко  
Александр ЯковлевичУниверситетская  
составляющая

## «Информационное общество»

- «общество знания, в котором главным условием благополучия каждого человека становится знание, полученное благодаря беспрепятственному доступу к информации и умению работать»

3

Аноприенко  
Александр ЯковлевичУниверситетская  
составляющая

## «Информационное общество»:

**Роль и ответственность системы образования**, призванной обеспечить уже не только высокий уровень грамотности населения, но и подготовить достаточное количество **высококвалифицированных специалистов**, в том числе **научную элиту**.

**Университеты** являются одной из важнейших составляющих процесса формирования информационного общества, **залогом его успешности** и одним из главных источников соответствующих изменений.

**Университеты** должны стать не только поставщиком кадров и образовательных услуг, но и **местом общения и передачи знаний**, местом где интенсивно рождается **новое знание**, место, где знания инициируют **инновационные процессы**.

4



Аноприенко  
Александр Яковлевич

Университетская  
составляющая



## «Информационное общество»:

### США:

Главное условие создания «силиконовых долин»:  
**наличие очень хорошего университета.**

Успех США в становлении информационного общества обусловлен во многом их выбором в пользу **относительно плохих школ и очень хороших университетов**, в отличие от большинства других развитых, но менее успешных стран с хорошими школами и относительно плохими университетами.

Считается, что «**позднее зажигание**» уже в университетской среде значительно **лучше, чем загубленные** в вузах школьные **вундеркинды**.

5



Аноприенко  
Александр Яковлевич

Университетская  
составляющая



Типичный американский «хороший университет»:

### Стэнфордский университет



Колыбель американской традиции университетских технопарков

Традиции, которые позволяют создавать магистрантам и аспирантам этого университета **крупнейшие компьютерные фирмы**, являющиеся эффективными **катализаторами** становления «**информационного общества**».

Такого рода прецеденты создают в перспективе буквально **лавину нововведений**. «Студенты Стэнфорда считаются предприимчивее студентов Йеля - но не из-за разницы в характерах, просто у студентов Йеля меньше **наглядных примеров**»

6



Аноприенко  
Александр Яковлевич

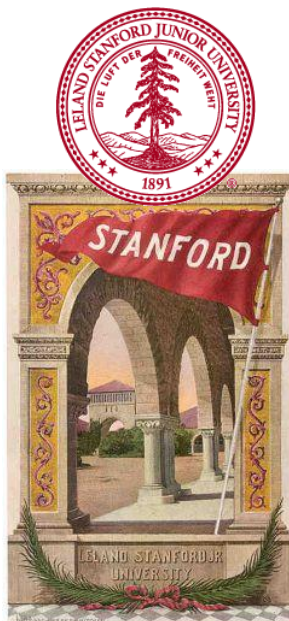
Университетская  
составляющая



1939: 2 магистра



1981: профессор  
(Джеймс Кларк) + 7 студентов  
и сотрудников



YAHOO!

1994: 2 аспиранта  
(Джерри Янг и Дэвид Файло)

Google

1998: 2 аспиранта  
(Сергей Брин и Ларри Пейдж)

8



Аноприенко  
Александр Яковлевич

Университетская  
составляющая



За пределами США, в том числе в **Украине**:

**Одна из самых серьезных проблем** заключается  
в количестве сил, которое необходимо потратить  
на учреждение компании и ее «запуск».

В США считается, что **стартапы**  
**по своей природе маргинальны**  
и создают их в основном  
относительно **бедные и**  
**не очень уверенные** люди  
в свободное время  
в малопривлекательных местах.



9



**Аноприенко**  
Александр Яковлевич

**Университетская**  
составляющая



«Молодые компании крайне хрупки...  
И общество со слишком жесткими законами  
все их уничтожит»

Но ... в информационную эпоху страны,  
**лишенные стартапов,**  
останутся **далеко позади** прочих  
подобно странам, пропустившим  
промышленную революцию.

10



**Аноприенко**  
Александр Яковлевич

**Университетская**  
составляющая



**В Украине:**

**более 60-ти** вузов готовят специалистов  
в области **компьютерных наук,**  
**около 60-ти** – в области **компьютерной инженерии,**  
и **45** – в области **программной инженерии.**

Эта и есть та «ударная сила», которая призвана  
формировать в стране **успешное информационное общество.**

Но, к сожалению, существующие в украинских вузах  
**финансовые и организационные условия не**  
**способствуют** ни развитию предпринимательской  
инициативы у студентов и аспирантов, ни формированию  
стартапов.

11



**Аноприенко**  
Александр Яковлевич

**Университетская**  
составляющая



**Постоянная утечка** лучших выпускников аспирантуры, магистратуры и бакалаврата украинских компьютерных факультетов **в зарубежные компании и университеты** не позволяет нарастить в Украине **критическую массу** высококвалифицированных и инициативных специалистов, **необходимую для формирования** успешного функционирования **современного информационного общества.**

12



**Аноприенко**  
Александр Яковлевич

**Университетская**  
составляющая



**Опыт**  
Донецкого национального технического университета (ДонНТУ):  
**С 1961 года – компьютерные технологии**



Гагарин сообщил: «Прошу доложить  
но Никите Сергеевичу Хрущеву, что  
льно, чувствую себя хорошо, травм

ека в космиче-  
грандиозные  
человечеством.

**12 апреля 1961 года в ДПИ**  
**самое первое** занятие с  
**использованием**  
**электронной**  
**вычислительной машины**  
**(АВМ МН-7)**



Советскому космонавту, впервые в мире совершившему космический полет,  
**майору ГАГАРИНУ Юрию Алексеевичу**

Дорогой Юрий Алексеевич!  
Мне доставляет большую радость горячо поздравить Вас с выдающимся героическим подвигом — первым космическим полетом на корабле-спутнике «Восток».

Совершенный Вами полет открывает новую страницу в истории человечества в покорении космоса и наполняет сердца советских людей великой радостью и гордостью за свою социалистическую Родину. От всего сердца поздравляю Вас со счастливым возвращением из космического путешествия на родную землю. Обнимаю Вас.  
До скорой встречи в Москве.

**Н. ХРУЩЕВ**

12 апреля 1961 года.

**12 апреля 1961 года — Т О В А Р И Щ ,**  
**ЗАПОМНИ ЭТОТ ДЕНЬ!**





Аноприенко  
Александр Яковлевич

Университетская  
составляющая



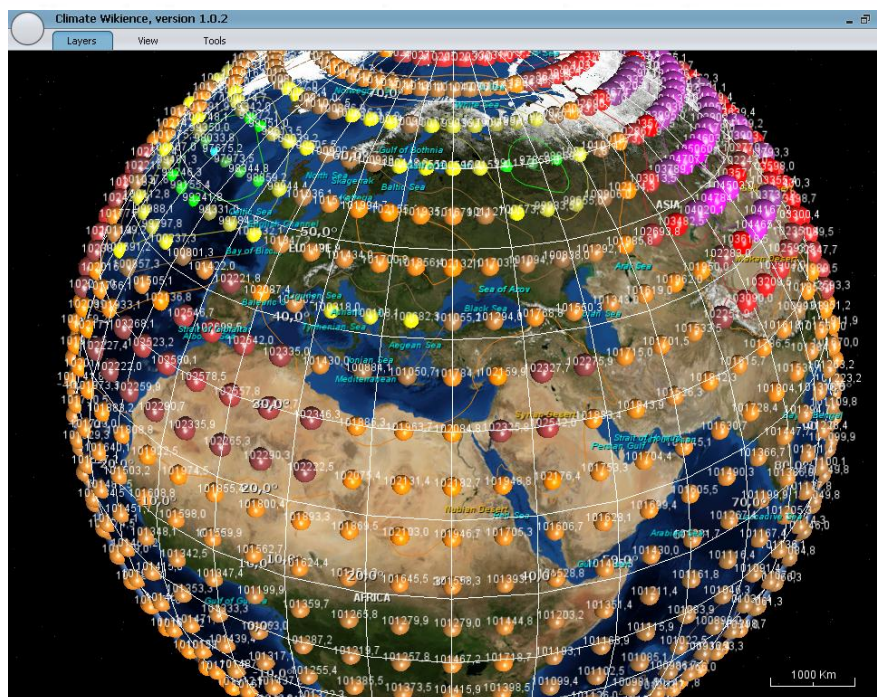
## Опыт ДонНТУ:

- С 1979 года: сотрудничество с немецкими университетами  
(Госпремия 2011 и суперкомпьютеры)
- С 1997 года: участие в проекте и ассоциации **URAN**  
(к 2007 г. современный Интернет)
- С 2000 года: технопарк ДонНТУ **УНИТЕХ**  
(некоторые успехи и борьба за выживание)
- С 2000 года: **портал магистров** ДонНТУ.  
(один из наиболее посещаемых и заметных  
научно-образовательных ресурсов  
в Украине и СНГ)
- С 2010 года: **суперкомпьютерный** центр
- С 2011 года: присоединение Института информатики и  
искусственного интеллекта  
(2011: почти треть всех принятых в ДонНТУ студентов  
- компьютерщики!)

14



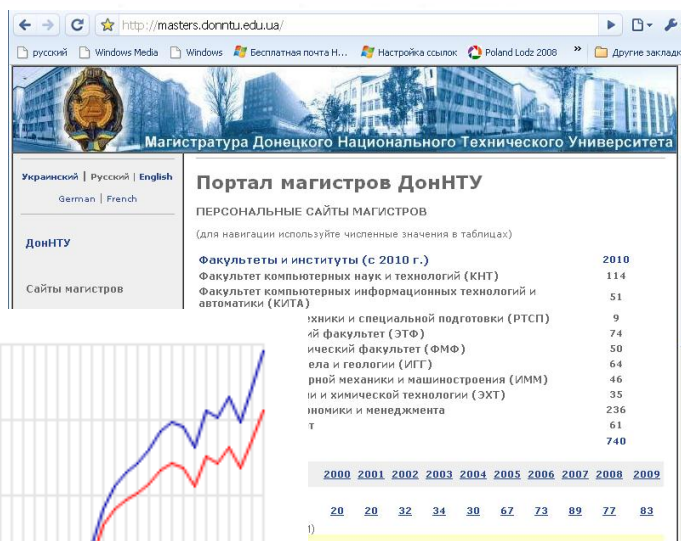
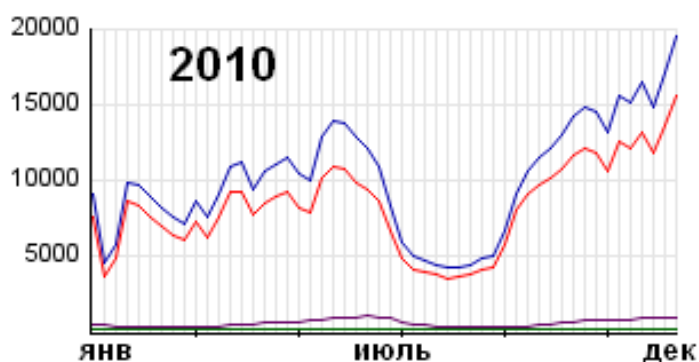
## CRGPU: Анализ и визуализация больших объемов глобальных данных (ua-us)





## Вводная лекция для студентов факультета КНТ

### Портал магистров ДонНТУ



Очередной рекорд  
конец 2010 г.:  
**20 тысяч**  
посетителей в сутки! 16



## Вводная лекция для студентов факультета КНТ

3 тысячи магистров:  
первый студент факультета,  
прошедший стажировку  
в штаб-квартире Microsoft,  
делится своим опытом

masters.donntu.edu.ua/2008/fvti/bublichenko/ind/ind



Бубличенко Александр Владимирович  
Факультет вычислительной техники и информатики

Тема выпускной работы магистра:

"Алгоритмы фрактального сжатия изображений: с



[Резюме](#) | [Автобиография](#) | [Автореферат](#) | [Библиотека](#) | [Ссылки](#) | [Отчет о поиске](#) | [Индивидуальное задание](#)

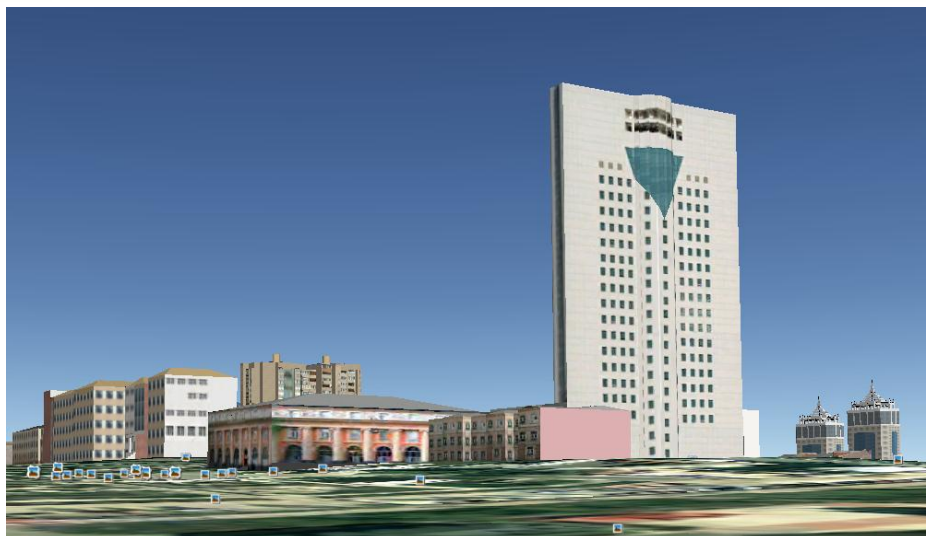
### Собеседования в ведущих западных IT-компаниях. Рекомендации для программистов

Проведение собеседований по приему на работу новых сотрудников в крупнейших западных международных компаниях, занимающихся информационными технологиями, уже давно стало искусством. Такие известные компании как Microsoft, IBM и другие сформировали определенную культуру проведения интервью, о которых ходят слухи уже не один десяток лет. Действительно, собеседование программиста требует неординарного подхода, чтобы за несколько часов адекватно оценить его способности. Еще большего искусства это требует, когда речь идет о собеседовании выпускников ВУЗов. Ведь задача собеседования состоит не только в определении квалификации кандидата, а и в раскрытии его будущего потенциала.

17



## Вводная лекция для студентов факультета КНТ



Студенты факультета КНТ: 3D Донецк  
в Google Планета Земля: с 2011 – Magenta... и др.

18



Аноприенко  
Александр Яковлевич

Университетская  
составляющая



**2011:** впервые 6 университетов Украины в 10% лучших (из 20-ти тыс.)  
впервые 5 – в тысяче лучших (по отдельным показателям)

## Ranking Web of World Universities

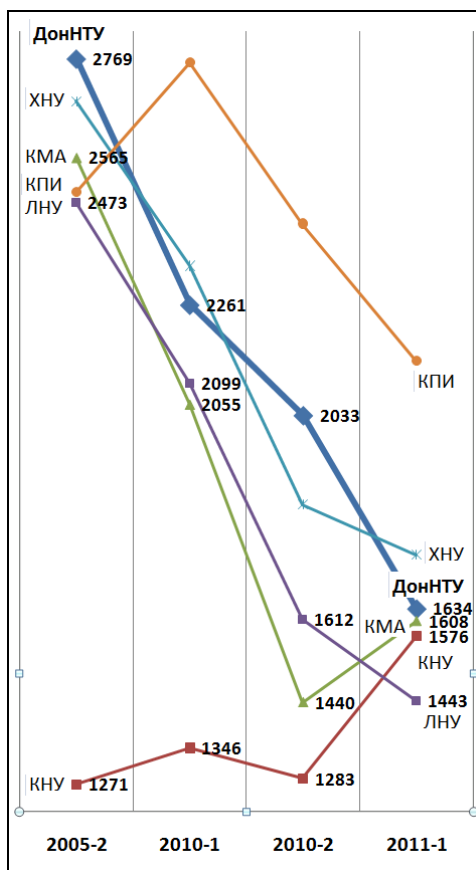
July 2011

### Rank of Universities of Ukraine

Universities 1 to 50 of 137

WORLD RANK	UNIVERSITY	POSITION			
		SIZE	VISIBILITY	RICH FILES	SCHOLAR
1321	National Taras Shevchenko University of Kyiv	1,272	2,533	1,365	1,226
1325	National Technical University of Ukraine Kiev Polytechnic Institute	1,349	2,929	1,761	868
1395	Kharkov National University VN Karazin	2,755	3,364	1,771	544
1409	Ivan Franko National University of Lviv *	2,836	4,323	1,117	795
1738	Lviv Polytechnic National University	2,551	6,130	2,634	785
1977	Donnetsk National Technical University	998	4,192	1,883	2,066
2397	Mohyla Academy University of Kiev	2,546	2,866	3,168	2,113





## Университетская составляющая



### Украинские университеты 2010-2011: прорыв в 1000 лучших

Но... по мнению западных экспертов **для** присутствия в **Топ-500** университет **должен** иметь **годовой бюджет минимум в 1 млрд \$ (а не 10 млн \$)!!!**

### Ranking Web of World Universities

Rank of Universities of Ukraine					
July 2011					
Universities 1 to 50 of 137					
WORLD RANK	UNIVERSITY	POSITION			
		SIZE	VISIBILITY	RICH FILES	SCHOLAR
1321	National Taras Shevchenko University of Kyiv	1,272	2,533	1,365	1,226
1325	National Technical University of Ukraine Kiev Polytechnic Institute	1,349	2,929	1,761	868
1395	Kharkov National University VN Karazin	2,755	3,364	1,771	544
1409	Ivan Franko National University of Lviv *	2,836	4,323	1,117	795
1738	Lviv Polytechnic National University	2,551	6,130	2,634	785
1977	Donnetsk National Technical University	998	4,192	1,883	2,066
2397	Mohyla Academy University of Kiev	2,546	2,866	3,168	2,113

20



Аноприенко  
Александр Яковлевич

## Ноокомпьютинг



### Пример России:

в прошедшем десятилетии

**фактически провалилась** федеральная целевая программа  
«**Электронная Россия**» (2002-2010 гг.: внедрение технологий  
«электронного правительства» **eGov 1.0**).

Планировалось направить **77 млрд руб.** (в ценах 2002 года).

Фактически было израсходовано лишь 21 млрд руб.  
ввиду неэффективности.

**В 2011 году в России** стартовала  
десятилетняя программа

«**Информационное общество**» (**eGov 2.0**):  
ежегодно: почти **400 млрд. руб.**, в т.ч.

120 млрд руб - из федерального бюджета

9 млрд руб. - средства субъектов федерации

200 млрд руб. - внебюджетные расходы

21



Аноприенко  
Александр Яковлевич

Университетская  
составляющая



## Что необходимо Украине (5 важнейших условий):

1. После стадионов, вокзалов и аэропортов (после 2012): **университеты** (новые корпуса и инфраструктура) !!!
2. Режим **максимального благоприятствования** развитию университетской составляющей высоких технологий (юридический, организационный, финансовый) через исследовательские центры, научные и технопарки, стартапы...
3. Создание **ассоциации** вузов (40-50) в области IT-технологий (аналог: ассоциация URAN).
4. Формирование **университетско-ориентированных венчурных фондов** и создание соревновательной, конкурентной и конкурентоспособной среды.
5. Формирование **сети IT-кластеров «Университет- IT-бизнес»**

22



Аноприенко  
Александр Яковлевич

Ноокомпьютинг



## Интернет

– техническое воплощение идей ноосферы...



23





**Аноприенко**  
Александр Яковлевич

**Ноокомпьютинг**






**Google earth**  
Around the World in 60 Seconds  
[www.oneworldmanystories.com](http://www.oneworldmanystories.com)


**2011**  
**Ноосфера**  
**от**  
**Google**

24

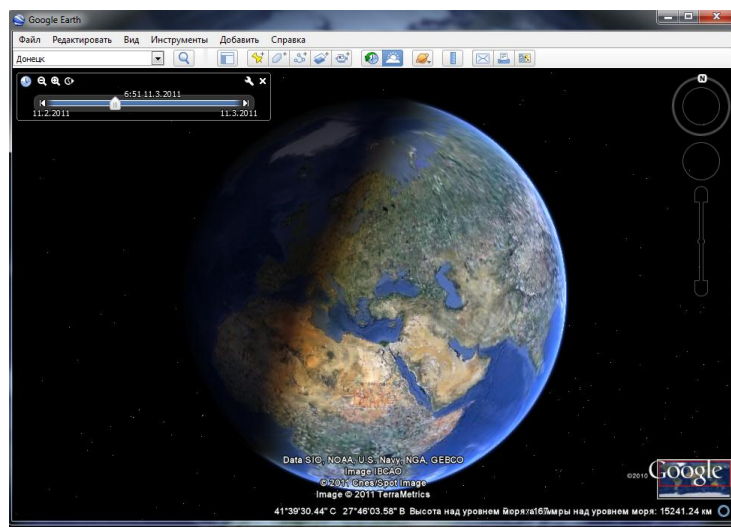


**Аноприенко**  
Александр Яковлевич

**Ноокомпьютинг**




## Ноосфера от Google: Планета Земля



Возможность работать со всей поверхностью Земли как с единым целостным объектом начиная от манипуляций со всем земным шаром (диаметр 12 тыс км) и заканчивая просмотром объектов, размер которых составляет порядка 10-ти см.


**Диапазон: в миллиард раз!!!**

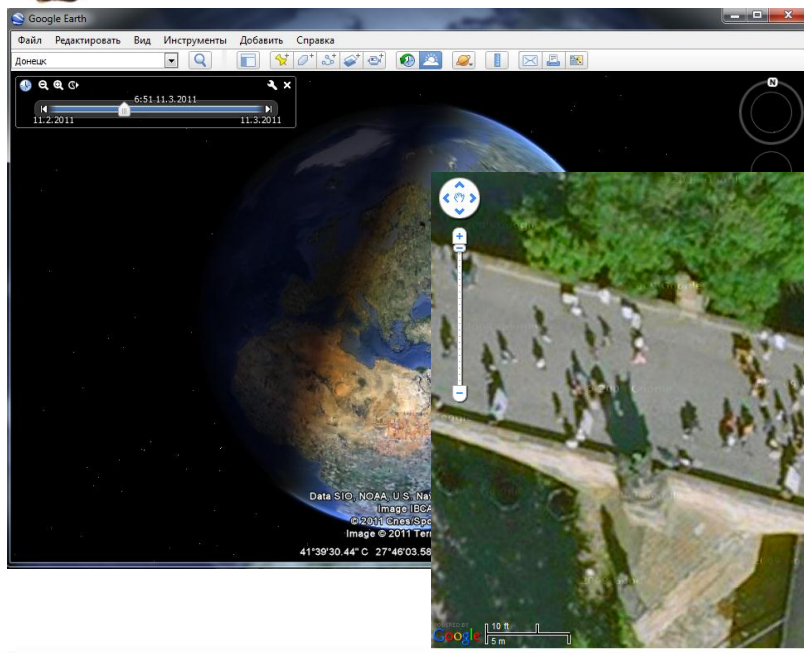
25



**Аноприенко**  
Александр Яковлевич

**Ноокомпьютинг**





**2011:**

Прага – 10 см



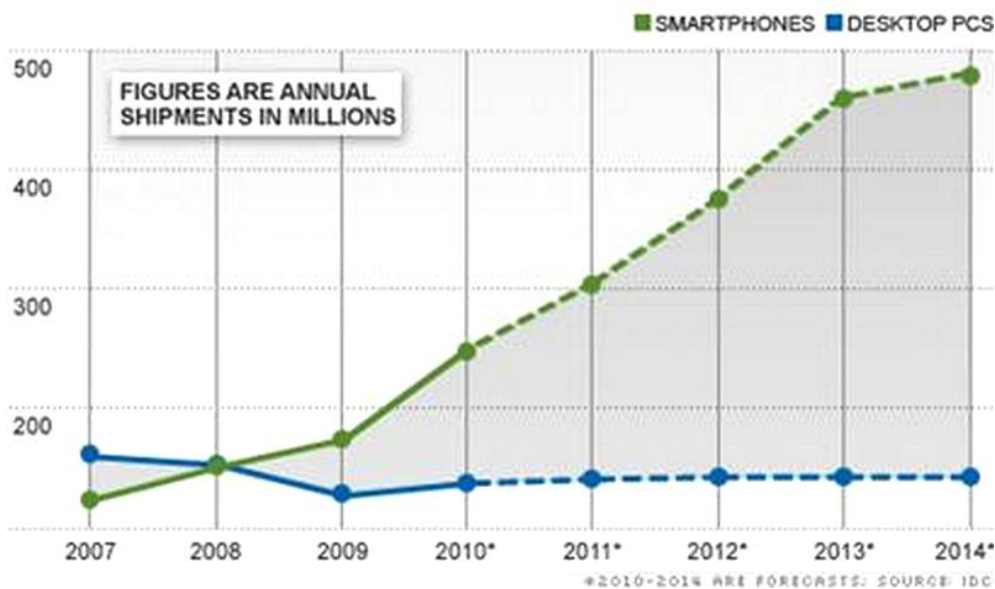


Аноприенко  
Александр Яковлевич

## Ноокомпьютинг



**2011: Сверхмобильные беспроводные смартфоны**  
с большим отрывом опережают проводные «десктопы»



28

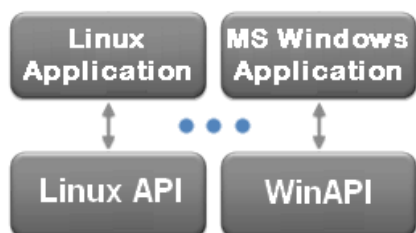


Аноприенко  
Александр Яковлевич

## Ноокомпьютинг



Традиционное  
программирование

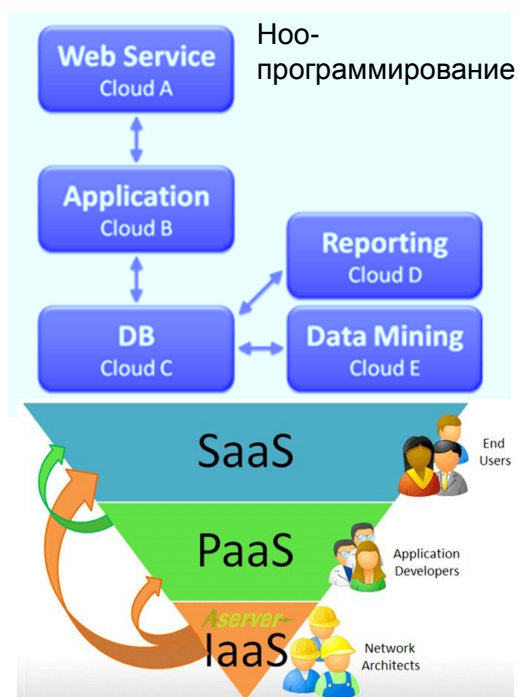


С 90-х:  
Метакомпьютинг...  
GRID...

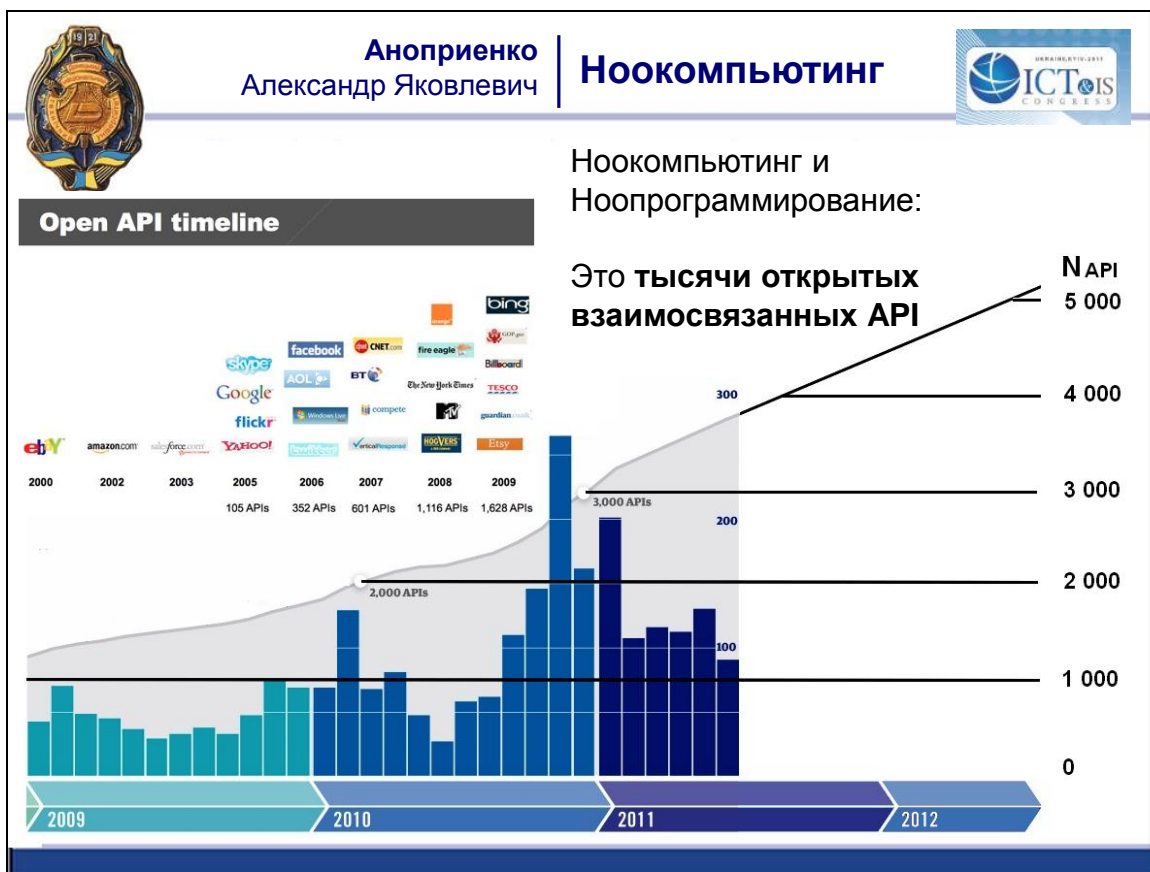
С конца 2000-х:  
**Облачные вычисления..**

С начала 2010-х:

**Ноокомпьютинг**







**Аноприенко**  
Александр Яковлевич

**Ноокомпьютинг**

**2011:**  
От Интернет  
(1 адрес на человека)  
к Ноонет (до 300 млн  
сетевых адресов  
на человека)

Интернет – это уже  
анахронизм...

**World IPv6 Day  
June 8, 2011**



Аноприенко  
Александр Яковлевич

Ноокомпьютинг



IBM Smarter Planet:  
Smart Computing etc.

2011:

Умные энергосети  
**Smart Grid**



32



Вводная лекция для студентов факультета КНТ

Сколько необходимо  
компьютерщиков и программистов ?

**И почему им так много платят ???**



**Нарастание сложности  
и разумности среды:**

Самолет Боинг-777 (1995 г.):  
(первый коммерческий авиалайнер,  
на 100 % разработанный на компьютерах)

**7 миллионов строк кода**

**Тысяча самолетов этого типа с 1995 года  
сохранили жизнь и здоровье ВСЕХ своих пассажиров!**

На тысячу самолетов **Боинг-707** (1958-1991 гг.): почти **200 катастроф**

На тысячу самолетов **Ту-154** (1968-1998 гг.): более **70-ти катастроф**

На 180 самолетов **Як-42** (1978-2010 гг.): **9 катастроф (569 погибших)**





## Вводная лекция для студентов факультета КНТ

Сколько необходимо  
компьютерщиков и программистов ?

**И почему им так много платят ???**



Самолет Боинг-777 (1995 г.):



7 миллионов строк кода



**Операционная система Android:**

**12 миллионов строк кода**

Операционная система MS Windows 7:

50 миллионов строк кода

Современный автомобиль класса «Люкс»: **100 миллионов строк кода**  
(почти половина стоимости: компьютерное оборудование  
и программное обеспечение)

34

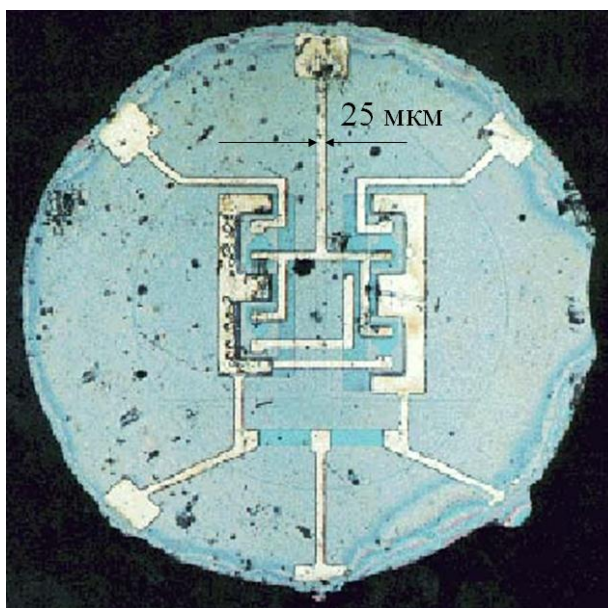


Аноприенко  
Александр Яковлевич

Ноокомпьютинг



## 2011 – 50 лет цифровой микросхеме



1961 – первая  
планарная микросхема

35



Аноприенко  
Александр Яковлевич

## Ноокомпьютинг



### If transistors were people

If the transistors in a microprocessor were represented by people, the following timeline gives an idea of the pace of Moore's Law.



Now imagine that those 1.3 billion people could fit onstage in the original music hall. That's the scale of Moore's Law.

36



Аноприенко  
Александр Яковлевич

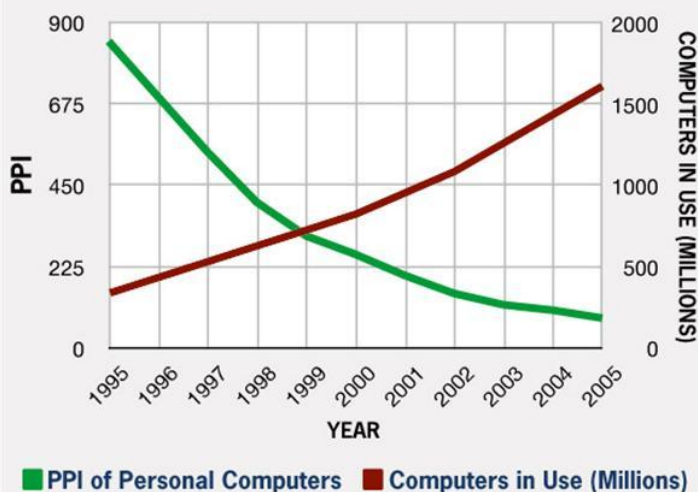
## Ноокомпьютинг



### COMPUTERS IN USE VS. PRODUCER'S PRICE INDEX (PPI)

As more consumers adopted personal computers, economies of scale rapidly reduced the cost of production.

Data sources: Source: International Data Corp., U.S. Bureau of Labor Statistics).



Снижение  
стоимости  
и рост  
количества  
компьютеров  
1995-2005:  
до 2 млрд...

37

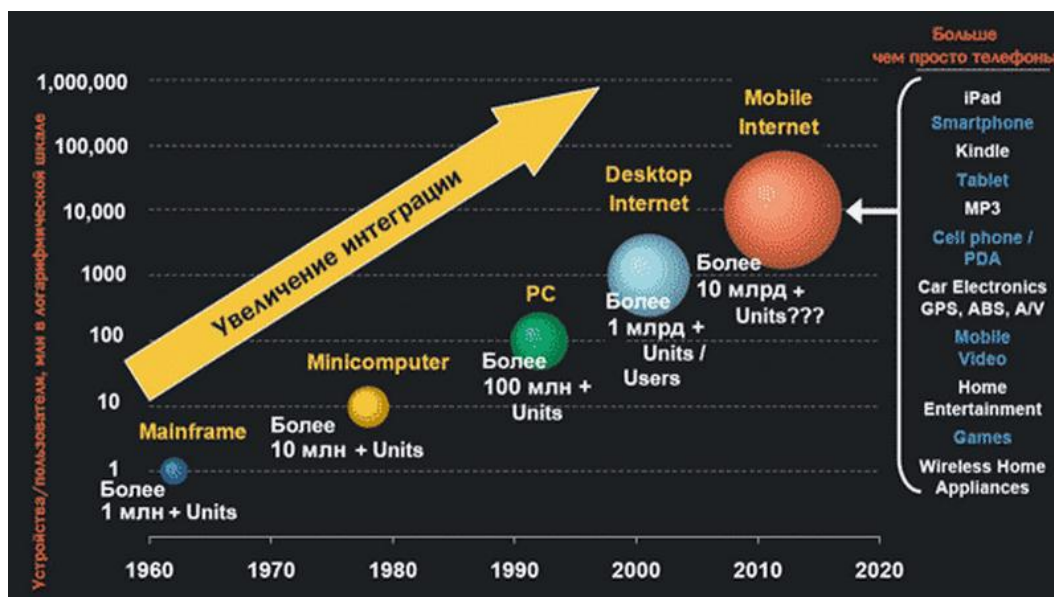


Аноприенко  
Александр Яковлевич

## Ноокомпьютинг



Рост количества компьютерных устройств: 2011 -> 10 млрд.



Аноприенко  
Александр Яковлевич

## Ноокомпьютинг



...Метакомпьютинг – Grid – Облачные вычисления – Ноокомпьютинг...





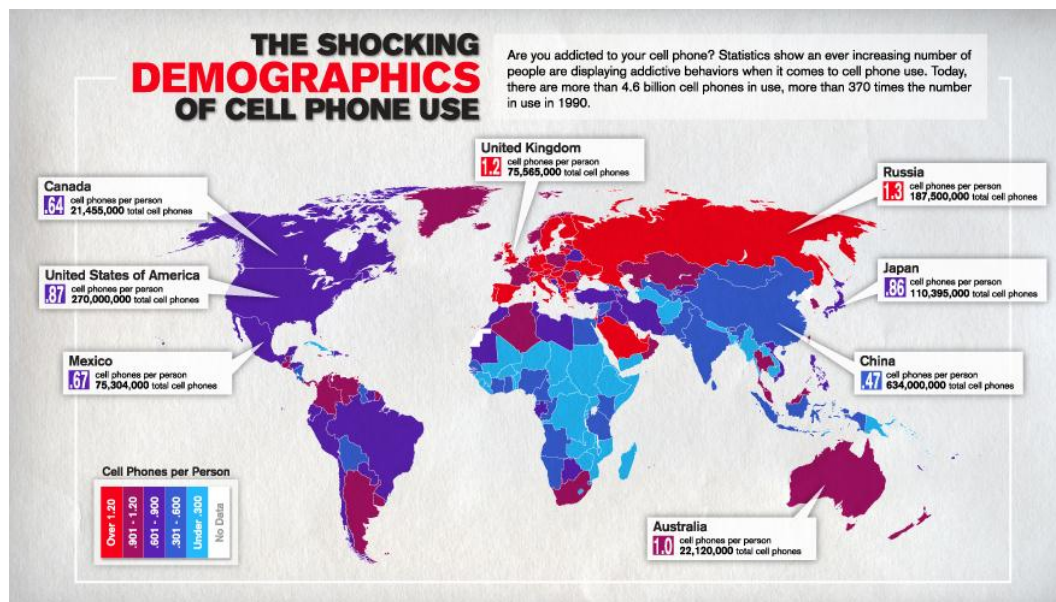


Аноприенко  
Александр Яковлевич

Ноокомпьютинг



Вернадский – Украина - ноосфера – **НООКОМПЬЮТИНГ...**



40



Аноприенко  
Александр Яковлевич

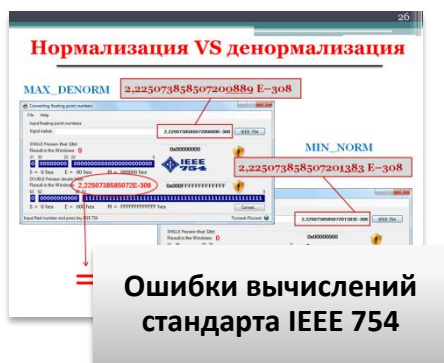
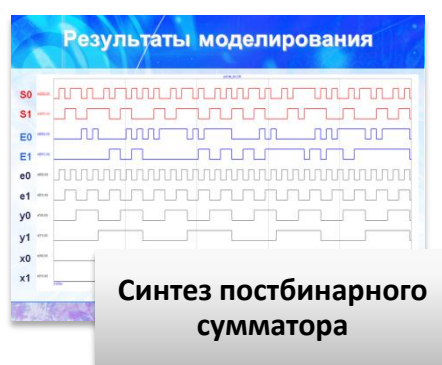
Ноокомпьютинг



Ноологизмы  
Ноопарадигма  
Ноопрограммирование  
**НООКОМПЬЮТИНГ**  
Ноомоделирование Нооология Ноографика  
Ноотехнологии Ноонет (ноосеть) Ноороботы  
Ноонетика Ноокибернетика  
**Ноосфера**  
Нооэволюция Ноогенезис Ноогеография  
Ноомодели Ноопространство Нообиблиотеки  
Ноотехнократия Ноотенденции Ноокультура  
**Нооэкономика**  
Ноократия Нооэтика  
Нооцивилизация  
Ноополисы

41

# исследования 2012





- 📖 Будущее IT-индустрии в Донецке и Украине: вызовы, возможности и перспективы, 161
- 📖 Постбинарный сопроцессор, 181
- 📖 Алгебра тетралогии. Реализация арифметических операций: синтез постбинарного сумматора, 187
- 📖 Проблемы компьютерных вычислений, вызванные использованием стандарта IEEE 754, 196
- 📖 Постбинарное округление, 211
- 📖 Погрешность представления чисел в постбинарных форматах, 220
- 📖 Операции с порядками постбинарных форматов чисел с плавающей запятой, 229
- 📖 Введение в постбинарную арифметику, 241
- 📖 Синтез постбинарных суммирующих, вычитающих и умножающих схем, 248
- 📖 Алгебра тетралогии. Синтез унарных логических операций и постбинарного исключаящего ИЛИ, 259
- 📖 Синтез блока сокращения разрядности порядка, 274
- 📖 Подготовка IT-профессионалов в украинских университетах: 50-летний опыт и следующие 50 лет, 280



Конференция «Донбасс-2020» 25 апреля 2012 г.

