

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Оренбургский государственный университет»

Кафедра промышленной электроники
и информационно-измерительной техники

С. А. Сильвашко

ИНФОРМАТИКА В ЭЛЕКТРОНИКЕ

Рекомендовано к изданию Редакционно-издательским советом федерального государственного бюджетного образовательного учреждения высшего образования «Оренбургский государственный университет» в качестве методических указаний для студентов, обучающихся по программам высшего образования по направлению подготовки 11.03.04 Электроника и нанoeлектроника

Оренбург
2015

УДК 004.3(076.5)
ББК 32.973-04я7
С 36

Рецензент – доцент, кандидат технических наук А. Н. Колобов

Сильвашко, С. А.

С 36 Информатика в электронике : методические указания к расчетно-графическим работам / С. А. Сильвашко; Оренбургский гос. ун-т. – Оренбург : ОГУ, 2015. – 40 с.

В методических указаниях кратко изложены сведения о кодировании чисел и реализации алгоритмов арифметических операций в арифметическо-логическом устройстве компьютера, о способах поиска информации в сети Интернет, а также приведены варианты индивидуальных заданий для выполнения студентами в часы самостоятельной работы расчетно-графических работ, предусмотренных рабочей программой дисциплины «Информатика в электронике».

Методические указания предназначены для использования студентами направления подготовки 11.03.04 Электроника и микроэлектроника профиля «Промышленная электроника» всех форм обучения.

УДК 004.3(076.5)
ББК 32.973-04я7

© Сильвашко С. А., 2015
© ОГУ, 2015

Содержание

1 Расчетно-графическая работа № 1. «Исследование алгоритмов реализации арифметических операций в ЭВМ»	4
1.1 Краткие сведения из теории	4
1.1.1 Кодирование чисел.....	4
1.1.2 Формы представления чисел в ЭВМ.....	7
1.1.3 Алгоритмы арифметических операций над числами, представленными в нормальной форме	11
1.1.4 Погрешности выполнения арифметических операций	23
1.2 Задание на выполнение расчетно-графической работы.....	25
1.3 Варианты исходных данных.....	26
1.4 Примерный перечень контрольных вопросов	26
1.5 Литература, рекомендованная для изучения	27
2 Расчетно-графическая работа № 2. «Поиск информации в глобальной сети Интернет»	28
2.1 Информационные ресурсы сети Интернет	28
2.2 Поиск информации с помощью информационных поисковых систем.....	32
2.3 Поиск учебной информации с использованием электронных библиотечных систем	36
2.4 Задание на выполнение расчетно-графической работы.....	36
2.5 Варианты индивидуальных заданий	38
2.6 Примерный перечень контрольных вопросов	39
2.7 Литература, рекомендованная для изучения	40

1 Расчетно-графическая работа № 1. «Исследование алгоритмов реализации арифметических операций в ЭВМ»

Целью работы является изучение алгоритмов реализации арифметических операций в арифметическо-логических устройствах ЭВМ (компьютеров), анализ погрешностей результатов арифметических вычислений.

1.1 Краткие сведения из теории

1.1.1 Кодирование чисел

Арифметические операции (действия) выполняют над числами. Десятичные числа в ЭВМ представляют многоразрядными двоичными кодами. Десятичные числа, используемые при решении различных практических задач с помощью компьютера, могут быть как положительными, так и отрицательными. Следовательно, необходимо предусмотреть способ кодирования знака числа.

Для представления знака числа в его двоичном коде используют символы из алфавита этой же системы счисления. В частности, знак «+» кодируют символом «0», а знак «-» – символом «1». Код знака числа записывают, как правило, в старшем разряде.

Например, двоичный код отрицательного двоичного числа $A = -110101_{(2)}$ будет иметь вид (рисунок 1):

$$[A]_{\text{ПК}} = 1, \underbrace{110101}$$

Знак числа Модуль числа

Рисунок 1

Квадратные скобки [] на рисунке 1 указывают на то, что справа записано не само двоичное число, а его *машинное изображение* (машинный код). Запятую в коде числа будем использовать для отделения знакового разряда от разрядов, отве-

денных для записи модуля числа. При таком кодировании целых чисел одним байтом (восемь бит) могут быть закодированы целые числа в интервале от -127 до $+127$, двумя байтами – от $-32\,767$ до $+32\,767$, четырьмя байтами – от $-2\,147\,483\,647$ до $+2\,147\,483\,647$ и т. д. Рассмотренный способ представления чисел называется **прямым кодом**.

Примечание – В памяти ЭВМ **положительные и отрицательные числа** (исходные данные, промежуточные и окончательные результаты вычислений) хранятся **в прямом коде**.

Использование прямых кодов чисел при выполнении арифметических операций в арифметическо-логических устройствах (АЛУ) процессоров (например, сложении отрицательных чисел, делении чисел) требует применения специальных устройств, что существенно усложняет АЛУ. С целью упрощения реализации арифметических операций, для алгебраического представления чисел (то есть для представления чисел с учетом их знака) применяют **специальные коды** – *обратный* или *дополнительный*. За счет этого облегчается определение знака результата операции, а операция вычитания числа заменяется на операцию сложения (следует отметить, что в ЭВМ все арифметические операции сводятся к операции сложения).

Примечание – **Положительные числа в прямом, обратном и дополнительном кодах имеют одинаковую форму представления**. Например, для положительного числа $A = 10111011_{(2)}$ можно записать: $[A]_{\text{ПК}} = [A]_{\text{ОК}} = [A]_{\text{ДК}} = 0,10111011$.

Переход от *прямого кода* к *обратному* для **отрицательных** чисел выполняется по правилу: *чтобы представить отрицательное двоичное число в обратном коде, нужно оставить в знаковом разряде прямого кода числа 1, а во всех значащих разрядах заменить «1» на «0» и «0» на «1»*. Такая операция называется **инвертированием**.

Это же правило справедливо и при переводе отрицательных чисел *из обратного кода в прямой*.

Пример 1. Найти обратный код числа $A = -1101011_{(2)}$.

Решение:

а) прямой код числа: $[A]_{\text{ПК}} = 1,1101011$;

б) обратный код числа: $[A]_{\text{OK}} = 1,0010100$.

Пример 2. Найти прямой код числа A , представленного в обратном коде:
 $[A]_{\text{OK}} = 1,1011011$.

Решение: прямой код числа A имеет вид: $[A]_{\text{ПК}} = 1,0100100$.

Переход от прямого кода к дополнительному для отрицательных чисел может быть выполнен двумя способами:

Правило 1. Чтобы перевести отрицательное число из прямого кода в дополнительный, необходимо сначала получить обратный код этого числа, а затем прибавить единицу к цифре в младшем разряде обратного кода числа.