# БУДУЩЕЕ IT-ИНДУСТРИИ в Донецке и Украине:

ВЫЗОВЫ, ВОЗМОЖНОСТИ  
и перспективы

**Аноприенко Александр Яковлевич**

Декан факультета  
компьютерных наук и технологий  
ДонНТУ

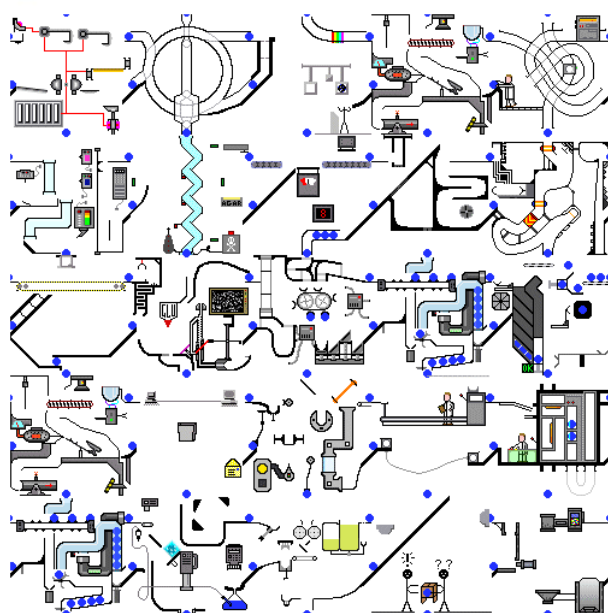


Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

**Главный вызов современности:**

**усложнение**  
окружающего мира  
– структурное и  
динамическое

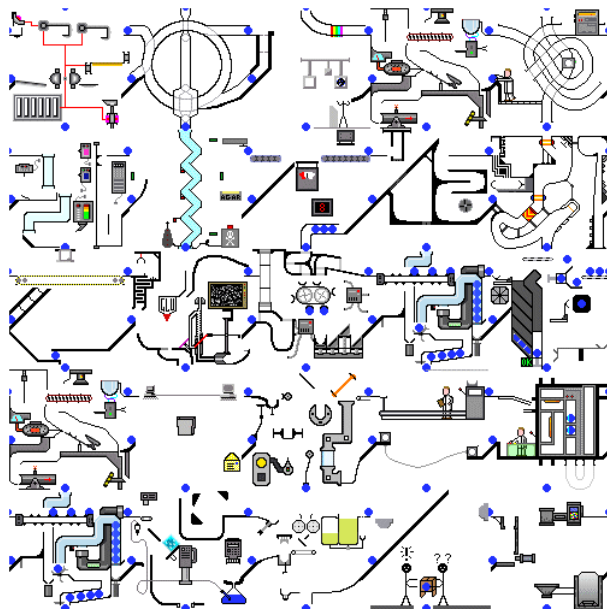




Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

## Главные **ВЫЗОВ** современности:



**усложнение**  
окружающего мира  
– структурное и  
динамическое

**Ответ:**  
тотальная  
**компьютеризация**  
(измеряемая в строках  
программного кода)



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

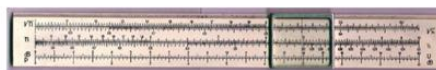
**Боевые  
самолеты:**  
**усложнение**  
программного  
обеспечения  
**на порядок**  
**каждые**  
**10 лет!**

**2000-е:**  
**F-35**  
**6 млн**  
**строк**  
**прогр.**  
**кода**  
**на борту**

Рост сложности программного обеспечения в соответствии с законом  
Августина: практически на порядок каждые 10 лет

До начала 1960-х: компьютеров на борту нет,  
логарифмическая линейка - основной инструмент

LOC  
- Lines Of Codes



1960-е



1000  
LOC  
F-4A



1970-е



50,000  
LOC  
F-15A



1980-е



300K  
LOC  
F-16C



1990-е



1.7M  
LOC  
F-22



2000-е



>6M  
LOC  
F-35



Software Engineering Institute

Carnegie Mellon

How Future Trends in Systems and Software  
Engineering Technologies Bode Well for Enabling the  
Military Mission  
Dr. Kenneth E. Nidiffer  
© 2008 Carnegie Mellon University



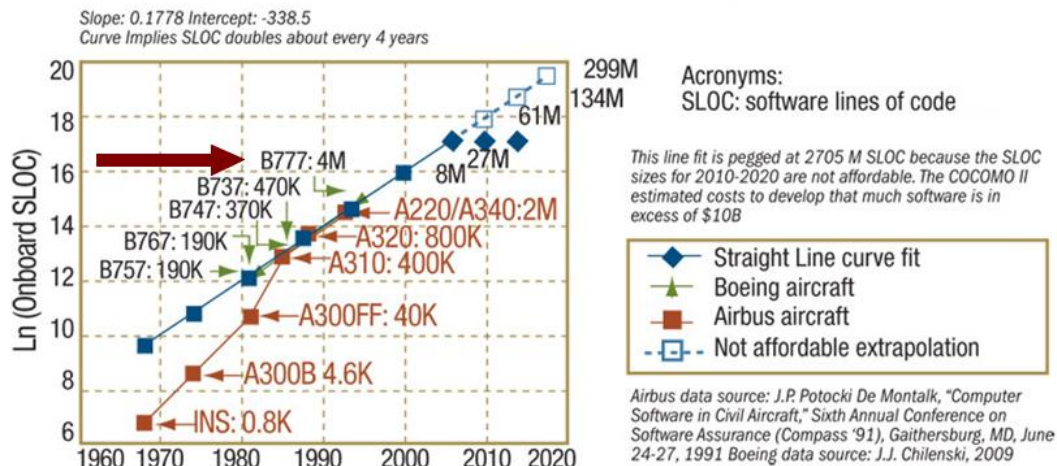
Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Рост сложности программного обеспечения в системах управления самолетов гражданской авиации

Начиная с 90-х годов (**Боинг-777**): миллионы строк программного кода

**2020 год: 300 млн. строк программного кода** на борту новых самолетов



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Рост сложности программного обеспечения в системах управления самолетов гражданской авиации



1995: Самолет **Боинг-777**

(первый коммерческий авиалайнер,  
на 100 % разработанный на компьютерах)

**7 миллионов строк кода**

Тысяча самолетов этого типа с 1995 года (за 12 лет)  
сохранили жизнь и здоровье **ВСЕХ** своих пассажиров!





Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

Уровень смертности на дорогах (по странам):

**Наивысший в России и Украине**

Страна	Число погибших в результате ДТП (на 100 тысяч жителей)
<b>Российская Федерация</b>	<b>21.1</b>
В среднем по Содружеству Независимых Государств (СНГ)	15.0
Польша	14.3
США	12.3
В среднем по Европейскому союзу	11.0
Венгрия	9.9
Новая Зеландия	8.6
Австралия	6.8
<b>Германия</b>	<b>5.4</b>

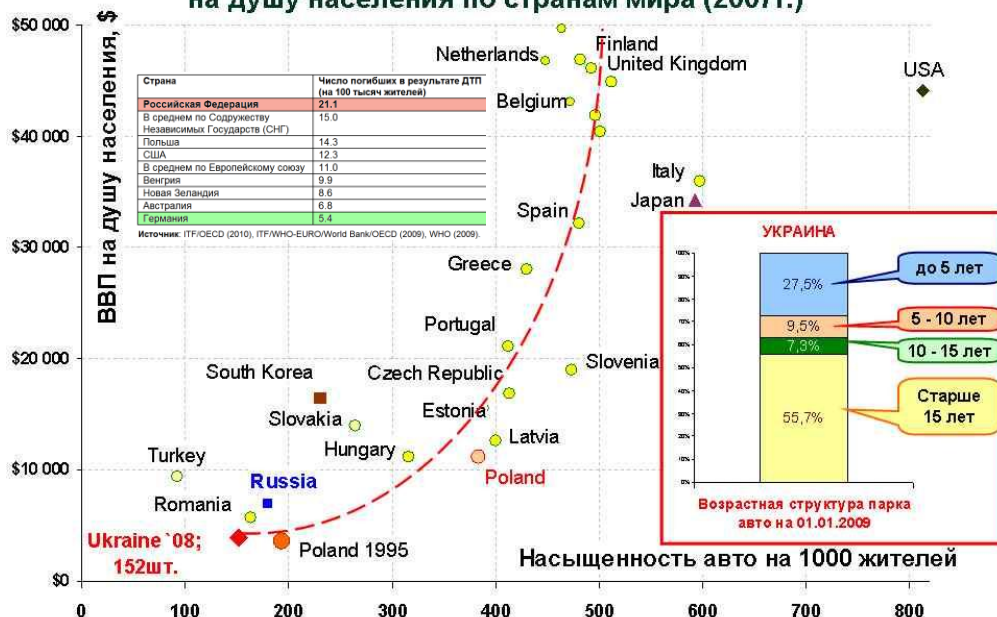
Источник: ITF/OECD (2010), ITF/WHO-EURO/World Bank/OECD (2009), WHO (2009).



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

**Взаимосвязь между насыщенностью автомобилями и ВВП  
на душу населения по странам мира (2007г.)**





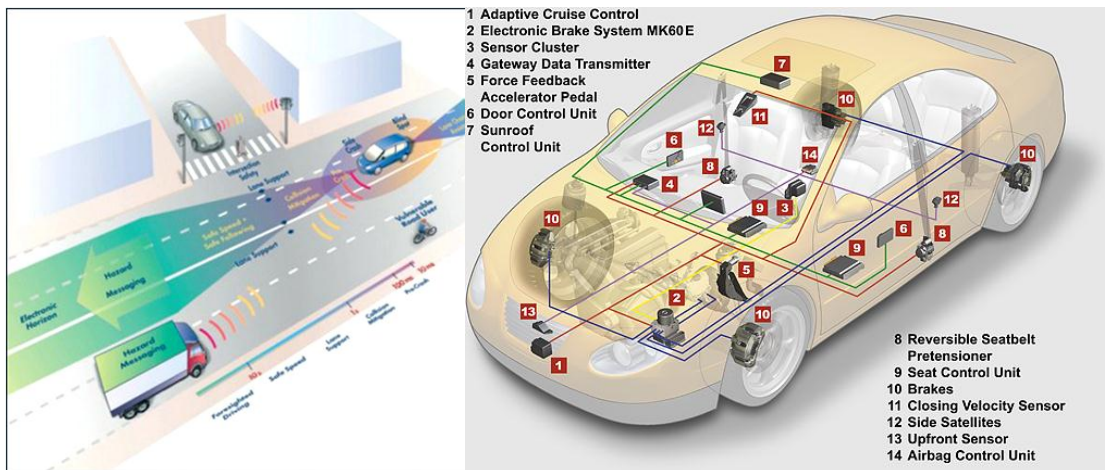


Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

Транспортная политика Европейского Союза направлена на **уменьшение смертности на дорогах на 50% к 2020 году:**

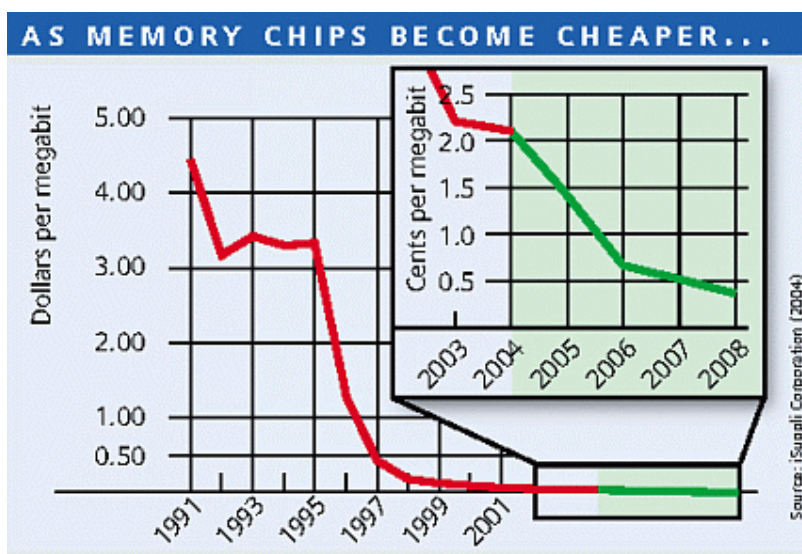
**К 2018 году 40% массовых автомобилей получат систему ADAS,** предотвращающую аварии. «Умные» системы ADAS непрерывно контролируют движение автомобиля, следя за полосой движения, дорожными знаками, светофорами..



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

**Удешевление мегабита цифровой памяти:**  
с долларов на центы (**в 100 раз с 1991 года**)



Источник: [http://www.siemens.com/innovation/en/publikationen/publications\\_pof/pof\\_fall\\_2004/software\\_articles/facts\\_and\\_forecasts.htm](http://www.siemens.com/innovation/en/publikationen/publications_pof/pof_fall_2004/software_articles/facts_and_forecasts.htm)

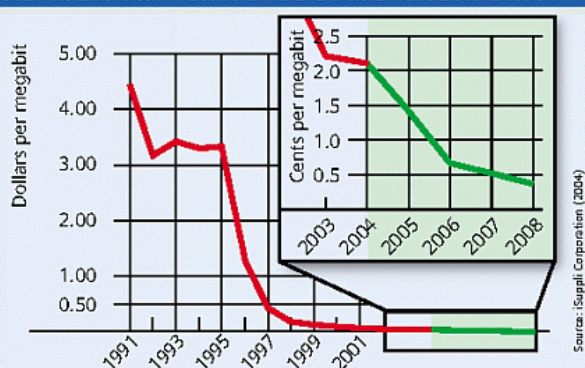


Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

## Усложнение автомобильной электроники

AS MEMORY CHIPS BECOME CHEAPER...



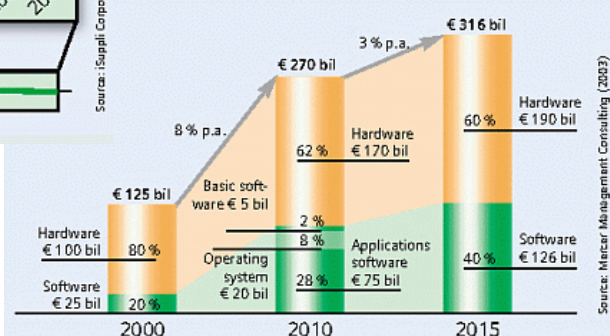
**7 миллионов строк кода**

Источник: [http://www.siemens.com/innovation/en/publikationen/publications\\_pof/pof\\_fall\\_2004/software\\_articles/facts\\_and\\_forecasts.htm](http://www.siemens.com/innovation/en/publikationen/publications_pof/pof_fall_2004/software_articles/facts_and_forecasts.htm)



Автомобили будущего (2010-х):  
**более 50 млн. строк кода**

SOFTWARE IS HITTING THE ROAD

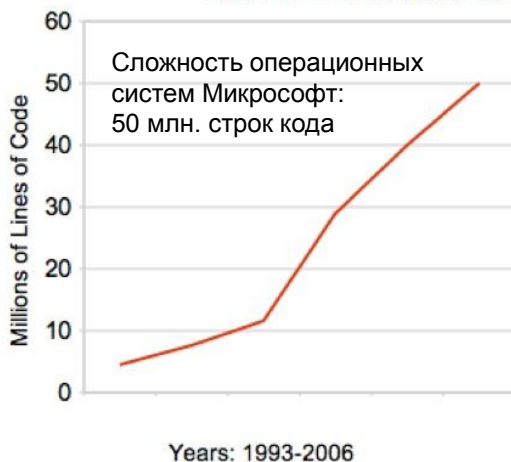


Аноприенко  
Александр Яковлевич

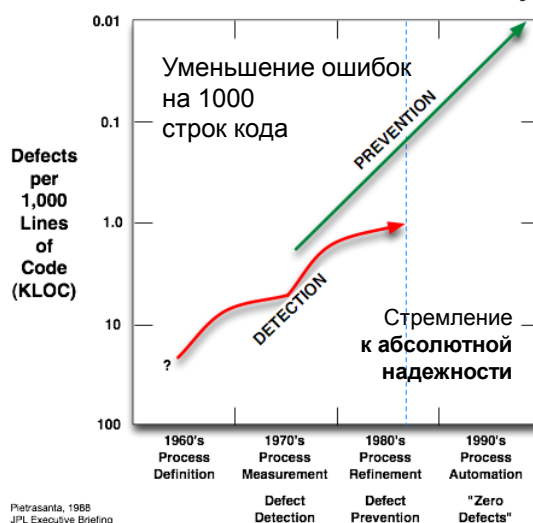
Будущее IT-индустрии  
в Донецке и Украине

Рост сложности программного кода сопровождается  
**повышением его надежности на порядок каждые 10 лет**

Lines of Code in Windows



The Evolution of Software Quality

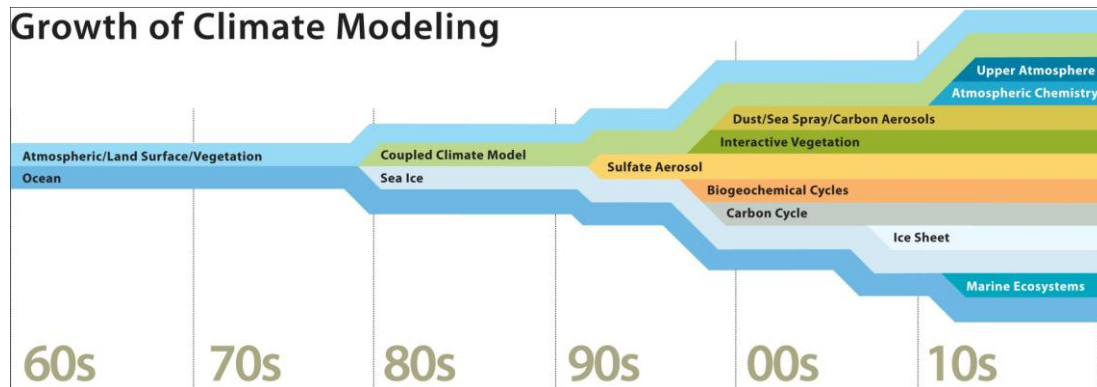




Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

**Тотальное моделирование  
и усложнение моделей,**  
в том числе **в моделировании климата**  
(толщина линий примерно соответствует объему кода)



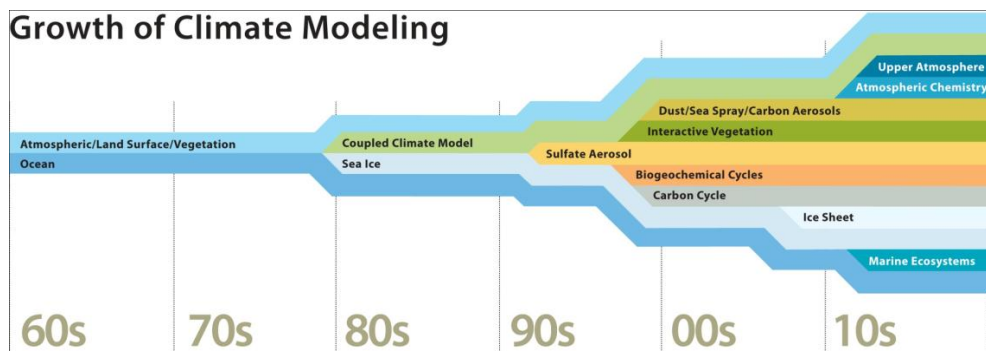
Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

**Моделирование климата:**



**ДонНТУ и университет Миннесоты**  
(США) в рамках совместного проекта  
разрабатывают уникальный **Интернет-сервис для  
анализа сверхбольших объемов  
климатических данных**, который позволит  
впервые объединить в единую информационную  
систему **более 700 характеристик, собранных  
спутниками США и Европы за последние 30 лет.**



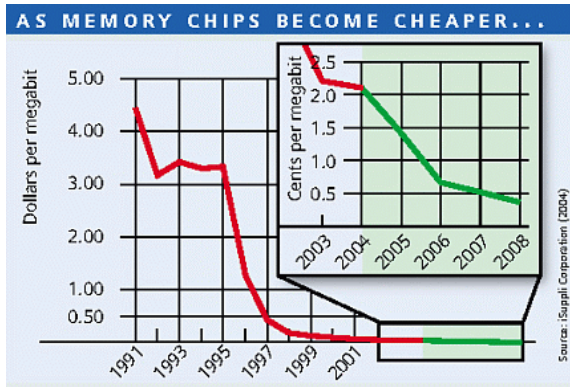




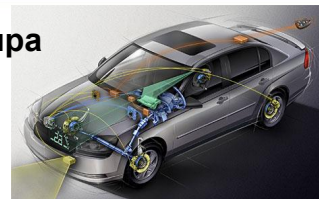
Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Интеллектуализация окружающего мира



7 миллионов строк кода



Автомобили будущего (2010-х):  
более 50 млн. строк кода

ОС «Андроид» 2012:  
12 млн. строк кода



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### ЭВОЛЮЦИЯ КОМПЬЮТЕРНОГО КОНТИНУУМА

Взгляд  
фирмы  
Интел на  
развитие  
компью-  
терного  
континуума:

«Эра  
всеобщей  
компью-  
теризации»  
это и есть,  
по-сути,  
начало  
ноо-  
компьютинга



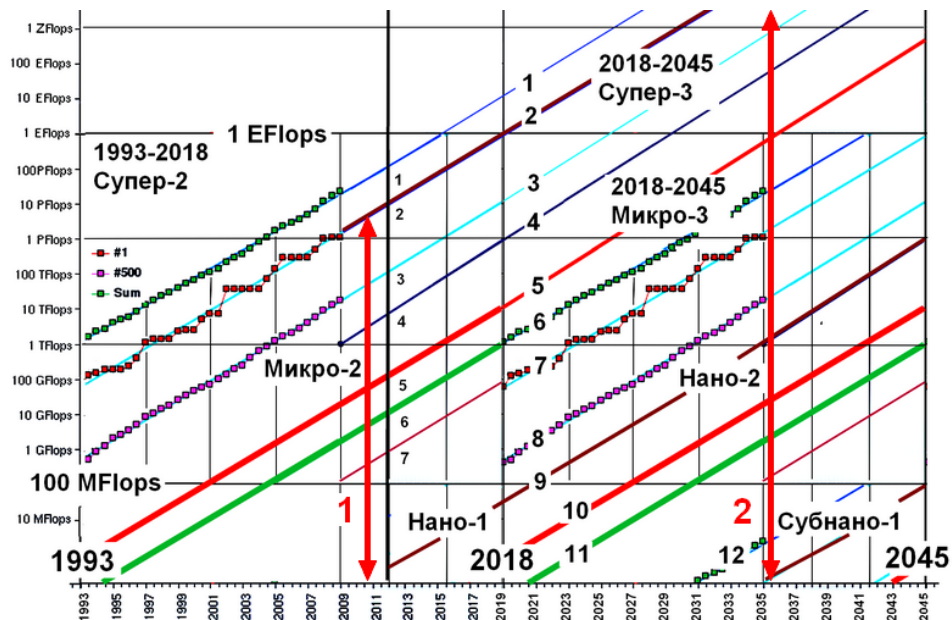


Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

Развитие и расширение компьютерного континуума:

1 – текущий; 2 – будущий, через 25 лет, дополненный нанокompьютерами



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

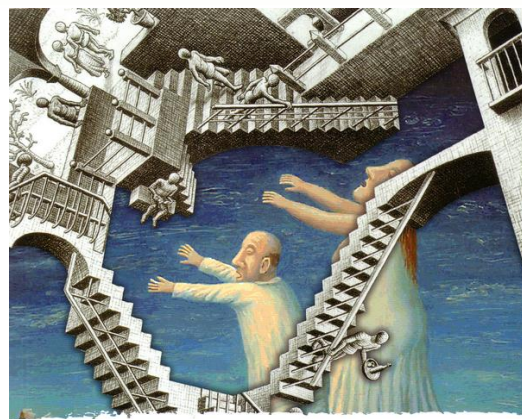
От этого теперь зависит все...

## «Глобальные тенденции-2025»

«Отдельные достижения в науке и  
технике на сегодня уже не так важны  
как **общая эффективность**  
**Национальной инновационной**  
**системы...**

Особое значение при этом имеет  
новое поколение компьютерных и  
Интернет-технологий...

**Скорое и существенное усвоение  
этих технологий обеспечит  
значительное экономическое  
преимущество»**



2025 ГОД МИР ПОСЛЕ  
КРИЗИСА

Доклад Национального  
разведывательного совета США







Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Динамика ВВП Украины в 1965-2005 гг.

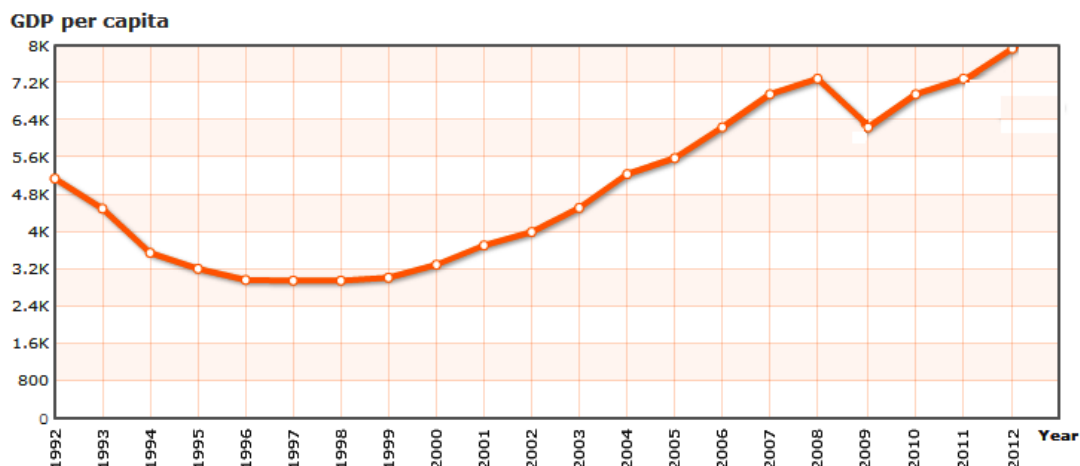


Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Динамика ВВП Украины за 20 лет независимости:

**А Украина?**





Аноприенко  
Александр Яковлевич

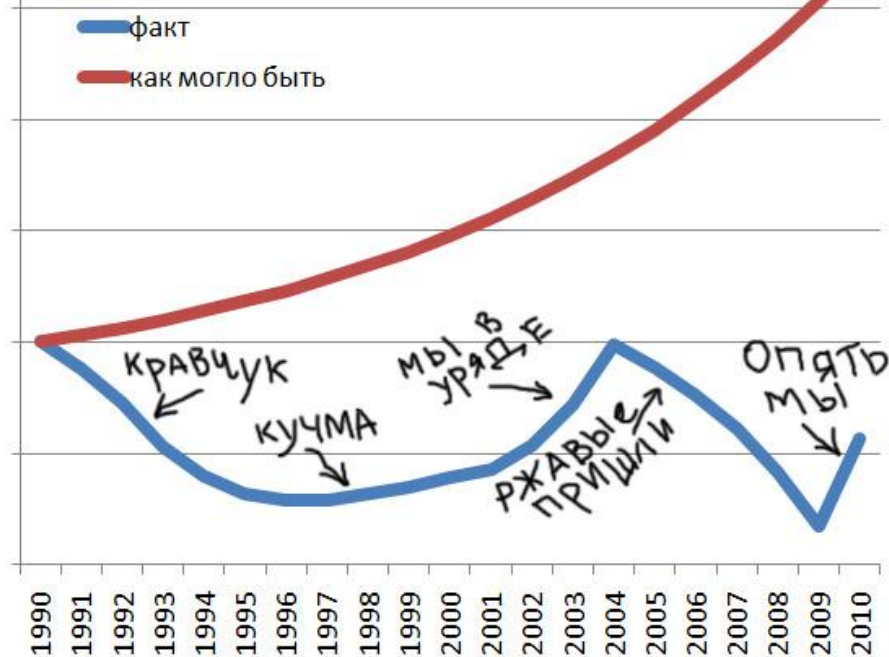
Будущее IT-индустрии  
в Донецке и Украине

В период кризиса  
2008-2009 гг.  
украинская  
экономика попала  
в пятерку  
худших в мире...

Может ли  
что-то  
в корне  
изменить  
ситуацию???



Донецкий взгляд на логику изменений в ВВП Украины ☺

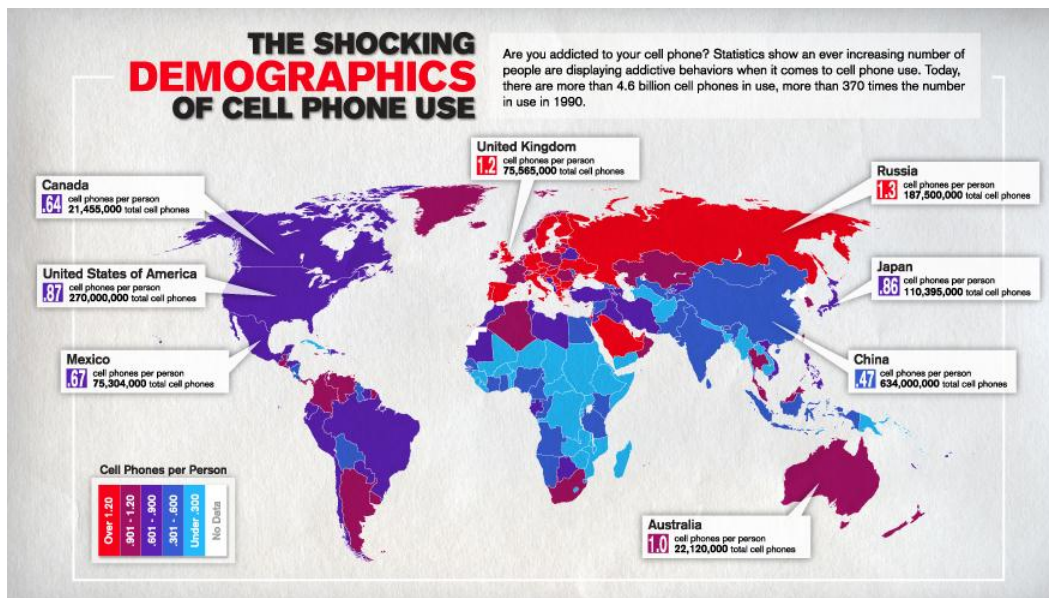




Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

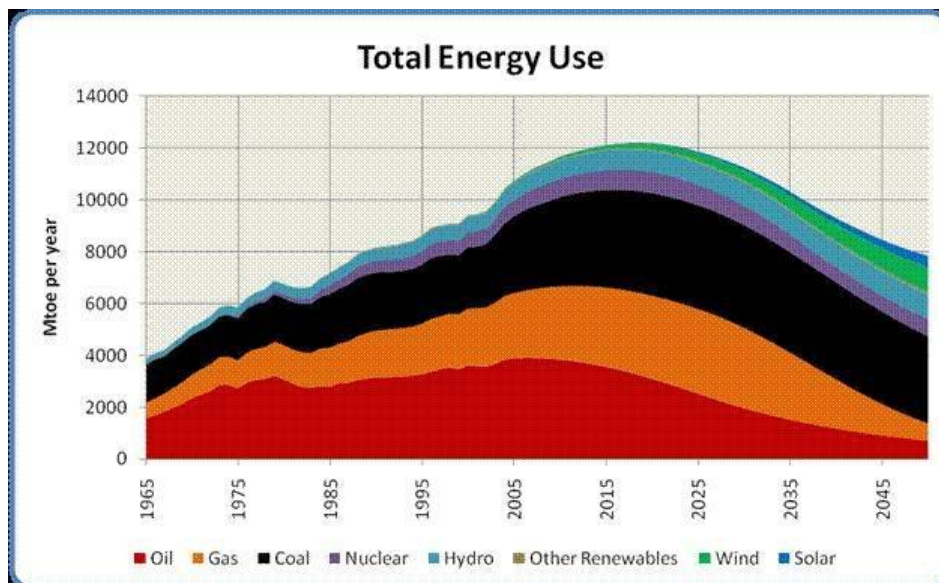
### Повод для украинского «оптимизма 2020» ☺



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Повод для «донецкого оптимизма 2020» ☺



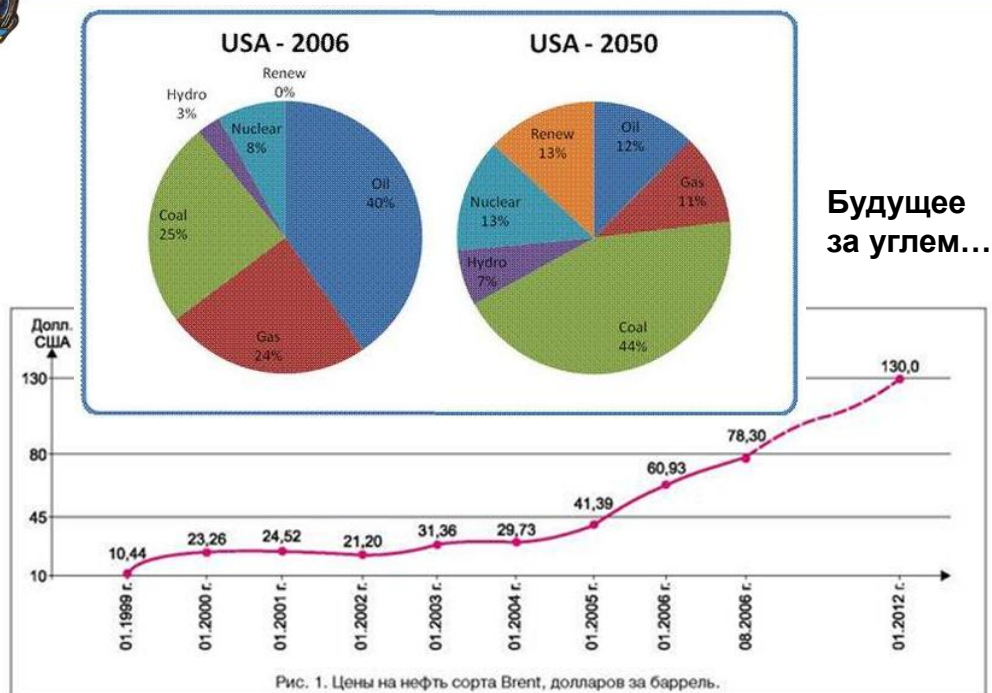
[www.paulchefurka.ca/WEAP2/Energy\\_Intensity\\_GDP\\_2050.html](http://www.paulchefurka.ca/WEAP2/Energy_Intensity_GDP_2050.html)





Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

Повод для «IT-оптимизма 2020»☺

**TOP-10**

САМЫХ ВЫСОКООПЛАЧИВАЕМЫХ  
СПЕЦИАЛИСТОВ ФЕВРАЛЯ

40 000	Генеральный директор
22 000	Коммерческий директор
21 000	Директор по маркетингу
20 000	Финансовый директор
19 600	Программист Java Developer
16 500	Программист Android Developer
15 600	iPhone программист
12 000	Программист C++ Developer
10 000	Бренд-менеджер



КТО СЕГОДНЯ ВОСТРЕБОВАН

2012



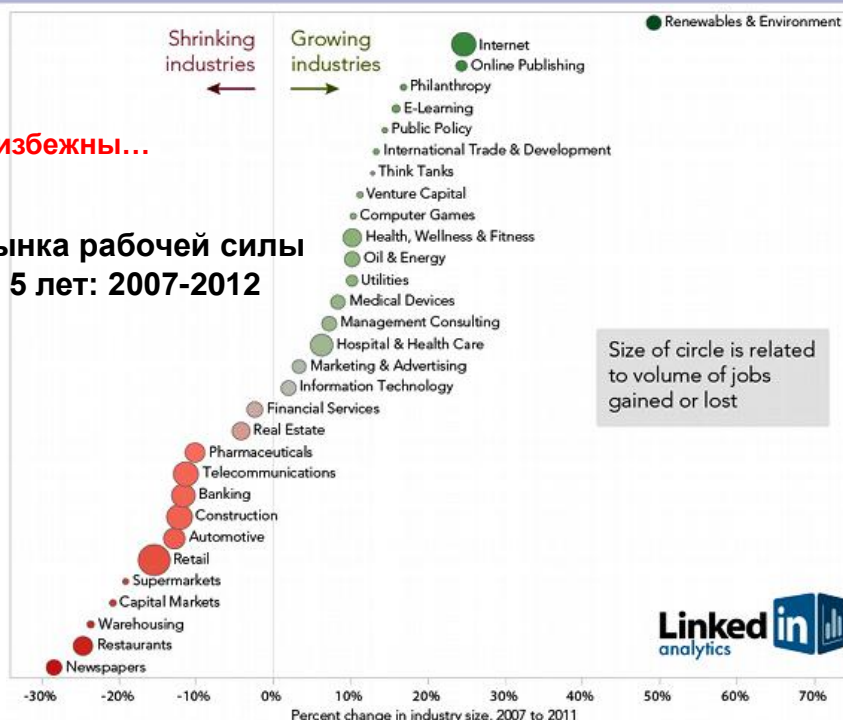


**Аноприенко  
Александр Яковлевич**

**Будущее IT-индустрии  
в Донецке и Украине**

**Изменения неизбежны...**

**Изменения рынка рабочей силы  
за последние 5 лет: 2007-2012**



**Аноприенко  
Александр Яковлевич**

**Будущее IT-индустрии  
в Донецке и Украине**

**Отраслевая структура экономики различных стран (в % ВВП)**

Страна	Сельское хозяйство	Промышленность	Услуги	ИКТ-сектор	ВВП, млрд. долл.
Весь мир	4,0	32,0	64,0	2,2	59 590
США	1,0	20,7	78,3	7,6	12 410
Евросоюз	2,2	27,3	70,5	4,0	12 180
Китай	14,4	53,1	32,5		8 172
Япония	1,3	25,3	73,4	4,0	3 914
Индия	20,6	28,1	51,3	...	3 699
Германия	1,1	28,6	70,3	4,3	2 454
Франция	2,5	21,4	76,1	4,0	1 822
Россия	5,0	35,0	60,0	1,0	1 539
Украина	22,5	33,2	44,3	1,0	319



Аноприенко  
Александр Яковлевич

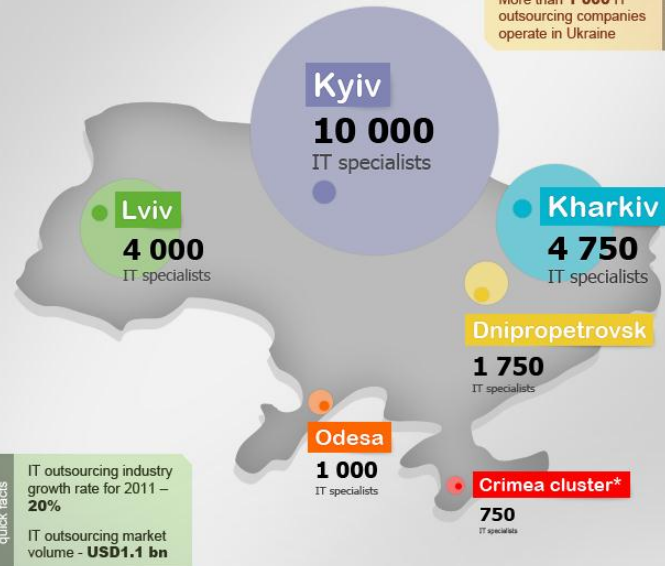
Будущее IT-индустрии  
в Донецке и Украине

Донбасс-2020:  
Назрела острая  
необходимость  
создания  
и ускоренного  
развития  
ИТ-кластера и  
в Донецке



**25 000** IT specialists  
are involved in IT outsourcing industry  
in Ukraine

Ukraine represents the  
**largest** IT outsourcing  
industry in Eastern  
Europe  
More than **1 000** IT  
outsourcing companies  
operate in Ukraine



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

В США по данным  
Национального портала по поиску  
работы CareerCast, в рейтинге 200  
лучших профессий на начало 2011  
года **1-е место занимает  
профессия «Разработчик  
программного обеспечения».**  
«Математик» — 2-е место,  
«Аналитик компьютерных систем»  
— 5-е место. В ТОП-30 вошли и  
другие компьютерные  
специальности.





Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине



## IT индустрия:

1. **Высокотехнологическая и высокорентабельная отрасль экономики.**
2. **Не требует от государства капитальных вложений, инвестиций, использования природных ресурсов.**
3. **Содействует реализации высокого образовательного, научного, интеллектуального и технологического потенциала страны.**



Міжнародний науковий конгрес  
з розвитку інформаційно-комунікаційних технологій  
та розбудови інформаційного суспільства в Україні

Київ, 17-18 листопада 2011 р.

Премьер-министр  
**Николай Азаров:**

Сегодня ВВП  
Украины достиг  
Уровня 300 млрд \$

**Эффективный рост  
в дальнейшем  
возможен только  
за счет новых  
технологий.  
Подготовка 15-ти  
тысяч ИТ-специалистов  
в год – слишком мало.**

**Потребуется в разы  
больше!**





Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Украина на пути к «стране высоких технологий»:

Количество IT-специалистов в Украине в конце 2010 года составило примерно **215 тыс. человек, из которых более 20-ти тыс. – сертифицированные высококлассные программисты, работающие на экспорт:**

1. Пятая часть рынка региона EMEA (Европа, Ближний Восток и Африка) принадлежит украинским программистам.
2. Из 10-ти ведущих аутсорсинговых IT-компаний Восточной и Центральной Европы 7 полностью украинские или имеют офисы в Украине.
3. Украина впервые появилась в Топ-12 стран-разработчиков игр в Европе.
4. **Украина в 2011 году вошла в пятерку мировых лидеров по объемам экспорта программной продукции**, уступая Индии, Китаю и России, но опережая 5-ю в этом рейтинге Бразилию



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Острый дефицит IT-специалистов:

Исходя из прогнозных расчетов темпов роста украинской IT-индустрии (35–40% в год, что наблюдается уже с 2011 года), **к 2015 году появится более 150-ти тысяч новых IT-вакансий**, из них **не менее 100 тыс.** в секторе разработки программного обеспечения на экспорт.

**Дефицит IT-специалистов при этом составит не менее 90 тыс. человек**

Фактически это означает, что выпуск украинскими вузами специалистов в области информационно-компьютерных технологий должен возрасти многократно за довольно небольшой промежуток времени.







Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### ИТ-кластер в Донецке – это необходимо для всего индустриального Донбасса!!!

Роль государства здесь трудно переоценить и она должна заключаться прежде всего в создании условий для **формирования и развития так называемых образовательных ИТ-кластеров** в тех городах, где есть **вузы с достаточно масштабным и высоким уровнем подготовки в области ИТ-специальностей** уже преподаются на высоком уровне.

«К числу этих городов традиционно относятся Киев, Харьков, Львов, а в последнее время – и **Донецк**»



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### ИТ и образование – главные условия инновационного развития Украины и Донбасса

В случае целенаправленных и согласованных усилий государства, бизнеса и образования **у Украины есть реальный шанс после 2015 года стать лидером в сфере информационных технологий** и достичь в последующие годы показателей ИТ-экспорта на уровне до 10 млрд. долл. в год, т.е. **увеличить высокотехнологичный экспорт практически на порядок** по сравнению с тем, что мы имеем сегодня.

Украина по потенциалу своего среднего и высшего образования **входит в десятку мировых лидеров**, а при надлежащей поддержке своего главного национального богатства – системы образования, ориентированной на чрезвычайно талантливый и изобретательный народ – **в состоянии войти к 2015 году в пятерку ведущих мировых образовательных лидеров**

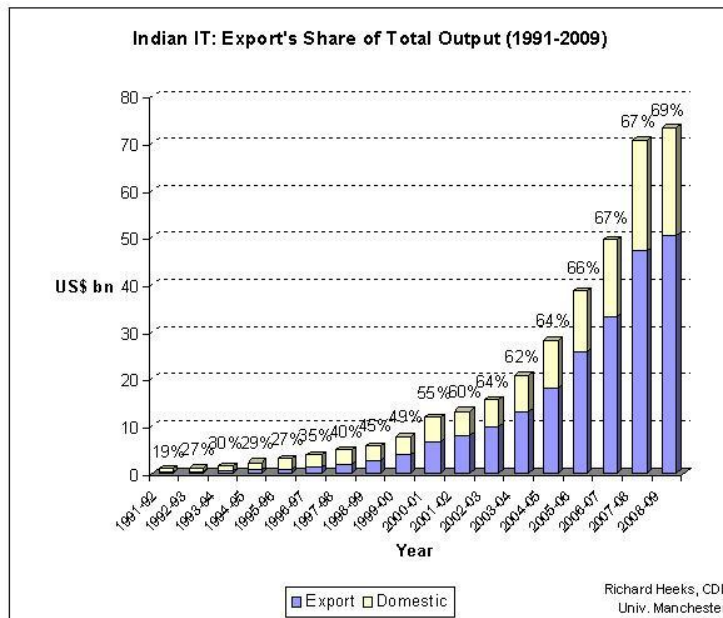




Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Феномен роста индустрии IT в Индии



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине



Вальтер Фольмер  
создатель инженерно-технического  
центра "Сименс-Украина":

**Донецк мог бы стать  
украинским Бангалором!!!**





Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Форум «Индустрия информационных технологий 2012»

В Донецке в СВЦ Эксподонбасс 18-21 сентября 2012 года впервые в рамках промышленной недели пройдет широкомасштабный форум **«Индустрия информационных технологий 2012»**

Есть основания надеяться, что это событие станет знаковым в развитии IT-индустрии в Донецке и Украине.



*для заметок*



ДонНТУ

# ПОСТБИНАРНЫЙ СОПРОЦЕССОР

ПАРАЛЛЕЛЬНЫЙ/КОНВЕЕРНЫЙ???

Аспирант кафедры КИ  
Иваница С.В.

17 / 02 / 2012

2

## Основные задачи

1. Форматы ( + постбинарная составляющая)
2. Структура постбинарного вычислительного устройства
3. Регистры сопроцессора
4. Система команд
5. Обработка особых случаев
6. Программирование команд



## Форматы данных

32 бита	64 бита	128 бит	256 бит	Вид данных	Пост-бинарный?
pb32	pb64	pb128	pb256	число	-
pb32/16p	pb64/32p	pb128/64p	pb256/128p		+
-	pb64/32f	pb128/64f	pb256/128f	дробь	-
-	-	pb128/32fp	pb256/64fp		+
-	pb64/32i	pb128/64i	pb256/128i	интервал	-
-	-	pb128/32ip	pb256/64ip		+

## Форматы данных

N	CF	MF	Формат	N	CF	MF	Формат
32	0	0	pb32	128	011	00000	pb128
		1	pb32/16p			00001	pb128/64f
64	01	00	pb64			00010	pb128/64i
		01	pb64/32f			00011	pb128/64p
		10	pb64/32i			00100	pb128/32fp
		11	pb64/32p			00101	pb128/32ip
256	0111	0000000000000000	pb256			0000000000000001	pb256/128f
		0000000000000001	pb256/128f			0000000000000010	pb256/128i
		0000000000000010	pb256/128i			0000000000000011	pb256/128p
		0000000000000011	pb256/128p			0000000000000100	pb256/64fp
		0000000000000100	pb256/64fp			0000000000000101	pb256/64ip
		0000000000000101	pb256/64ip				

## Форматы данных

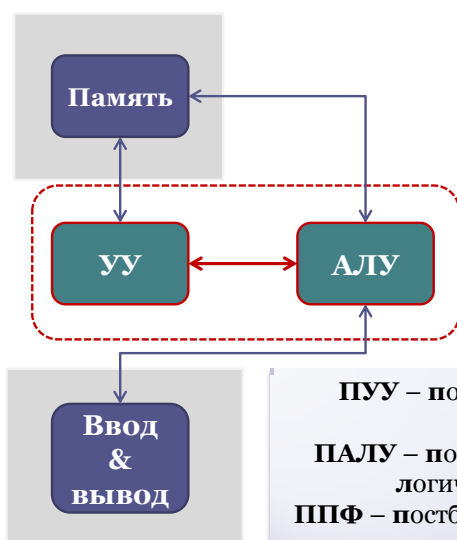


**S** – знак; **E** – порядок; **M** – мантисса;  
**MF** – модификатор формата; **CF** – код формата

<b>n+1</b>	<b>S + E + M</b>	<b>CF + MF</b>
32	30	2 (1 + 1)
64	60	4 (2 + 2)
128	120	8 (3 + 5)
256	240	16 (4 + 12)

## Общая структура

### Фон-неймановская



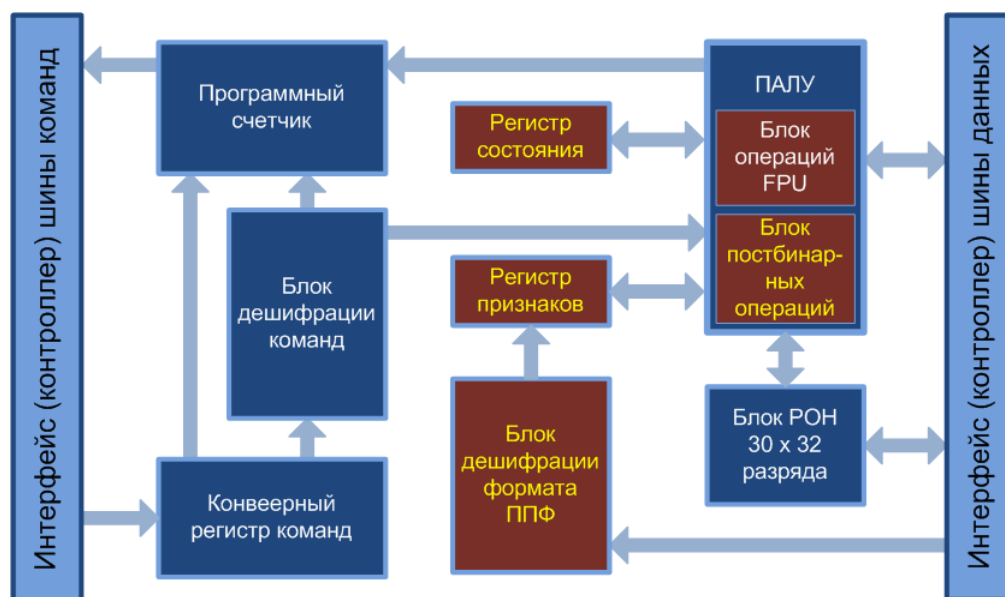
### Постбинарная



**ПУУ** – постбинарное устройство управления;  
**ПАЛУ** – постбинарное арифметико-логическое устройство;  
**ППФ** – постбинарный преобразователь формата

# Архитектура ядра

## Условная!!



# П о с т б и н а р н ы й преобразователь формата

**Назначение:** Передача данных в направлении  
РОН(ПАЛУ) <-> Память по адресу <А>

**Формат шины данных:** 16/8 бит

## Режимы работы:

1. Анализ кода формата CF
2. Разборка (сборка) формата данных с учетом кода формата CF
3. Фиксация модификации данных MF
4. Передача разобранного формата в РОН(ПАЛУ)
5. Передача собранного формата в память





## Первоочередные задачи

1. Оптимальный алгоритм формирования модификатора формата и смещения при выборке операнда
2. Согласование обмена данными 30-разрядного РОНа и 32-разрядного РОПа
3. Блок согласования форматов (!!!!)
4. Организация РОН в виде стека

*для заметок*



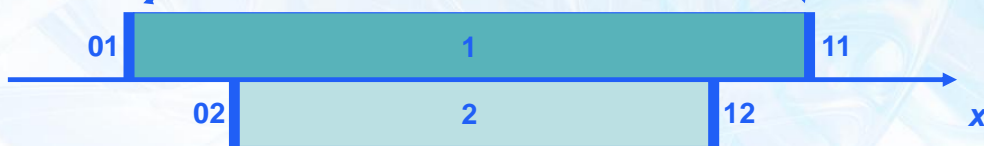


## Интервальное представление тетрадовых значений

$$T \in \{0, 1, M, A\} \quad x \in Z$$

$$T_{01} \in \{0, 1, M \rightarrow 0, A \rightarrow 0\}$$

$$T_{11} \in \{0, 1, M \rightarrow 1, A \rightarrow 1\}$$



$$T_{02} \in \{0, 1, M \rightarrow 0, A \rightarrow 1\}$$

$$T_{12} \in \{0, 1, M \rightarrow 1, A \rightarrow 0\}$$

$$[a_1; a_2] + [b_1; b_2] = [a_1 + b_1; a_2 + b_2]$$

3

## Алгоритм выполнения операции сложения для двух операндов

**Шаг 1:** Преобразование тетрадовых операндов к интервальному типу данных с формированием соответствующих границ интервала:

$$\min = \{0, 1, M \leftarrow 0, A \leftarrow 1\}, \max = \{0, 1, M \leftarrow 1, A \leftarrow 0\};$$

**Шаг 2:** Получение границ результирующего интервала, используя формулу сложения интервалов

$$[a_1; a_2] + [b_1; b_2] = [a_1 + b_1; a_2 + b_2]$$

**Шаг 3:** Путем побитового сопоставляя min и max границ результирующего интервала, сформировать результирующий тетракод, придерживаясь следующих правил:

- если значение текущего бита не изменилось, бит переносится в результат:  $(0 \rightarrow 0) \rightarrow 0; (1 \rightarrow 1) \rightarrow 1$ .

- если значение текущего бита изменилось, то в текущем бите результата формируется значение множественности (M) или неопределенности (A):  $(0 \rightarrow 1) \rightarrow M; (1 \rightarrow 0) \rightarrow A$ .

4



## Сформированная таблица сложения

$0 + 0 = 0$	$1 + A = AM$
$0 + 1 = 1$	$1 + M = MA$
$0 + A = A$	$A + A = A0$
$0 + M = M$	$A + M = 1$
$1 + 1 = 10$	$M + M = M0$

5

## Ассоциативность сложения или переход к сложению трех операндов

$$\begin{aligned} &\underline{a + b + c} = \\ &(a + b) + c = \\ &a + (b + c) \end{aligned}$$

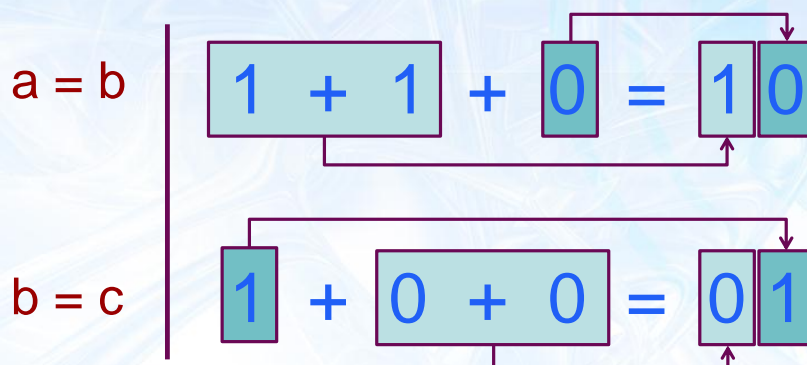
- $1. 1+1+A = \begin{cases} (1+1)+A = 10+A=1A \\ 1+(A+1) = 1+AM=1A \end{cases}$
- $2. 1+1+M = \begin{cases} (1+1)+M = 10+M=1M \\ 1+(1+M) = 1+MA=1M \end{cases}$
- $3. A+M+1 = \begin{cases} (A+M)+1 = 1+1=10 \\ A+(M+1) = A+MA=10 \end{cases}$
- $4. A+M+M = \begin{cases} (A+M)+M = 1+M=MA \\ A+(M+M) = A+M0=MA \end{cases}$
- $5. A+A+M = \begin{cases} (A+A)+M = A0+M=AM \\ A+(A+M) = A+1=AM \end{cases}$

6



## Сложение трех двоичных операндов

$$\underline{a + b + c}, \quad a, b, c \in \{0, 1\}$$



7

## Сложение трех операндов-тетритов

$$\underline{a + b + c}, \quad a, b, c \in \{0, 1, A, M\}$$

- |                     |                     |                      |
|---------------------|---------------------|----------------------|
| 1. $1 + 1 + 0 = 10$ | 5. $A + A + 0 = A0$ | 9. $M + M + 0 = M0$  |
| 2. $1 + 1 + 1 = 11$ | 6. $A + A + 1 = A1$ | 10. $M + M + 1 = M1$ |
| 3. $1 + 1 + A = 1A$ | 7. $A + A + A = AA$ | 11. $M + M + A = MA$ |
| 4. $1 + 1 + M = 1M$ | 8. $A + A + M = AM$ | 12. $M + M + M = MM$ |

$$13. 1 + A + M = 10$$

8

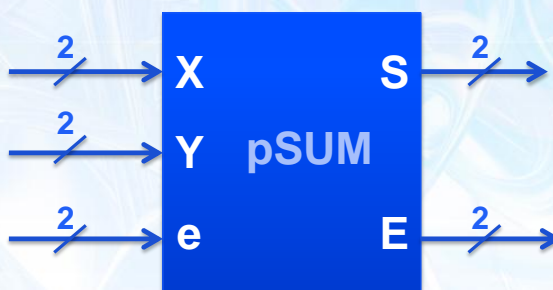
# ПОСТБИНАРНЫЙ ОДНОТЕТРИТНЫЙ СУММАТОР pSUM

## Условное обозначение и кодирование тетрита

Кодирование тетрита  $t$   
парой бит  $b[1:0]$

$b1$	$b0$	$t$
0	0	A
0	1	0
1	0	1
1	1	M

$$X + Y + e = (E)S$$



## Таблица истинности (начало)

№	X[1]	X[0]	Y[1]	Y[0]	e[1]	e[0]	E[1]	E[0]	S[1]	S[0]	Примечание
0	0	0	0	0	0	0	0	0	0	0	$A+A+A=AA$
1	0	0	0	0	0	1	0	0	0	1	$A+A+0=A0$
2	0	0	0	0	1	0	0	0	1	0	$A+A+1=A1$
3	0	0	0	0	1	1	0	0	1	1	$A+A+M=AM$
4	0	0	0	1	0	0	0	0	0	1	$A+0+A=A0$
5	0	0	0	1	0	1	0	1	0	0	$A+0+0=0A$
6	0	0	0	1	1	0	0	0	1	1	$A+0+1=AM$
7	0	0	0	1	1	1	0	1	1	0	$A+0+M=01$
8	0	0	1	0	0	0	0	0	1	0	$A+1+A=A1$
9	0	0	1	0	0	1	0	0	1	1	$A+1+0=AM$
10	0	0	1	0	1	0	1	0	0	0	$A+1+1=1A$
11	0	0	1	0	1	1	1	0	0	1	$A+1+M=10$
12	0	0	1	1	0	0	0	0	1	1	$A+M+A=AM$
13	0	0	1	1	0	1	0	1	1	0	$A+M+0=01$
14	0	0	1	1	1	0	1	0	0	1	$A+M+1=10$
15	0	0	1	1	1	1	1	1	0	0	$A+M+M=MA$
16	0	1	0	0	0	0	0	0	0	1	$0+A+A=A0$

...

11

## Таблица истинности (окончание)

...

№	X[1]	X[0]	Y[1]	Y[0]	e[1]	e[0]	E[1]	E[0]	S[1]	S[0]	Примечание
47	1	0	1	1	1	1	1	1	1	0	$1+M+M=M1$
48	1	1	0	0	0	0	0	0	1	1	$M+A+A=AM$
49	1	1	0	0	0	1	1	1	0	0	$M+A+0=MA$
50	1	1	0	0	1	0	1	0	0	1	$M+A+1=10$
51	1	1	0	0	1	1	1	1	0	0	$M+A+M=MA$
52	1	1	0	1	0	0	0	1	1	0	$M+0+A=01$
53	1	1	0	1	0	1	0	1	1	1	$M+0+0=0M$
54	1	1	0	1	1	0	1	1	0	0	$M+0+1=MA$
55	1	1	0	1	1	1	1	1	0	1	$M+0+M=M0$
56	1	1	1	0	0	0	1	0	0	1	$M+1+A=10$
57	1	1	1	0	0	1	1	1	0	0	$M+1+0=MA$
58	1	1	1	0	1	0	1	0	1	1	$M+1+1=1M$
59	1	1	1	0	1	1	1	1	1	0	$M+1+M=M1$
60	1	1	1	1	0	0	1	1	0	0	$M+M+A=MA$
61	1	1	1	1	0	1	1	1	0	1	$M+M+0=M0$
62	1	1	1	1	1	0	1	1	1	0	$M+M+1=M1$
63	1	1	1	1	1	1	1	1	1	1	$M+M+M=MM$

12



## Результирующие зависимости

$$E1 = x1 x0 \# y0 e0 + x1 y1 \# y0 + x1 y1 \# e0 + x1 x0 y1 + x1 e1 + y1 e1;$$

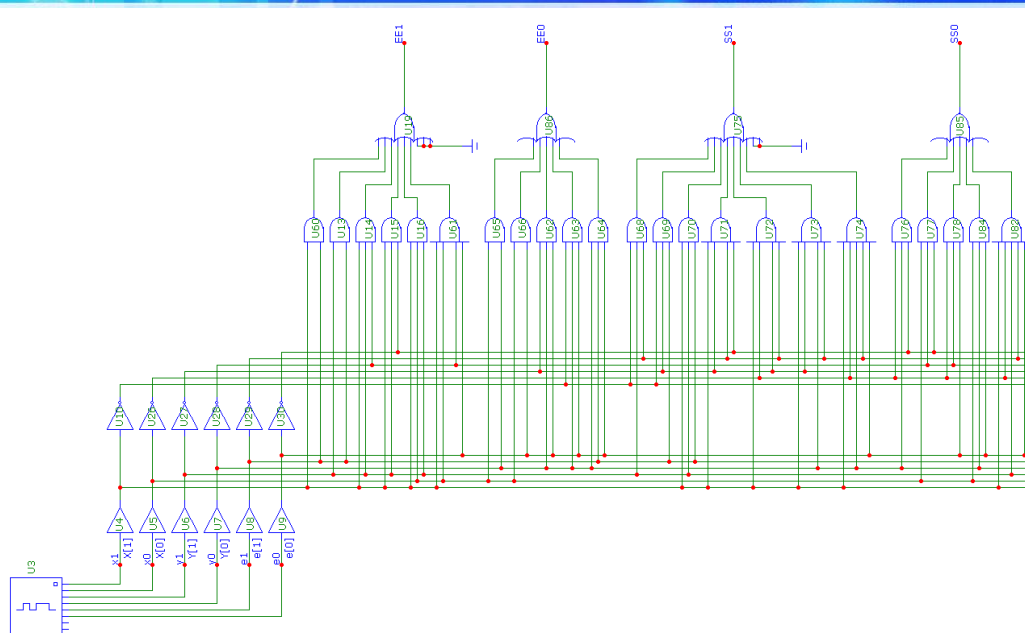
$$E0 = \# y1 y0 e0 + \# x1 y0 e0 + y0 e1 e0 + x0 y0 + x0 e0;$$

$$S1 = x1 \# x0 y0 \# e1 e0 + x1 \# y1 \# e1 \# e0 + x1 \# x0 \# y0 \# e1 + x1 \# y1 y0 \# e1 + \# x1 y1 \# e1 + \# x1 \# y1 e1 + x1 y1 e1;$$

$$S0 = x1 \# x0 y1 \# e1 e0 + \# x0 y0 \# e0 + x0 \# y0 \# e0 + \# x0 \# y0 e0 + x0 y0 e0.$$

13

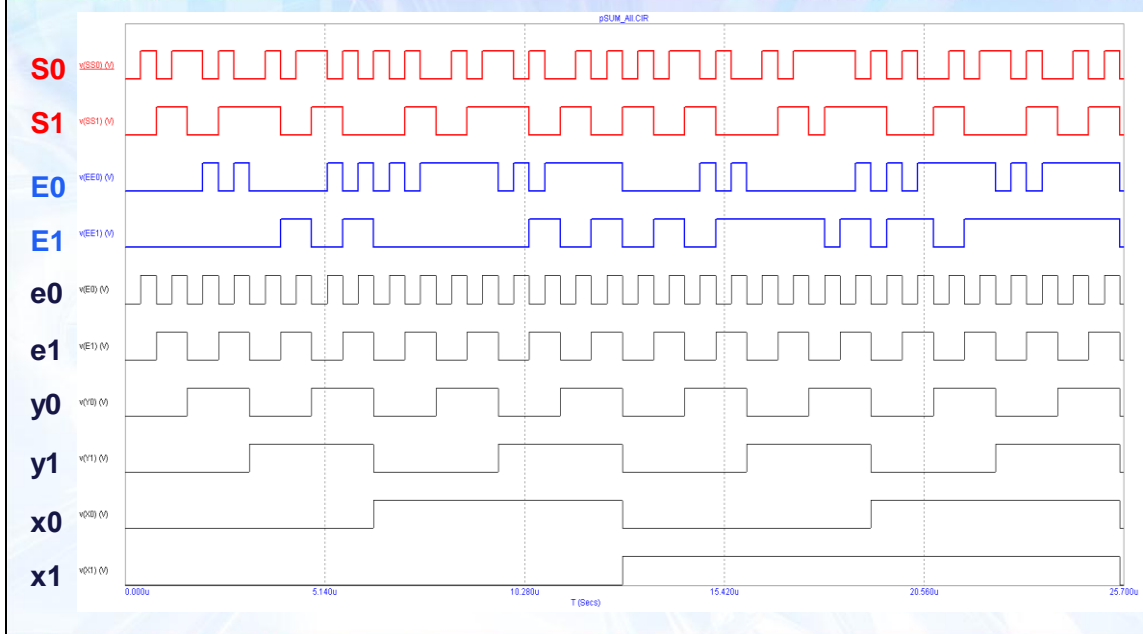
## Схема сумматора



14



## Результаты моделирования



15

## ЗАКЛЮЧЕНИЕ

## Выводы



1. Возможность **сложения тетракодов**, представляющих мантиссу и порядок постбинарного числа в формате с плавающей запятой;
2. Реализация **постбинарного однотетритного сумматора** с возможностью каскадирования до необходимой разрядности



1. Как сохранить **нормированность** суммы?!!
2. Открытая проблема: преобразование тетракода в **дополнительный код** (для реализации операции вычитания)

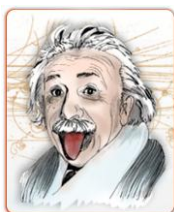
## Задачи для реализации алгоритма сложения чисел

Реализация алгоритма сложения чисел, представленных в постбинарном формате с плавающей запятой

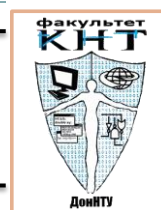
- |            |  |
|------------|--|
| <b>+</b>   | 1. Распаковка формата ( $e_{a,b,c}$ – порядок и $m_{a,b,c}$ – мантисса операндов <b>a</b> , <b>b</b> и суммы <b>c</b> ); |
| <b>+</b>   | 2. Обеспечение условия $e_a \geq e_b$ ;  |
| <b>+</b>   | 3. Проверка условия $e_a - e_b \rightarrow e_a \gg e_b$ ;  |
| <b>?</b>   | 4. Нормализация в случае выполнения условия 3;   |
| <b>?</b>   | 5. Масштабирование мантиссы в случае невыполнения условия 3;   |
| <b>++?</b> | 6. Сложение: $m_a + m_b = m_c$ ;   |
| <b>?</b>   | 7. Реакция на переполнение и антипереполнение;   |
| <b>?</b>   | 8. Округление результата   |

**ДонНТУ**

# **ПРОБЛЕМЫ КОМПЬЮТЕРНЫХ ВЫЧИСЛЕНИЙ, ВЫЗВАННЫЕ ИСПОЛЬЗОВАНИЕМ СТАНДАРТА IEEE 754**



Классификация ошибок  
(+ немного теории) с примерами  
получения «неожиданных» результатов

**04/2012**

Аспирант кафедры КИ  
**Иваница С.В.**

2

## **Актуальность??**

1. Стандарт IEEE 754 широко применяется в технике и программировании
2. Большинство современных микропроцессоров изготавливаются с аппаратной реализацией представления вещественных переменных в формате IEEE 754
3. Язык программирования и программист не могут изменить эту ситуацию, поскольку

**иного представления вещественного  
числа в микропроцессоре  
не существует !!!**





## Алгоритм преобразования

### Исходное число

0,123456789

- 1)** Преобразование исходной десятичной дроби к двоичной (часто бесконечной)
- 2)** Приведение полученного двоичного числа к нормализованному виду в двоичной системе
- 3)** Преобразование двоичного нормализованного числа в 32-битный формат стандарта IEEE 754:
  - формирование знака;
  - формирование смещенного порядка;
  - формирование остатка от мантииссы (усеченного сигнификанта)

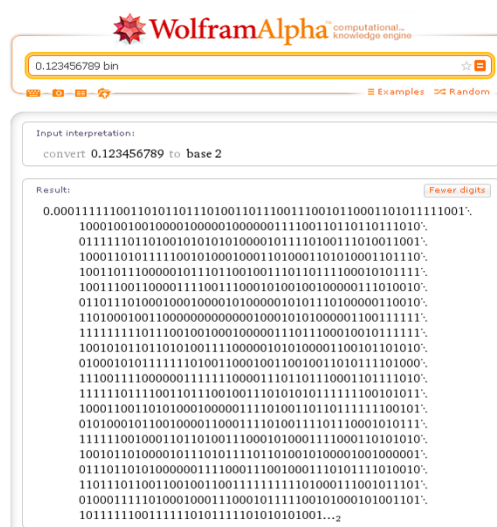
## Формирование двоичной дроби

<http://www.wolframalpha.com>

Для перевода числа **0,123456789**  
в бесконечную двоичную дробь  
воспользуемся сервисом  
**Wolfram Alpha!**

$$0,123456789_{10} =$$

= 0,00011111001101011011101001101110011100...,



## Нормализованный вид числа

$$\pm m \times p^{\pm \text{exp}}$$

$p$  – основание системы счисления

$1 \leq m < p$  – мантисса числа

$\text{exp}$  – порядок числа

Исходное число:

0,00011111001101011011101001101110011100

Нормализованное число:

1,1111001101011011101001101110011100  $\times 2^{\text{exp}}$   
 $\text{exp} = -100$   
 $m = 1,1111001101011011101001101110011100$

## Упаковка в 32-битный формат IEEE

$m_2 = 1,1111001101011011101001101110011100...$

$\text{exp}_2 = -100$

### Первый этап: формирование знака

Число может быть «+» или «-».

Поэтому отводится 1 бит для обозначения знака числа:

0 – положительное; 1 – отрицательное.

Это самый старший бит в 32-битной последовательности.

На 1 этапе 32-битная последовательность имеет вид:



## Упаковка в 32-битный формат IEEE

$$m_2 = 1,1111001101011011101001101110011100...$$

$$\exp_2 = -100_2 = -4_{10}$$

### Второй этап: формирование смещенного порядка

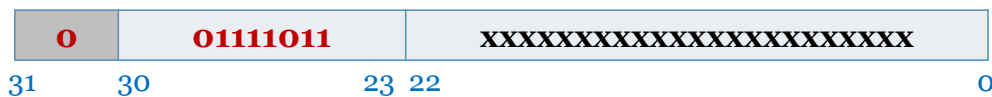
Экспонента может быть, как и число, со знаком + или -. Для определения знака экспоненты, чтобы не вводить еще один бит знака, добавляют смещение к экспоненте в половину байта

$$+127_{10} (0111\ 1111_2)$$

Смещенная экспонента равна

$$-4 + 127 = 123 = 1111011_2$$

На 2 этапе 32-битная последовательность имеет вид:



## Упаковка в 32-битный формат IEEE

$$m_2 = 1,1111001101011011101001101110011100...$$

$$\exp_2 = -100_2 = -4_{10}$$

### Третий этап: формирование остатка от мантиссы

У нормализованной двоичной мантиссы первый бит всегда равен 1, так как число лежит в диапазоне  $1 \leq m < 2$ . Поэтому в отведенные 23 бита записывают остаток от мантиссы.

В нашем случае остаток от мантиссы равен

$$1111001101011011101001101110011100...$$

(ушла единица перед запятой).

## Упаковка в 32-битный формат IEEE

$$m_2 = 1,1111001101011011101001101110011100...$$

$$\text{exp}_2 = -100_2 = -4_{10}$$

### Четвертый этап: округление мантииссы

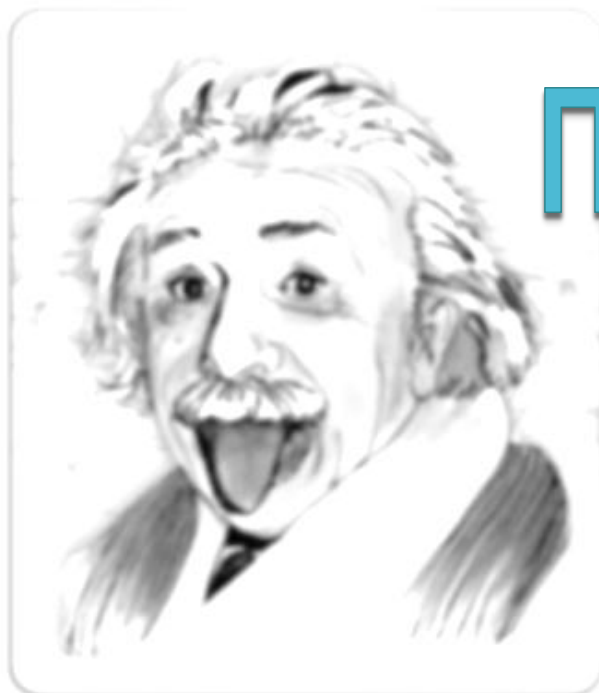
В 23 битное поле числа остаток от мантииссы  
записывается с округлением  
(механизм округления по умолчанию – к ближайшему целому):

$$11111001101011011101001 | 101110011100... \approx$$

$$11111001101011011101001 + 1 \approx$$

$$\approx 11111001101011011101010$$

На **заключительном этапе** 32-битная последовательность имеет вид:



**Подводные  
камни  
IEEE 754**



## Классификация ошибок

1. **Опасная редукция**

Ошибки, связанные с точностью представления вещественных чисел в формате IEEE754

2. **Дикие ошибки**

Ошибки, связанные с неправильным приведением типов данных

3. **Циклические дыры**

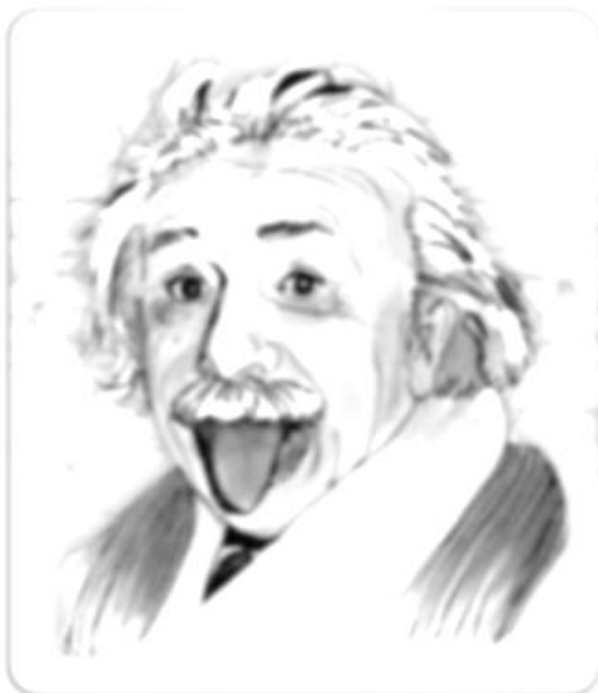
Ошибки, вызванные сдвигом мантисс

4. **Грязный ноль**

Ошибки вызванные округлением

5. **Числа убийцы**

Ошибки, возникающие на границе нормализованных/денормализованных чисел



## Опасная редукция

**Данная ошибка  
всегда присутствует  
в компьютерных  
вычислениях!!!**

## Опасная редукция (пример)

```

New_1.Program
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace New_1
{
    class Program
    {
        static void Main(string[] args)
        {
            float a, b, c;

            a = 123456789;
            b = 123456788;

            c = a - b;
        }
    }
}
    
```

Ожидаемый (правильный)  
результат:

$$123456789 - 123456788 = 1$$

Имя	Значение	Тип
args	{string[0]}	string[]
a	123456792.0	float
b	123456784.0	float
c	8.0	float

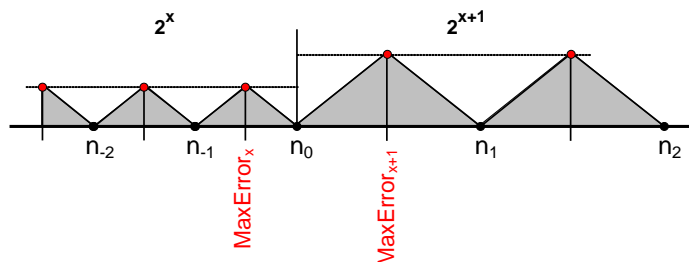
Абсолютная ошибка: **+7**

**Почему результат оказался неправильным???**

## Точность представления IEEE 754



Числа в формате IEEE754 представляют конечное множество,  
на которое отображается бесконечное множество  
вещественных чисел



Шаг чисел равен  
величине наименьшего  
разряда:

$$2^{(E-m-off)}$$

## Точность представления IEEE 754

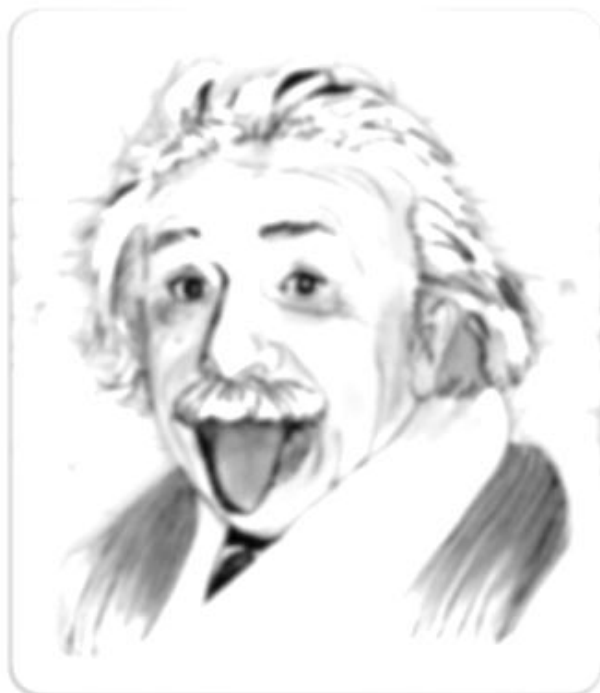
Числа с плавающей запятой обеспечивают относительную точность:

$m+1$ ,  $m$  двоичных чисел;

$\lg(2^{m+1})$ ,  $\lg(2^m)$  десятичных чисел!!

Относительная точность десятичного числа для **single**:  
**23-24** двоичных и **7-8** десятичных цифр

Относительная точность десятичного числа для **double**:  
**51-52** двоичных и **15-16** десятичных цифр



## Дикие ОШИБКИ

Эти ошибки вызваны  
тем, что исходное  
число представленное  
в форматах single и  
double уже не является  
одним и тем же  
числом!!!

## Дикие ошибки (пример)

```
namespace New_1
{
    class Program
    {
        static void Main(string[] args)
        {
            float a, b, c;

            a = 1.0f;
            b = 3.0f;
            c = a / b;

            float d1 = (c - 1 / 3) * 1.0e9f;
            float d2 = (c - 1f / 3f) * 1.0e9f;
            double d3 = Convert.ToDouble((c - 1.0 / 3.0) * 1.0e9);
        }
    }
}
```

Переменные и  
промежуточные  
результаты  
компьютерных  
вычислений должны  
быть приведены к  
одному типу данных

Имя	Значение	Тип
args	{string[0]}	string[]
a	1.0	float
b	3.0	float
c	0.333333343	float
d1	333333344.0	float
d2	0.0	float
d3	9.9341074810688212	double

## Дикие ошибки (продолжение)

```
namespace New_1
{
    class Program
    {
        static void Main(string[] args)
        {
            float a, b, c;

            a = 1;
            b = 3;
            c = a / b;
            c = c - 1 / 3;
        }
    }
}
```

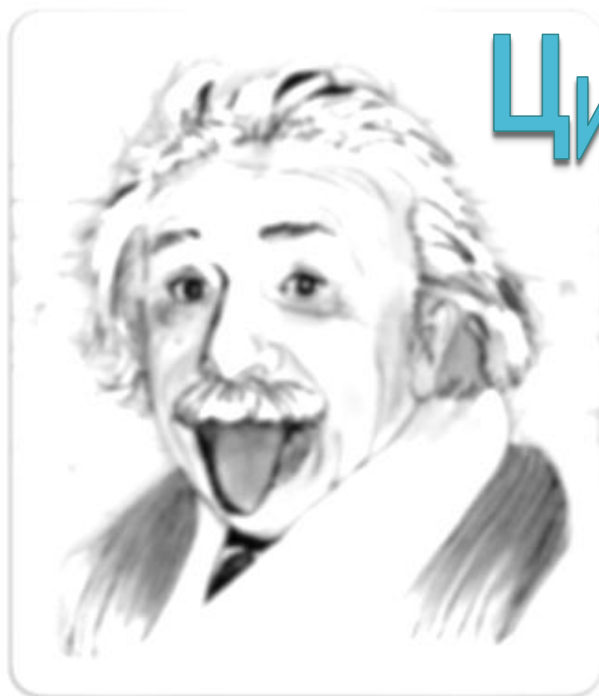
Ожидаемый результат: **0**

Имя	Значение	Тип
args	{string[0]}	string[]
a	1.0	float
b	3.0	float
c	0.333333343	float

Промежуточный результат  $1/3$  в строке  
 $c = c - 1/3$  будет иметь тип double

Ошибки вычислений вызванные **неверным приведением типов данных** (или его отсутствия) – «дикие», так как они связаны с незнанием стандартов и теории программирования (т.е. с плохим фундаментальным образованием)





# Циклические дыры

Эти ошибки связаны с потерей точности результата при неполном пересечении мантисс чисел на числовой оси!!!