Правило 2. Чтобы перевести отрицательное число из прямого кода в дополнительный, необходимо оставить в знаковом разряде 1, а во всех значащих разрядах (слева направо) заменить «1» на «0» и «0» на «1», за исключением последней единицы и следующих за ней нулей.

При переходе от дополнительного кода к прямому применяют те же правила.

Пример 3. Найти дополнительный код числа $A = -1101100_{(2)}$.

Решение:

а) прямой код числа имеет вид: $[A]_{\text{ПК}} = 1,1101100$;

б) запишем дополнительный код числа A :

1) по первому правилу

$$[A]_{\text{OK}} = 1,0010011$$

$$+ \quad \quad 1$$

$$[A]_{\text{ДК}} = 1,0010100;$$

2) по второму правилу

$$[A]_{\text{ПК}} = 1,1101100,$$

$$[A]_{\text{ДК}} = 1,0010100.$$

Пример 4. Найти прямой код числа A , представленного в дополнительном коде: $[A]_{\text{ДК}} = 1,1101110$.

Решение:

$$[A]_{\text{ДК}} = 1,1101110;$$

$$[A]_{\text{ПК}} = 1,0010010.$$

1.1.2 Формы представления чисел в ЭВМ

Оперативное запоминающее устройство (ОЗУ) компьютера реализуют в виде совокупности отдельных *адресуемых ячеек памяти*, длина которых составляет **один байт** (восемь бит – восемь двоичных разрядов). Адресом ячейки является ее порядковый номер в ОЗУ (нумерация ячеек памяти начинается с нуля). Таким образом, **минимальная длина двоичного кода (машинного кода) данных**, с которыми оперирует процессор компьютера, составляет 8 бит. В зависимости от разрядности процессора, он может оперировать с данными в объеме *одного, двух, четырех* и т. д. байт. Если длина машинного кода превышает один байт, то такой код принято называть **машинным словом**. Таким образом, машинное слово может быть двух-, четырех-, восьмибайтовым. Если длина машинного слова превышает один байт, то оно занимает в памяти компьютера несколько подряд расположенных ячеек. При этом *адресом* машинного слова является *адрес первого байта*.

Используемые в практике человеческой деятельности действительные числа могут быть *целыми, дробными точными, рациональными, иррациональными, дробными приближенными*. Числа могут быть *положительными и отрицательными*. Числа могут быть очень маленькими (например, масса атома) и очень большими (например, масса Земли). Единого *оптимального* представления для всевозможных действительных чисел создать невозможно. Поэтому в современных компьютерах используют две формы записи чисел – **естественная** (*с фиксированной запятой*) и **нормальная** (*с плавающей запятой*). При естественной форме действительное число записывают в том виде, в каком оно существует, например: 3852409 – целое число, 0,402984 – правильная дробь, 54,08356 – неправильная дробь и т. п. При использовании формы с фиксированной запятой в разрядной сетке компьютера (под *разрядной сеткой* ЭВМ понимают совокупность элементов памяти, служащих для хранения многоразрядных двоичных чисел) необходимо предусмотреть место для хранения целой части и дробной части числа. Поскольку длина разрядной сетки компьютера ограничена разрядностью его процессора (в современных вычислительных устройствах – 8, 16, 32 или 64 разряда), то такая форма

представления чисел имеет ряд **недостатков**. От положения запятой в разрядной сетке машины зависят такие ее важные характеристики, как точность и диапазон представляемых чисел. Поэтому компьютеры, в которых используется форма с фиксированной запятой, характеризуются малым диапазоном представления чисел и ограниченной точностью.

Однако наряду с указанными недостатками форма с фиксированной запятой имеет и ряд **достоинств**: позволяет построить сравнительно несложные операционные устройства с высоким быстродействием.

В современных компьютерах форма с фиксированной запятой применяется для представления целых чисел. При этом разрядная сетка имеет вид, показанный на рисунке 2.

знак числа	2^{n-1}	2^{n-2}	...	2^1	2^0
---------------	-----------	-----------	-----	-------	-------

Рисунок 2

Если для записи числа отводится n двоичных разрядов, то минимальное целое число, представляемое в ЭВМ в форме с фиксированной запятой, равно «0», а максимальное – $2^n - 1$.

Действительные числа в математике представляются *конечными или бесконечными дробями*, т. е. точность представления чисел практически не ограничена. Однако в компьютерах из-за ограниченной длины разрядной сетки *бесконечные или очень длинные числа усекаются до некоторой длины*, что приводит к появлению *погрешности вычислений*.

Применение для записи чисел *формы с плавающей запятой* позволяет уменьшить указанную погрешность, а также существенно расширить диапазон представления чисел.

Из математики известно, что любое число X можно записать в показательной (нормальной) форме вида $X = M K^P$, где M – мантисса числа, K – основание системы счисления, P – порядок числа (целое число), определяющий положение запятой в

числе. При этом мантисса и порядок представляются символами (цифрами) системы счисления с основанием K , знак числа совпадает со знаком мантиссы.

Например, десятичное число 4235,25 может быть записано так: $4235,25 = 0,423525 \cdot 10^4 = 0,0423525 \cdot 10^5 = 423525 \cdot 10^{-2}$ и т. п. Аналогично можно записать двоичное число 1011,01: $1011,01 = 0,101101 \cdot 2^{100} = 0,0101101 \cdot 2^{101} = 101101 \cdot 2^{-10}$.

Очевидно, такое представление не однозначно. На практике с целью устранения неоднозначности используют *нормализованную форму представления чисел с плавающей запятой* – форму, для которой справедливо условие $K^{-1} \leq |M| < 1$. Для двоичных чисел это означает, что мантисса должна быть правильной дробью и после запятой (в старшем разряде мантиссы) должна располагаться единица.

Нормализация мантиссы обеспечивается за счет подбора порядка числа. В рассмотренных примерах числа $0,423525 \cdot 10^4$ и $0,101101 \cdot 2^{100}$ являются нормализованными.

Если «плавающая» запятая расположена в мантиссе перед первой значащей цифрой, то при фиксированном количестве разрядов в разрядной сетке ЭВМ, отведенных под мантиссу, обеспечивается запись максимального количества значащих цифр числа, т. е. обеспечивается максимально возможная точность получения результата вычислений.

Действительные числа в компьютерах различных типов записываются по-разному, тем не менее, существует несколько международных стандартных форматов, различающихся по точности, но имеющих одинаковую структуру. Рассмотрим в качестве примера распределение разрядов в 32-разрядной сетке, наиболее часто применяемое в современных компьютерах (рисунок 3).