## Пример циклической дыры в АСУ

```
namespace New_1
{
    class Program
    {
        static void Main(string[] args)
        {
            float a; // вес таблетки в кг
            float c; // продукции в бункере в кг
            long n = 10000000; // количество циклов

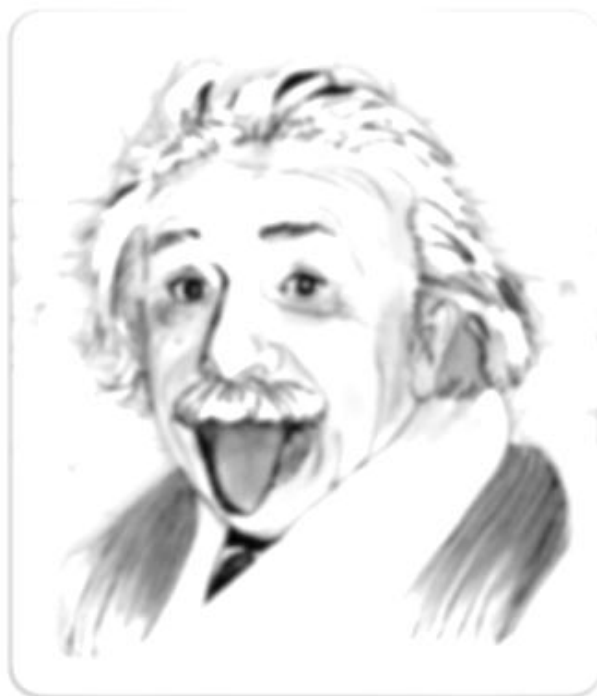
            c = 300; // исходный вес бункера
            a = 0.00001f; // вес таблетки

            // одна таблетка забирается фасовочной машиной
            for (long i = 1; i < n; i++) c = c - a;
        }
    }
}
```

В результате нам крупно повезло, что это был фармацевтический цех, а не Запорожская АЭС!!

Если числа отличаются более чем в  $2^{23}$  (для **single**) и  $2^{52}$  (для **double**), то операции сложения и вычитания между этими числами **невозможны!**

Локальные		
Имя	Значение	Тип
args	{string[0]}	string[]
a	0.00001	float
c	300.0	float
n	10000000	long



# Грязный ноль

Эти ошибки вызваны  
спецификой  
округления чисел при  
представлении их в  
формате с плавающей  
запятой!!!

## Грязный ноль

**Грязный ноль** – ситуация, когда пользователь или программа считают, что переменная не равная нулю – равна нулю!

```
namespace WindowsForm1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button_Click(object sender, EventArgs e)
        {
            double a, b, c;

            a = Convert.ToDouble(textBox_op1.Text);

            b = 0.123456789012346 + 1E-16;

            textBox_op2.Text = b.ToString();

            c = a - b;
            textBox_res.Text = c.ToString();
        }
    }
}
```

Очень часто эта ошибка  
возникает  
в интерфейсе  
«оператор-машина»

Исходная переменная и ее отображение в окне Windows это разные числа.  
Это не ошибка формата IEEE 754, это ошибка Windows.



# Числа УБИЙЦЫ

Эти ошибки  
возникают при работе  
с числами,  
находящимися на  
границе  
нормализованного/  
денормализованного  
представления  
чисел!!!

## Нормализация VS денормализация

MAX\_DENORM

2,225073858507200889 E-308

The screenshot shows the 'Converting floating point numbers' application. It displays the IEEE 754 representation for two extreme values. For MAX\_DENORM (2,225073858507200889 E-308), the SINGLE Precision float (32bit) result is 0, and the DOUBLE Precision (double 64bit) result is 2,2250738585072E-308. For MIN\_NORM (2,225073858507201383 E-308), the SINGLE Precision float (32bit) result is 0, and the DOUBLE Precision (double 64bit) result is 2,2250738585072E-308. Red arrows point from the text 'MAX\_DENORM' and 'MIN\_NORM' to their respective input fields. A red arrow points from the DOUBLE Precision result of MIN\_NORM to a red equals sign (=) below it.

## Пример: баг №53632 для РНР

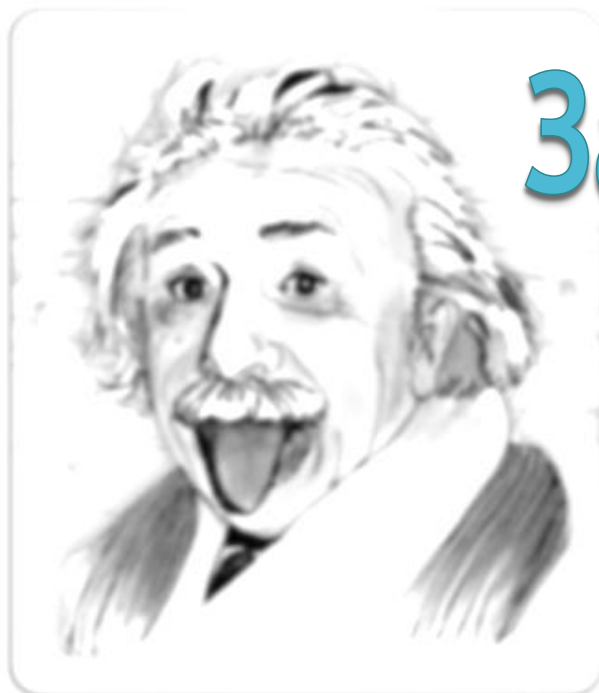
<http://bugs.php.net/53632>

```
< html>  
< body>  
< ?php $d = 2.2250738585072011e-308; ?>  
...  
< /body>  
< /html>
```

Сообщение  
о баге поступило  
**30.12.2010**,  
исправлено  
разработчиком  
**10.01.2011**.

Ввод числа **2.2250738585072011e-308** вызывал зависание  
процесса почти со 100% загрузкой CPU

Так как РНР препроцессор используют большинство серверов, то у  
любого пользователя сети **в течение 10 дней** была возможность  
«вырубить» практически любой сервер.



## Заключение

**Мнение о том, что при  
вычислениях с  
плавающей точкой  
результат не выходит за  
пределы относительной  
погрешности  
представления  
наибольшего числа  
является  
ОШИБОЧНЫМ!!**



## Выводы

1. Вышеперечисленные ошибки складываются друг с другом в процессе вычислений
2. Такие ошибки как грязный ноль и опасная редукция могут сделать погрешность вычислений неприемлемой
3. Особое внимание при программировании компьютерных вычислений программист должен обратить на близкие к нулю результаты

Некоторые специалисты считают, что формат чисел IEEE 754 представляет угрозу человечеству:

**IEEE754-тика угрожает человечеству**

(Юровицкий В.М. МФТИ, РГСУ, Москва)

<http://www.yur.ru/science/computer/IEEE754.htm>

## Инновационные идеи

- **АПРОКСИМЕТИКА**  
(**Юровицкий В.М.**, Россия)
- **ПОСТБИНАРНЫЕ ФОРМАТЫ**  
(**Аноприенко А.Я.**, Украина)
- **???????????**

Если вы знаете другие инновационные идеи в области представления вещественных чисел – **сообщите!!**

ДонНТУ

# ПОСТБИНАРНОЕ ОКРУГЛЕНИЕ

К вопросу представления вещественных чисел в  
постбинарных форматах

3/05/2012

Аспирант кафедры КИ: **Иваница С.В.**

2

## Девиз постбинарных вычислений

*В математической практике нет ничего более хлопотного, что более всего досаждало бы и мешало вычислителям, чем перемножение, деление или извлечение квадратных и кубических корней больших чисел. Выполнение этих операций **не только приводит к значительным потерям времени, но и сопряжено с такой массой скрытых ошибок**, что я начал размышлять о поисках надежного и удобного средства устранения подобных помех*

**Джон Непер (1616 г.)**

## Об округлении...

*Округленные числа всегда лгут*

**Сэмюэль Джонсон (1750 г.)**

*Я буду говорить в округленных числах,  
не абсолютно точно, но не настолько далеко  
от истины, чтобы изменить  
реальный результат*

**Томас Джефферсон (1824 г.)**

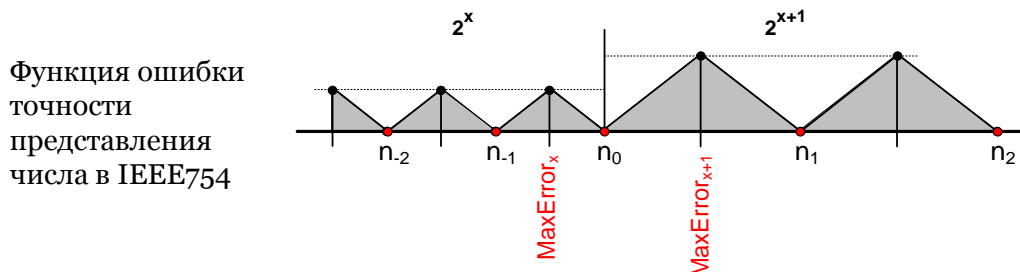
## Постановка задачи

1. Виды округления чисел с плавающей запятой
2. Оценка погрешности при округлении чисел с плавающей запятой IEEE 754
3. Расчет относительной погрешности при округлении исходного значения к ближайшему числу
4. **Применение стандартных округлений к постбинарным форматам**
5. **Постбинарное округление**
6. **Расчет шага округления**
7. Выводы

## Принцип кодирования чисел с плавающей запятой



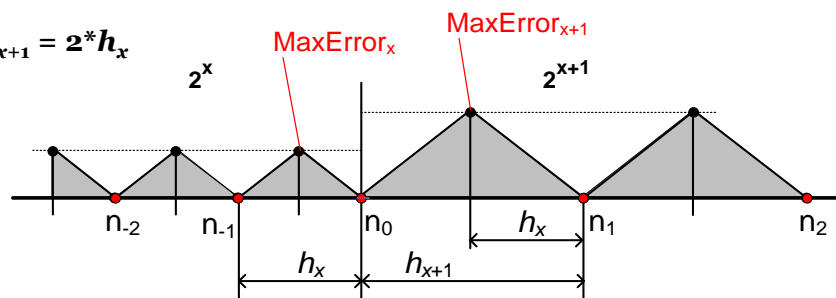
Числа представленные в формате IEEE754 представляют конечное множество, на которое отображается бесконечное множество вещественных чисел



## Максимальная абсолютная ошибка для чисел в формате IEEE754

$h$  – шаг чисел:  $h_{x+1} = 2 * h_x$

$$h = 2^{Ex_{10} - off - m}$$



**Абсолютная максимальная ошибка  $\Delta$**  для числа в формате IEEE754 равна в пределе половине шага чисел:

$$\Delta = \frac{1}{2} h = \frac{1}{2} \left( 2^{Ex_{10} - off - m} \right) = 2^{Ex_{10} - off - m - 1}$$

$Ex_{10}$  – десятичное значение поля порядка;  
**off** – смещение порядка;  
**m** – количество разрядов поля мантиисы



## Максимальная относительная ошибка для чисел в формате IEEE754

**Относительная максимальная ошибка  $\delta$**  для числа в формате IEEE754:

$$\delta = \frac{h}{|F|} \cdot 100\%$$

$F$  – десятичное число, полученное из числа IEEE754 соответствующей точности;  
 $M$  – десятичное число, полученное из поля мантиссы.

$$|F_d| = 2^{(1-off)} \cdot \frac{M}{2^m} \quad \text{– для денормализованного числа;}$$

$$|F_n| = 2^{(Ex_{10}-off)} \cdot \left(1 + \frac{M}{2^m}\right) \quad \text{– для нормализованного числа.}$$

$$\delta_d = \frac{1}{2 \cdot M} \cdot 100\%$$

$$\delta_n = \frac{1}{2^m + M} \cdot 100\%$$

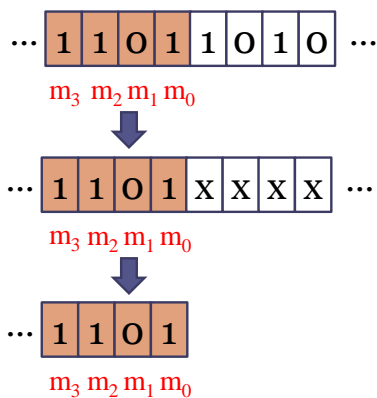
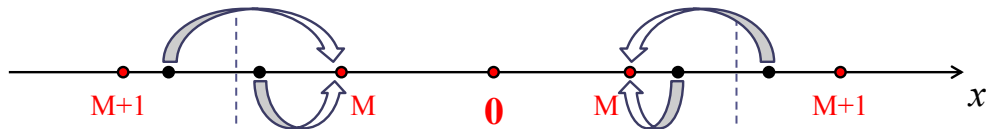
## Виды округления чисел IEEE754

Стандарт IEEE754 предусматривает четыре способа округления чисел:

1. Стремящееся к ближайшему числу  
(к ближайшей базовой точке)
2. Стремящееся к нулю
3. Стремящееся к  $+\infty$
4. Стремящееся к  $-\infty$

## Округление к нулю

**Округление к нулю** – округление к **всегда меньшей** базовой точке:

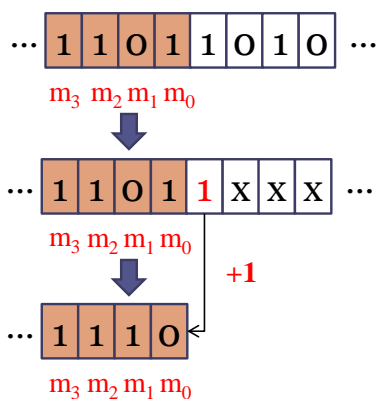
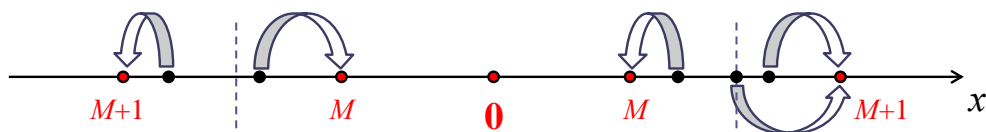


Округление к нулю используют в различных устройствах.

При округлении к нулю **нужно просто отбросить незначащие разряды числа**, поэтому этот способ самый легкий в аппаратной реализации.

## Округление к ближайшему числу

**Округление к ближайшему числу** – округление к **ближайшей** базовой точке:

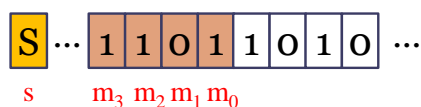
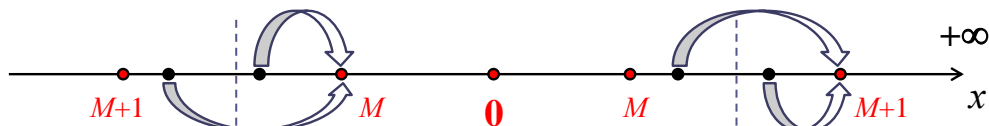


Округление к ближайшему числу в сопроцессоре используется по умолчанию.

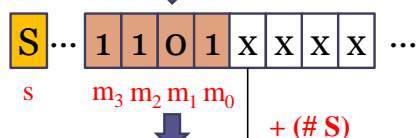
При округлении к ближайшему числу **нужно к мантиссе прибавить значение первого незначащего разряда числа**.

## Округление к $+\infty$

**Округление к  $+\infty$**  – округление к базовой точке, находящейся ближе к  $+\infty$ :

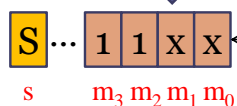


Округление к  $+\infty$  применяется при кодировании интервальных чисел.



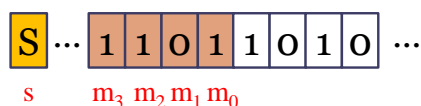
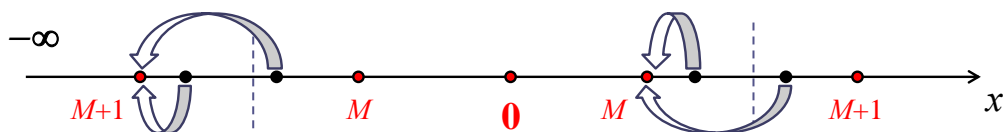
При округлении к  $+\infty$  :

- если число  $>0$  – **нужно к мантиссе всегда прибавлять 1;**
- если число  $<0$  – **нужно просто отбросить незначимые разряды.**

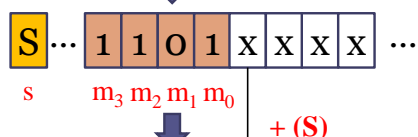


## Округление к $-\infty$

**Округление к  $-\infty$**  – округление к базовой точке, находящейся ближе к  $-\infty$ :

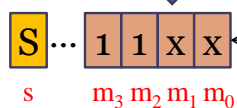


Округление к  $-\infty$  также применяется при кодировании интервальных чисел.



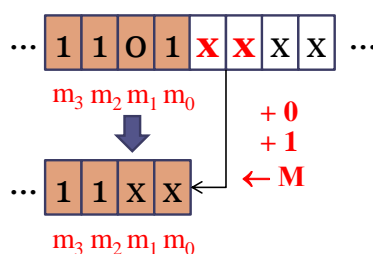
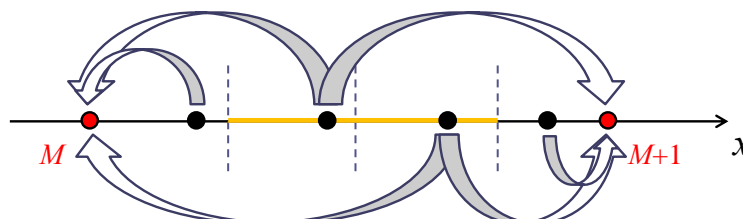
При округлении к  $+\infty$  :

- если число  $>0$  – **нужно просто отбросить незначимые разряды;**
- если число  $<0$  – **нужно к мантиссе всегда прибавлять 1.**



## Постбинарное округление

**Постбинарное округление** – округление, оперирующее значением  $\frac{1}{4}$  шага между базовыми точками:



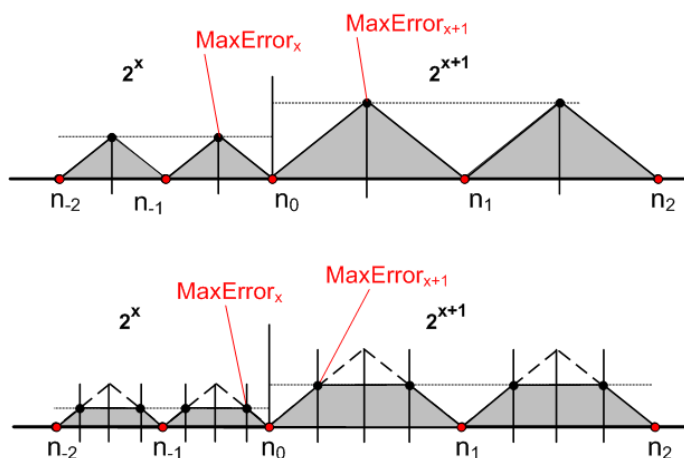
При **постбинарном округлении**

рассматриваются два ближайших незначащих разряда:

- **00** – округление к **меньшей базовой точке** (т.е. к ближайшему числу);
- **01** или **10** – занесение в младший разряд мантиисы **значения «множественности»**;
- **11** – округление к **большей базовой точке** (т.е. к ближайшему числу).

## Оценка погрешностей при постбинарном округлении

**Величина ошибки округления уменьшилась в два раза!!**



$$p\Delta = \frac{1}{4} h = 2^{Ex_{10} - off - m - 2}$$

$$p\delta = \frac{h}{2 \cdot |F|} \cdot 100\%$$

$$p\delta_n = \frac{1}{2 \cdot (2^m + M)} \cdot 100\%$$

$$p\delta_d = \frac{1}{4 \cdot M} \cdot 100\%$$

## Расстояние $h$ между соседними базовыми точками

$h_{\min}$  – расстояние между минимальным денормализованным числом и нулем :

$F1 = 0 \ 00..000 \ 00000..000$

$F2 = 0 \ 00..000 \ 00000..001$

$$h_d = 2^{1-off-m}$$

Формат	$h_{\min} = F2 - F1$ (с точностью 30 значащих цифр)
pb16	$5,96046447753906250 \times 10^{-8}$
single	$1,40129846432481707092372958329 \times 10^{-45}$
pb32	$5,60519385729926828369491833316 \times 10^{-45}$
double	$4,94065645841246544176568792868 \times 10^{-324}$
pb64	$7,90505033345994470682510068589 \times 10^{-323}$
pb128	$1,65764483057613442839665637331 \times 10^{-4963}$
pb256	$1,82862336051135490395060395989 \times 10^{-157892}$

Полученные значения неизменны для всей области денормализованных чисел!!

## Расстояние $h$ между соседними базовыми точками

$h_{\max}$  – расстояние между максимальным нормализованным и предшествующим его числом :

$F1 = 0 \ 111..110 \ 1111..110$

$F2 = 0 \ 111..110 \ 1111..111$

$$h_n = 2^{Ex_{10}-off-m}$$

Формат	$h_{\max} = F2 - F1$ (с точностью 30 значащих цифр)
pb16	32,0
single	$2,02824096036516704239472512860 \times 10^{+31}$
pb32	$8,11296384146066816957890051441 \times 10^{+31}$
double	$1,99584030953471981165637271304 \times 10^{+292}$
pb64	$3,19334449525555169865019634086 \times 10^{+293}$
pb128	$2,93291457624204329451374018399 \times 10^{+4900}$
pb256	$1,54087578874600654289200380744 \times 10^{+157760}$



## Выводы

1. При стандартном округлении происходят значительные потери значимости в числах в области максимального представления
2. Округление к нулю «противопоказано» к применению в постбинарных форматах!!
3. Интервальное округление границ к  $\pm\infty$  приводит к неоправданному увеличению ширины интервала
4. Постбинарное округление способно уменьшить относительную погрешность представления числа в 2 раза!
5. Наличие «множественности» в младшем разряде мантиссы можно свести к интервалу гарантированно содержащему исходное (реальное) значение

## для заметок



ДонНТУ

# ПОГРЕШНОСТЬ ПРЕДСТАВЛЕНИЯ ЧИСЕЛ В ПОСТБИНАРНЫХ ФОРМАТАХ

## Гистограммы

относительной погрешности представления чисел  
в постбинарных форматах  
при различных способах округления

10/08/2012

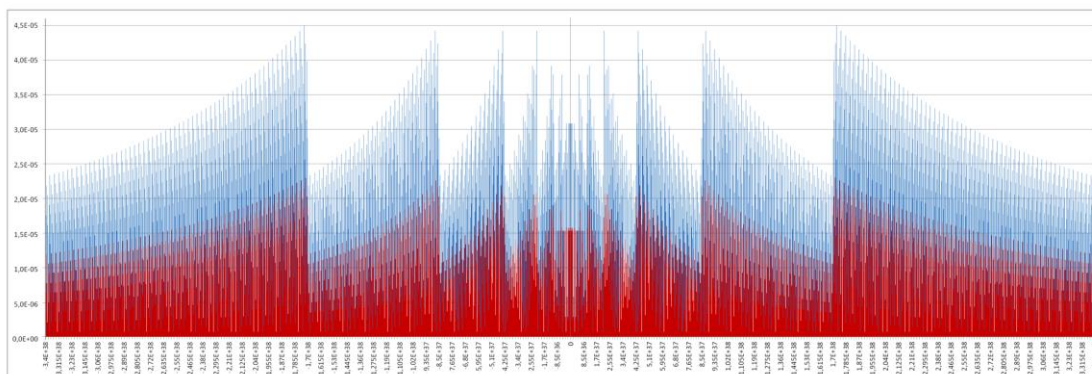
аспирант кафедры КИ **Иваница С.В.**  
магистрант кафедры КИ **Котов Е.И.**

2

## В качестве предисловия...

**Округленные числа всегда лгут!**  
Но насколько?  
И можно ли доверять таким числам?

Ответ на эти вопросы лежит в оценке  
относительной ошибки представления  
чисел в форматах с плавающей запятой!!!



Такой «**кровавый закат**» получился при вычислении погрешности на всей области определения чисел одинарной точности (rb32)  $\pm 3,4e+38$  с шагом  $1,7e+35$  для двух способов округления (8000 измерений!!!)

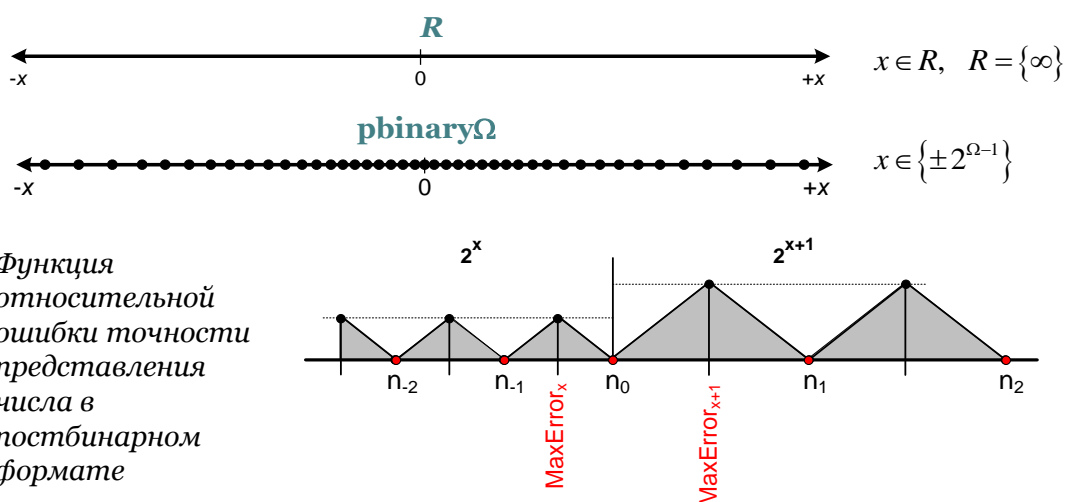
При этом значение относительной погрешности не превышает **0,000045 %** !!!!

## Постановка задачи

1. Погрешность представления чисел с плавающей запятой
2. Оценка погрешности при округлении чисел с плавающей запятой
3. Расчет максимальной относительной погрешности
4. Максимально возможные ошибки округления чисел в постбинарных форматах
5. Погрешности представления между соседними базовыми точками
6. Погрешности представления в различных диапазонах чисел
7. Выводы

## Принцип кодирования чисел с плавающей запятой

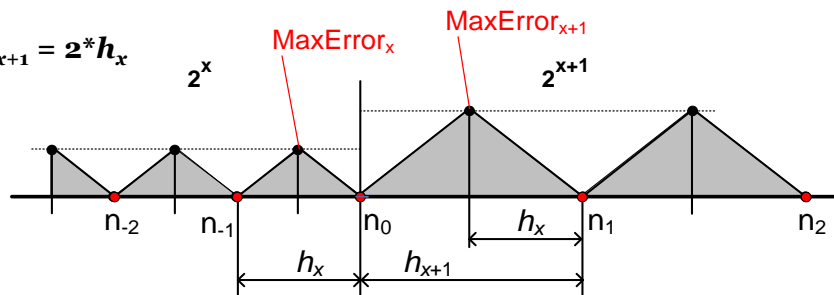
*Числа, представленные в постбинарном формате, являются конечным множеством, на которое отображается бесконечное множество вещественных чисел*



## Максимальная абсолютная ошибка для чисел в постбинарном формате

$h$  – шаг чисел:  $h_{x+1} = 2 \cdot h_x$

$$h = 2^{Ex_{10}-off-m}$$



**Абсолютная максимальная ошибка  $\Delta$**

равна в пределе половине шага чисел:

$$\Delta = \frac{1}{2} h = \frac{1}{2} (2^{Ex_{10}-off-m}) = 2^{Ex_{10}-off-m-1}$$

$Ex_{10}$  – десятичное значение поля порядка;

**off** – смещение порядка;

**m** – количество разрядов поля мантиссы

## Максимальная относительная ошибка для чисел в постбинарных форматах

**Относительная максимальная ошибка  $\delta$ :**

$$\delta = \frac{\Delta}{|F|} \cdot 100\%$$

$F$  – десятичное число, полученное из постбинарного формата соответствующей точности;

$M$  – десятичное число, полученное из поля мантиссы.

$$|F_d| = 2^{(1-off)} \cdot \frac{M}{2^m} \quad \text{– для денормализованных чисел;}$$

$$|F_n| = 2^{(Ex_{10}-off)} \cdot \left(1 + \frac{M}{2^m}\right) \quad \text{– для нормализованных чисел.}$$

$$\delta_d = \frac{1}{2 \cdot M} \cdot 100\%$$

$$\delta_n = \frac{1}{2^{m+1} + 2M} \cdot 100\%$$

## Сравниваем постбинарные и бинарные форматы

Величины относительной ошибки числа  $\delta'$ , представленного в бинарном формате и относительной ошибки числа  $\delta$ , представленного в аналогичном ему постбинарном формате (т. е. форматы с равными порядками и разрядностью) связаны следующим соотношением:

$$\delta = 2^{id} \cdot \delta'$$

где  $id$  — разрядность идентификатора формата `rbinary`, являющаяся фактически разницей между разрядностями мантисс форматов `binary` и `rbinary`.

За счет «укороченной» мантиссы формат `rbinary` имеет абсолютную ошибку представления числа, **большую в  $2^{id}$  раза**, чем у формата `binary` аналогичной точности.

**Т.е. числа в постбинарном формате представляются менее точно в  $2^{id}$  раза!!!**

## Максимально возможные ошибки представления чисел

Значения максимальной возможной ошибки представления для чисел форматов `binary32` (светлая строка) и `rbinary32` (темная строка)

Поле формата, 16 с/с	Число, 10 с/с	Абсолютная ошибка $\Delta$ , 10 с/с	Относительная, ошибка $\delta$ , %
0000 0004	$2^{-147} \approx 5,605194 \times 10^{-45}$	$2^{-150} \approx 7,006492 \times 10^{-46}$	12,5
		$2^{-148} \approx 2,802597 \times 10^{-45}$	50
0080 0004	$\approx 1,175495 \times 10^{-38}$	$2^{-150} \approx 7,006492 \times 10^{-46}$	$\approx 5,96046 \times 10^{-6}$
		$2^{-148} \approx 7,006492 \times 10^{-46}$	$\approx 2,38418 \times 10^{-5}$
1C84 C24C	$\approx 8,785252 \times 10^{-22}$	$2^{-94} \approx 5,04871 \times 10^{-29}$	$\approx 5,7468 \times 10^{-6}$
		$2^{-92} \approx 2,019484 \times 10^{-28}$	$\approx 2,29872 \times 10^{-5}$
3F7F FFFC	$\approx 0,999998$	$2^{-25} \approx 2,980232 \times 10^{-8}$	$\approx 2,98024 \times 10^{-6}$
		$2^{-23} \approx 1,192093 \times 10^{-7}$	$\approx 1,1921 \times 10^{-5}$
42FA F0A4	$\approx 125,47$	$2^{-18} \approx 3,814697 \times 10^{-6}$	$\approx 3,04033 \times 10^{-6}$
		$2^{-16} \approx 1,525879 \times 10^{-5}$	$\approx 1,21316 \times 10^{-5}$
7F7F FFFC	$\approx 3,402823 \times 10^{+38}$	$2^{+103} \approx 1,01412 \times 10^{+31}$	$\approx 2,98023 \times 10^{-6}$
		$2^{+105} \approx 4,05648 \times 10^{+31}$	$\approx 1,19209 \times 10^{-5}$



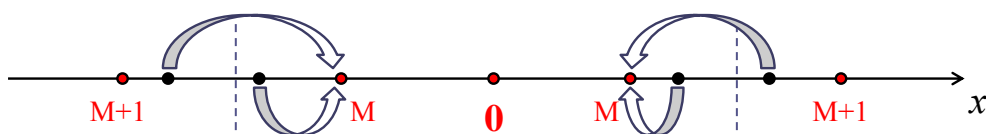
## Максимально возможные ошибки представления чисел

Значения максимальной возможной ошибки представления для чисел форматов **binary64** (светлая строка) и **rbinary64** (темная строка)

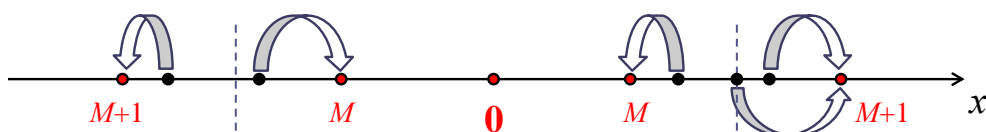
Поле формата*, 16 с/с	Число, 10 с/с	Абсолютная ошибка $\Delta$ , 10 с/с	Относительная, ошибка $\delta$ , %
0000 0000 0000 0010	$2^{-1070} \approx 7,9050503334599 \times 10^{-323}$	$2^{-1075} \approx 2,470328229 \times 10^{-324}$	3,125
		$2^{-1071} \approx 3,952525167 \times 10^{-323}$	50
0010 0000 0000 0010	$\approx 2,22507385850721 \times 10^{-308}$	$2^{-1075}$	$\approx 1,11022 \times 10^{-14}$
		$2^{-1071}$	$\approx 1,77636 \times 10^{-13}$
3FEF FFFF FFFF DCD0	$\approx 0,99999999999999$	$2^{-54} \approx 5,551115123 \times 10^{-17}$	$\approx 5,55112 \times 10^{-15}$
		$2^{-50} \approx 8,881784197 \times 10^{-16}$	$\approx 8,88178 \times 10^{-14}$
7FE0 0000 0000 0010	$\approx 8,98846567431161 \times 10^{+307}$	$2^{+970} \approx 9,979201548 \times 10^{+291}$	$\approx 1,110223 \times 10^{-14}$
		$2^{+974} \approx 1,596672248 \times 10^{+293}$	$\approx 1,776357 \times 10^{-13}$
7FEF FFFF FFFF FFF0	$\approx 1,79769313486231 \times 10^{+308}$	$2^{+970}$	$\approx 5,551115 \times 10^{-15}$
		$2^{+974}$	$\approx 8,881784 \times 10^{-14}$

## Рассматриваемые способы округления

**Округление к нулю** – округление к всегда меньшей базовой точке:



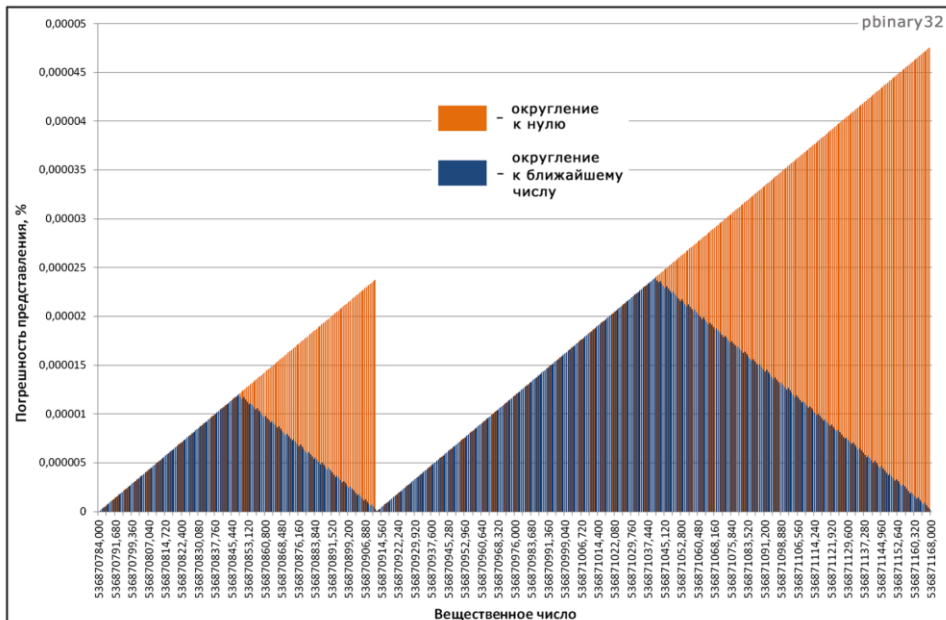
**Округление к ближайшему числу** – округление к ближайшей базовой точке:



11

## Погрешность на переходе увеличения порядка

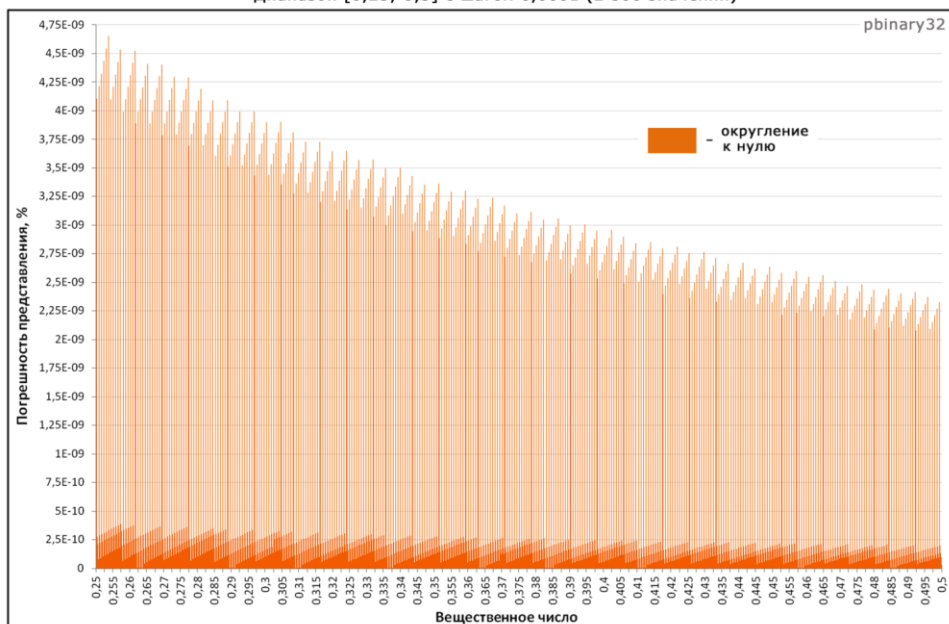
Диапазон [536870784; 536871168] с шагом 0,768 (500 значений)



12

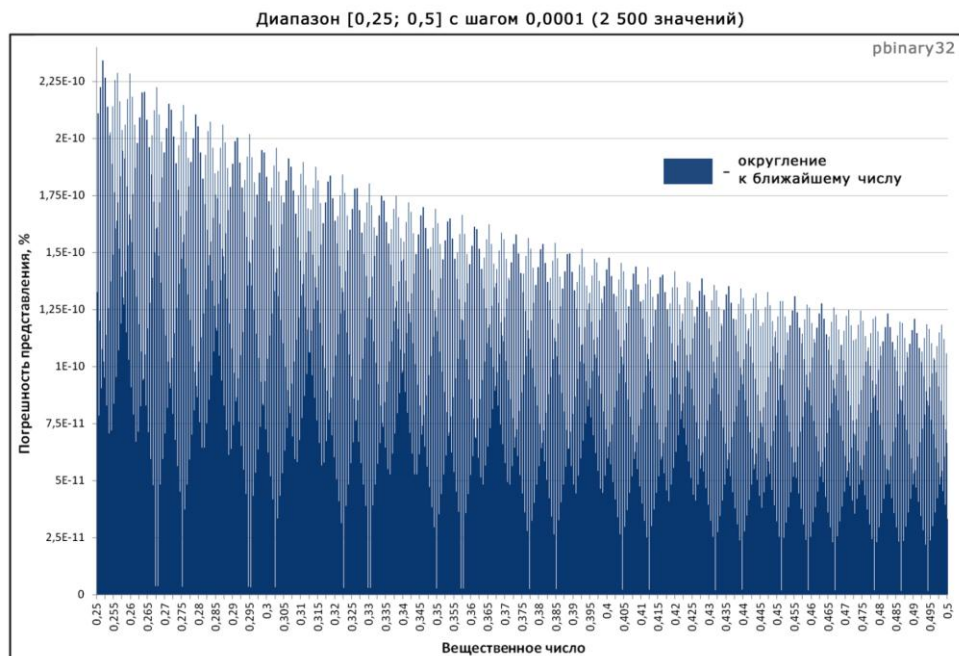
## Сужаем диапазон!