Первый бит двоичного кода используется для кодирования *знака мантиссы* (а, следовательно, и знака числа). В следующей группе бит (в рассматриваемом примере – 8 бит (разряды с 24-го по 31-й)) записывают код порядка числа, а оставшиеся 23 бита используются для записи мантиссы числа.

Длины (число разрядов) порядка и мантиссы в разных ЭВМ могут отличаться, но в каждой ЭВМ они имеют строго фиксированное значение.

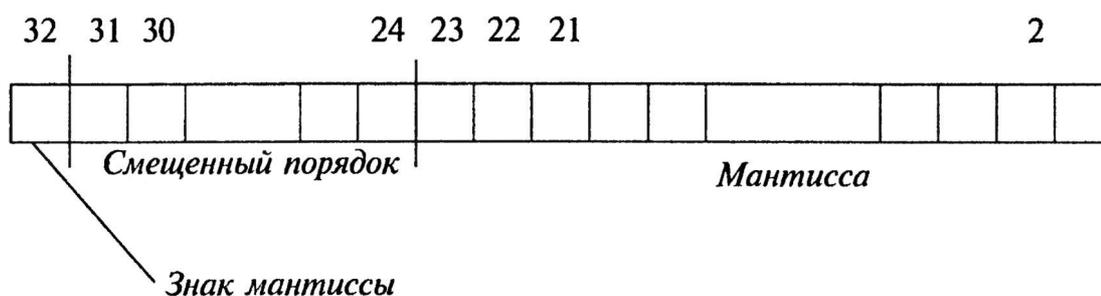


Рисунок 3

Порядок числа, в общем случае, может быть как *положительным*, так и *отрицательным*. Но чтобы в разрядной сетке не выделять один разряд для записи знака порядка, используют *смещенный порядок*. Смещенный порядок (SP) представляет собой *алгебраическую сумму истинного порядка (P)* и некоторой *константы (S)*, равной *абсолютной величине максимального по модулю отрицательного порядка*, называемой *смещением* ($SP = P + S$).

Смещение S выбирается таким образом, чтобы смещенный порядок принимал только положительные значения и при этом минимальному значению порядка соответствовал «0». Например, если для порядка отводится восемь разрядов, то в разрядной сетке можно записать двоичные числа в диапазоне от 00000000 до 11111111, то есть всего 256 различных двоичных чисел. Выбрав в качестве смещения $S = 256 : 2 = 128$, можно кодировать порядок в диапазоне значений от -128 до 127 . При этом смещенный порядок будет принимать значения от 0 ($-128+128$) до 255 ($127+128$). Чтобы определить истинный порядок числа, необходимо из смещенного порядка, записанного в разрядной сетке, вычесть смещение ($P = SP - S$). Например, если смещенный порядок равен $SP = 23$, то порядок числа равен $P = 23 - 128 = -105$.

Пример 5. Записать двоичное число $X = -1101001,01_{(2)}$ в формате разрядной сетки, представленной на рисунке 3.

Решение:

а) число X , записанное в нормальной форме, имеет вид:

$$X = -0,110100101 \cdot 2^{111};$$

б) порядок числа, представленного в нормализованной форме, в десятич-

ной системе счисления равен: $P = 7$;

в) определим смещенный порядок: $SP = 7 + 128 = 135$ (в двоичном коде $SP = 10000111$);

г) запишем заданное число в формате разрядной сетки (рисунок 3):

11000011111010010100000000000000.

Диапазон представления чисел в форме с плавающей запятой значительно шире диапазона представления чисел в форме с фиксированной запятой, а погрешность вычислений при той же длине разрядной сетки значительно ниже, так как она определяется весом самого младшего разряда мантиссы. Однако **быстродействие ЭВМ** при обработке чисел с плавающей запятой **гораздо ниже**, чем при обработке чисел с фиксированной запятой, а АЛУ – более сложно в реализации.

Использование смещенного порядка позволяет производить операции над порядками как над беззнаковыми числами, что упрощает операции сравнения, сложения и вычитания порядков, а также упрощает операцию сравнения самих нормализованных чисел.

1.1.3 Алгоритмы арифметических операций над числами, представленными в нормальной форме

Все арифметические операции в компьютере выполняются над числами, представленными в двоичной системе счисления по правилам этой системы счисления. В частности, результаты выполнения арифметического сложения и умножения представлены на рисунке 4.

Как было отмечено ранее, для упрощения технической реализации АЛУ компьютера в нем предусмотрено выполнение только одной арифметической операции – **сложение**. Выполнение таких арифметических операций, как вычитание, умножение и деление осуществляется по различным алгоритмам, приводящим эти операции к операции сложения чисел.

Рассмотрим основные алгоритмы выполнения арифметических операций над числами, представленными в форме с плавающей запятой.

$0 + 0 = 0$	$0 \times 0 = 0$
$0 + 1 = 1$	$0 \times 1 = 0$
$1 + 0 = 1$	$1 \times 0 = 0$
$1 + 1 = 10$	$1 \times 1 = 1$

Рисунок 4

1.1.3.1 Сложение чисел. Числа, используемые при сложении в качестве операндов (слагаемых), могут быть как положительными, так и отрицательными. Поэтому в компьютере реализуется алгебраическая операция сложения над числами, представленными в обратном или дополнительном кодах.

Для сложения двух чисел, представленных **в нормальной форме**, необходимо предварительно **выровнять их порядки**, так как вес одноименных разрядов мантисс чисел с разными порядками неодинаков. Очевидно, что **при выравнивании порядков сдвиг** мантиссы одного из чисел **возможен только вправо** (что соответствует сдвигу запятой влево), так как сдвиг влево может привести к появлению **целой части, что не соответствует условию нормализации числа**. При сдвиге мантиссы вправо увеличивается порядок числа. Следовательно, сдвигать вправо нужно мантиссу числа, имеющего **меньший порядок**.

Алгоритм алгебраического сложения чисел в нормальной форме (в форме с плавающей запятой) состоит в выполнении следующей последовательности действий:

1) записать слагаемые (представленные в двоичной системе счисления) в форме с плавающей запятой, выполняя условие нормализации числа;

2) проверить равенство порядков чисел.

Если в качестве операндов участвуют числа A и B , то данную операцию математически можно записать следующим образом:

$$\Delta P = P_A - P_B, \quad (1)$$

где P_A и P_B , соответственно, порядки чисел A и B .

Если $\Delta P = 0$, то порядки чисел одинаковы и операцию сдвига нужно пропустить. Если $\Delta P < 0$, то $P_A < P_B$ и мантиссу числа A нужно сдвинуть вправо на $|\Delta P|$ разрядов. Если же $\Delta P > 0$, то вправо на ΔP разрядов сдвигают мантиссу числа B ;

3) после выравнивания порядков чисел записать обратные или дополнительные коды мантисс;

4) сложить мантиссы. При этом результат будет получен, соответственно, в обратном или дополнительном коде;

5) записать результат в прямом коде;

6) представить результат в нормальной форме (с учетом значения большего порядка). При необходимости нормализовать результат.

Правила сложения чисел в обратных и дополнительных кодах отличаются, поэтому рассмотрим каждое из них в отдельности.

Правило 1. Чтобы получить результат сложения двух чисел, мантиссы которых представлены в обратных кодах, необходимо сложить обратные коды мантисс по правилам двоичной арифметики, рассматривая при этом знаковые разряды как обычные числовые разряды. Если при сложении появляется единица переноса из знакового разряда, то ее следует прибавить к цифре в младшем разряде ранее полученного результата.

Последнее обстоятельство (добавление единицы переноса в младший разряд) увеличивает время выполнения операций сложения. Поэтому в ЭВМ чаще реализуют операцию сложения над мантиссами чисел, представленных в дополнительных кодах. Сформулируем второе правило сложения чисел.

Правило 2. Чтобы получить результат сложения двух чисел, мантиссы которых представлены в дополнительных кодах, необходимо сложить дополнительные коды мантисс по правилам двоичной арифметики, рассматривая при этом знаковые разряды как обычные числовые разряды. Если при сложении появляется единица переноса из знакового разряда, то ее следует проигнорировать.

Рассмотрим некоторые примеры.

Пример 6. Сложить числа $A = -5,5_{(10)}$ и $B = 11_{(10)}$, используя алгоритмы, реализованные в АЛУ ЭВМ.

Решение. Переведем операнды в двоичную систему счисления: $A = -101,1_{(2)}$, $B = 1011_{(2)}$.

Запишем числа A и B в форме с плавающей запятой, одновременно выполняя их нормализацию:

$$A = -0,1011 \cdot 2^{11},$$

$$B = 0,1011 \cdot 2^{100}.$$

Поскольку порядки операндов разные, необходимо выполнить их выравнивание до большего. Найдем разность порядков $\Delta P = P_A - P_B = P_A + (-P_B)$. Поскольку при вычислении ΔP используется операция вычитания, порядки должны быть представлены в обратном или дополнительном кодах. Запишем дополнительные коды порядков, предварительно выровняв число разрядов, отводимых для записи значащих цифр порядков. Так как порядки – целые числа, то выравнивание числа разрядов заключается в добавлении требуемого числа нулей слева в меньшем (по модулю) порядке. С учетом изложенного:

$$[P_A]_{\text{дк}} = 0,011; [P_B]_{\text{дк}} = 1,100.$$

В приведенной выше записи запятая использована для условного отделения знакового разряда от разрядов, используемых для записи значащих цифр числа.

Найдем алгебраическую сумму порядков:

$$\begin{array}{r} [P_A+P_B]_{\text{дк}} = 0,011 \\ + \\ \quad \quad \quad \underline{1,100} \\ \quad \quad \quad 1,111 \end{array}$$

$$[P_A+P_B]_{\text{пк}} = 1,001.$$

Таким образом $\Delta P = -1$, следовательно мантиссу числа A нужно сдвинуть вправо на один разряд. Окончательно запишем

$$A = -0,01011 \cdot 2^{100},$$

$$B = 0,10110 \cdot 2^{100}.$$

Запишем прямые коды мантисс операндов: $[M_A]_{\text{пк}} = 1,01011$; $[M_B]_{\text{пк}} = 0,10110$.

Выполним сложение с использованием обратных кодов чисел. Обратные коды мантисс имеют вид: $[M_A]_{\text{OK}} = 1,10100$; $[M_B]_{\text{OK}} = 0,10110$.

$$\begin{array}{r} [M_{A+M_B}]_{\text{OK}} = 1,10100 \\ + \\ \quad \underline{0,10110} \\ \quad 0,01010 \\ + \\ \quad \quad \underline{1} \\ \quad \quad 0,01011 \end{array}$$

$$[M_{A+M_B}]_{\text{ПК}} = 0,01011,$$

$$(A+B) = 0,01011 \cdot 2^{100} = 101,1_{(2)} = 5,5_{(10)}.$$

Выполним сложение с использованием дополнительных кодов чисел. Дополнительные коды мантисс имеют вид: $[M_A]_{\text{ДК}} = 1,10101$; $[M_B]_{\text{ДК}} = 0,10110$.

$$\begin{array}{r} [M_{A+M_B}]_{\text{ДК}} = 1,10101 \\ + \\ \quad \underline{0,10110} \\ \quad 0,01011 \end{array}$$

$$[M_{A+M_B}]_{\text{ПК}} = 0,01011,$$

$$(A+B) = 0,01011 \cdot 2^{100} = 101,1_{(2)} = 5,5_{(10)}.$$

И в первом, и во втором случае получен правильный результат сложения чисел A и B .

Пример 7. Сложить числа $A = 27_{(10)}$ и $B = 9_{(10)}$, представленные в нормальной форме.

Решение. Запишем операнды в двоичной системе счисления: $A = 11011_{(2)}$, $B = 1001_{(2)}$.

Запишем числа A и B в нормальной форме:

$$A = 0,11011 \cdot 2^{101}; B = 0,1001 \cdot 2^{100}.$$

Найдем разность порядков ΔP :

$$[P_A]_{\text{ДК}} = 0,101, [P_B]_{\text{ДК}} = 1,100,$$

$$\begin{array}{r} [P_{A+P_B}]_{\text{ДК}} = 0,101 \\ + \\ \quad \underline{1,100} \\ \quad 0,001 \end{array}$$

$$[P_{A+P_B}]_{\text{ПК}} = 0,001.$$

Таким образом $\Delta P = 1$, следовательно сдвигать вправо на один разряд необходимо мантиссу числа B . Окончательно запишем

$$A = 0,11011 \cdot 2^{101},$$

$$B = 0,01001 \cdot 2^{101}.$$

Так как оба операнда – числа положительные, то прямые и дополнительные коды мантисс совпадают:

$$[M_A]_{\text{ПК}} = [M_A]_{\text{ДК}} = 0,11011; [M_B]_{\text{ПК}} = [M_B]_{\text{ДК}} = 0,01001.$$

Выполним сложение мантисс:

$$\begin{array}{r} [M_{A+M_B}]_{\text{ДК}} = 0,11011 \\ + \\ \underline{0,01001} \\ 1,00100 \end{array}$$

Как видно из примера, результат сложения двух положительных чисел оказался отрицательным, чего быть не должно. Отрицательный результат при сложении двух положительных чисел, равно как и положительный результат при сложении двух отрицательных чисел, свидетельствуют о том, что имело место **переполнение разрядной сетки**. Чтобы избежать переполнения, достаточно сдвинуть мантиссы операндов на один разряд вправо, одновременно увеличив порядок на единицу. Выполним указанную корректировку и повторим операцию сложения мантисс:

$$A = 0,011011 \cdot 2^{110},$$

$$B = 0,001001 \cdot 2^{110},$$

$$\begin{array}{r} [M_{A+M_B}]_{\text{ДК}} = 0,011011 \\ + \\ \underline{0,001001} \\ 0,100100 \end{array}$$

$$[M_{A+M_B}]_{\text{ПК}} = 0,100100,$$

$$(A+B) = 0,100100 \cdot 2^{110} = 100100_{(2)} = 36_{(10)}.$$

Результат сложения правильный.