Диапазон [0,25; 0,5] с шагом 0,0001 (2 500 значений)



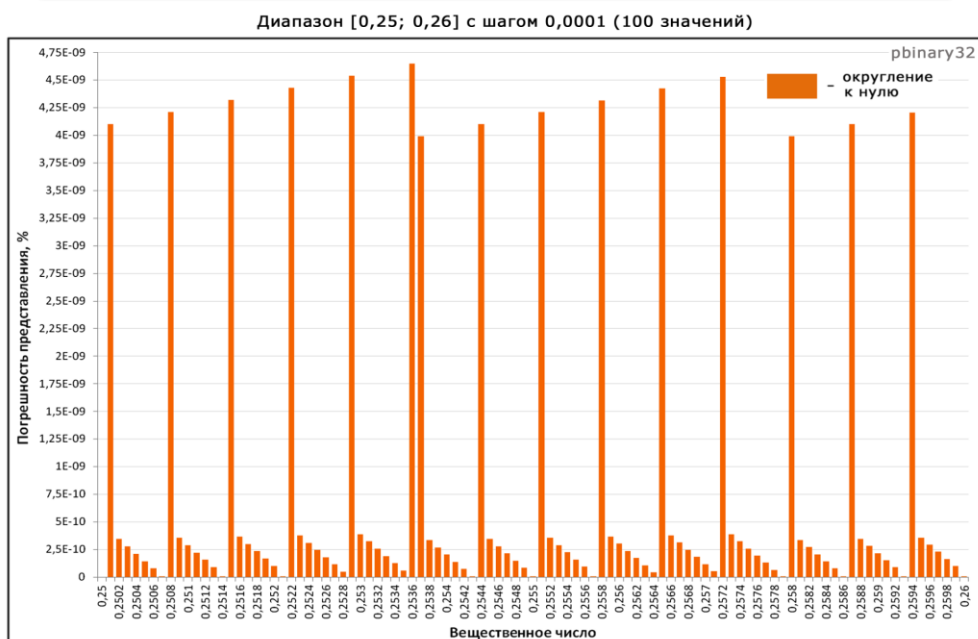
В диапазоне [0,25; 0,5] находится **2 097 152** базовых точек!!!

## Сужаем диапазон!



В диапазоне [0.25; 0.5] находится **2 097 152** базовых точек!!!

## Еще уже!

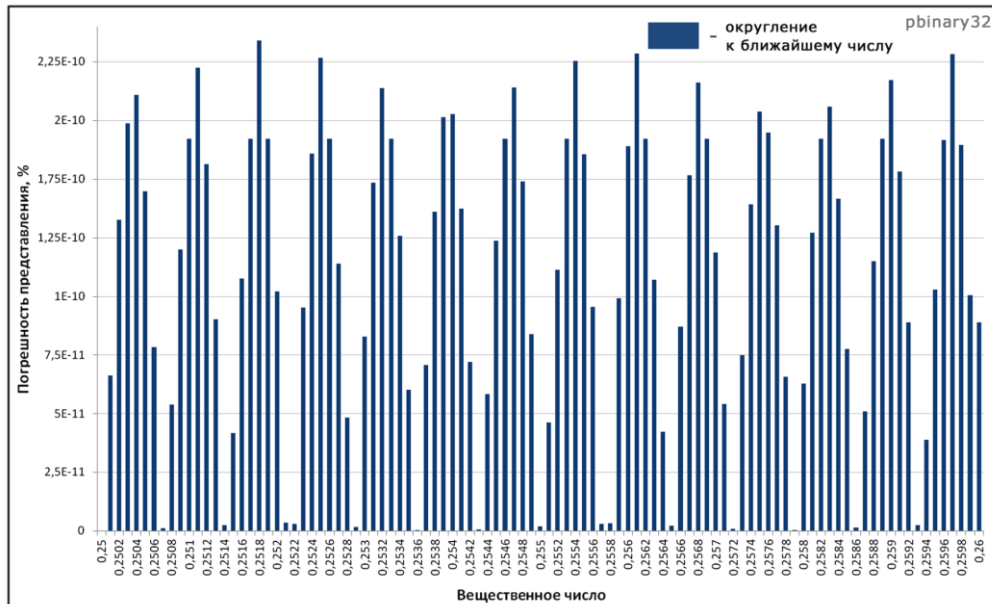


В диапазоне [0.25; 0.26] находится **41 943** базовых точек!!!

15

## Еще уже!

Диапазон [0,25; 0,26] с шагом 0,0001 (100 значений)

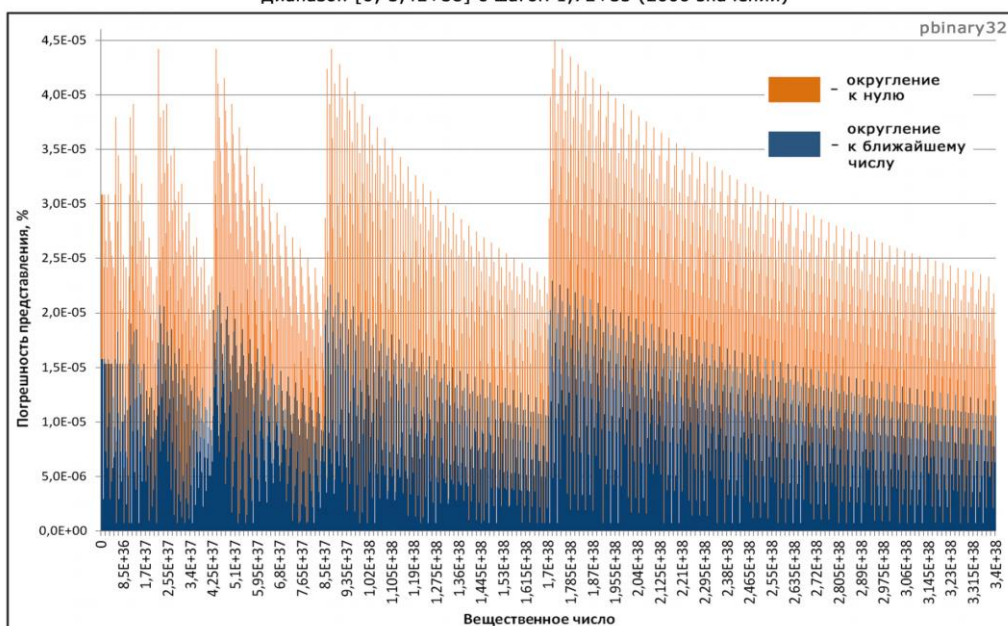


В диапазоне [0.25; 0.26] находится **41 943** базовых точек!!!

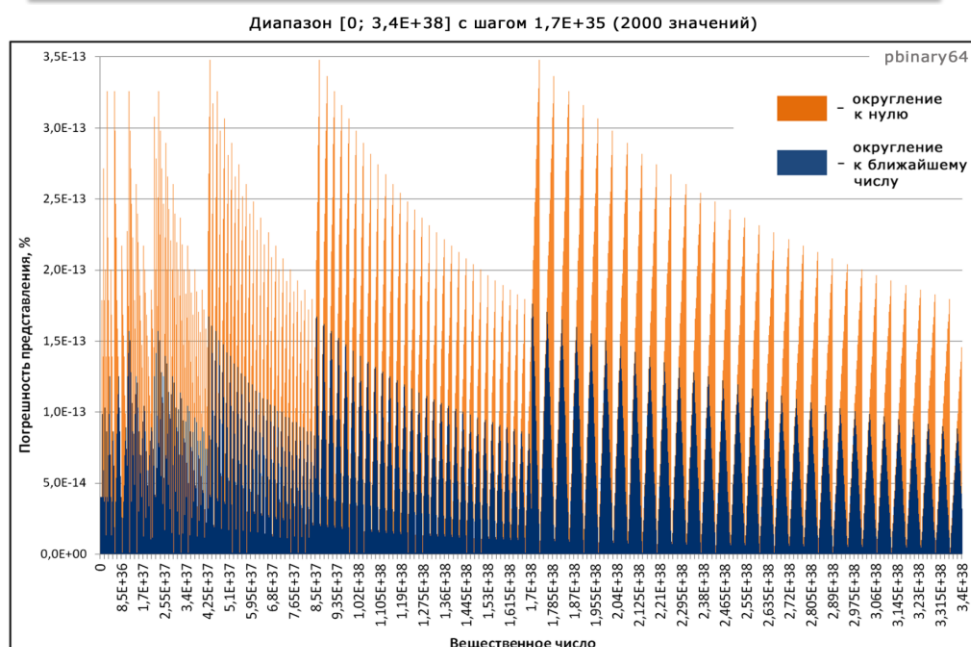
16

## Весь диапазон 32-разрядного формата (положительная полуось)

Диапазон [0; 3,4E+38] с шагом 1,7E+35 (2000 значений)



## Предыдущий диапазон, но для 64-разрядного формата



## Выводы

1. При стандартном округлении происходят потери значимости в числах в области максимального удаления от базовых точек
2. Округление к нулю «противопоказано» к применению в постбинарных форматах (как и в форматах IEEE754)!!
3. Потеря точности (увеличение абсолютной погрешности) в постбинарных форматах компенсируется следующими возможностями:
  - Динамическое наращивание разрядности (наличие служебной информации в «теле» формата)
  - Применение гибкого (постбинарного) округления чисел (расширенный кодовый базис)



ДонНТУ

# Операции с порядками постбинарных форматов чисел с плавающей запятой

К вопросу арифметики чисел с плавающей запятой,  
представленных в постбинарных форматах  
произвольной точности

20/08/2012

Аспирант кафедры КИ: **Иваница С.В.**

2

## Представление нормализованного числа

$$A = 10,25 \rightarrow +1010,01_2 = \underbrace{+1,01001}_{\substack{S \text{ Significatum} \\ Ch}} \times 2^{+3}$$

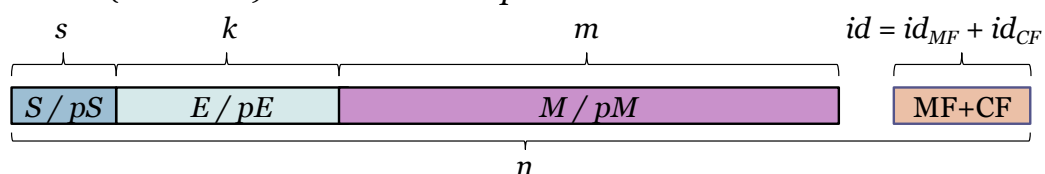
**S** (*sign*) – знак числа

**Significatum** – сигнификант (значащая часть)  
нормализованного числа

**p** (*exponent*) – порядок числа

**Ch** – (*characterisation*) основание характеристики

**M** (*mantissa*) – мантисса нормализованного числа



## Пример умножения чисел одинарной точности (single)

$$a = \pm M_1 \times 2^{p_1} \quad b = \pm M_2 \times 2^{p_2} \quad \text{IEEE754}$$

$$a \times b = \pm (M_1 \times M_2) \times 2^{p_1 + p_2}$$

1. Определить знак произведения
2. Умножить поля мантиссы
3. Сложить поля порядков

$$a = 1.25 \quad b = -2 \quad c = a \times b = -2.5$$

$a =$	<div>0</div>	<div>01111111</div>	<div>010000000000000000000000</div>	$= 1,25$
$\times$	$\oplus$	$+$	$\times$	
$b =$	<div>1</div>	<div>10000000</div>	<div>000000000000000000000000</div>	$= -2$
$c =$	<div>1</div>	<div>10000000</div>	<div>010000000000000000000000</div>	$= -2,5$

## Представление порядка в формате с отрицательным нулем

$$E \neq p$$

$p$  (exponent) – порядок числа

$E$  (biased exponent) – формат смещенного порядка с отрицательным нулем (ОН)

$$p = \underbrace{\pm .xxx \dots xx}_k \quad E = p + Q^{(-)}$$

$Q^{(-)}$  – смещение порядка с отрицательным нулем  $Q^{(-)} = 2^{k-1} - 1$

$$E = \begin{matrix} k \\ \text{0000} \dots \text{000} \end{matrix} \quad E = p + 2^{k-1} - 1 = 0 \Rightarrow p = 1 - 2^{k-1} = -Q^{(-)} \quad (1)$$

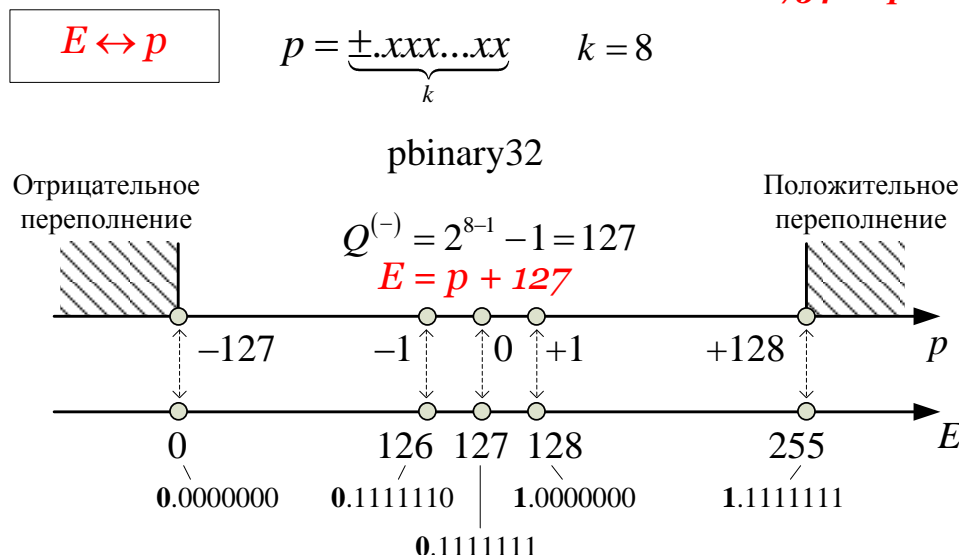
...

$$\begin{matrix} \text{1111} \dots \text{111} \end{matrix} \quad E = p + 2^{k-1} - 1 = 2^k - 1 \Rightarrow p = 2^k - 2^{k-1} = Q^{(-)} + 1 \quad (2)$$

5

## Представление порядка в формате с отрицательным нулем (ОН)

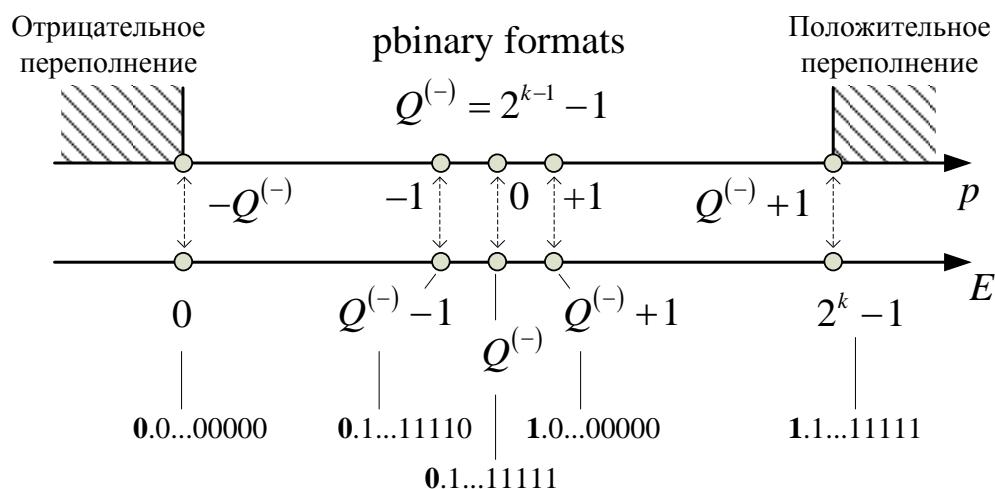
IEEE754 == pbinary



6

## Представление порядка в формате с отрицательным нулем (ОН)

Общая схема соответствия  $E$  и  $p$  для всех форматов



Ноль в знаковом разряде – признак отрицательного числа

## Сложение порядков в формате с ОН

$$\begin{aligned} (p_A + p_B)^{(-)} &= (p_A + p_B) + Q = (E_A - Q + E_B - Q) + Q = E_A + E_B - Q = \\ &= E_A + E_B - Q + 1 - 1 = (E_A + E_B + 1) - (Q + 1) \end{aligned} \quad (3)$$

**Для сложения порядков в формате с ОН необходимо:**

- Учитывать входной перенос равный 1;
- Проинвертировать «знаковый» разряд суммы (равносильно вычитанию  $(Q+1)$ );
- Зафиксировать наличие (или отсутствие) переполнения ПП: положительного (П+) или отрицательного (П-).

**Примеры для binary32 (IEEE754):**

$$p_A + p_B = (E_A + E_B + 1) - 128$$

$E_A$  – порядок операнда A;

$C_{-1}$  – входной перенос;

$E_B$  – порядок операнда B;

$C_7$  – старший «знаковый» разряд;

$E_C$  – порядок суммы;

$C_8$  – выходной перенос;

$k = 8$  – количество бит;

$Q = 127$  – смещение формата.

## Примеры сложения порядков

**Примеры для binary32 (IEEE754)**

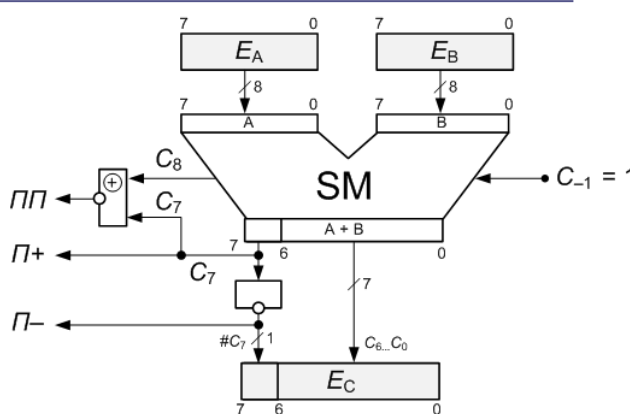
$p_A + p_B = 0$ $E_A + E_B = 127$	$-127 \leq p_A + p_B \leq 128$ $0 \leq E_A + E_B \leq 255$	$p_A + p_B > 128$ (П+) $E_A + E_B > 255$ (П+)	$p_A + p_B < -127$ (П-) $E_A + E_B < 0$ (П-)
$p_A = 6_{10}$ $p_B = -6_{10}$ $E_A = 133_{10} = 1000\ 0101_2$ $E_B = 121_{10} = 0111\ 1001_2$	$p_A = 50_{10}$ $p_B = -22_{10}$ $E_A = 177_{10} = 1011\ 0001_2$ $E_B = 105_{10} = 0110\ 1001_2$	$p_A = 120_{10}$ $p_B = 9_{10}$ $E_A = 247_{10} = 1111\ 0111_2$ $E_B = 136_{10} = 1000\ 1000_2$	$p_A = -80_{10}$ $p_B = -48_{10}$ $E_A = 47_{10} = 0010\ 1111_2$ $E_B = 79_{10} = 0100\ 1111_2$
$\begin{array}{r} 1000\ 0101 \rightarrow E_A \\ + 0111\ 1001 \rightarrow E_B \\ \hline 1 \rightarrow C_{-1} \end{array}$	$\begin{array}{r} 1011\ 0001 \rightarrow E_A \\ + 0110\ 1001 \rightarrow E_B \\ \hline 1 \rightarrow C_{-1} \end{array}$	$\begin{array}{r} 1111\ 0111 \rightarrow E_A \\ + 1000\ 1000 \rightarrow E_B \\ \hline 1 \rightarrow C_{-1} \end{array}$	$\begin{array}{r} 0010\ 1111 \rightarrow E_A \\ + 0100\ 1111 \rightarrow E_B \\ \hline 1 \rightarrow C_{-1} \end{array}$
$E_C = 0111\ 1111_2 = 127_{10}$ $p_C = 127_{10} - 127_{10} = 0$	$E_C = 1001\ 1011_2 = 155_{10}$ $p_C = 155_{10} - 127_{10} = 28_{10}$	$E_C = 0$ $p_C = 0 - 127_{10} \neq 129_{10}$	$E_C = 1111\ 1111_2 = 255_{10}$ $p_C = 128_{10} - 127_{10} \neq -128_{10}$

## Выводы по сложению порядков

$$III = \begin{cases} 0, & \text{if } C_7 \neq C_8 \\ 1, & \text{if } C_7 = C_8 \end{cases} \Rightarrow III = \#(C_7 \oplus C_8);$$

$$(\Pi+) = \begin{cases} 0, & \text{if } C_7 = 0 \\ 1, & \text{if } C_7 = 1 \end{cases} \Rightarrow (\Pi+) = C_7; \quad (\Pi-) = \begin{cases} 0, & \text{if } C_7 = 1 \\ 1, & \text{if } C_7 = 0 \end{cases} \Rightarrow (\Pi-) = \#C_7;$$

## Схема устройства, выполняющего сложение порядков для формата *single IEEE754*



## Вычитание порядков в формате с ОН

$$\begin{aligned} \left( p_A - p_B \right)^{(-)} &= \left( p_A - p_B \right) + Q = \left( E_A - Q - E_B - Q \right) + Q = E_A - E_B - Q = \\ &= E_A - E_B - Q + 1 - 1 = \left( E_A - E_B + 1 \right) - (Q + 1) = \left( E_A + (\#E_B) \right) - (Q + 1) \end{aligned} \quad (4)$$

**Для сложения порядков в формате с ОН необходимо:**

- Учитывать входной перенос равный нулю;
- Выполнить сложение порядка-вычитаемого с инверсией порядка-вычитателя;
- Проинвертировать «знаковый» разряд суммы (равносильно вычитанию  $(Q+1)$ );
- Обеспечить возможность проверки  $p_A = p_B$ ,  $p_A > p_B$ ,  $p_A < p_B$ ;
- Зафиксировать наличие (или отсутствие) переполнения **ПП**: положительного (**П+**) или отрицательного (**П-**).

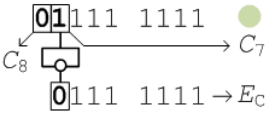
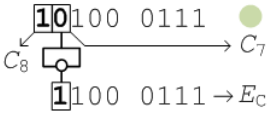
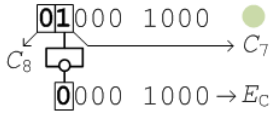
### Примеры для *binary32* (IEEE754):

$$p_A - p_B = (E_A + (\#E_B)) - 128$$



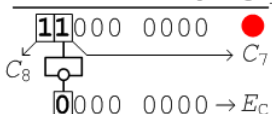
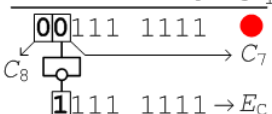
## Примеры вычитания порядков

Примеры для binary32 (IEEE754)

$p_A - p_B = 0$ $p_A = p_B$ $E_A - E_B = 127$ $p_A = 6_{10} \quad p_B = 6_{10}$ $E_A = 133_{10} = 1000\ 0101_2$ $E_B = 133_{10} = 1000\ 0101_2$	$-127 \leq p_A - p_B \leq 128$ $p_A > p_B$ $0 \leq E_A - E_B \leq 255$ $p_A = 50_{10} \quad p_B = -22_{10}$ $E_A = 177_{10} = 1011\ 0001_2$ $E_B = 105_{10} = 0110\ 1001_2$	$-127 \leq p_A - p_B \leq 128$ $p_A < p_B$ $0 \leq E_A - E_B \leq 255$ $p_A = -110_{10} \quad p_B = 9_{10}$ $E_A = 17_{10} = 0001\ 0001_2$ $E_B = 136_{10} = 1000\ 1000_2$
$\begin{array}{r} 1000\ 0101 \rightarrow E_A \\ + 0111\ 1010 \rightarrow \#E_B \\ \hline 0 \rightarrow C_{-1} \end{array}$ 	$\begin{array}{r} 1011\ 0001 \rightarrow E_A \\ + 1001\ 0110 \rightarrow \#E_B \\ \hline 0 \rightarrow C_{-1} \end{array}$ 	$\begin{array}{r} 0001\ 0001 \rightarrow E_A \\ + 0111\ 0111 \rightarrow \#E_B \\ \hline 0 \rightarrow C_{-1} \end{array}$ 
$E_C = 01111111_2 = 127_{10}$ $p_C = 127_{10} - 127_{10} = 0$	$E_C = 11001011_2 = 199_{10}$ $p_C = 199_{10} - 127_{10} = 72_{10}$	$E_C = 00001000_2 = 8_{10}$ $p_C = 8 - 127_{10} = -119_{10}$

## Примеры вычитания порядков

Примеры для binary32 (IEEE754)

$p_A - p_B > 128 \text{ (} \Pi+ \text{)}$ $p_A \gg p_B$ $E_A - E_B > 255 \text{ (} \Pi+ \text{)}$ $p_A = 120_{10} \quad p_B = -9_{10}$ $E_A = 247_{10} = 1111\ 0111_2$ $E_B = 118_{10} = 0111\ 0110_2$	$p_A - p_B < -127 \text{ (} \Pi- \text{)}$ $p_A \ll p_B$ $E_A - E_B < 0 \text{ (} \Pi- \text{)}$ $p_A = -103_{10} \quad p_B = 25_{10}$ $E_A = 24_{10} = 0001\ 1000_2$ $E_B = 152_{10} = 1001\ 1000_2$
$\begin{array}{r} 1111\ 0111 \rightarrow E_A \\ + 1000\ 1001 \rightarrow \#E_B \\ \hline 0 \rightarrow C_{-1} \end{array}$ 	$\begin{array}{r} 0001\ 1000 \rightarrow E_A \\ + 0110\ 0111 \rightarrow \#E_B \\ \hline 0 \rightarrow C_{-1} \end{array}$ 
$E_C = 0$ $p_C = 0 - 127_{10} \neq 129_{10}$	$E_C = 11111111_2 = 128_{10}$ $p_C = 128_{10} - 127_{10} \neq -128_{10}$

## Выводы по вычитанию порядков

$$\Pi\Pi = \begin{cases} 0, & \text{if } C_7 \neq C_8 \\ 1, & \text{if } C_7 = C_8 \end{cases} \Rightarrow \Pi\Pi = \#(C_7 \oplus C_8);$$

$$(\Pi+) = \begin{cases} 0, & \text{if } C_7 = 0 \\ 1, & \text{if } C_7 = 1 \end{cases} \Rightarrow (\Pi+) = C_7; \quad (\Pi-) = \begin{cases} 0, & \text{if } C_7 = 1 \\ 1, & \text{if } C_7 = 0 \end{cases} \Rightarrow (\Pi-) = \#C_7;$$

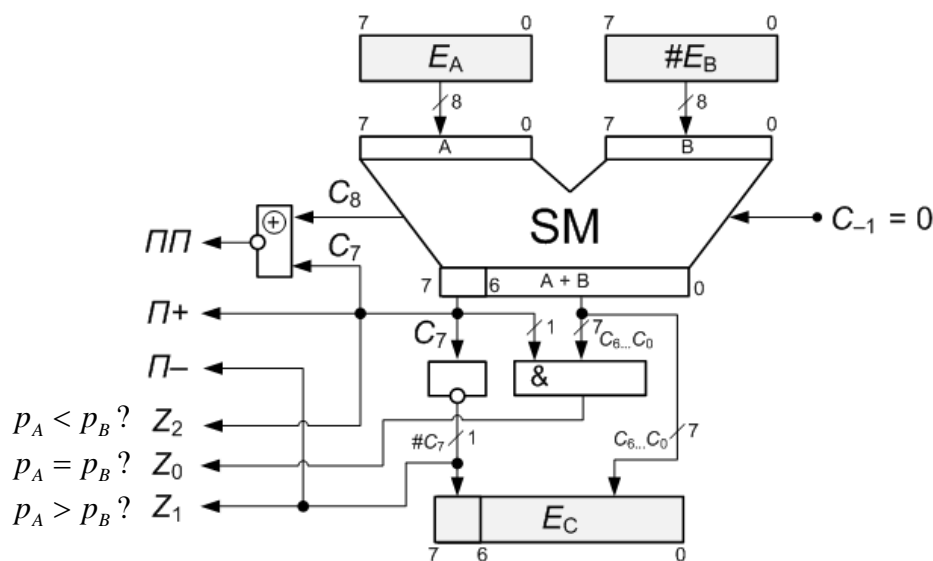
$$p_A = p_B ? \Rightarrow Z_0 = \begin{cases} 1, & \text{if } C_7 \div C_0 = 1 \\ 0, & \text{else} \end{cases} \Rightarrow Z_0 = (C_7 \wedge C_6 \wedge \dots \wedge C_0);$$

$$p_A > p_B ? \Rightarrow Z_1 = \begin{cases} 1, & \text{if } C_7 = 0 \\ 0, & \text{if } C_7 = 1 \end{cases} \Rightarrow Z_1 = \#C_7;$$

$$p_A < p_B ? \Rightarrow Z_2 = \begin{cases} 1, & \text{if } C_7 = 1 \\ 0, & \text{if } C_7 = 0 \end{cases} \Rightarrow Z_2 = C_7.$$

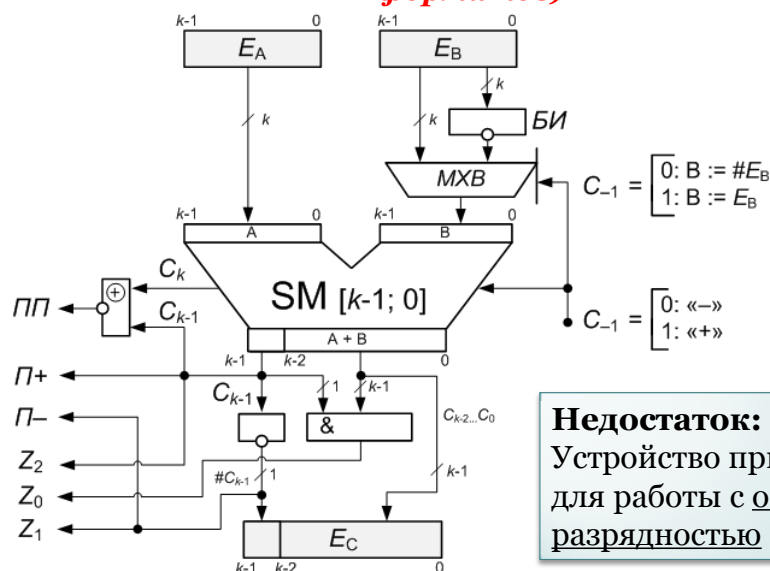
## Схема вычитания

Схема устройства, выполняющего вычитание порядков  
для формата single IEEE754



## Объединенная схема

*Схема устройства, выполняющего сложение и вычитание  $k$ -разрядных порядков (для бинарных и постбинарных форматов)*



### Недостаток:

Устройство пригодно только для работы с определенной разрядностью порядков!

## Постбинарные форматы

*Постбинарные форматы (rbinary) имеют следующие особенности по организации и работе с порядками:*

- Разрядность порядка и его смещение такие же как и у соответствующих бинарных форматов;
- Возможность динамического наращивания/сокращения разрядности формата, в том числе – **изменение разрядности порядка** (новое смещение, перерасчет смещенного значения, расчет новых границ диапазонов для оценки переполнения);
- Наличие двух принципов кодирования порядков: бинарное и постбинарное.

*Постбинарные форматы – форматы для хранения и обработки числовых данных в математическом постбинарном сопроцессоре!!*

### Важнейшая задача:

Разработка одного устройства для всех постбинарных форматов (!!!), выполняющего сложение и вычитание  $K$ -разрядных порядков и умеющего их обрабатывать (фиксирование переполнения и сравнение порядков)

## Критерии выбора разрядности устройства

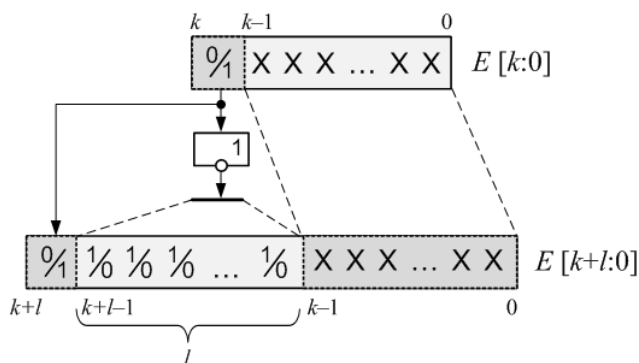
Разрядность формата	Разрядность порядка	Приращение к разрядности порядка от предыдущего формата
16	5	—
32	8	+ 3
64	11	+ 3
128	15	+ 4
256	20	+ 5
512	26	+ 6
1024	32	+ 6

Таким образом,  $K = 32$  – достаточная разрядность для устройства обработки порядков!!

## Принципы наращивания разрядности порядка

Наращивание порядка на  $l$  разрядов:  
 $E[k:0] \rightarrow E[k+l:0]$

1. Занесение значения старшего разряда меньшего порядка в старший разряд большего порядка;
2. Занесение значений оставшейся части меньшего порядка в младшие разряды большего порядка;
3. Заполнение оставшихся разрядов большего порядка значениями, противоположными значению старшего разряда.



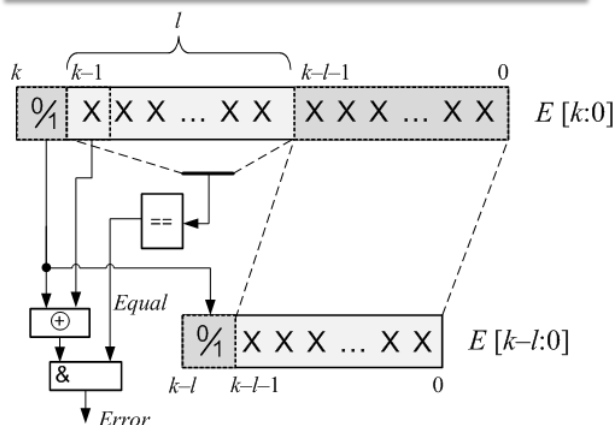
Исключение:  
нулевое поле порядка!

Только для форматов нормализованных чисел  
с плавающей запятой (т.е.  $E \neq 0000..00$ )!!

# Принципы сокращения разрядности порядка

1. Занесение значения старшего разряда большего порядка в старший разряд меньшего порядка;
2. Выделение поля значений перед старшим разрядом большего порядка с разрядностью, равной разнице разрядностей рассматриваемых порядков;
3. В случае равенства всех значений указанного поля, отличных от старшего разряда большего порядка (сокращение корректно) – заполнение оставшихся разрядов меньшего порядка значениями аналогичных разрядов старшего разряда. В иных случаях (значения старшего разряда и поля совпадают или значения поля не равны) – сокращение порядков некорректно.

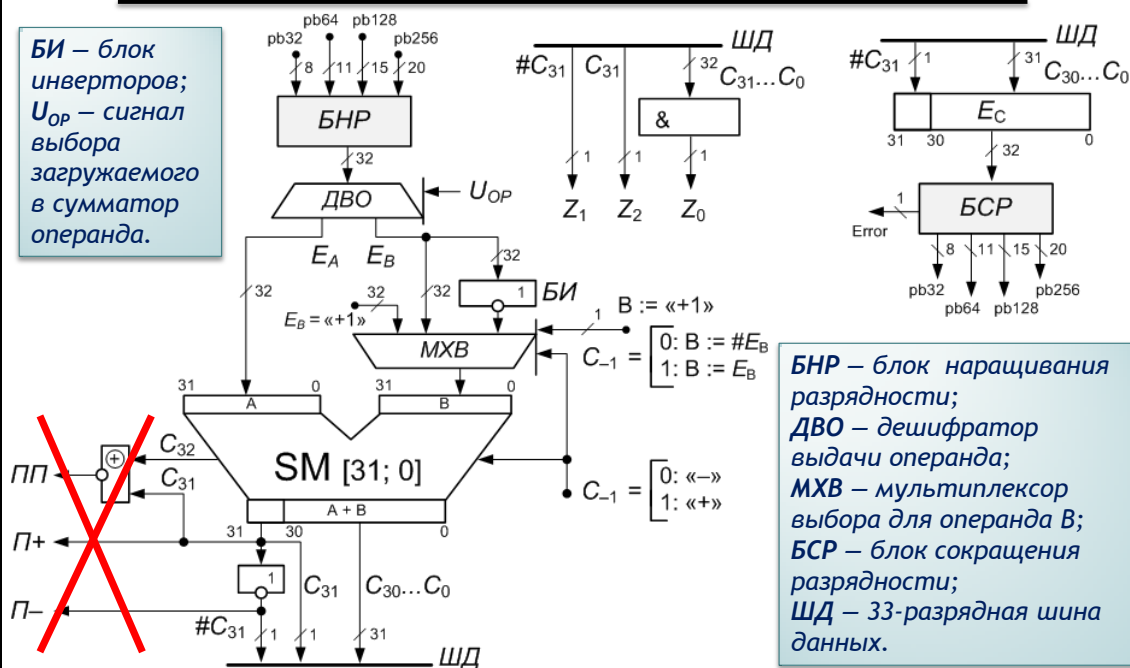
Сокращение порядка на  $l$  разрядов:  
 $E[k : 0] \rightarrow E[k - l : 0], (l < k)$



При **Equal** = 1 – равные значения разрядов.  
 При **Error** = 1 – сокращение некорректно!!

## Схема устройства, выполняющего операции с 32-разрядными порядками

**БИ** – блок  
инверторов;  
**U<sub>ор</sub>** – сигнал  
выбора  
загружаемого  
в сумматор  
операнда.



**БНР** – блок наращивания разрядности;  
**ДВО** – дешифратор выдачи операнда;  
**МХВ** – мультиплексор выбора для операнда В;  
**БСР** – блок сокращения разрядности;  
**ШД** – 33-разрядная шина данных.



<i>Tempum</i>	Пара битов
<b>0</b>	<b><u>0</u> 1</b>
<b>1</b>	<b><u>1</u> 0</b>

- При постбинарном кодировании работа с 32-разрядными порядками осуществляется в 64-разрядном устройстве**

Данная концепция работает только с тетритами неполной группы  $t = \{0, 1\}$ . Для полной группы  $t = \{0, 1, A, M\}$  данная концепция требует ряда уточнений и дополнительных проверок!!

The diagram illustrates the PS-1000 computer architecture, divided into a main unit and two peripheral units.

**Main Unit:**

- ПБНР (Post-Binary Number Register):** Receives data from the bus (pb64p, pb24p, pb128p, pb256p) and outputs 64 bits to the ДВО.
- ДВО (Digital-to-Word Converter):** Receives 64 bits from ПБНР and outputs 64 bits to the БИ.
- БИ (Binary Incrementer):** Takes 64 bits from ДВО and outputs 64 bits to the МХВ.
- МХВ (Microprogrammed Control Word):** Receives 64 bits from БИ and outputs 2 bits to the рSM.
- рSM [63; 0] (Post-Binary Summation Module):** Receives 64 bits from ДВО and 2 bits from МХВ. It outputs 64 bits to the bus (C65-64) and 61 bits to the БИ.
- БИ (Binary Incrementer):** Receives 61 bits from рSM and outputs 64 bits to the МХВ.
- МХВ (Microprogrammed Control Word):** Receives 64 bits from БИ and outputs 2 bits to the рSM.
- Bus (C65-64):** Connects the main unit to the peripheral units.

**Peripheral Units:**

- ШД (Data Bus):** A 65-bit data bus connecting the main unit to the peripheral units.
- ПБСР (Post-Binary Summation Register):** Receives data from the bus (pb32p, pb64p, pb128p, pb256p) and outputs 64 bits to the bus (C61...C0).
- ШД (Data Bus):** A 65-bit data bus connecting the peripheral units to the main unit.

**Legend:**

- рSM** – постбинарный 64-разрядный сумматор;
- ПБНР** – постбинарный блок наращивания разрядности;
- ПБСР** – постбинарный блок сокращения разрядности;
- ШД** – 65-разрядная шина данных.

## Заключение

### Выводы:

1. Рассмотрены алгоритмы сложения, вычитания и сравнения порядков с фиксацией переполнения и определения его направления;
2. Решен вопрос обработки порядков различных форматов в одном устройстве;
3. Уточнена возможность наращивания порядков без пересчета смещения и расчета нового значения порядка большей разрядности (**разработка блока наращивания/сокращения постбинарных порядков не рассмотрена, поскольку очевидна**);
4. Определены критерии возможности перехода к порядку меньшей разрядности;
5. Разработано устройство работы с порядками в контексте постбинарного кодирования.

### Актуальные вопросы:

1. Гарантировано ли **не появление** разрядов неоднозначностей в постбинарных порядках?
2. Вопрос **арбитража** разрядности порядка результата?

## для заметок



Донецкий национальный технический университет

# Введение в постбинарную арифметику

октябрь, 2012

Аспирант кафедры КИ  
Иваница С.В.

## Тетракоды и бинарные коды:

2

 $B \in \{0,1\}$  <sup>*b - бум*</sup> - бинарный код  $B = \{b_i \mid b_i = 0 \vee b_i = 1\}$ 
 $T \in \{0,1,A,M\}$  <sup>*t - тетра*</sup> - тетракод  $T = \{t_i \mid t_i = 0 \vee t_i = 1 \vee t_i = A \vee t_i = M\}$ 
 $T^B \in \{0,1\}$  - «бинарный» тетракод

101101  
100101  
011101  
100110

 $T^B \equiv B$ 
 $T^{N1} \in \{0,1,M\}$  - нормированный тетракод **1-го рода**
 $T^{N1} \Rightarrow \{0,1\}\{M_k\}$  ← Поле M

M  
110M  
100MM  
MMMMM

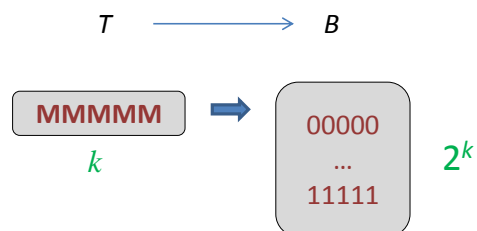
 $T^{N2} \in \{0,1,M,A\}$  - нормированный тетракод **2-го рода**
 $T^{N2} \Rightarrow \{0,1\}\{M_k\}\{A_l\}$  ← Поле M ← Поле A

MAA  
110MAAA  
10MMA  
MMMMAA

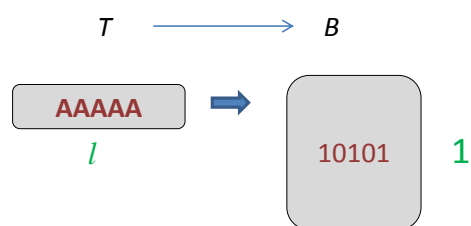
3

**Множественность и неопределенность:***Множественность*

$$M = \begin{bmatrix} 0, \\ 1 \end{bmatrix} \Rightarrow M = \{t \mid t = 0 \wedge t = 1\}$$

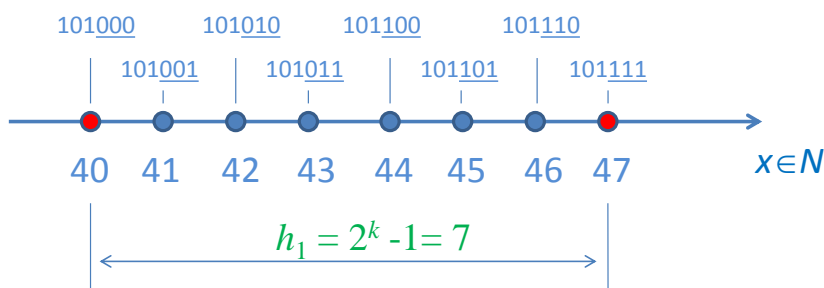
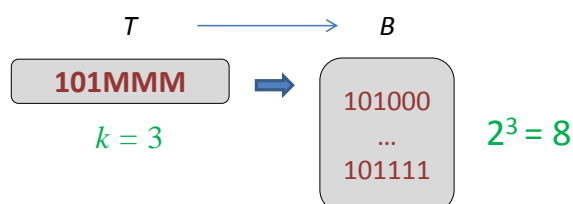
*Неопределенность*

$$A = \begin{bmatrix} 0, \\ 1 \end{bmatrix} \Rightarrow A = \{t \mid t = 0 \vee t = 1\}$$



$$A_i = Rnd(0,1)$$

4

**Тетракоды и целочисленные интервалы:***Пример для  $T^{N1}$* 

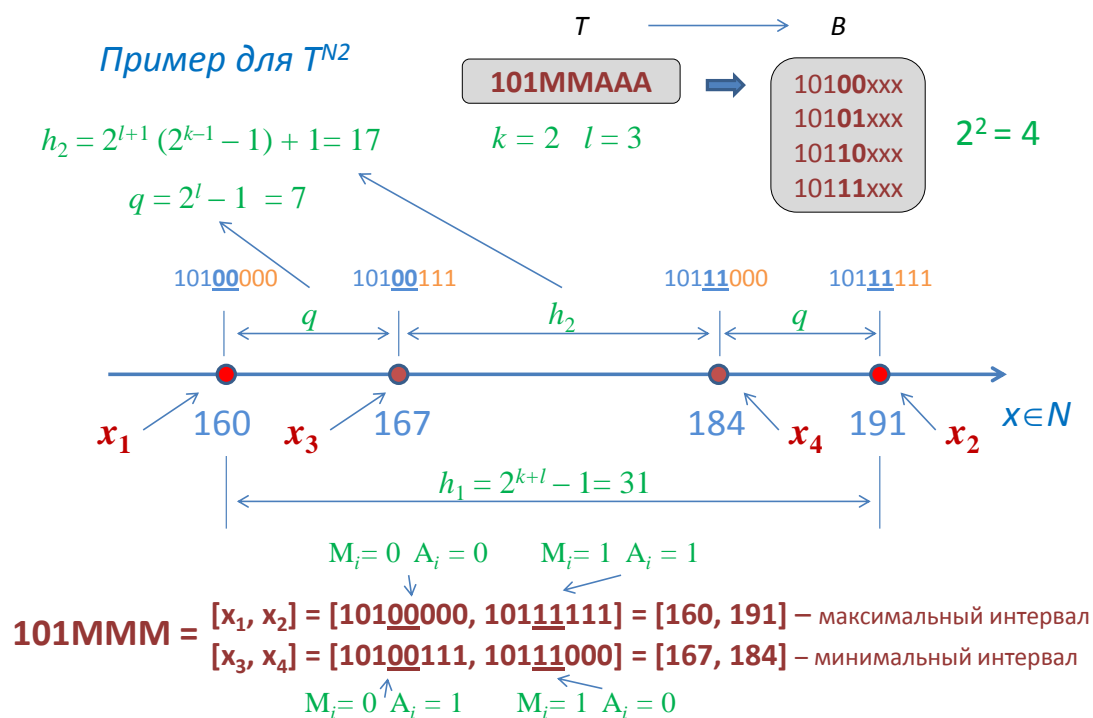
$$101MMM = [101000, 101111] = [40, 47]$$

$$M_i = 0$$

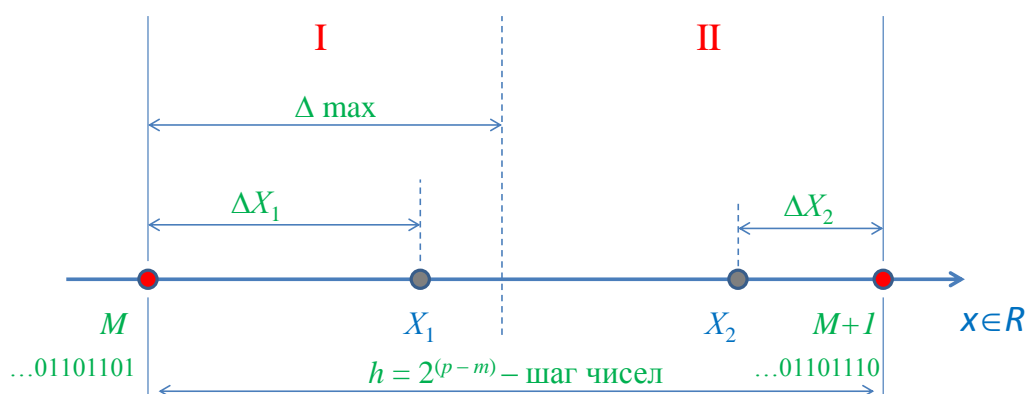
$$M_i = 1$$

5

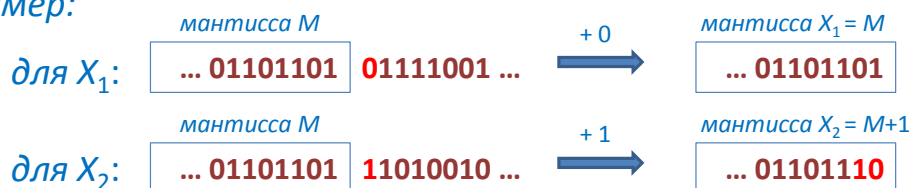
## Тетракоды и целочисленные интервалы:



6

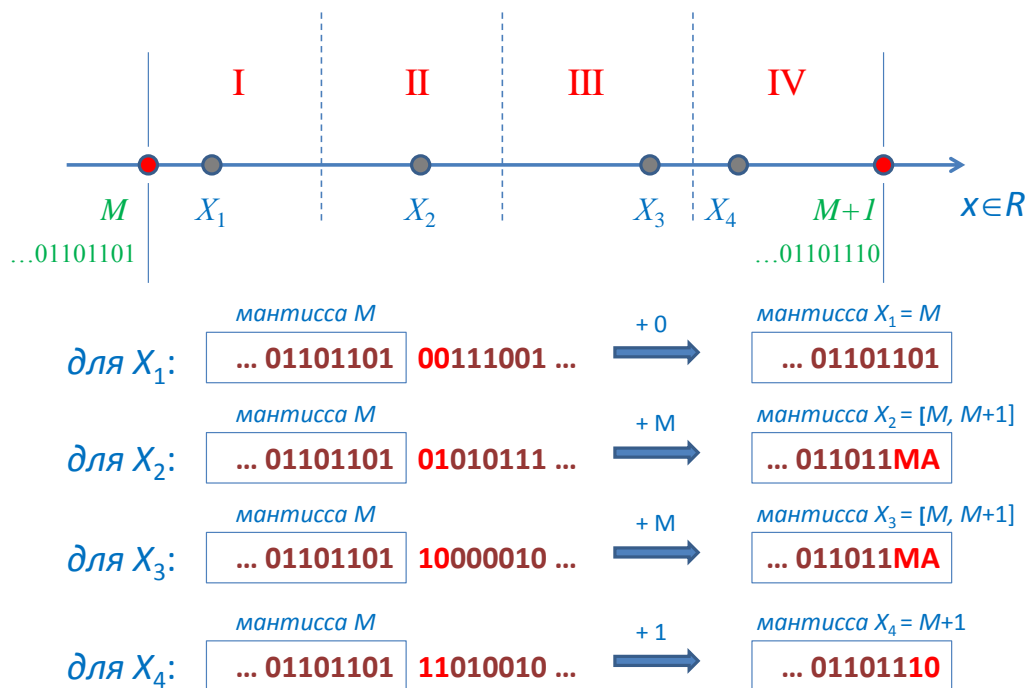
Округление вещественных чисел  
в формате с плавающей запятой:

Пример:





## Постбинарное округление вещественных чисел в формате с плавающей запятой:



## ЧАСТЬ 2

## Пример постбинарного вычитания чисел в формате rb32

141,2 – 14321,125

9

**Представление чисел:**

single: +141,1999969482421875

$141,2_{10} =$

0	10000110	00011010011001100110011	
0	10000110	000110100110011001101	11

pb32 : +141,20001220703125

single: +14321,125

$14321,125_{10} =$

0	10001100	101111111000100100000000	
0	10001100	101111111000100100000	00

pb32 : +14321,125

10

**Выравнивание порядков:**

single:  $\downarrow$

$141,2_{10} =$

0	10001100	00000100011010011001101	110011
0	10001100	000001000110100110011	001101

pb32 :  $\uparrow$

В числе **141,2** мантисса сдвинулась  
на шесть разрядов вправо (с учетом скрытой единицы)  
и произошло округление (Показаны выпавшие разряды мантиссы).  
Порядки чисел сравнялись.

single:  $\downarrow$

$14321,125_{10} =$

0	10001100	101111111000100100000000	
0	10001100	101111111000100100000	

pb32 :  $\uparrow$

11

### Производим вычитание мантисс:

Учитываем скрытую единицу

single:

$$\begin{array}{r}
 14321,125_{10} = 1 \ 10111111100010010000000 \\
 - 141,2_{10} = 0 \ 00000100011010011001101 \\
 \hline
 1 \ 10111011000111110110011
 \end{array}$$

Учитываем скрытую единицу

pb32 :

$$\begin{array}{r}
 14321,125_{10} = 1 \ 101111111000100100000 \\
 - 141,2_{10} = 0 \ 000001000110100110011 \\
 \hline
 1 \ 101110110001111101101
 \end{array}$$

12

### Получаем результат:

Скрываем старшую единицу в мантиссе, добавляем порядок и формируем знак

single:    − 14179,9248046875

$$\begin{array}{l}
 1 \ 10000110 \ 10111011000111110110\underline{0}11 \\
 1 \ 10000110 \ 10111011000111110110\underline{1}
 \end{array}$$

pb32 :    − 14179,92578125

Правильный результат:  $141,2 - 14321,125 = -141179,925$ 

Получили формат pb32 идентичный формату single

Форматы идентичны!!!  
Однако десятичные числа не равны!!!

**Объяснения и выводы :**

Числа с плавающей запятой обеспечивают относительную точность **некоторого количества десятичных цифр**, которое определяется по формуле ( $m$  – разрядность мантиссы формата):

$$\left\lfloor \lg 2^{m+1} \right\rfloor$$

Цветом выделено количество цифр для каждого формата которое может быть представлено точно:

single:  $\left\lfloor \lg 2^{24} \right\rfloor = 7$       – 14179,925

pb32:  $\left\lfloor \lg 2^{22} \right\rfloor = 6$       – 14179,925

Поскольку постбинарный формат **pb32** выводится на экран **средствами бинарного компьютинга**, то он представляется как **float** (добавляются два ноля в младшие разряды) и к нему применяются приемы округления как к числу одинарной точности.

В конечной версии программной модели постбинарной арифметики все операции по кодированию в двоично-десятичном направлении будут специально разработаны для постбинарных форматов!

**для заметок**

Донецкий национальный  
технический университет

# СИНТЕЗ ПОСТБИНАРНЫХ СУММИРУЮЩИХ, ВЫЧИТАЮЩИХ И УМНОЖАЮЩИХ СХЕМ

К вопросу  
аппаратной реализации  
постбинарной арифметики

17/10/2012

Аспирант кафедры КИ: **Иваница С.В.**

2

## Введение

$T \in \{0, 1, A, M\}$  – тетракод.

$B \in \{0, 1\}$  – двоичный код.

**Тетракод  $T$** , как способ представления (кодирования) количественных значений постбинарного компьютеринга выражается (что на сегодняшний день является единственно возможным) средствами бинарного компьютеринга, т. е. с **помощью двоичного кода  $B$** .

Поэтому в основе разработки **постбинарной арифметики** с плавающей запятой лежат принципы аппаратной реализации **двоичной плавающей арифметики**

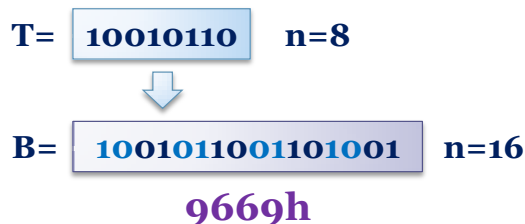


## Отображение тетракодов на множестве двоичных чисел

Таблица  
соответствия разряда тетрита к  
паре двоичных разрядов

Тетрит Т [о]	Биты В [1:0]
<b>A</b>	<b>00</b>
<b>0</b>	<b>01</b>
<b>1</b>	<b>10</b>
<b>M</b>	<b>11</b>

$$T[n] \rightarrow B[2n]$$

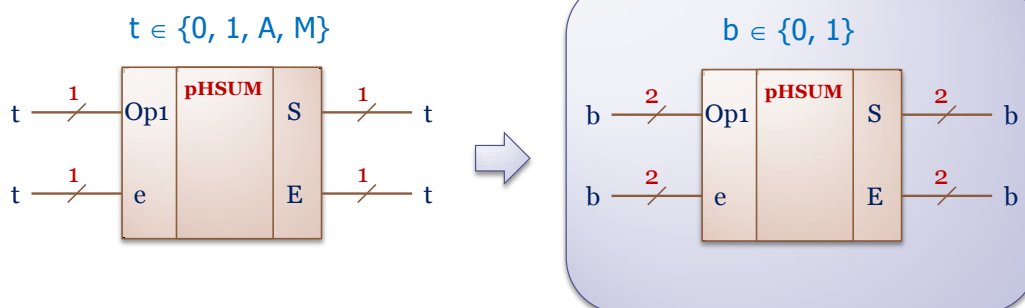


N / {1, 2, 3, 4, 8, D, E} — для нормированных тетракодов

Все показанные дальше схемы работают с **битами**, т. е.  
с **битовым представлением** тетракода!

## Постбинарные однетрититные суммирующие схемы

### Постбинарный полусумматор pHSUM



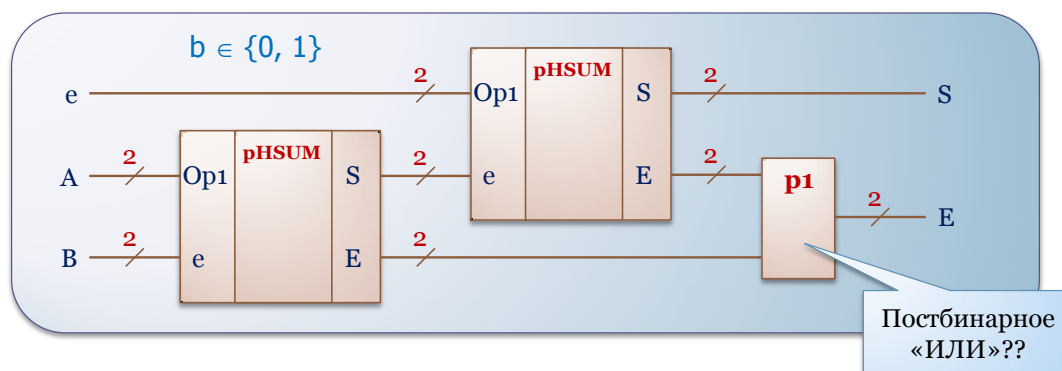
**Op1** — слагаемое,  
**e** — входной перенос,  
**S = Op1 + e** — сумма,  
**E** — выходной перенос.

$$\begin{aligned} E0 &= Op1[0] \vee e0; & S0 &= \overline{Op1[0]} \oplus e0; \\ E1 &= Op1[1] \wedge e1; & S1 &= Op1[1] \oplus e1. \end{aligned}$$

## Постбинарные однотетритные суммирующие схемы

### Схема подключения постбинарного полусумматора

$$S = A + B + e$$



## Постбинарное ИЛИ

### Постбинарное «ИЛИ» в тетритах

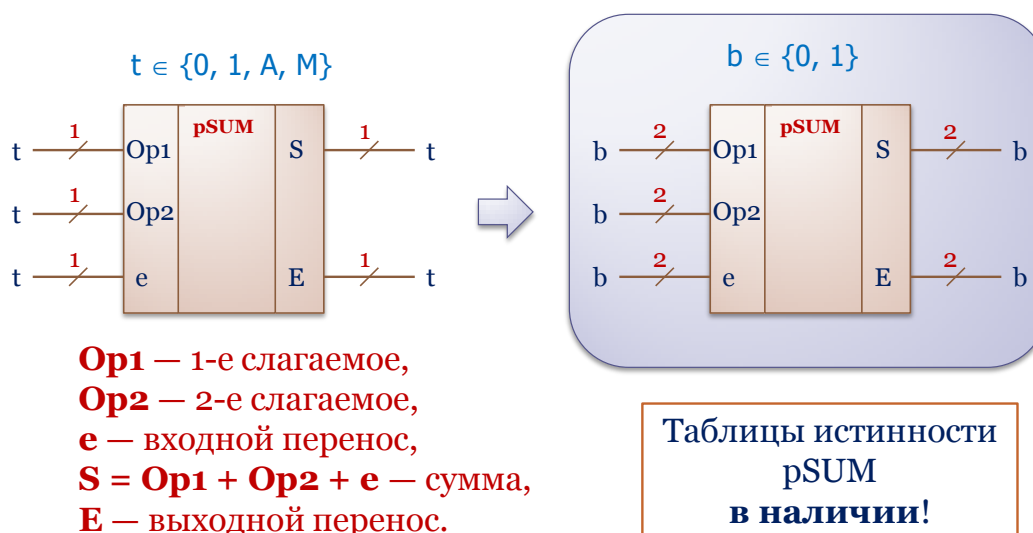
t1	t2	$t1 \vee t2$	t1	t2	$t1 \vee t2$	t1	t2	$t1 \vee t2$	t1	t2	$t1 \vee t2$
0	0	0	1	0	1	A	0	A	M	0	M
0	1	1	1	1	1	A	1	1	M	1	1
0	A	A	1	A	1	A	A	A	M	A	1
0	M	M	1	M	1	A	M	1	M	M	M

### Постбинарное «ИЛИ» в битах

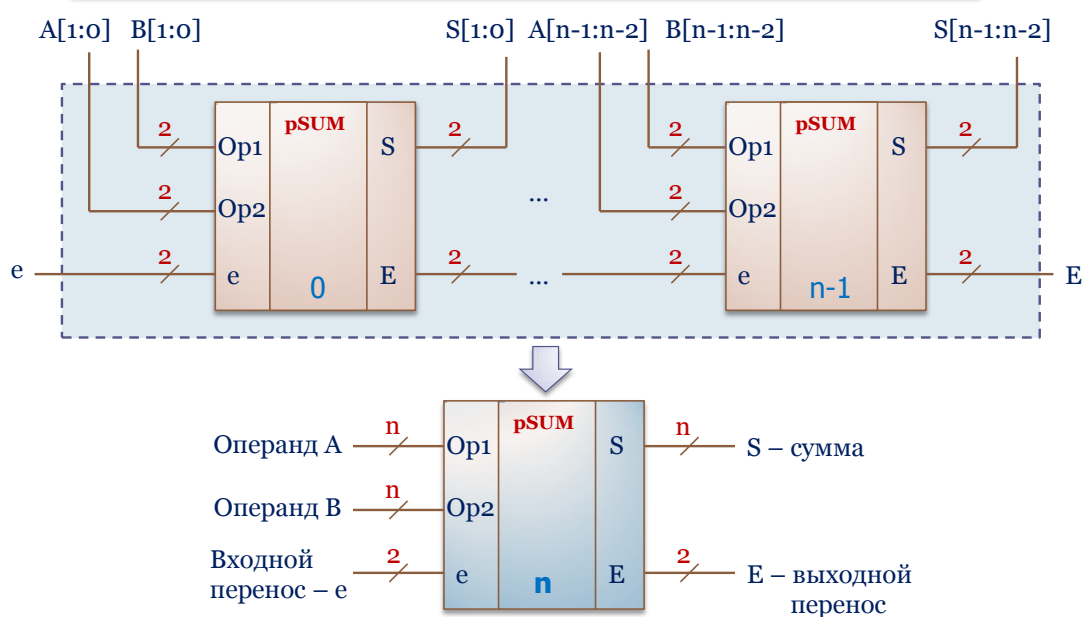
b1	b2	$b1 \vee b2$	b1	b2	$b1 \vee b2$	b1	b2	$b1 \vee b2$	b1	b2	$b1 \vee b2$
01	01	01	10	01	10	00	01	00	11	01	11
01	10	10	10	10	10	00	10	10	11	10	10
01	00	00	10	00	10	00	00	00	11	00	10
01	11	11	10	11	10	00	11	10	11	11	11

## Постбинарные однотетритные суммирующие схемы

### Постбинарный сумматор pSUM

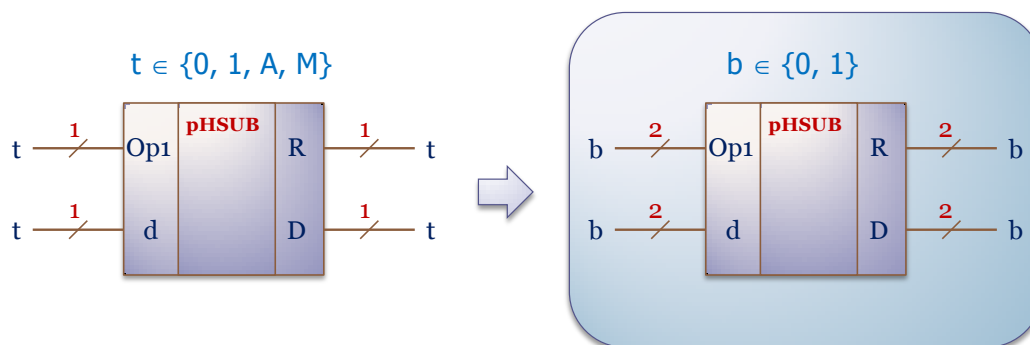


## Постбинарный n-разрядный сумматор



## Постбинарные одноканальные вычитающие схемы

### Постбинарный полувычитатель pHSUB

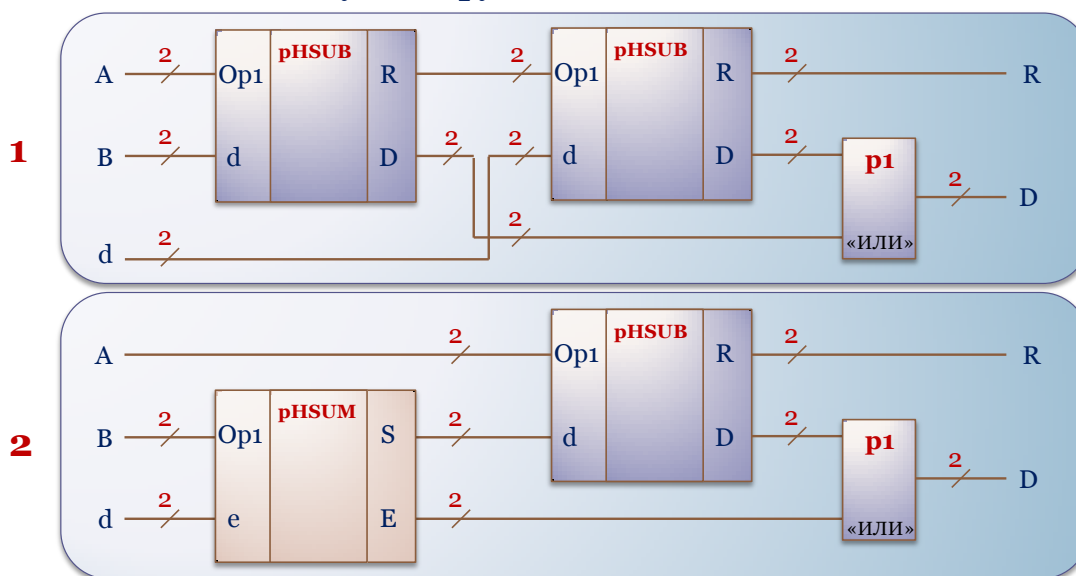


**Op1** — уменьшаемое,  
**d** — заем из младшего разряда,  
**R = Op1 - d** — разность,  
**D** — заем из старшего разряда.

$$\begin{aligned} D0 &= \overline{\text{Op1}[0]} \vee d0; & R0 &= \overline{\text{Op1}[0]} \oplus d0; \\ D1 &= \overline{\text{Op1}[1]} \wedge d1; & R1 &= \text{Op1}[1] \oplus d1. \end{aligned}$$

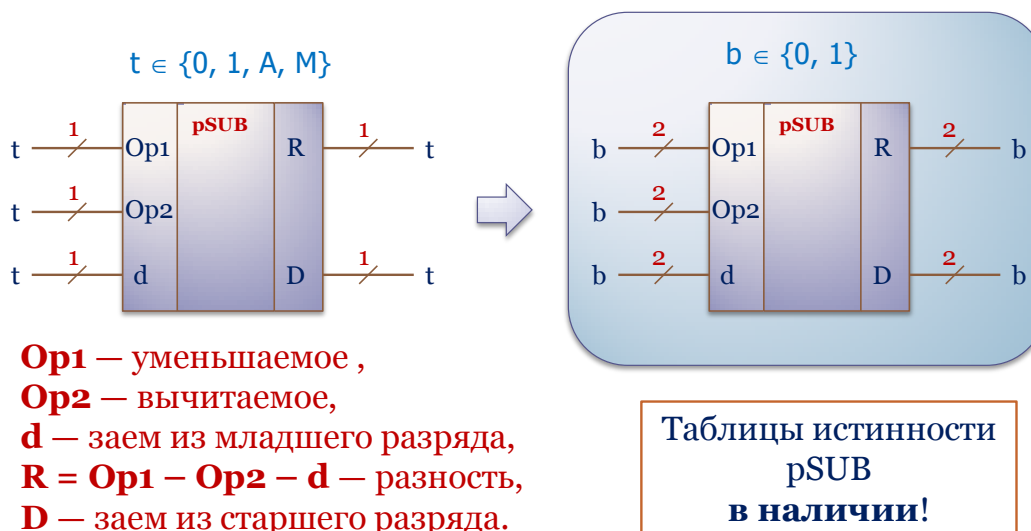
## Схемы подключения полувычитателя

Реализуемая функция:  $R = A - B - d$   $b \in \{0, 1\}$

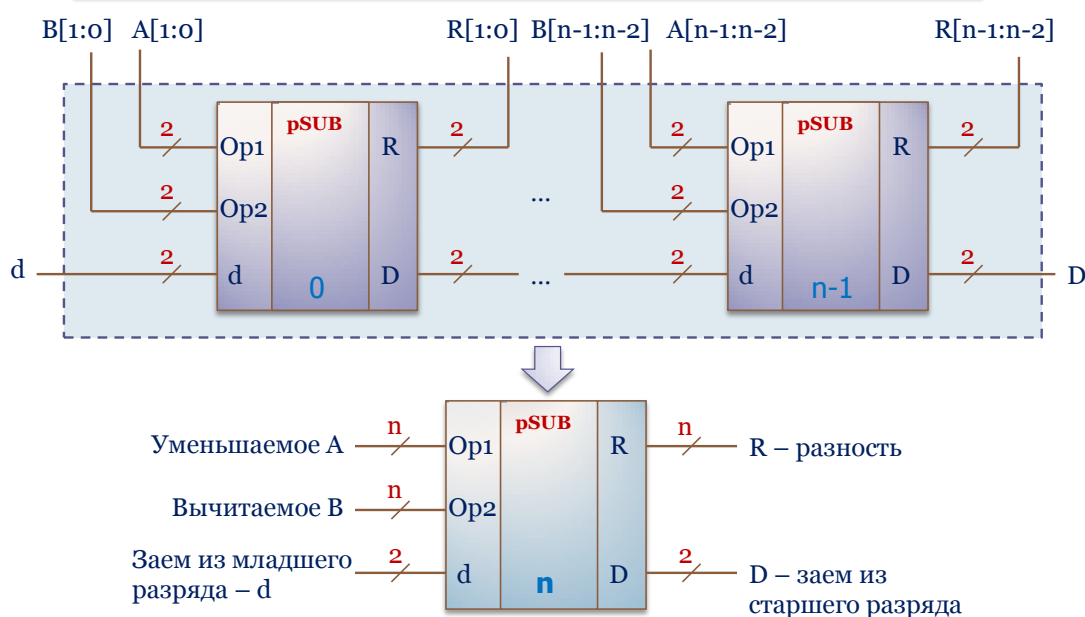


## Постбинарные однотетритные вычитающие схемы

### Постбинарный вычитатель pSUB



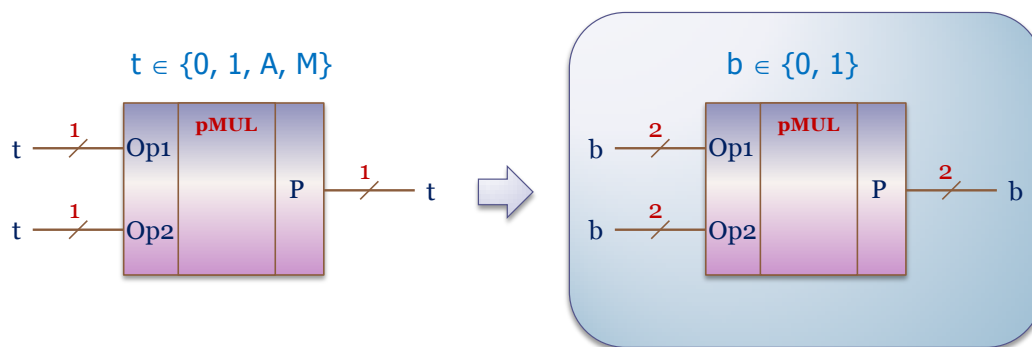
## Постбинарный n-разрядный вычитатель





## Постбинарные однотетритные умножающие схемы

### Постбинарный умножитель pMUL



**Op1** — множитель,  
**Op2** — множитель,  
**P = Op1 \* Op2** — произведение.

$$P_0 = Op1[0] \vee Op2[0];$$

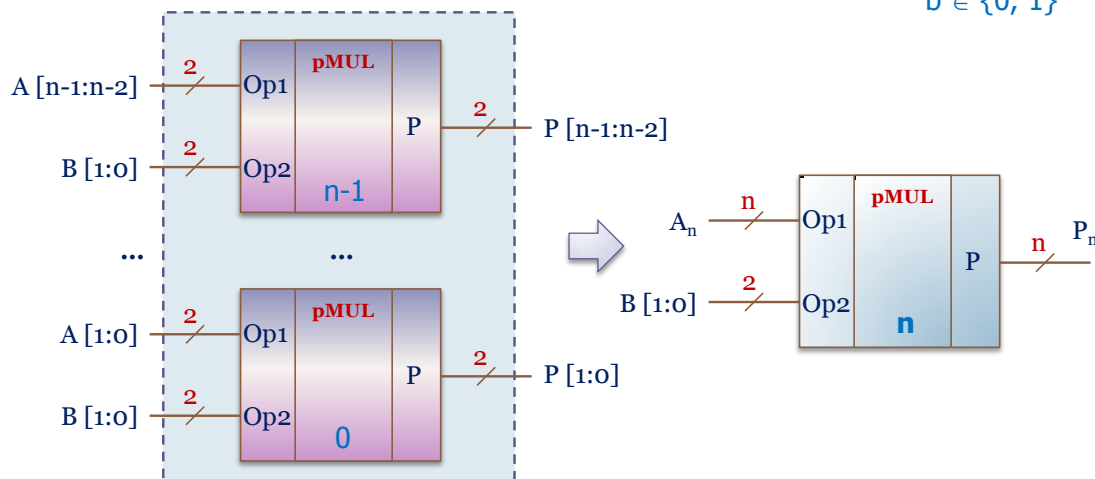
$$P_1 = Op1[1] \wedge Op2[1].$$

## Постбинарный n-разрядный умножитель

Постбинарный умножитель реализуется как произведение младшего разряда (тетрита) второго множителя на все разряды первого множителя

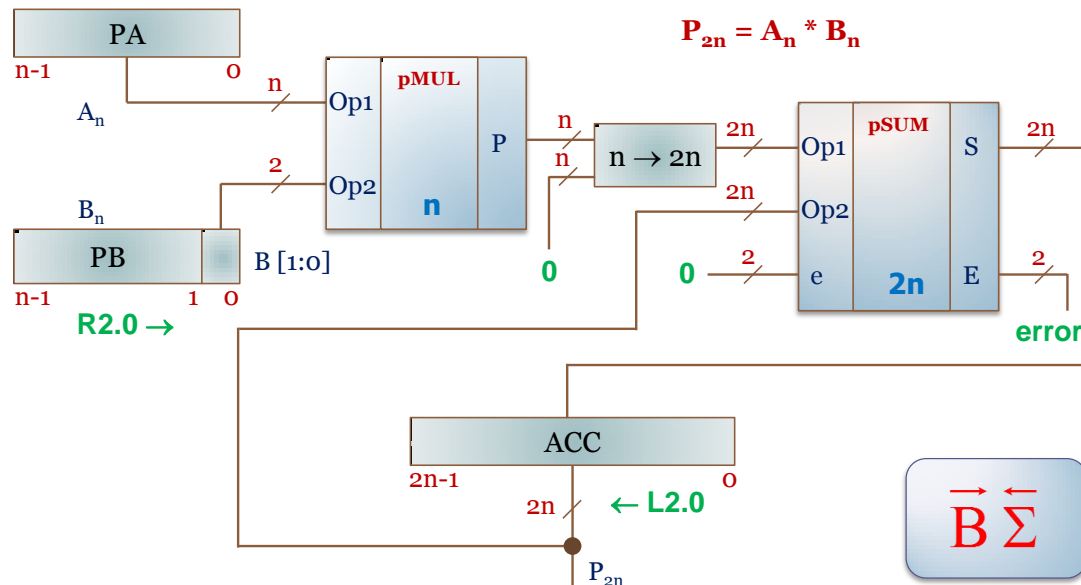
$$P_n = A_n * B_{1-0}$$

$$b \in \{0, 1\}$$



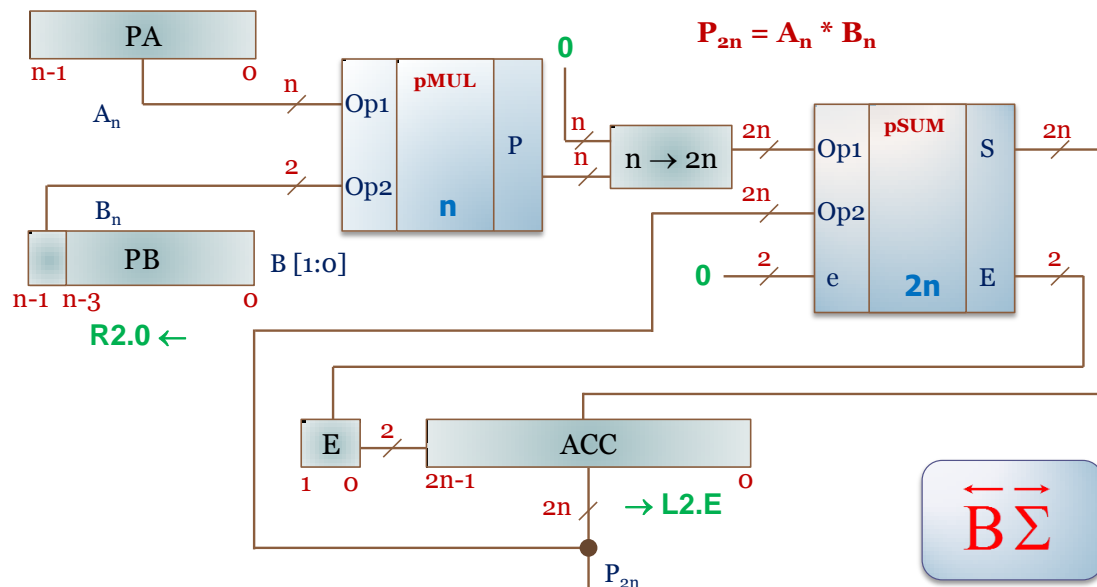
15

## Схема умножения n-разрядных тетракодов



16

## Схема умножения n-разрядных тетракодов



## Постбинарное И

Постбинарное «И» в тетритах

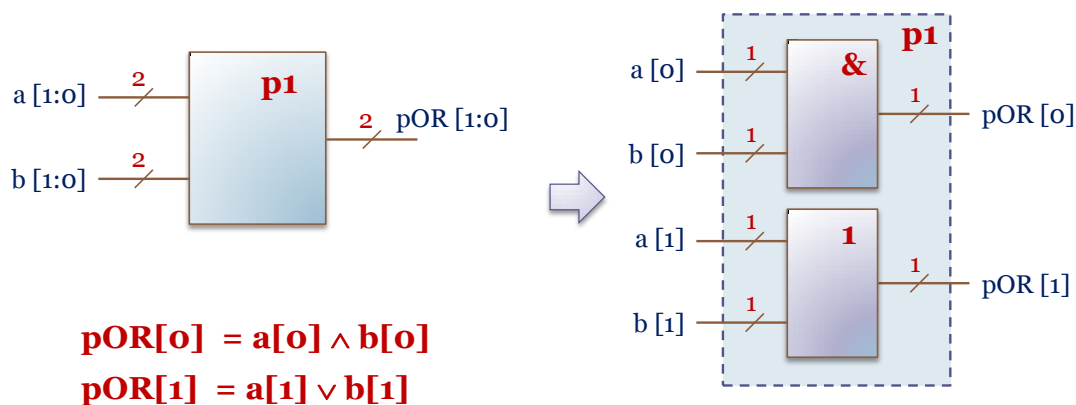
t1	t2	t1 & t2	t1	t2	t1 & t2	t1	t2	t1 & t2	t1	t2	t1 & t2
0	0	0	1	0	0	A	0	0	M	0	0
0	1	0	1	1	1	A	1	A	M	1	M
0	A	0	1	A	A	A	A	A	M	A	0
0	M	0	1	M	M	A	M	0	M	M	M

Постбинарное «И» в битах

b1	b2	b1 & b2	b1	b2	b1 & b2	b1	b2	b1 & b2	b1	b2	b1 & b2
01	01	01	10	01	01	00	01	01	11	01	01
01	10	01	10	10	10	00	10	00	11	10	11
01	00	01	10	00	00	00	00	00	11	00	01
01	11	01	10	11	11	00	11	01	11	11	11

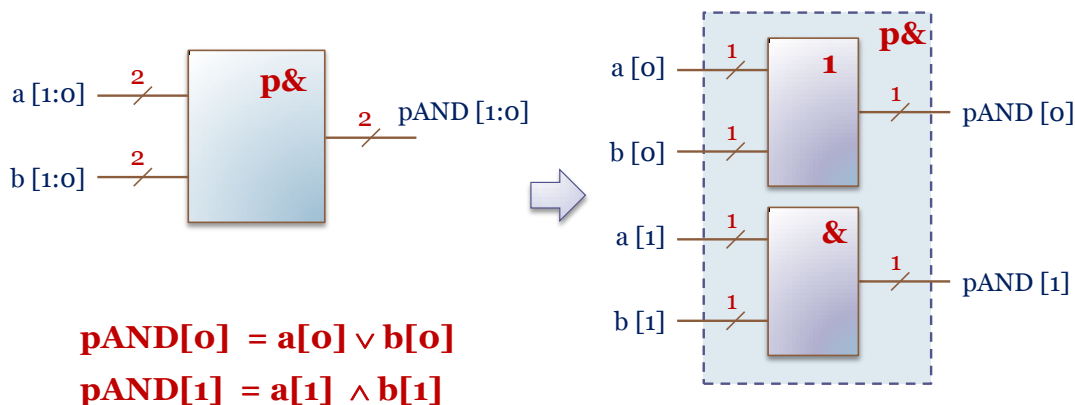
## Синтез постбинарной логической функции ИЛИ

$$pOR[1:0] = a[1:0] \vee b[1:0]$$



## Синтез постбинарной логической функции И

$$pAND[1:0] = a[1:0] \wedge b[1:0]$$



## Синтез постбинарной логической функции НЕ

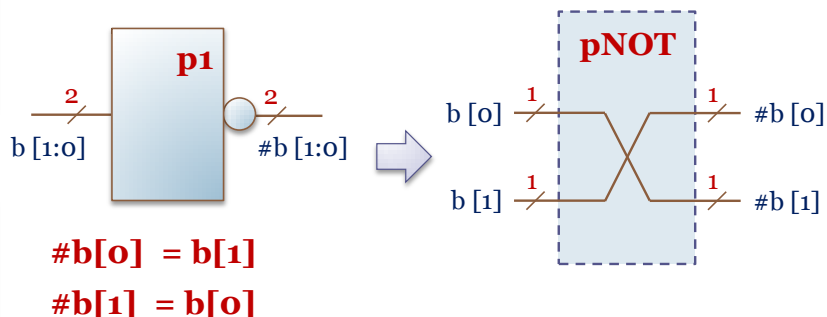
$$pNOT(b[1:0])$$

в тетритах

t	#t
0	1
1	0
A	A
M	M

в битах

b	#b
0 1	1 0
1 0	0 1
0 0	0 0
1 1	1 1



## Итоговые таблицы истинности

### Арифметика

+	0	A	M	1
0	0	A	M	1
A	A	A0	1	AM
M	M	1	M0	MA
1	1	AM	MA	10

×	0	A	M	1
0	0	0	0	0
A	0	A	0	A
M	0	0	M	M
1	0	A	M	1

### Логика

∨	0	A	M	1
0	0	A	M	1
A	A	A	1	1
M	M	1	M	1
1	1	1	1	1

∧	0	A	M	1
0	0	0	0	0
A	0	A	0	A
M	0	0	M	M
1	0	A	M	1

## Выводы

1. Построены схемы **n-разрядного сумматора** с помощью таблиц истинности или с использованием схемы соединения двух полусумматоров. **Выполнен синтез полусумматора.**
2. Построены схемы **n-разрядного вычитателя** с помощью таблиц истинности или с использованием схем соединения двух полувывчитателей или одного полусумматора и одного полувывчитателя. **Выполнен синтез полувывчитателя.**
3. Построена схема **2n-разрядного блока умножения** n-разрядных множителей с использованием множителя и аккумулирующего 2n-разрядного сумматора.
4. Подтверждены разработанные ранее таблицы постбинарных **логических операций конъюнкции и дизъюнкции.**
5. **Синтезированы блоки постбинарных «И» и «ИЛИ».**





ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ  
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

**АЛГЕБРА ТЕТРАЛОГИКИ**

*Синтез унарных  
логических операций и  
исключающего ИЛИ*

аспирант кафедры КИ  
Иваница С.В.