1.1.3.2 Умножение двух чисел. В АЛУ компьютеров нашли реализацию следующие два алгоритма умножения многоразрядных чисел:

- 1) умножение, начиная с младших разрядов множителя;
- 2) умножение, начиная со старших разрядов множителя.

В обоих случаях операция умножения состоит из ряда последовательных операций сложения частных произведений. Основное отличие алгоритмов состоит в направлении сдвига суммы частных произведений (или множимого). Операциями сложения управляют разряды множителя: если в каком-то разряде множителя находится единица, то к сумме частных произведений добавляется множимое с соответствующим сдвигом; если в разряде множителя находится нуль, то сдвиг производится без прибавления множимого.

При умножении двух чисел, представленных в форме с плавающей запятой, необходимо выполнить три основных операции: *определить знак произведения, алгебраически сложить порядки, перемножить мантиссы*. В АЛУ ЭВМ все эти операции реализуются разными функциональными узлами.

Если $M_A = 0$, $a_1a_2\dots a_n$ – мантисса множимого и $M_B = 0$, $b_1b_2\dots b_n$ – мантисса множителя (n -разрядные двоичные числа), то математически операцию умножения мантисс можно записать следующим образом

$$M_C = M_A M_B = (\dots((b_n \cdot 0, a_1a_2\dots a_n) \cdot 2^{-1} + b_{n-1} \cdot 0, a_1a_2\dots a_n) \cdot 2^{-1} + \dots + b_1 \cdot 0, a_1a_2\dots a_n) \cdot 2^{-1}, \quad (2)$$

где M_C – мантисса произведения.

Выражение (2) описывает алгоритм умножения, в котором умножение начинается с младших разрядов *множителя* и сумма частных произведений сдвигается вправо (умножение на 2^{-1} соответствует сдвигу вправо на один разряд).

Примечание – При выполнении операции умножения используют **прямые коды** мантисс сомножителей.

Алгоритм выполнения операции умножения чисел состоит в следующем:

1) определить **знак произведения**. Для этих целей применить операцию сложения по модулю «2» (логический элемент «исключающее ИЛИ») к цифрам в знаковых разрядах сомножителей, то есть

$$Z_{NC} = Z_{NA} \oplus Z_{NB}. \quad (3)$$

Результат операции сложения по модулю «2» определяется следующим образом:

$$0 \oplus 0 = 0,$$

$$0 \oplus 1 = 1,$$

$$1 \oplus 0 = 1,$$

$$1 \oplus 1 = 0;$$

2) определить **порядок произведения**, для чего алгебраически сложить порядки сомножителей, представленные обратными или дополнительными кодами:

$$P_C = P_A + P_B; \quad (4)$$

3) перемножить мантиссы чисел без учета знакового разряда.

В АЛУ компьютера множимое и множитель хранятся в соответствующих регистрах. Их умножение реализуется с помощью сумматора (логического устройства, реализующего арифметическое сложение двух операндов). Перед началом умножения во все разряды сумматора помещают нули, после чего все дальнейшие операции выполняют в соответствии с выражением (2) следующим образом:

а) если в младшем разряде множителя находится 1, то к содержимому сумматора прибавить множимое, а если 0 – то к содержимому сумматора прибавить нули во всех разрядах. Результат сложения (первое частное произведение) сдвинуть вправо на один разряд;

б) если во втором (справа) разряде множителя находится 1, то прибавить к содержимому сумматора множимое, а если 0 – то прибавить нули. Второе частное произведение сдвинуть вправо на один разряд и т. д.

В общем случае суммирование со сдвигом выполнять n раз, где n – число разрядов в мантиссе множителя;

4) записать окончательный результат умножения в форме с плавающей запятой с учетом знака и порядка произведения, полученных ранее.

Пример 8. Найти результат умножения чисел A и B , если $A = 18_{(10)}$, $B = -5,5_{(10)}$. Использовать представление двоичных чисел в форме с плавающей запятой.

Решение. Запишем операнды A и B в двоичной системе счисления:

$$A = 10010_{(2)}, B = -101,1_{(2)}.$$

В форме с плавающей запятой операнды имеют вид:

$$A = 0,10010 \cdot 2^{101}; B = -0,10110 \cdot 2^{11}.$$

Запишем прямые коды мантисс операндов и дополнительные коды порядков:

$$[M_A]_{\text{ПК}} = 0,10010,$$

$$[M_B]_{\text{ПК}} = 1,10110,$$

$$[P_A]_{\text{ДК}} = 0,101,$$

$$[P_B]_{\text{ДК}} = 0,011.$$

1) Определим знак произведения:

$$Z_{\text{НС}} = 0 \oplus 1 = 1 \text{ (то есть результат умножения будет отрицательным).}$$

2) Определим порядок произведения:

$$\begin{array}{r} [P_C]_{\text{ДК}} = 0,0101 \\ + \\ \underline{0,0011} \\ 0,1000 \end{array}$$

$$[P_C]_{\text{ПК}} = 0,1000, P_C = 1000_{(2)}.$$

3) Перемножим мантиссы операндов (результаты пошагового выполнения алгоритма показаны на рисунке 5).

Запишем результат умножения чисел в нормальной форме:

$$C = -0,0110001100 \cdot 2^{1000}.$$

$$\text{Таким образом, } C = -1100011_{(2)} = -99_{(10)}.$$

1.1.3.3 Деление двух чисел, представленных в форме с плавающей запятой.

Известны несколько алгоритмов деления чисел, применяемых в АЛУ компьютеров. Наибольшее распространение получили два из них – **алгоритм деления без восстановления остатка** и **алгоритм деления с восстановлением остатка**. Остановимся на первом из них.

При делении двух чисел, представленных в форме с плавающей запятой, необходимо выполнить три основных операции: *определить знак частного, определить порядок результата (частного), разделить мантиссу делимого на мантиссу делителя*.

	$M_B = 0,10110$
00000	1011 <u>0</u>
+ <u>00000</u>	
00000·2 ⁻¹	
000000	1011 <u>0</u>
+ <u>10010</u>	
100100·2 ⁻¹	
0100100	10 <u>1</u> 10
+ <u>10010</u>	
1101100·2 ⁻¹	
01101100	10 <u>1</u> 10
+ <u>00000</u>	
01101100·2 ⁻¹	
001101100	<u>1</u> 0110
+ <u>10010</u>	
110001100·2 ⁻¹	
0110001100	

Рисунок 5

Знак частного, как и в случае умножения чисел, находят с помощью операции сложения по модулю «2» (3), применяя ее к цифрам в знаковых разрядах исходных кодов операндов.

Для нахождения **порядка результата** необходимо *вычесть из порядка делимого порядок делителя* (то есть сложить порядки операндов, представленные в дополнительных или обратных кодах).

При делении мантисс, если модуль мантиссы делимого больше модуля мантиссы делителя, результат деления (то есть частное) будет больше единицы, что приведет к получению неверного результата. Чтобы избежать этого, до начала деления необходимо обеспечить выполнение условия $|M_A| < |M_B|$, где M_A – мантисса делимого, M_B – мантисса делителя. В этом случае результат деления M_C будет представлять собой правильную дробь.

Если для исходных операндов указанное выше условие не выполняется, то до начала деления (до нахождения порядка результата) необходимо выполнить операцию масштабирования мантиссы делимого. На практике масштабирование заключается в сдвиге мантиссы делимого на один разряд вправо. При этом порядок делимого, соответственно, должен быть увеличен на единицу.

Рассмотрим сущность *алгоритма деления без восстановления остатка*. Предположим, что необходимо разделить число A (делимое) на число B (делитель), записанные в нормальной форме. Результатом деления будет число $C = \frac{A}{B}$.

Процедура деления состоит из следующих операций:

1) записать числа A и B , представленные в двоичной системе счисления, в форме с плавающей запятой и проверить выполнение условия $|M_A| < |M_B|$. Если условие выполняется, то перейти к следующей операции, если нет – то выполнить операцию масштабирования мантиссы делимого, а затем перейти к следующей операции;

2) записать прямые коды мантиссы делимого и делителя;

3) определить **знак частного**, для чего применить операцию сложения по модулю «2» (3) к цифрам в знаковых разрядах исходных кодов операндов;

4) определить **порядок частного**, для чего вычесть из порядка делимого порядок делителя ($P_C = P_A - P_B = P_A + (-P_B)$);

5) получить **прямой** и **дополнительный коды** делителя. При этом, *независимо от того, положительным или отрицательным числом является делитель, его прямой код должен быть записан как для положительного числа (сохранить модуль мантиссы и в знаковом разряде записать «0»)*, а **дополнительный** – как для отрицательного числа (в знаковом разряде нужно записать «1» и преобразовать модуль мантиссы по правилам получения дополнительного кода);

6) удвоить код мантиссы делимого (сдвинуть **мантиссу делимого** влево на один разряд) и вычесть из полученного результата делитель (то есть прибавить его в дополнительном коде). В результате этой операции будет получен первый **остаток** α_1 . Если он положительный ($\alpha_1 > 0$), то записать в старший разряд результата (пер-

вый разряд справа от запятой в мантиссе частного) «1» (то есть $c_1 = 1$), а если отрицательный ($\alpha_1 < 0$) – то записать «0» (то есть $c_1 = 0$).

Примечание – Результат деления (частное) ищем в виде: $M_C = 0, c_1 c_2 c_3 \dots c_n$, где n – число значащих цифр после запятой, зависящее от требуемой точности получения результата;

7) сдвинуть влево на один разряд остаток α_1 и, если он **был** положительным, то вычесть из полученного результата делитель, в противном случае – прибавить к полученному результату делитель. Определить знак второго **остатка** α_2 и если он положительный ($\alpha_2 > 0$), то записать во второй разряд результата после запятой «1» ($c_2 = 1$), а если отрицательный ($\alpha_2 < 0$) – то записать «0» ($c_2 = 0$);

8) сдвинуть влево на один разряд остаток α_2 и, если он **был** положительным, то вычесть из полученного результата делитель, в противном случае – прибавить к полученному результату делитель и т. д.

Алгоритм выполняется до тех пор, пока очередной остаток α_i не станет равным нулю, либо пока не будет достигнута требуемая точность результата деления;

9) записать результат деления в нормальной форме с учетом ранее полученных значений знака и порядка результата (**результат будет представлен в прямом коде**).

Пример 9. Разделить число $A = -7_{(10)}$ на число $B = 1,2_{(10)}$.

Решение.

$$1) A = -111_{(2)} = -0,111 \cdot 2^{11}, B \approx 1,0011_{(2)} = 0,10011 \cdot 2^1.$$

Как видно, условие $|M_A| < |M_B|$ не выполняется, поэтому сдвинем мантиссу делимого на один разряд вправо. Окончательно получим: $A = -0,0111 \cdot 2^{100}$, $B = 0,10011 \cdot 2^1$.

2) Запишем прямые коды мантисс делимого и делителя: $[M_A]_{ПК} = 1,01110$; $[M_B]_{ПК} = 0,10011$.

3) Определим знак частного:

$$Z_{HC} = 1 \oplus 0 = 1.$$

4) Найдем порядок частного.

$$[P_A]_{ПК} = 0,100, [P_B]_{ПК} = 0,001, [-P_B]_{ПК} = 1,001, [-P_B]_{ДК} = 1,111, \text{ тогда}$$

$$[P_C]_{\text{ДК}} = 0,100$$

+

$$\frac{1,111}{0,011}$$

$$[P_C]_{\text{ПК}} = 0,011, P_C = 11_{(2)}.$$

5) Запишем требуемые коды для мантииссы делителя:

$$[M_B]_{\text{ПК}} = 0,10011,$$

$$[M_B]_{\text{ДК}} = 1,01101.$$

6) Выполним деление мантиисс чисел (результаты пошагового выполнения алгоритма показаны на рисунке 6).

Запишем полученный результат деления:

$$[M_C]_{\text{ПК}} \approx 1,1011110.$$

В нормальной форме результат деления будет представлен следующим образом: $C = \frac{A}{B} \approx -0,101111 \cdot 2^{11}$. Окончательно $C \approx -101,111_{(2)} = -5,875_{(10)}$.