24.10.2012

**Синтез унарных  
логических операций  
инверсной группы  
(SWAP GROUP)**

## Отрицание около неопределенности и множественности (традиционная инверсия)

в тетритах

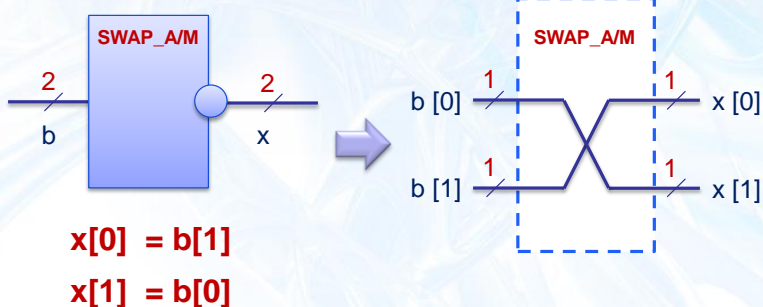
t	SWAP_A/M(t)
0	1
1	0
A	A
M	M

SWAP\_A/M(t)



в битах

b	x
0 1	1 0
1 0	0 1
0 0	0 0
1 1	1 1



3

## Отрицание около нуля и единицы

в тетритах

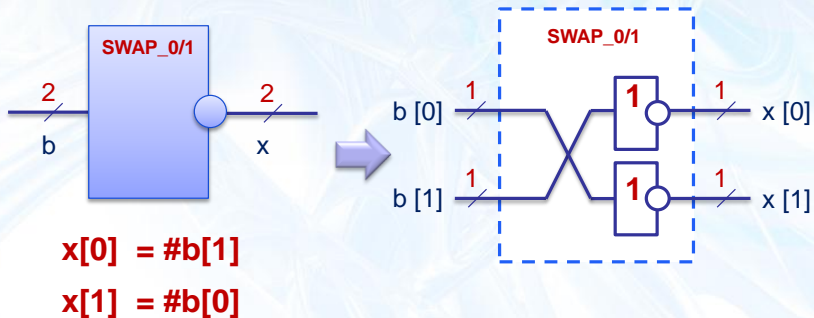
t	SWAP_0/1(t)
0	0
1	1
A	M
M	A

SWAP\_0/1(t)



в битах

b	x
0 1	0 1
1 0	1 0
0 0	1 1
1 1	0 0



4



## Вероятностное отрицание по оси False

в тетритах

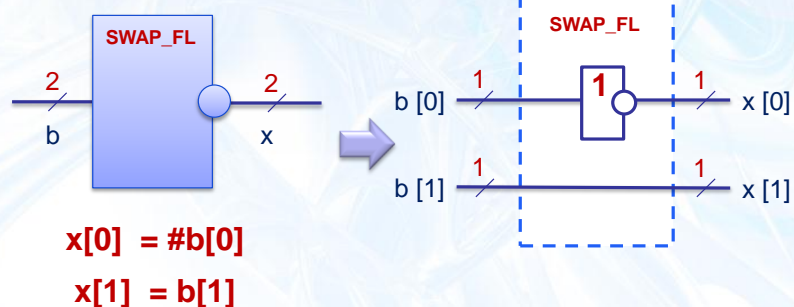
t	SWAP_FL(t)
0	A
1	M
A	0
M	1

SWAP\_FL(t)



в битах

b	x
0 1	0 0
1 0	1 1
0 0	0 1
1 1	1 0



5

## Вероятностное отрицание по оси True

в тетритах

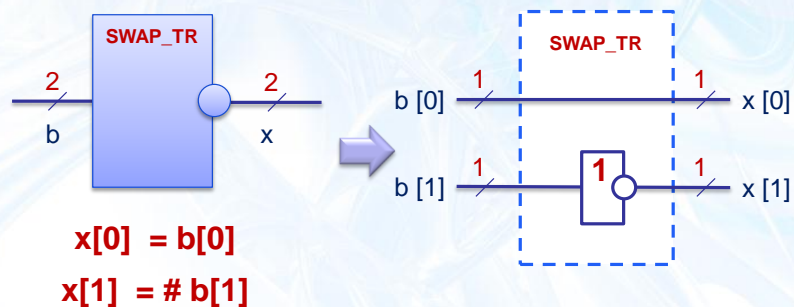
t	SWAP_FL(t)
0	M
1	A
A	1
M	0

SWAP\_TR(t)



в битах

b	x
0 1	1 1
1 0	0 0
0 0	1 0
1 1	0 1



6

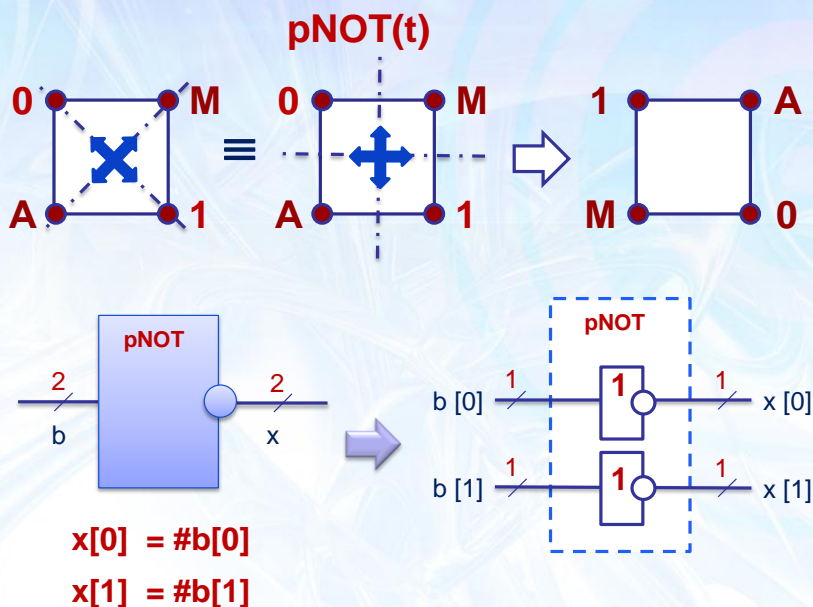
## Полное (глобальное) отрицание. Постбинарная инверсия

в тетритах

t	pNOT(t)
0	1
1	0
A	M
M	A

в битах

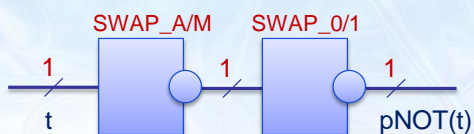
b	x
0 1	1 0
1 0	0 1
0 0	1 1
1 1	0 0



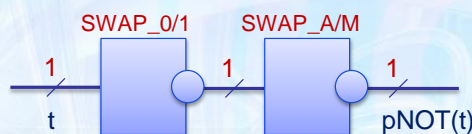
7

## Постбинарная инверсия

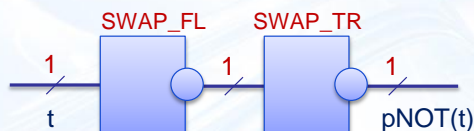
**pNOT(t)**



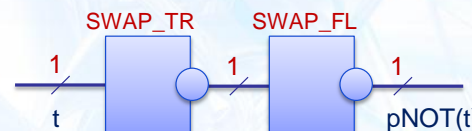
$$pNOT(t) \equiv SWAP\_0/1(SWAP\_A/M(t))$$



$$pNOT(t) \equiv SWAP\_A/M(SWAP\_0/1(t))$$



$$pNOT(t) \equiv SWAP\_FL(SWAP\_TR(t))$$



$$pNOT(t) \equiv SWAP\_TR(SWAP\_FL(t))$$

8



## Закон двойного отрицания

*Над всеми видами инверсии справедлив*

**ЗАКОН ДВОЙНОГО ОТРИЦАНИЯ**

$$t \leftrightarrow \text{SWAP}_\Psi (\text{SWAP}_\Psi(t))$$

$$\Psi \in \{A/M, 0/1, FL, TR\}$$

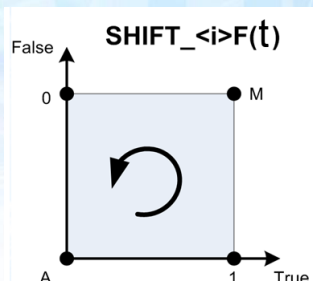
$$t \leftrightarrow \text{pNOT} (\text{pNOT}(t))$$

9

## Синтез унарных логических операций сдвиговой группы (SHIFT GROUP)



## Сдвиговые операции



### SHIFT\_<i>F(t)

– циклический сдвиг (поворот, вращение) **вперед** на  $i$ ;

$i \bmod 4 = 0$  – тождественная функция, повторитель (полный оборот)

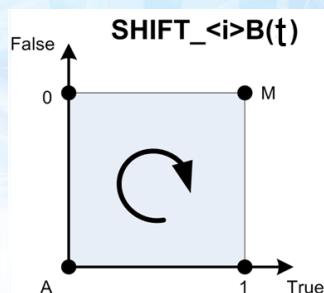
$i \bmod 4 = 1$  – циклический сдвиг вперед на 1 (1/4 оборота)

$i \bmod 4 = 2$  – циклический сдвиг вперед на 2 (1/2 оборота)

$i \bmod 4 = 3$  – циклический сдвиг вперед на 3 (3/4 оборота)

11

## Сдвиговые операции



### SHIFT\_<j>B(t)

– циклический сдвиг (поворот, вращение) **назад** на  $j$ ;

$j \bmod 4 = 0$  – тождественная функция, повторитель (полный оборот)

$j \bmod 4 = 1$  – циклический сдвиг **назад** на 1 (1/4 оборота)

$j \bmod 4 = 2$  – циклический сдвиг **назад** на 2 (1/2 оборота)

$j \bmod 4 = 3$  – циклический сдвиг **назад** на 3 (3/4 оборота)

12

## Эквивалентность сдвиговых операций (в пределах одного оборота)

$$t \equiv \text{SHIFT\_0B}(t) \equiv \text{SHIFT\_0F}(t)$$

$$\text{SHIFT\_1B}(t) \equiv \text{SHIFT\_3F}(t)$$

$$\text{SHIFT\_2B}(t) \equiv \text{SHIFT\_2F}(t)$$

$$\text{SHIFT\_3B}(t) \equiv \text{SHIFT\_1F}(t)$$

13

## Эквивалентность логических операций сдвиговой и инверсной групп

$$t \leftrightarrow \text{SHIFT\_0B}(t) \leftrightarrow \text{любое двойное отрицание}$$

$$\text{SHIFT\_1B}(t) \equiv \text{SHIFT\_3F}(t) \leftrightarrow \text{SWAP\_FL}(t)$$

$$\text{SHIFT\_2B}(t) \equiv \text{SHIFT\_2F}(t) \leftrightarrow \text{pNOT}(t)$$

$$\text{SHIFT\_3B}(t) \equiv \text{SHIFT\_1F}(t) \leftrightarrow \text{SWAP\_TR}(t)$$

Операции сдвиговой группы эквивалентны  
вероятностным отрицаниям и  
полной постбинарной инверсии

14



## Законы сдвиговых операций

Закон  $k$ -ого сдвига в  $k$ -ичной логике, согласно которому  $k$  сдвигов вперед/назад равносильны повторению (утверждению)

$$\text{SHIFT}_{<i>\vartheta} \leftrightarrow X \text{ при } \vartheta \in \{B, F\}, i \bmod 4 = 0$$

$$t \leftrightarrow \text{SHIFT}_{0B}(t) \leftrightarrow \text{SHIFT}_{4B}(t) \leftrightarrow \text{SHIFT}_{8B}(t) \leftrightarrow \dots$$

$$t \leftrightarrow \text{SHIFT}_{0F}(t) \leftrightarrow \text{SHIFT}_{4F}(t) \leftrightarrow \text{SHIFT}_{8F}(t) \leftrightarrow \dots$$

15

## Законы сдвиговых операций

Закон цикличности, согласно которому  $k$  сдвигов вперед/назад приводит к эквивалентным операциям, если

$$\text{SHIFT}_{<i>\vartheta} \leftrightarrow \text{SHIFT}_{<j>\vartheta} \\ \text{при } \vartheta \in \{B, F\}, i \bmod 4 = j \bmod 4$$

$$\text{SHIFT}_{1B}(t) \leftrightarrow \text{SHIFT}_{5B}(t) \leftrightarrow \text{SHIFT}_{9B}(t) \leftrightarrow \dots$$

$$\text{SHIFT}_{2F}(t) \leftrightarrow \text{SHIFT}_{6F}(t) \leftrightarrow \text{SHIFT}_{10F}(t) \leftrightarrow \dots$$

$$\text{SHIFT}_{3B}(t) \equiv \text{SHIFT}_{7F}(t) \leftrightarrow \text{SHIFT}_{11F}(t) \leftrightarrow \dots$$

16

# Минимизация / максимизация неопределенности и множественности

## Минимизация неопределенности

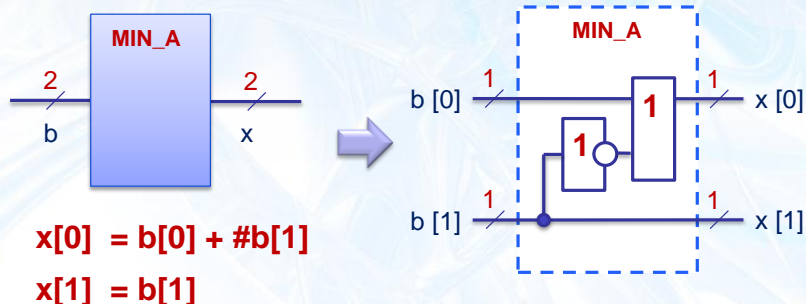
в тетритах

t	MIN_A(t)
0	0
1	1
A	0
M	M

в битах

b	x
0 1	0 1
1 0	1 0
0 0	0 1
1 1	1 1

MIN\_A(t)





## Максимизация неопределенности

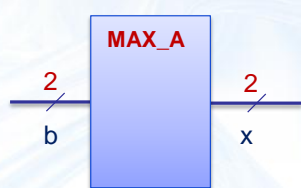
в тетритах

t	MAX_A(t)
0	0
1	1
A	1
M	M

в битах

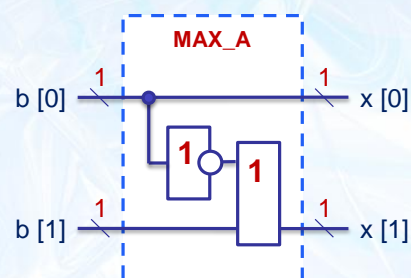
b	x
0 1	0 1
1 0	1 0
0 0	1 0
1 1	1 1

MAX\_A(t)



$$x[0] = b[0]$$

$$x[1] = \#b[0] + b[1]$$



19

## Минимизация множественности

в тетритах

t	MIN_M(t)
0	0
1	1
A	A
M	0

в битах

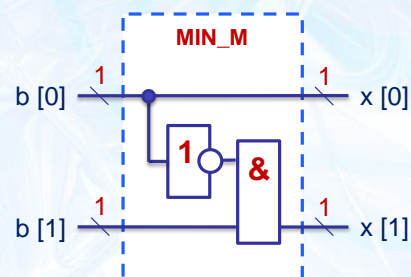
b	x
0 1	0 1
1 0	1 0
0 0	0 0
1 1	0 1

MIN\_M(t)



$$x[0] = b[0]$$

$$x[1] = \#b[0] \& b[1]$$



20



## Максимизация множественности

в тетридах

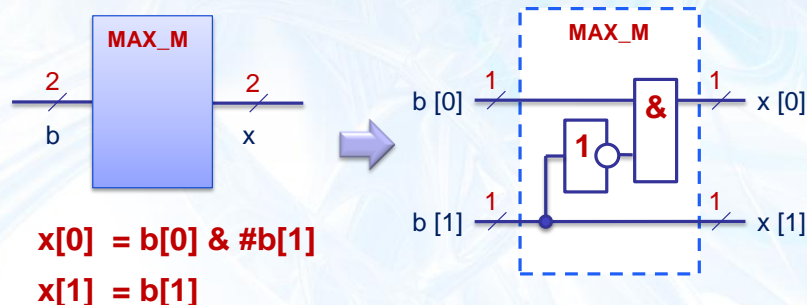
t	MAX_M(t)
0	0
1	1
A	A
M	1

MAX\_M(t)



в битах

b	x
0 1	0 1
1 0	1 0
0 0	0 0
1 1	1 0



21

## Применение минимизации и максимизации

Операции, **MIN\_A**, **MAX\_A**, **MIN\_M**, **MAX\_M** применяются для сведения тетракода **T** к границам минимального интервала  $[T_1, T_2]$  и максимального интервала  $[T_3, T_4]$ :

$$T_1 = \text{MIN\_M}(\text{MIN\_A}(T)) = \text{MIN\_A}(\text{MIN\_M}(T))$$

$$T_2 = \text{MAX\_M}(\text{MAX\_A}(T)) = \text{MAX\_A}(\text{MAX\_M}(T))$$

$$T_3 = \text{MIN\_M}(\text{MAX\_A}(T)) = \text{MAX\_A}(\text{MIN\_M}(T))$$

$$T_4 = \text{MAX\_M}(\text{MIN\_A}(T)) = \text{MIN\_A}(\text{MAX\_M}(T))$$

22

# Постбинарное исключающее ИЛИ

## Определение

**Правило для бинарной логики:**

результат равен 0, если оба операнда равны;  
во всех остальных случаях результат равен 1.

**Правило для постбинарной логики:**

результат равен 0, если оба операнда равны;  
во всех остальных случаях результат нулю не равен.

**Определение исключающего ИЛИ:**

исключающее ИЛИ – сумма по модулю 2, т. е. алгебраическая сумма операндов без переносов.



## Таблица истинности

Арифметика

+	0	A	M	1
0	0	A	M	1
A	A	A0	1	AM
M	M	1	M0	MA
1	1	AM	MA	10

Логика

pXOR	0	A	M	1
0	0	A	M	1
A	A	0	1	M
M	M	1	0	A
1	1	M	A	0

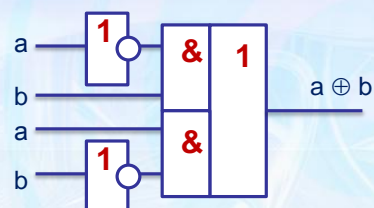
25

## Функция XOR и pXOR

Для бинарной логики:

Все логические операции – бинарные:

$$a \oplus b = (a \wedge \#b) \vee (\#a \wedge b)$$

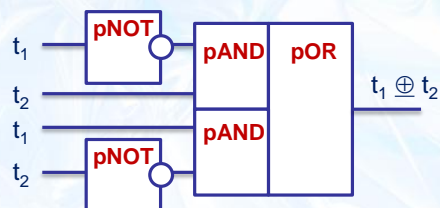


Для постбинарной логики:

Все логические операции – постбинарные:

$$t_1 \oplus t_2 = (t_1 \wedge \#t_2) \vee (\#t_1 \wedge t_2),$$

где  $\#t \equiv \text{pNOT}(t)$



26

## Свойства рXOR

Для XOR:

$$a \oplus b = b \oplus a \quad (1)$$

$$a \oplus 0 = a \quad (2)$$

$$a \oplus a = 0 \quad (3)$$

$$a \oplus 1 = \#a \quad (4)$$

$$(a \oplus b) \oplus b = a \quad (5)$$

$$\#a \oplus b = \#b \oplus a \quad (6)$$

$$(a \oplus b) c = ac \oplus bc \quad (7)$$

Для рXOR:

Свойства справедливы и для рXOR, если все операции отрицания – постбинарные **рNOT**

27

## Подведение ИТОГОВ

## Выводы

1. Проведен синтез ВСЕХ ранее определенных операций тетралогии.
2. Реализована обобщающая операция  $pNOT$ .
3. Реализована операция  $pXOR$  с сопоставлением свойств операции бинарной  $XOR$ .
4. Показаны законы операций сдвига и инверсии.
5. Выполнено сопоставление унарных операций инверсной и сдвиговой групп.

### Важный момент

**Назначение символов  
для обозначения постбинарных логических операций  
(составить таблицу соответствия)**

29

## для заметок





ДонНТУ

# Синтез блока сокращения разрядности порядка

К вопросу аппаратной реализации арифметики чисел с плавающей запятой в постбинарных форматах произвольной точности

07/11/2012

Аспирант кафедры КИ: **Иваница С.В.**

2

## Постановка задачи

1. С учетом принципов сокращения разрядности порядков **разработать блок**, выполняющий сокращение порядка-результата (32 тетрита или 64 двоичных бита) с анализом возможности «упаковки» порядка начиная с формата наименьшей точности;
2. Возможность сокращения порядка в одном блоке **для форматов различной точности**;
3. Учитывать **арбитраж сокращения**: при возможности сокращать к порядку формата меньшей разрядности (меньшего класса точности);
4. Осуществить **возможность внешнего управления** для получения определенной разрядности порядка (зависимость от блока сокращения разрядности манитессы);
5. Представлять порядки тетракодов **двоичными** (один тетрит – два бита).

3

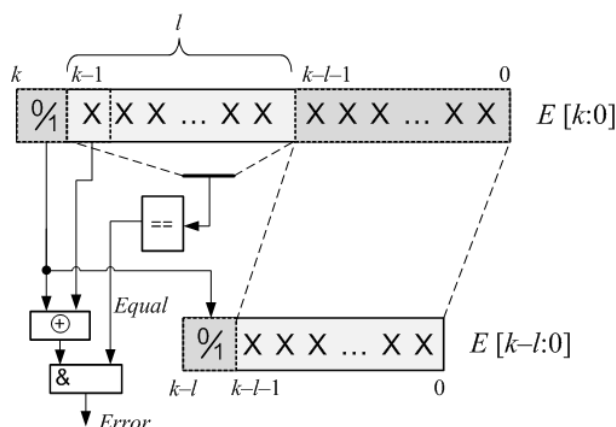
## Обобщенная схема сокращения разрядности порядка

Для постбинарных форматов:  
 $0, 1 \in T$  (тетракод).

Для бинарных форматов:  
 $0, 1 \in B$  (бинарный код).

Сокращение порядка на  $l$  разрядов:

$$E[k:0] \rightarrow E[k-l:0], (l < k)$$



При **Equal** = 1 – равные значения поля тетритов (т.е. область сокращения состоит из одинаковых значений).

При **Error** = 1 – сокращение некорректно!! (т.е. старший (знаковый) тетрит не отличается от значений сокращаемого поля).

Исключение – значение нуля: все разряды (и мантиссы в том числе) – тетранули.

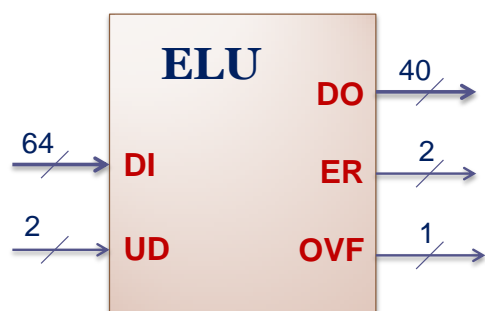
4

## Алгоритм сокращения разрядности порядка

1. Занесение значения старшего разряда большего порядка в старший разряд меньшего порядка.
2. Выделение поля значений перед старшим разрядом большего порядка с разрядностью, равной разнице разрядностей рассматриваемых порядков.
3. В случае равенства всех значений указанного поля, отличных от старшего разряда большего порядка (сокращение корректно) – заполнение оставшихся разрядов меньшего порядка значениями аналогичных разрядов старшего разряда. В иных случаях (значения старшего разряда и поля совпадают или значения поля не равны) – сокращение порядков некорректно.

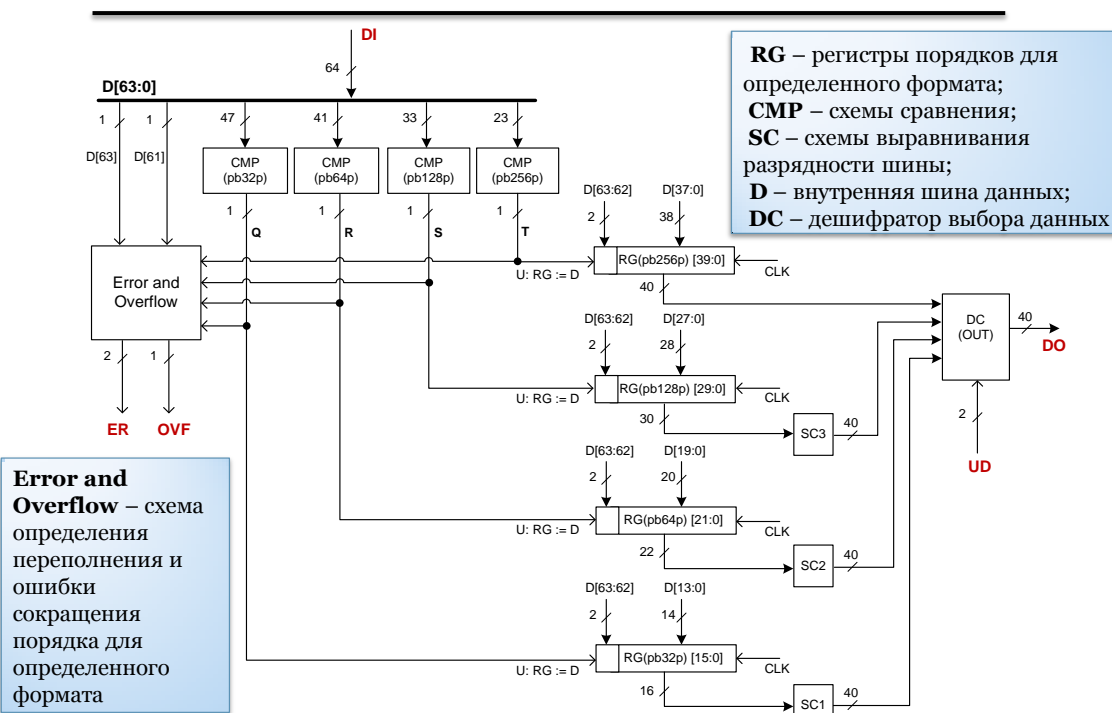
## Условно-графическое обозначение схемы блока Exponent Lower Unit

Все разряды – битовые эквиваленты тетрады  
(порядок представлен в тетракаде)



**DI** – входная шина (выход блока обработки порядков);  
**DO** – выходная шина для сокращенного порядка (к регистрам хранения результата);  
**UD** – сигналы управления дешифратором вывода;  
**ER** – сигнал ошибки вывода порядка для определенного формата;  
**OVF** – сигнал переполнения (overflow) или ошибки сокращения порядков (при невозможности вывода порядка для всех форматов).

## Схема ELU



7

## Схема сравнения СМР

### СМР – схема сравнения на равенство для тетракода

Для реализации **схем сравнения** (в том числе на равенство) для двух тетритов, принимающих значения тетрануля и тетраединицы и представленных парами бит, можно рассматривать каждый нечетный бит тетракода (старший в паре бит, кодирующей один тетрит). В этом случае схемы сравнения для тетракодов реализуются как и для двоичных кодов.

### Схема сравнения на равенство двух «бинарных» тетракодов:

$$t_i, t_{i+1} \in T(0, 1) \quad b_0, b_1 \in B$$

$E_{cv} = 1$  – при равных значениях тетритов:

$(b_1)t_i$	$(b_1)t_{i+1}$	$E_{cv}$
0	0	1
0	1	0
1	0	0
1	1	1

$$E_{cv_i} = \overline{(b_1)_{t_i} \oplus (b_1)_{t_{i+1}}}$$

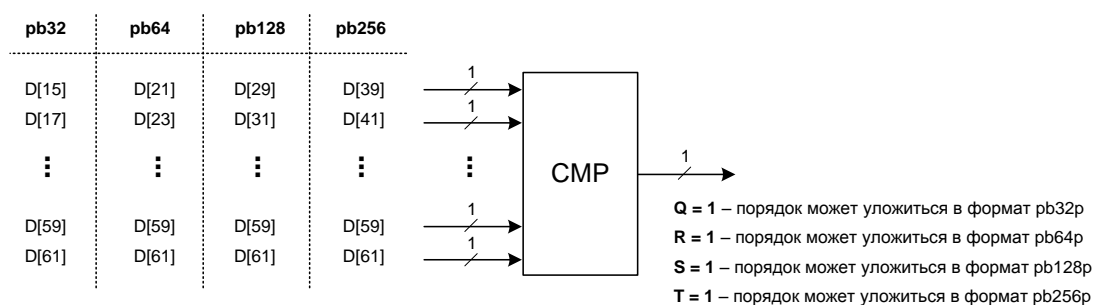
$$\text{Для всего тетракода: } E_{cv} = \bigwedge_{i=1}^{n-1} E_{cv_i}$$

8

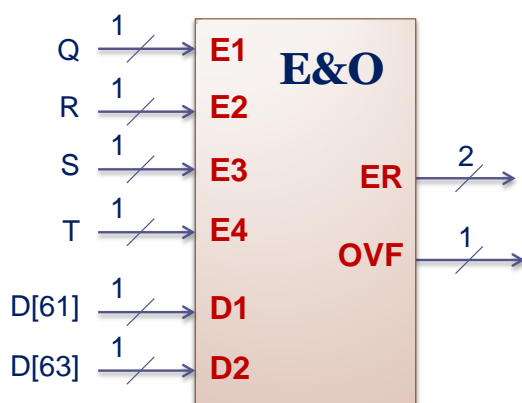
## Реализация СМР

### Обобщенная схема

Для каждой схемы свое количество разрядов (47, 41, 33, 23) и собственный выходной сигнал Q, R, S, T.



## Блок Error & Overflow



**E1-E4** – входы для сигналов возможности сокращения (1 – сокращение возможно, 0 – сокращение невозможно);  
**D1-D2** – входы для проверки на неравенство;  
**ER** – сигнал ошибки вывода порядка для определенного формата;  
**OVF** – сигнал переполнения (overflow) или ошибки сокращения порядков (при невозможности вывода порядка для всех форматов).

## Таблица истинности и уравнения для сигналов блока Error & Overflow

	32	64	128	256			
nE	Q	R	S	T	ER[1]	ER[0]	OVF
1	1	1	1	1	0	0	0
1	0	1	1	1	0	1	0
1	0	0	1	1	1	0	0
1	0	0	0	1	1	1	0
1	0	0	0	0	*	*	1
0	*	*	*	*	*	*	1

**nE** – сигнал не равенства D1 и D2, т.е. если **nE** = 1, то **D1 ≠ D2**:

$$nE = \overline{E_{cv}} = D_1 \oplus D_2$$

$$nE = D[61] \oplus D[63]$$

**ER = 0h** – можно сокращать во все форматы;

**ER = 1h** – можно сокращать во все, кроме **одного** младшего формата;

**ER = 2h** – можно сокращать во все, кроме **двух** младших форматов;

**ER = 3h** – можно сокращать во все, кроме **трех** младших форматов;

$$ER[1] = \bar{R}$$

$$ER[0] = \bar{Q} \& R + \bar{S}$$

$$OVR = \bar{nE} + \bar{T} = \overline{D[61] \oplus D[63]} + \bar{T}$$



## Порядок выдачи данных

1. Опрос сигнала **OVR**. Если  $OVR = 1$ , то фиксация ошибки вычислений и останов.
2. Если  $OVR = 0$  – опрос сигнала **ER**.
3. Сопоставление значения **ER** с **аналогичным сигналом** в блоке сокращения разрядности мантисс.
4. В результате сопоставления – **формирование сигнала UD** для выборки необходимого порядка:

UD[1]	UD[0]	Операция
0	0	<b>DO := RG(pb32)</b>
0	1	<b>DO := RG(pb64)</b>
1	0	<b>DO := RG(pb128)</b>
1	1	<b>DO := RG(pb256)</b>

## Выводы

1. Рассмотрены принципы сокращения порядка и алгоритм работы блока сокращения разрядности.
2. Разработана общая схема блока сокращения разрядности. Получены схемы элементов и необходимые зависимости.
3. Соблюдены необходимые требования, позволяющие согласовать работу данного блока и блока сокращения разрядности мантиссы.
4. Учтены исключительные ситуации, например в случае получения нулевого значения.
5. Синтез данного блока является завершающим этапом в разработке блока обработки порядков для постбинарных форматов.



18-21 СЕНТЯБРЯ

IT-INDUSTRY 2012

ФОРУМ «ИНДУСТРИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ 2012»

# Подготовка IT- профессионалов в украинских университетах:

## 50-летний опыт и следующие 50 лет

**Аноприенко Александр Яковлевич**

Декан факультета  
компьютерных наук и технологий  
Донецкого национального технического университета (ДонНТУ)  
Директор Технопарка ДонНТУ УНИТЕХ



**Аноприенко  
Александр Яковлевич**

Подготовка IT-  
профессионалов



18-21 СЕНТЯБРЯ

IT-INDUSTRY 2012

ФОРУМ «ИНДУСТРИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ 2012»

Начало традиции форумов  
индустрии информационных технологий  
в главном регионе традиционной индустрии  
можно считать символическим началом  
реального **вступления Украины в информационное общество...**

### «Информационное общество»

- это «общество знания, в котором  
**главным условием благополучия  
каждого человека становится знание,**  
полученное благодаря  
беспрепятственному доступу к  
информации и умению работать»





Аноприенко  
Александр Яковлевич

Подготовка IT-  
профессионалов



18-21 СЕНТЯБРЯ  
**IT-INDUSTRY 2012**  
ФОРУМ «ИНДУСТРИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ 2012»

## «Информационное общество»:

### Мировые тенденции и «настроения»

Университеты являются одной из важнейших составляющих процесса формирования информационного общества, **залогом его успешности** и одним из главных источников соответствующих изменений.



Университеты должны стать не только поставщиком кадров и образовательных услуг, но и **местом общения и передачи знаний**, местом где интенсивно рождается **новое знание**, место, где знания инициируют **инновационные процессы**.



## IT-конференция «i-Community»

### Начало подготовки в области IT в ДонНТУ:



**12 апреля 1961 года**

Инициатором начала IT-подготовки для студентов ДонНТУ выступил заведующий кафедры «Автоматика и телемеханика» **С.Р. Буачидзе – выпускник Каннского политехнического института (Франция)**, изначально заложивший ориентацию на европейские стандарты в IT-образовании в ДонНТУ



С. Р. Буачидзе



А.Я. Аноприенко, В.А. Святный



Вычислительная техника  
и информатика в ДонНТУ:  
люди, события, факты  
Первые 50 лет

Обчислювальна техніка  
та інформатика в ДонНТУ:  
люди, події, факти.  
Перші 50 років

Computer Engineering & Computer Science  
at DonNTU: People, Facts, & Events  
The First 50 Years



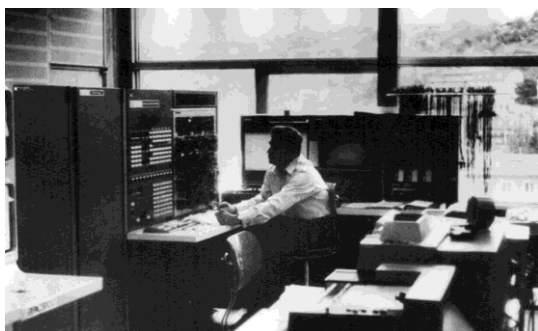
**12 апреля 1961 года — Т О В А Р И Щ,  
ЗАПОМНИ ЭТОТ ДЕНЬ!**



Universität Stuttgart

Опыт европейского сотрудничества в  
области подготовки IT-специалистов

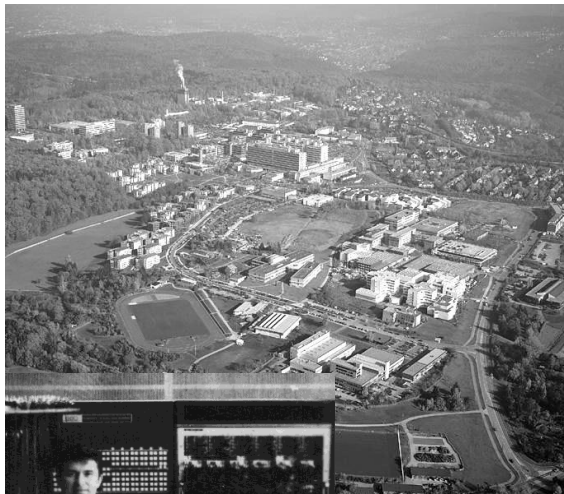
Научное и образовательное **сотрудничество ДонНТУ и Штуттгартского университета** в области информационно-компьютерных технологий **началось в 1974 году** с 10-месячной стажировки декана факультета вычислительной техники и автоматизированных систем управления (ВТ и АСУ) **В.А. Святого** в Институте системной динамики и управляющей техники (ISR)



Партнеры



Universität Stuttgart



**С 1989 года**  
систематический  
обмен визитами и  
студентами



Universität Stuttgart

Опыт европейского сотрудничества в  
области подготовки IT-специалистов

## Hermit Fires Up as Fastest Civil Supercomputer in Europe

Партнеры

By Rich Brueckner | Published February 24, 2012



Universität Stuttgart



Директор HLRS  
профессор  
Михаэль Реш  
— почетный  
доктор ДонНТУ



В феврале 2012 года в Штуттгартском центре суперкомпьютерных вычислений (HLRS) введен в строй **самый мощный в Германии** (второй в Европе) **суперкомпьютер Cray XE6** (производительность 1 петафлопс). В 2012 году производительность будет доведена до 5 петафлопс. Это **самый мощный компьютер в Европе** из числа тех, которые используются в гражданских и промышленных целях.

6





Universität Stuttgart

Опыт европейского сотрудничества в  
области подготовки IT-специалистов

## Партнеры



**В 2010 году в ДонНТУ введен в эксплуатацию первый суперкомпьютер NEC Xeon Linux Cluster** (3-й в Украине, 4 Тфлопса, в 2006-2008 гг. входил в Top500 мира и Top100 Европы), **переданный суперкомпьютерным центром Штуттгартского университета** для использования в совместных исследованиях и учебном процессе для подготовки IT-специалистов

7



Universität Stuttgart

Опыт европейского сотрудничества в  
области подготовки IT-специалистов

**Примеры прочих результатов сотрудничества  
ДонНТУ и Штуттгартского университета  
в подготовке IT-специалистов:**

**Штуттгартский университет**

**ДонНТУ**



В 1993 году введен в эксплуатацию **первый в Европе университетский веб-сервер**



С конца 90-х годов лидирующие позиции в Германии в подготовке **IT-специалистов на английском языке**

В 1996 году введен в эксплуатацию **первый в Донбассе университетский веб-сервер**

С 2000 года развивается **портал магистров ДонНТУ** – один из наиболее посещаемых университетских веб-ресурсов постсоветского пространства

С 90-х функционирует уникальный английский технический факультет для подготовки **IT-специалистов на английском языке**

8





Universität Stuttgart

Опыт европейского сотрудничества в  
области подготовки IT-специалистов

### Потенциал ДонНТУ в подготовке IT-специалистов:

После объединения с Государственным университетом информатики и искусственного интеллекта в 2011 году 10 кафедр ДонНТУ готовят IT-специалистов практически по всем основным направлениям подготовки



**IT-студенты – это почти треть всех студентов ДонНТУ, принятых в на 1-й курс в 2011 году, в том числе:**

Компьютерная инженерия – более 100

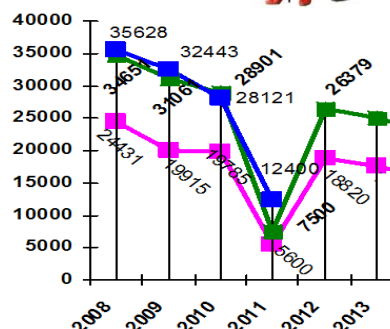
Компьютерные науки – около 100

Программная инженерия – 100

...

Магистры факультета

компьютерных наук и технологий – более 100



Есть **достаточный потенциал** для дальнейшего роста объемов подготовки

9

Аноприенко  
Александр ЯковлевичБудущее IT-индустрии  
в Донецке и Украине

### IT индустрия:

1. **Высокотехнологическая и высокорентабельная отрасль экономики.**
2. **Не требует от государства капитальных вложений, инвестиций, использования природных ресурсов.**
3. **Содействует реализации высокого образовательного, научного, интеллектуального и технологического потенциала страны.**



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### Острый дефицит IT-специалистов:

Исходя из прогнозных расчетов темпов роста украинской IT-индустрии (35–40% в год, что наблюдается уже с 2011 года), **к 2015 году появится более 150-ти тысяч новых IT-вакансий**, из них **не менее 100 тыс.** в секторе разработки программного обеспечения на экспорт.

**Дефицит IT-специалистов при этом составит не менее 90 тыс. человек**

Фактически это означает, что **выпуск украинскими вузами специалистов в области информационно-компьютерных технологий должен возрасти многократно** за довольно небольшой промежуток времени.



Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

### **IT-кластер в Донецке – это необходимо для всего индустриального Донбасса!!!**

Роль государства здесь трудно переоценить и она должна заключаться прежде всего в создании условий для **формирования и развития так называемых образовательных IT-кластеров** в тех городах, где есть **вузы с достаточно масштабным и высоким уровнем подготовки в области IT-специальностей** уже преподаются на высоком уровне.

**«К числу этих городов** традиционно относятся Киев, Харьков, Львов, а в последнее время – **и Донецк»**





Аноприенко  
Александр Яковлевич

Будущее IT-индустрии  
в Донецке и Украине

## ИТ и образование – главные условия инновационного развития Украины и Донбасса

В случае целенаправленных и согласованных усилий государства, бизнеса и образования у Украины **есть реальный шанс после 2015 года стать лидером в сфере информационных технологий** и достичь в последующие годы показателей ИТ-экспорта на уровне до 10 млрд. долл. в год, т.е. **увеличить высокотехнологичный экспорт практически на порядок** по сравнению с тем, что мы имеем сегодня.

Украина по потенциалу своего среднего и высшего образования входит в десятку мировых лидеров, а при надлежащей поддержке своего главного национального богатства – системы образования, ориентированной на чрезвычайно талантливый и изобретательный народ – **в состоянии войти к 2015 году в пятерку ведущих мировых образовательных лидеров**



Аноприенко  
Александр Яковлевич

Подготовка IT-  
профессионалов




18-21 СЕНТЯБРЯ  
**IT-INDUSTRY 2012**  
ФОРУМ «ИНДУСТРИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ 2012»

**2012, июль: впервые**  
3 украинских университета  
**в тысяче лучших**  
по интегральным показателям

**RANKING WEB  
OF UNIVERSITIES**

### Ukraine

ranking	World Rank	University
1	713	<a href="#">National Technical University of Ukraine Kyiv (Kiev) Polytechnic Institute</a>
2	973	<a href="#">National Taras Shevchenko University of Kyiv (Kiev)</a>
3	982	<a href="#">Donetsk National Technical University</a>
4	1469	<a href="#">Kharkov National University VN Karazin</a>
5	1703	<a href="#">Lviv Polytechnic National University</a>



**Аноприенко**  
Александр Яковлевич

Подготовка IT-профессионалов

18-21 СЕНТЯБРЯ  
**IT-INDUSTRY 2012**  
ФОРУМ «ИНДУСТРИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ 2012»

**2012: впервые**  
3 украинских университета  
**в тысяче лучших**

**RANKING WEB  
OF UNIVERSITIES**

July 2012

**Rank of Universities of Ukraine**

First | Previous | **Next** | Last | Universities 1 to 50 of 321

WORLD RANK	UNIVERSITY	POSITION			
		PRESENCE	IMPACT	OPENNESS	EXCELLENCE
712	National Technical University of Ukraine Kyiv (Kiev) Polytechnic Institute	962	934	62	2,389
972	National Taras Shevchenko University of Kyiv (Kiev)	725	2,508	544	1,421
981	Donetsk National Technical University	1,828 ↓	1,155 ↑	193 ↑	2,869 ↓
1469	Kharkov National University VN Karazin	3,965	3,955	359	1,697
1704	Lviv Polytechnic National University	858	4,686	381	3,150
1710	National Pedagogical University MP Dragomanova	327	1,725	1,243	5,228
1811	Nezhinskii State University Nikolai Gogol	69	3,698	648	5,228



**Посвящение в студенты факультета КНТ**

**RANKING WEB  
OF UNIVERSITIES**



Крупнейший Всемирный рейтинг вузов Вебометрикс

**ДонНТУ в августе 2012 года:**

- 1. Впервые вошел в Топ-1000 лучших вузов мира (среди 20-ти тысяч ведущих вузов мира)**
- 2. Впервые вошел в Топ-500 лучших вузов Европы (среди 5-ти тысяч европейских вузов)**
- 3. Впервые вошел в Топ-100 восточно-европейских университетов.**
- 4. Вошел в тройку лучших университетов Украины**



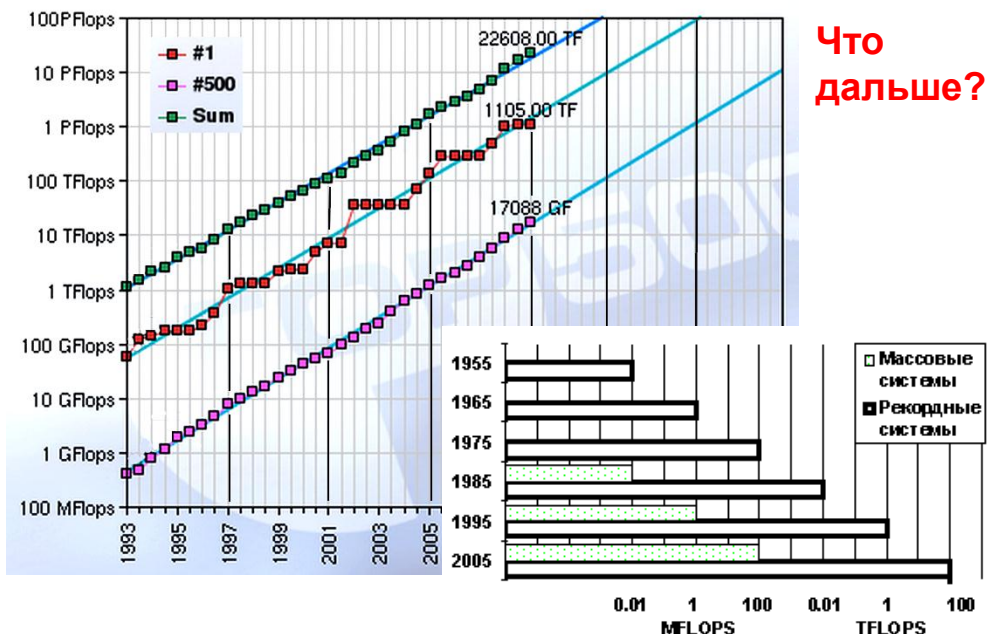


Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии  
в прошлом, настоящем и будущем

1996: 1 ТФлопс

2005: 100 ТФлопс



Аноприенко  
Александр Яковлевич

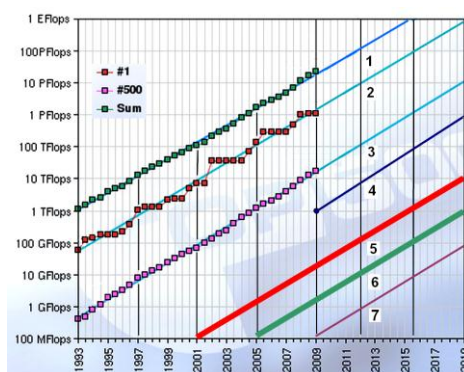
Компьютерные науки и технологии:  
следующие 50 лет

Суперкомпьютерная гонка и экзамасштаб  
(Е-масштаб):

**$10^{18}$  к 2018 году**

**Миллиард миллиардов**

операций с плавающей запятой в секунду



Prefixes

Kilo –  $10^3$

Mega –  $10^6$

Giga –  $10^9$

Tera –  $10^{12}$

Peta –  $10^{15}$

Exa –  $10^{18}$

Zetta –  $10^{21}$



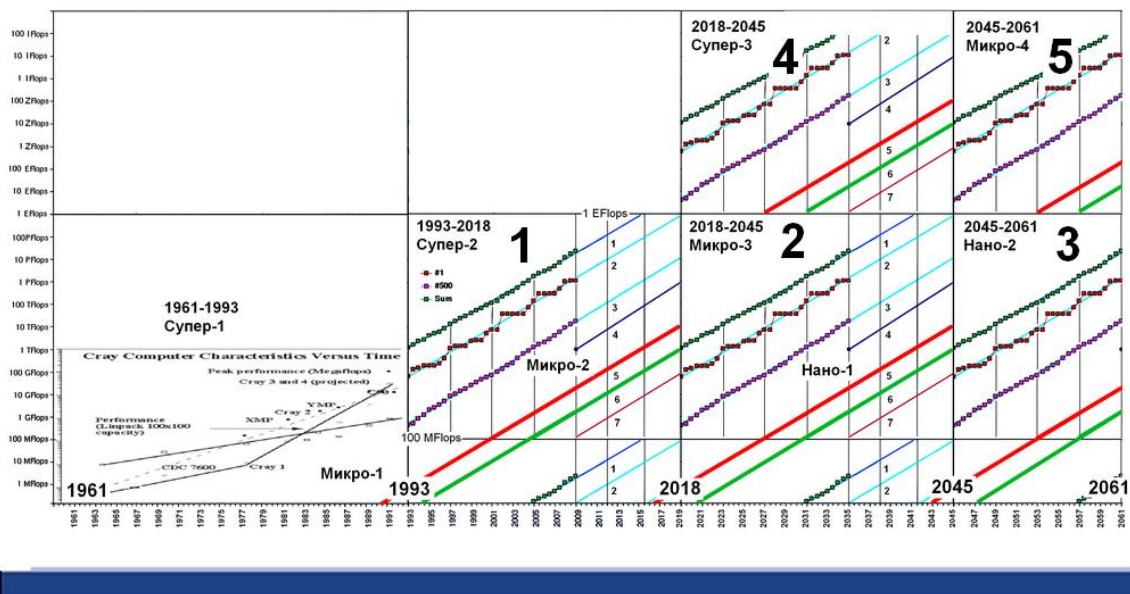


Аноприенко  
Александр Яковлевич

Компьютерные науки и технологии:  
следующие 50 лет

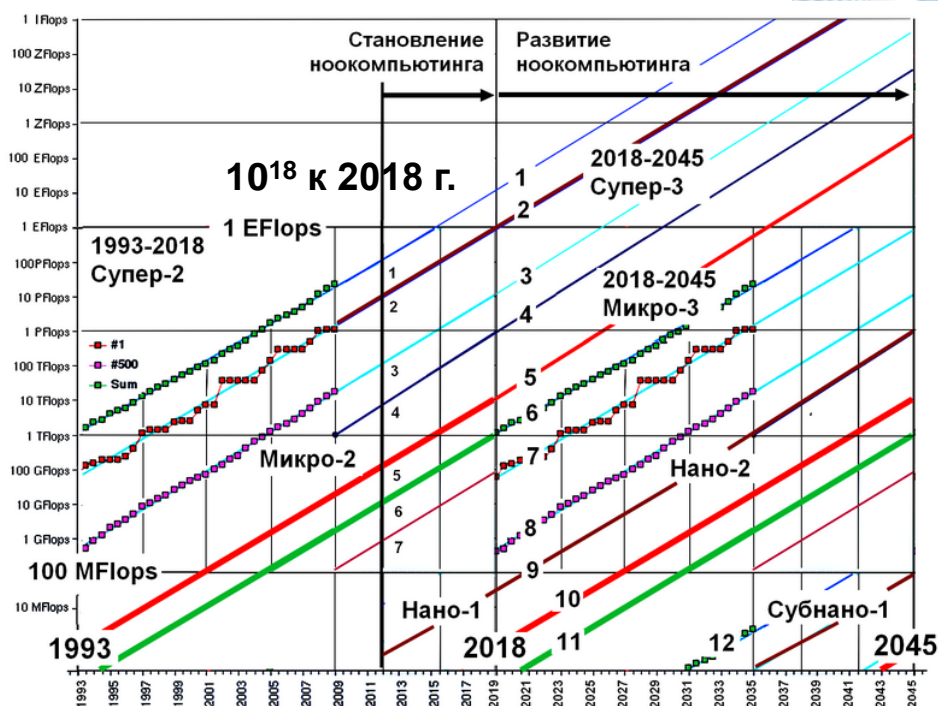
1961-2011-2061 !!!

**Периодическая таблица** на следующие 50 лет:  
3 потока (супер, микро, нано), 5 основных прогнозных блоков



Аноприенко  
Александр Яковлевич

Ноокомпьютинг:  
следующие 50 лет



**для заметок****для заметок**

## **ЗАКЛЮЧЕНИЕ**

Проведенные в 2010–2012 гг. исследования позволили получить следующие основные результаты:

1. Обоснованы актуальность и закономерный характер перехода к постбинарному компьютерному как следующему этапу в развитии компьютерных технологий.
2. Проанализированы закономерности эволюции технических средств компьютерного за последние 50 лет, на базе чего сформирован прогноз на следующие 50 лет, позволяющий, в частности, прогнозировать новый уровень интеллектуализации окружающей среды, который может быть обозначен как ноокомпьютер, а также – появление и динамику новых классов компьютеров.
3. Проанализирована эффективность интервальных вычислений на базе различных вариантов современного программного обеспечения и рассмотрены возможности их дальнейшего развития, в первую очередь, в контексте постбинарного компьютерного.
4. Детально разработаны и исследованы правила выполнения логических операций тетралогии.
5. Исследованы и проанализированы недостатки стандартных форматов представления действительных чисел (представление чисел с плавающей запятой в соответствии со стандартом IEEE 754–2008).
6. Предложены и исследованы постбинарные форматы точного представления действительных чисел, позволяющие контролировать текущую точность и в соответствии с этим гибко управлять разрядностью представления чисел с плавающей запятой.
7. Разработаны правила преобразования бинарных форматов в постбинарные и наоборот.
8. Разработаны алгоритмы выполнения постбинарных арифметических операций, в том числе такие, которые обеспечивают гибкую разрядность компьютерного

представления чисел в соответствии с реальной необходимостью.

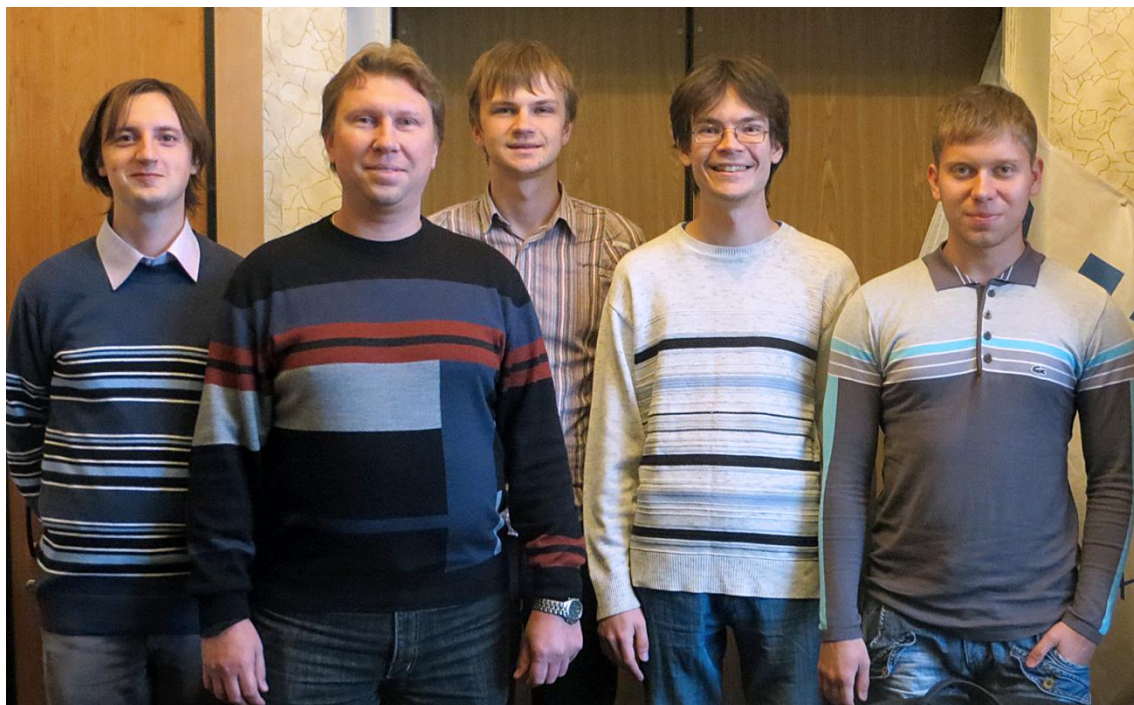
9. Разработано программное обеспечение, реализующее бинарные и постбинарные вычисления с гибкой разрядностью и позволяющее провести комплексные исследования эффективности постбинарного представления чисел.
10. Предложена концепция постбинарного процессора и/или сопроцессора и синтезированы соответствующие схемные решения.
11. Предложены и обоснованы такие новые компьютерные инструменты (и связанные с ними направления исследований) как ноограммы, ноографика и ноомоделирование, которые в совокупности с постбинарным компьютерингом и другими, связанными с ними направлениями, могут расцениваться как основные составляющие ноокомпьютинга.
12. Проанализированы процессы усложнения современной техногенной среды и насыщения ее компьютерными средствами и программным кодом, на основании чего сформулированы задачи в области компьютерных исследований и разработок, совершенствования инфраструктуры и подготовки кадров в области компьютерных наук и технологий на региональном и национальном уровнях.

В ходе исследований 2010–2012 гг. авторами опубликовано и подготовлено к печати в общей сложности 48 публикаций, перечень которых приведен далее в разделе «Список публикаций». Тексты большинства из этих работ представлены в электронном архиве ДонНТУ, проиндексированы ведущими поисковыми системами и, соответственно, постоянно доступны широкому кругу ученых, преподавателей и студентов для ознакомления и эффективной работы с ними. Авторы надеются, что эти материалы в совокупности с данной монографией будут способствовать росту интереса к постбинарному компьютерингу и расширению круга его исследователей и разработчиков.





**Главное, что объединяет участников семинара — любовь к науке!**  
Жажда новых познаний, обсуждение мировых научных достижений и стремление к собственным открытиям создает непередаваемую атмосферу «научного потенциала» на семинарах факультета КНТ



**Счастливые лица докладчиков.**  
На каждом семинаре минимум пять докладов!






**Положительные отзывы, теплые слова, а главное — улыбки на лицах участников семинара, позволяют говорить об успешности данной формы работы со студентами, магистрантами и аспирантами факультета КНТ**




**Семинар «в самом разгаре».**  
Дискуссии, доклады, рекомендации и обмен опытом

**для заметок**




A vertical yellow bar on the left side of the page contains a blue circle and a small white dot. The page is ruled with horizontal blue lines. A thick dark red line is positioned below the title.

**для заметок**




A vertical yellow bar on the left side of the page contains a blue circle and a small white dot. The page is ruled with horizontal blue lines. A thick dark red line is positioned below the title.

**для заметок**



A vertical yellow bar on the left side of the page contains a blue circle and a small white dot. The page is ruled with horizontal blue lines.

**для заметок**



A vertical yellow bar on the left side of the page contains a blue circle and a small white dot. The page is ruled with horizontal blue lines.

## СПИСОК ПУБЛИКАЦИЙ

1. Аноприенко А. Я., Иваница С. В. Постбинарный компьютеринг и интервальные вычисления в контексте кодологической эволюции. — Донецк, ДонНТУ, УНИТЕХ, 2011. — 248 с.  
<http://ea.donntu.edu.ua/handle/123456789/7544>
2. Аноприенко А. Я., Святный В. А. Вычислительная техника и информатика в ДонНТУ: люди, события, факты. Первые 50 лет. — Донецк: ДонНТУ, УНИТЕХ, 2011. — 264 с., ил.  
<http://ea.donntu.edu.ua/handle/123456789/2812>
3. Аноприенко А. Я. Программная модель представления чисел в постбинарных форматах / А. Я. Аноприенко, С. В. Иваница, Е. И. Котов // Искусственный интеллект. Интеллектуальные системы ИИ-2012: материалы Международной научно-технической конференции (пос. Кацивели, АР Крым, 1–5 октября 2012 года). Донецьк: ІПШ «Наука і освіта», 2012. — С. 9–12.  
[http://www.iai.donetsk.ua/conf/2012/AI\\_2012\\_tez.pdf](http://www.iai.donetsk.ua/conf/2012/AI_2012_tez.pdf)
4. Аноприенко А. Я. Аппаратная реализация преобразователя чисел в постбинарный формат / А. Я. Аноприенко, С. В. Иваница, С. В. Кулибаба // Искусственный интеллект. Интеллектуальные системы ИИ-2012: материалы Международной научно-технической конференции (пос. Кацивели, АР Крым, 1–5 октября 2012 года). Донецьк: ІПШ «Наука і освіта», 2012. — С. 13–16.  
[http://www.iai.donetsk.ua/conf/2012/AI\\_2012\\_tez.pdf](http://www.iai.donetsk.ua/conf/2012/AI_2012_tez.pdf)
5. Иваница С. В. Методы округления чисел с плавающей запятой, представленных в постбинарных форматах / С. В. Иваница // Искусственный интеллект. Интеллектуальные системы ИИ-2012: материалы Международной

научно-технической конференции (пос. Кацивели, АР Крым, 1–5 октября 2012 года). Донецьк: ІППІ «Наука і освіта», 2012. — С. 33–36.

[http://www.iai.donetsk.ua/conf/2012/AI\\_2012\\_tez.pdf](http://www.iai.donetsk.ua/conf/2012/AI_2012_tez.pdf)

6. Аноприенко А. Я. Интервальный анализ и его применение при расчетах параметров серверных компьютерных систем / А. Я. Аноприенко, С. В. Иваница, Хамза Аль Рабаба // Научный журнал «Радіоелектроніка, інформатика, управління». — Запорожье, 2012.
7. Аноприенко А. Я. Будущее IT-индустрии в Донецке и Украине: вызовы, возможности и перспективы // Донбасс-2020: Материалы VI научно-практической конференции. Донецк, 24–26 апреля 2012 г. — Донецк, Донецкий национальный технический университет, 2012. С. 18–27.
8. Анопрієнко О. Я. Принцип роботи, структура і моделювання блоку перетворювача форматів у складі постбінарного співпроцесора / О. Я. Анопрієнко, С. В. Іваниця, С. В. Кулібаба // Міжнародний науково-технічний журнал «Інформаційні технології та комп'ютерна інженерія». — Вінниця, 2012.
9. Аноприенко А. Я. Преобразователь вещественных десятичных чисел в постбинарные форматы с плавающей запятой / А. Я. Аноприенко, С. В. Иваница, Е. И. Котов // Научные труды Донецкого национального технического университета. Серия «Проблемы моделирования и автоматизации проектирования динамических систем» (МАП-2012). Донецк: ДонНТУ, — 2012.
10. Иваница С. В. Оценка погрешности представления вещественных чисел в постбинарных форматах с плавающей запятой / С. В. Иваница // Научно-теоретический журнал «Искусственный интеллект», № 4. — Донецк, 2012. С. 32–44.

[http://www.iai.donetsk.ua/conf/2012/AI\\_4\\_2012.pdf](http://www.iai.donetsk.ua/conf/2012/AI_4_2012.pdf)



11. Иваница С. В. Поддержка динамической разрядности постбинарных форматов чисел с плавающей запятой / С. В. Иваница, С. В. Кулибаба, А. Я. Анопrienко // «Информатика и компьютерные технологии», сборник трудов VIII международной научно-технической конференции студентов, аспирантов и молодых ученых — 18–19 сентября 2012 г., Донецк, ДонНТУ. — 2012. В 2-х томах, Т. 2. — С. 74–78.  
<http://csconf.donntu.edu.ua/arxiv/>
12. Иваница С. В. Оценка погрешности представления вещественных чисел в постбинарных форматах с плавающей запятой / С. В. Иваница, Е. И. Котов, А. Я. Анопrienко // «Информатика и компьютерные технологии», сборник трудов VIII международной научно-технической конференции студентов, аспирантов и молодых ученых — 18–19 сентября 2012 г., Донецк, ДонНТУ. — 2012. В 2-х томах, Т. 2. — С. 46–50.  
<http://csconf.donntu.edu.ua/arxiv/>
13. Анопrienко А. Я., Иваница С. В., Аль Рабаба Хамза Анализ времени выполнения арифметических операций над интервальными числами в СКА Mathematica / А. Я. Анопrienко, С. В. Иваница, Хамза Аль Рабаба // Научные труды Донецкого национального технического университета. Серия: «Информатика, кибернетика и вычислительная техника» (ИКВТ-2012). — Донецк: ДонНТУ, 2012.
14. Анопrienко А. Я. Вызовы и перспективы развития высшего образования в области компьютерных наук и технологий // Материалы III всеукраинской научно-технической конференции «Информационные управляющие системы и компьютерный мониторинг (ИУС и КМ 2012)» — 17–18 апреля 2012 г., Донецк, ДонНТУ, 2012. С. 12–15.  
<http://ea.donntu.edu.ua/handle/123456789/14483>
15. Анопrienко А. Я. Гибкая разрядность и постбинарные форматы представления вещественных чисел / А. Я. Анопrienко, С. В. Иваница // Вестник Инженерной

Академии Украины. Теоретический и научно-практический журнал Инженерной Академии Украины. Выпуск 1. — Киев, 2012. — С. 92–98.

<http://ea.donntu.edu.ua:8080/jspui/handle/123456789/14490>

16. Иваница С. В. Преобразователь вещественных чисел с регулируемой точностью / «Информационные управляющие системы и компьютерный мониторинг – 2012» (ИУС КМ – 2012) / С. В. Иваница, С. В. Кулибаба, А. Я. Аноприенко // Материалы III Всеукраинской научно-технической конференции студентов, аспирантов и молодых ученых — 16–18 апреля 2012 г. Донецк, ДонНТУ. — 2012. С. 621–624.
17. Аноприенко А. Я., Иваница С. В. Представление интервальных чисел средствами постбинарного кодирования // Научно-теоретический журнал «Искусственный интеллект», № 1, 2012. С. 6–16.
18. Аноприенко А. Я., Иваница С. В. Особенности представления вещественных чисел в постбинарных форматах // Научно-теоретический журнал «Математичні машини і системи», № 3. — Киев, 2012. — С. 49–60.
19. Аноприенко А. Я., Иваница С. В., Бурлака Е. В. Программная реализация постбинарного кодирования интервалов // Научные труды Донецкого национального технического университета. Серия «Проблемы моделирования и автоматизации проектирования динамических систем» (МАП-2011). Выпуск 10 (197): Донецк: ДонНТУ, — 2011. С. 207–214.
20. Аноприенко А. Я. Университет в современном информационном пространстве: тенденции, рейтинги и опыт развития портала магистров ДонНТУ // Научные труды Донецкого национального технического университета. Серия: «Информатика, кибернетика и вычислительная техника» (ИКВТ-2011). Выпуск 13 (185). — Донецк: ДонНТУ, 2011. С. 224–235.

<http://ea.donntu.edu.ua/handle/123456789/2773>

21. Анопрієнко О. Я., Іваниця С. В., Соловей О. О. Розробка та дослідження серверних модулів для порталу моделювання ДонНТУ // Научные труды Донецкого национального технического университета. Серия «Проблемы моделирования и автоматизации проектирования динамических систем» (МАП-2011). Выпуск 10 (197): Донецк: ДонНТУ, — 2011. С. 183–190.  
<http://ea.donntu.edu.ua/handle/123456789/5828>
22. Аноприенко А. Я. Ноокомпьютинг // Материалы VI международной научно-технической конференции «Информатика и компьютерные технологии» — 22–23 ноября 2011 г. Т.1. Донецк, ДонНТУ. — 2011. С. 10–23.  
<http://ea.donntu.edu.ua/handle/123456789/2804>
23. Аль Абабнех Хасан, Иваниця С. В., Аноприенко А. Я. Расчет характеристик серверных компьютерных систем // Информатика и компьютерные технологии / Материалы VII международной научно-технической конференции студентов, аспирантов и молодых ученых — 22–23 ноября 2011 г. Т.1. Донецк, ДонНТУ. — 2011. С. 123–128.  
<http://csconf.donntu.edu.ua/arxiv/>
24. Меркулов А. В., Иваниця С. В., Аноприенко А. Я. Методы контроля стабильности корней многочленов с интервальными коэффициентами // Информатика и компьютерные технологии / Материалы VII международной научно-технической конференции студентов, аспирантов и молодых ученых — 22–23 ноября 2011 г. Т.1. Донецк, ДонНТУ. — 2011. С. 228–232.  
<http://csconf.donntu.edu.ua/arxiv/>
25. Кулибаба С. В, Иваниця С. В. Программная модель преобразователя вещественных чисел в постбинарные форматы // Информатика и компьютерные технологии / Материалы VII международной научно-технической конференции студентов, аспирантов и молодых ученых — 22–23 ноября 2011 г. Т.1. Донецк, ДонНТУ. — 2011. С. 214–219.  
<http://csconf.donntu.edu.ua/arxiv/>

26. Иваница С. В., Соловей О. О., Аноприенко А. Я. Разработка и исследование серверных модулей для портала моделирования ДонНТУ // Информатика и компьютерные технологии / Материалы VII международной научно-технической конференции студентов, аспирантов и молодых ученых — 22–23 ноября 2011 г. Т.1. Донецк, ДонНТУ. — 2011. С. 118–122.  
<http://ea.donntu.edu.ua/handle/123456789/3937>
27. Иваница С. В., Аль Рабаба Хамза, Аноприенко А. Я. Интервальный анализ и его применение в рамках традиционных и постбинарных вычислений // Информатика и компьютерные технологии / Материалы VII международной научно-технической конференции студентов, аспирантов и молодых ученых — 22–23 ноября 2011 г. Т.1. Донецк, ДонНТУ. — 2011. С. 249–255.  
<http://csconf.donntu.edu.ua/arxiv/>
28. Бурлака Е. В., Иваница С. В., Аноприенко А. Я. Программная модель отображения тетракода на множествах действительных и интервальных чисел // Информатика и компьютерные технологии / Материалы VII международной научно-технической конференции студентов, аспирантов и молодых ученых — 22–23 ноября 2011 г. Т.1. Донецк, ДонНТУ. — 2011. С. 336–342.  
<http://csconf.donntu.edu.ua/arxiv/>  
<http://ea.donntu.edu.ua/handle/123456789/4081>
29. Аноприенко А. Я., Иваница С. В., Кулибаба С. В. Представление постбинарных форматов чисел с плавающей запятой в контексте интервальных вычислений // Научные труды Донецкого национального технического университета. Серия: «Информатика, кибернетика и вычислительная техника» (ИКВТ-2011). Выпуск 14 (188). — Донецк: ДонНТУ, 2011. С. 44–49.  
[http://www.nbuu.gov.ua/portal/natural/Npdntu\\_ikot/2011\\_14/](http://www.nbuu.gov.ua/portal/natural/Npdntu_ikot/2011_14/)
30. Иваница С. В., Меркулов А. В. Методы контроля точности результирующих интервалов при вычислении интерваль-

ных полиномиальных функций // Научные труды Донецкого национального технического университета. Серия: «Информатика, кибернетика и вычислительная техника» (ИКВТ-2011). Выпуск 14 (188). — Донецк: ДонНТУ, 2011. С. 55–60.

[http://www.nbuu.gov.ua/portal/natural/Npdntu\\_ikot/2011\\_14/](http://www.nbuu.gov.ua/portal/natural/Npdntu_ikot/2011_14/)

31. Иваница С. В., Меркулов А. В. Особенности вычисления интервальных полиномов с контролем точности результирующих интервалов // Моделирование и компьютерная графика / Материалы IV международной научно-технической конференции — 5–8 октября 2011 г. Донецк, ДонНТУ. — 2011. С. 119–126.

<http://ea.donntu.edu.ua/handle/123456789/3021>

32. Аноприенко А. Я., Иваница С. В., Кулибаба С. В. Особенности представления постбинарных вещественных чисел в контексте интервальных вычислений и развития аппаратного обеспечения средств компьютерного моделирования // Моделирование и компьютерная графика / Материалы IV международной научно-технической конференции — 5–8 октября 2011 г. Донецк, ДонНТУ. — 2011. С. 13–19.

<http://ea.donntu.edu.ua/handle/123456789/2885>

33. Аноприенко А. Я., Иваница С. В. Особенности реализации постбинарных логических операций / А.Я. Аноприенко, С.В. Иваница // Научно-теоретический журнал «Искусственный интеллект», № 2, 2011. С. 110–121.

34. Аноприенко А. Я. Ноографика и ноомоделирование // Материалы четвертой международной научно-технической конференции «Моделирование и компьютерная графика» 5–8 октября 2011 года, Донецк, ДонНТУ, 2011. С. 321–324.

<http://ea.donntu.edu.ua/handle/123456789/2796>

35. Аноприенко А. Я. Компьютерные науки и технологии: следующие 50 лет // Материалы II всеукраинской научно-технической конференции «Информационные управляющие системы и компьютерный мониторинг (ИУС и КМ



2011)» — 12–13 апреля 2011 г., Донецк, ДонНТУ, 2011. Т.1. С. 7–22.

<http://ea.donntu.edu.ua/handle/123456789/3360>

36. Аноприенко А. Я. Университетская составляющая в формировании информационного общества в Украине и мире // Міжнародний науковий конгрес з розвитку інформаційно-комунікаційних технологій та розбудови інформаційного суспільства в Україні, м. Київ, 17–18 листопада 2011 р. Тези доповідей. С. 10–11.

<http://ea.donntu.edu.ua/handle/123456789/2797>

37. Аноприенко А. Я. Ноокомпьютинг и будущее информационно-компьютерной инфраструктуры // Міжнародний науковий конгрес з розвитку інформаційно-комунікаційних технологій та розбудови інформаційного суспільства в Україні, м. Київ, 17–18 листопада 2011 р. Тези доповідей. С. 12–13.

<http://ea.donntu.edu.ua/handle/123456789/2800>

38. Аноприенко А. Я. Будущее компьютерных технологий в контексте технической и кодо-логической эволюции // Вестник Инженерной Академии Украины. Теоретический и научно-практический журнал Инженерной Академии Украины. Выпуск 3–4, 2011, с. 108–113.

<http://ea.donntu.edu.ua/handle/123456789/3606>

39. Аноприенко А. Я., Гранковский В. А., Иваница С. В. Пример Румпа в контексте традиционных, интервальных и постбинарных вычислений // Научные труды Донецкого национального технического университета. Серия «Проблемы моделирования и автоматизации проектирования динамических систем» (МАП-2011). Выпуск 9 (179): Донецк: ДонНТУ, 2011. С. 324–343.

<http://ea.donntu.edu.ua/handle/123456789/1287>

40. Иваница С. В., Аноприенко А. Я. Реализация логических операций над элементами тетракодов / С. В. Иваница, А. Я. Аноприенко / «Інформаційні управляючі системи та комп'ютерний моніторинг» (ІУС КМ – 2011) // Матеріали

- II Всеукраинской научно-технической конференции студентов, аспирантов и молодых ученых — 11–13 апреля 2011 г. Т.2. Донецк, ДонНТУ. — 2011. С. 198–202.
41. Иваница С. В., Аноприенко А. Я. Особенности реализации операций тетралогии / С. В. Иваница, А. Я. Аноприенко // Научные труды Донецкого национального технического университета. Серия: «Информатика, кибернетика и вычислительная техника» (ИКВТ-2011). Выпуск 13 (185). — Донецк: ДонНТУ, 2011. С. 134–140.  
<http://ea.donntu.edu.ua:8080/jspui/handle/123456789/4143>
42. Иваница С. В., Меркулов А. В., Аноприенко А. Я. Интервальные вычисления в математических пакетах Scilab и Mathematica // Информатика и компьютерные технологии / Материалы VI международной научно-технической конференции студентов, аспирантов и молодых ученых — 23–25 ноября 2010 г. Т.1. Донецк, ДонНТУ. — 2010. С. 240–246.  
<http://csconf.donntu.edu.ua/wp-content/uploads/2011/10/>
43. Гранковский В. А., Иваница С. В., Аноприенко А. Я. Подходы к сложным вычислениям в различных математических пакетах // Информатика и компьютерные технологии / Материалы VI международной научно-технической конференции студентов, аспирантов и молодых ученых — 23–25 ноября 2010 г. Т.2. Донецк, ДонНТУ. — 2010. С. 135–140.  
<http://csconf.donntu.edu.ua/wp-content/uploads/2011/10/>
44. Аноприенко А. Я., Иваница С. В. Реализация интервальных вычислений средствами математического пакета SciLab с использованием интервального расширения Int4Sci // «Донбасс-2020: перспективы развития глазами молодых ученых»: Материалы V научно-практической конференции. Донецк, 25–27 мая 2010 г. — Донецк, ДонНТУ Министерства образования и науки, 2010. С. 629–633.
45. Аноприенко А. Я., Иваница С. В. Интервальные вычисления и перспективы их развития в контексте кодо-

логической эволюции // Научные труды Донецкого национального технического университета. Серия «Проблемы моделирования и автоматизации проектирования динамических систем» (МАП-2010). Выпуск 8 (168): Донецк: ДонНТУ, 2010. С. 150–160.

<http://ea.donntu.edu.ua/handle/123456789/991>

46. Аноприенко А. Я., Иваница С. В. Особенности постбинарного кодирования на примере интервального представления результатов вычислений по формуле Бэйли-Боруэйна-Плаффа / А. Я. Аноприенко, С. В. Иваница // Научные труды Донецкого национального технического университета. Серия: «Информатика, кибернетика и вычислительная техника» (ИКВТ-2010). Выпуск 11 (164). — Донецк: ДонНТУ, 2010. С. 19–23.

[http://nbuv.gov.ua/portal/natural/Npdntu\\_ikot/2010\\_11/1\\_3.pdf](http://nbuv.gov.ua/portal/natural/Npdntu_ikot/2010_11/1_3.pdf)

<http://ea.donntu.edu.ua/handle/123456789/2823>

47. Аноприенко А. Я. Вызовы времени и постбинарный компьютеринг // Информатика и компьютерные технологии / Материалы VI международной научно-технической конференции – 23–25 ноября 2010 г. Т. 1. Донецк, ДонНТУ. — 2010. С. 13–31.

<http://ea.donntu.edu.ua/handle/123456789/2794>

48. Аноприенко А. Я., Коноплева А. П. Моделирование постбинарных клеточных автоматов // «Моделирование и информационные технологии». Сборник научных трудов. Специальный выпуск по материалам международной научной конференции «Моделирование-2010» (13–14 мая 2010 года). — Киев, НАН Украины, Институт проблем моделирования в энергетике им. Г. Е. Пухова, 2010. С. 162–170.

<http://ea.donntu.edu.ua/handle/123456789/3679>



**Научное издание**

***АНОПРИЕНКО Александр Яковлевич***  
***ИВАНИЦА Сергей Васильевич***

**ТЕТРАЛОГИКА, ТЕТРАВЫЧИСЛЕНИЯ  
И НООКОМПЬЮТИНГ**  
**Исследования 2010–2012**

**ISBN 978-966-8248-40-5**

Дизайн обложки *Иваница Сергей Васильевич*

Редактор *Аноприенко Александр Яковлевич*  
Компьютерная верстка *Иваница Сергей Васильевич*

Издательство: ООО «Технопарк ДонНТУ УНИТЕХ»

Свидетельство о внесении субъекта издательского дела в государственный реестр издателей, изготовителей и распространителей издательской продукции: ДК № 1017 от 21.08.2002.

83000, г. Донецк, ул. Артема, 58, к.1.311  
Тел.: (062) 304-90-19

Подписано к печати 24.11.2012. Формат 60x84 1/32  
Усл. печ. л. 12. Печать лазерная.  
Тираж 500 экз.

**Отпечатано в типографии «Цифровая типография»**  
**Адрес: г. Донецк, ул. Челюскинцев, 291а**  
**Тел.: +38 062 388 07 30**





### **Иваница Сергей Васильевич**

ассистент кафедры  
компьютерной инженерии  
Донецкого национального технического  
университета



### **Аноприенко Александр Яковлевич**

профессор кафедры  
компьютерной инженерии  
Донецкого национального технического  
университета



В монографии представлены результаты исследований и разработок в области теоретического обоснования и практической реализации тетралогии как наиболее эффективного варианта реализации постбинарной логики и тетравычислений и как наиболее перспективного варианта постбинарных арифметических операций. Рассмотрены также особенности перехода к ноокомпьютингу, раскрывающегося в качестве следующего этапа в развитии информационно-компьютерных технологий.

Материалы монографии предназначены для научного и образовательного использования и ориентированы на преподавателей, исследователей, специалистов и студентов, специализирующихся в области компьютерных наук и технологий.