1.1.4 Погрешности выполнения арифметических операций

Выбор длины разрядной сетки ЭВМ, а также формы представления чисел в ЭВМ тесно связаны с обеспечением заданной точности вычислений. Основными причинами, приводящими к возникновению погрешностей вычислений при выполнении различных арифметических операций в АЛУ компьютера, могут быть:

1) неточное задание исходных данных (операндов), связанное либо с ограниченной длиной разрядной сетки компьютера, либо с наличием неточностей перевода чисел из одной системы счисления в другую;

2) использование приближенных методов вычислений, приводящих к появлению методической погрешности (например, при вычислении интегралов);

3) округление результатов элементарных операций, что в свою очередь может привести к появлению накопленных погрешностей;

4) сбои в работе компьютера;

5) ошибки оператора при вводе исходных данных.

Различают два вида погрешностей вычислений – *абсолютная* и *относительная*.

Сложение, сдвиги	Комментарии
1,01110	$[M_A]_{ПК}$
0,11100	$[M_A]_{ПК} \cdot 2^1$
+	
<u>1,01101</u>	
0,01001	$\alpha_1 > 0, c_1 = 1$
0,10010	$\alpha_1 \cdot 2^1$
+	
<u>1,01101</u>	
1,11111	$\alpha_2 < 0, c_2 = 0$
1,11110	$\alpha_2 \cdot 2^1$
+	
<u>0,10011</u>	
0,10001	$\alpha_3 > 0, c_3 = 1$
1,00010	$\alpha_3 \cdot 2^1$
+	
<u>1,01101</u>	
0,01111	$\alpha_4 > 0, c_4 = 1$
0,11110	$\alpha_4 \cdot 2^1$
+	
<u>1,01101</u>	
0,01011	$\alpha_5 > 0, c_5 = 1$
0,10110	$\alpha_5 \cdot 2^1$
+	
<u>1,01101</u>	
0,00011	$\alpha_6 > 0, c_6 = 1$
0,00110	$\alpha_6 \cdot 2^1$
+	
<u>1,01101</u>	
1,10011	$\alpha_7 < 0, c_7 = 0$
и т. д.	

Рисунок 6

Абсолютная погрешность вычислений – это разность между истинным значением результата вычислений (в частности – в десятичной системе счисления) $C_{ист}$ и значением, полученным в результате вычислений на компьютере $C_{комп}$:

$$\Delta C = C_{ист} - C_{комп}. \quad (5)$$

Относительная погрешность вычислений – это отношение абсолютной погрешности вычислений к истинному значению результата вычислений, умноженному на 100%:

$$\delta_C = \frac{\Delta C}{C_{ист}} \cdot 100\%. \quad (6)$$

Например, при делении числа $A = -7_{(10)}$ на число $B = 1,2_{(10)}$ (пример 9 в п. п. 1.1.3.3) в десятичной системе счисления получим: $C_{ист} = -5,83(3) \approx -5,833$. Результат деления с использованием алгоритмов ЭВМ (в том же примере) – $C_{комп} = -5,875$. Тогда абсолютная и относительная погрешности вычислений составят:

$$\Delta C = C_{ист} - C_{комп} = -5,833 - (-5,875) = 0,042,$$

$$\delta_C = \frac{\Delta C}{C_{ист}} \cdot 100\% = \frac{0,042}{5,833} \cdot 100\% = 0,72\%.$$

Погрешность вычислений в рассмотренном примере вызвана неточностью перевода числа B из десятичной системы счисления в двоичную.

1.2 Задание на выполнение расчетно-графической работы

1.2.1 Выполнить арифметические операции в двоичной системе счисления над числами, представленными в форме с плавающей запятой, используя алгоритмы, применяемые в ЭВМ:

- а) сложить числа A и B ;
- б) умножить число A на число B ;
- в) разделить число A на число B .

Результаты вычислений записать с учетом формата разрядной сетки (рисунок 3) и в десятичной системе счисления.

1.2.2 Оценить точность выполнения арифметических операций в АЛУ компьютера, сравнивая полученные результаты с результатами аналогичных вычислений

в десятичной системе счисления.

1.2.3 Проанализировать результаты выполненной работы и сформулировать выводы.

1.3 Варианты исходных данных

Таблица 1 – Варианты исходных данных

Номера вариантов	Операнды		Номера вариантов	Операнды	
	$A_{(10)}$	$B_{(10)}$		$A_{(10)}$	$B_{(10)}$
1	- 3,25	2,2	14	- 6,5	7,4
2	3,6	4,25	15	6,75	7,7
3	3,75	2,3	16	7,9	5,25
4	- 4,0	2,7	17	7,25	8,1
5	- 4,25	5,4	18	- 7,5	6,2
6	4,5	5,2	19	7,75	8,8
7	- 4,8	3,5	20	- 5,0	6,4
8	5,0	3,7	21	5,2	6,5
9	- 5,25	6,6	22	- 8,3	6,75
10	5,5	4,2	23	- 5,75	7,6
11	5,75	7,3	24	9,7	2,25
12	6,0	4,8	25	- 4,25	6,8
13	- 6,2	5,5	26	6,5	-4,4

1.4 Примерный перечень контрольных вопросов

1.4.1 Что называется системой счисления. В чем различие между позиционной и непозиционной системами счисления?

1.4.2 Как перевести число из десятичной в двоичную систему счисления?

1.4.3 Как перевести число из двоичной в десятичную систему счисления?

1.4.4 Как перевести число из системы счисления с основанием 2 в систему счисления с основанием 2^n и наоборот?

1.4.5 Как представляется число в разрядной сетке компьютера в форме с фиксированной запятой (с плавающей запятой)?

1.4.6 Как получить смещенный порядок при представлении числа в форме с плавающей запятой?

1.4.7 Пояснить алгоритм сложения чисел, представленных в форме с плавающей запятой.

1.4.8 Пояснить алгоритм умножения чисел, представленных в форме с плавающей запятой.

1.4.9 Пояснить алгоритм деления чисел, представленных в форме с плавающей запятой.

1.4.10 Что является причинами возникновения погрешностей вычислений при реализации арифметических операций на ЭВМ?

1.5 Литература, рекомендованная для изучения

1.5.1 Елович, И. В. Информатика : учебник для студ. высш. учеб. заведений / И. В. Елович, И. В. Кулибаба ; под редакцией Г. Г. Раннева. – М. : Издательский центр «Академия», 2011. – 400 с. (Сер. Бакалавриат). – ISBN 978-5-7695-7975-2.

1.5.2 Савельев, А. Я. Арифметические и логические основы цифровых автоматов : учебник / А. Я. Савельев. – М. : Высш. школа, 1980. – 255 с.

2 Расчетно-графическая работа № 2. «Поиск информации в глобальной сети Интернет»

Целью работы является развитие способности осуществления поиска информации с использованием информационных ресурсов глобальной сети Интернет, а также представления ее в требуемом формате с использованием компьютерных технологий.

2.1 Информационные ресурсы сети Интернет

Человеческое общество за годы своего существования накопило огромные массивы информации. Этот информационный ресурс рассредоточен в библиотеках (учебных, научных и др.), базах данных, архивах и других источниках информации. Без применения средств автоматизации найти необходимую информацию в короткие сроки в этом бескрайнем океане бесценных знаний практически невозможно. Неоценимую помощь в поиске информации по требуемой тематике может оказать глобальная компьютерная сеть Интернет. Она объединяет миллионы источников информации общего пользования практически по всем возможным темам.

При поиске учебной (учебно-методической) информации можно воспользоваться сервисами, обеспечивающими доступ к *образовательным ресурсам Интернета*. Все образовательные ресурсы сети Интернет можно разделить на следующие виды: *федеральные; региональные; учебное книгоиздание и образовательная пресса; конференции, выставки, конкурсы, олимпиады; инструментальные программные средства; электронные библиотеки, словари, энциклопедии* и ряд других.

Вид окна сайта **Федеральных образовательных ресурсов** (<http://www.nios.ru/FederalEducationResources/>) показан на рисунке 7. Окно содержит ссылки на другие сайты с информацией, относящейся к образованию.

Студентов вузов может заинтересовать сайт *информационной системы «Единое окно доступа к образовательным ресурсам»* (<http://window.edu.ru>) (рисунок 8). На этом сайте в свободном доступе (можно скачать бесплатно или читать онлайн и

без регистрации) имеется подборка учебной литературы для высших учебных заведений (профессиональное образование). Для удобного поиска учебной литературы по интересующей тематике она сгруппирована по нескольким направлениям: «Гуманитарное и социальное образование», «Математика и естественно-научное образование», «Образование в области техники и технологий» и др.

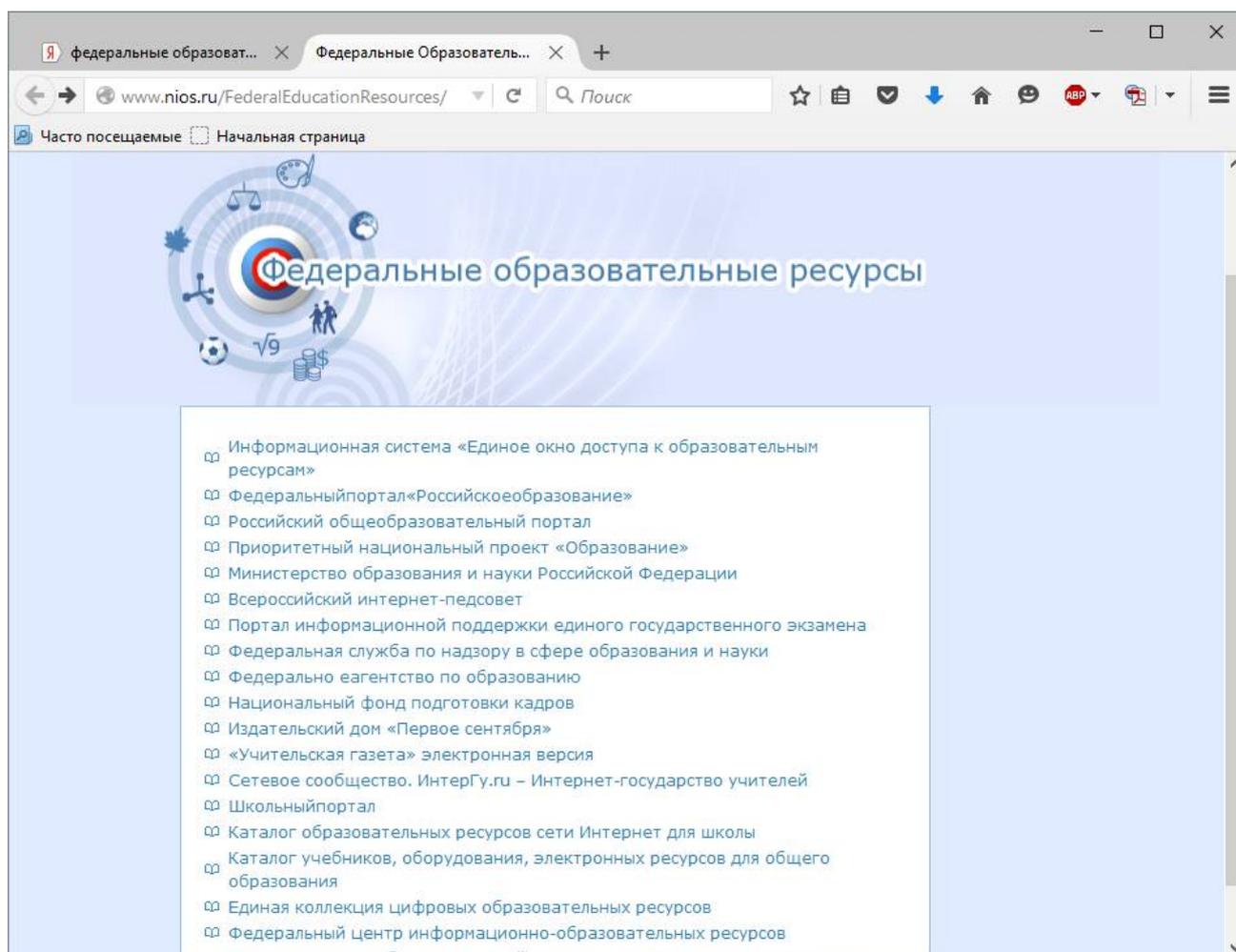


Рисунок 7

Поиск необходимой учебной литературы на сайте информационной системы «Единое окно доступа ...» целесообразно начинать с открытия каталога (рисунок 8), и далее перемещаться по ссылкам до интересующего раздела (например, **Профессиональное образование** → **Образование в области техники и технологий** → **Электроника. Радиотехника** → **Радиотехнические цепи и сигналы**). В разделе имеются ссылки на учебные издания, доступ к которым разрешен, а также приводится краткая аннотация для каждого издания. При поиске информации по ключе-

вым словам (по фамилии автора или по названию учебного издания) можно использовать простой или расширенный поиск. Для этого в правой верхней части окна имеется поле для ввода ключевых слов и кнопка для выбора вида поиска (простого или расширенного).

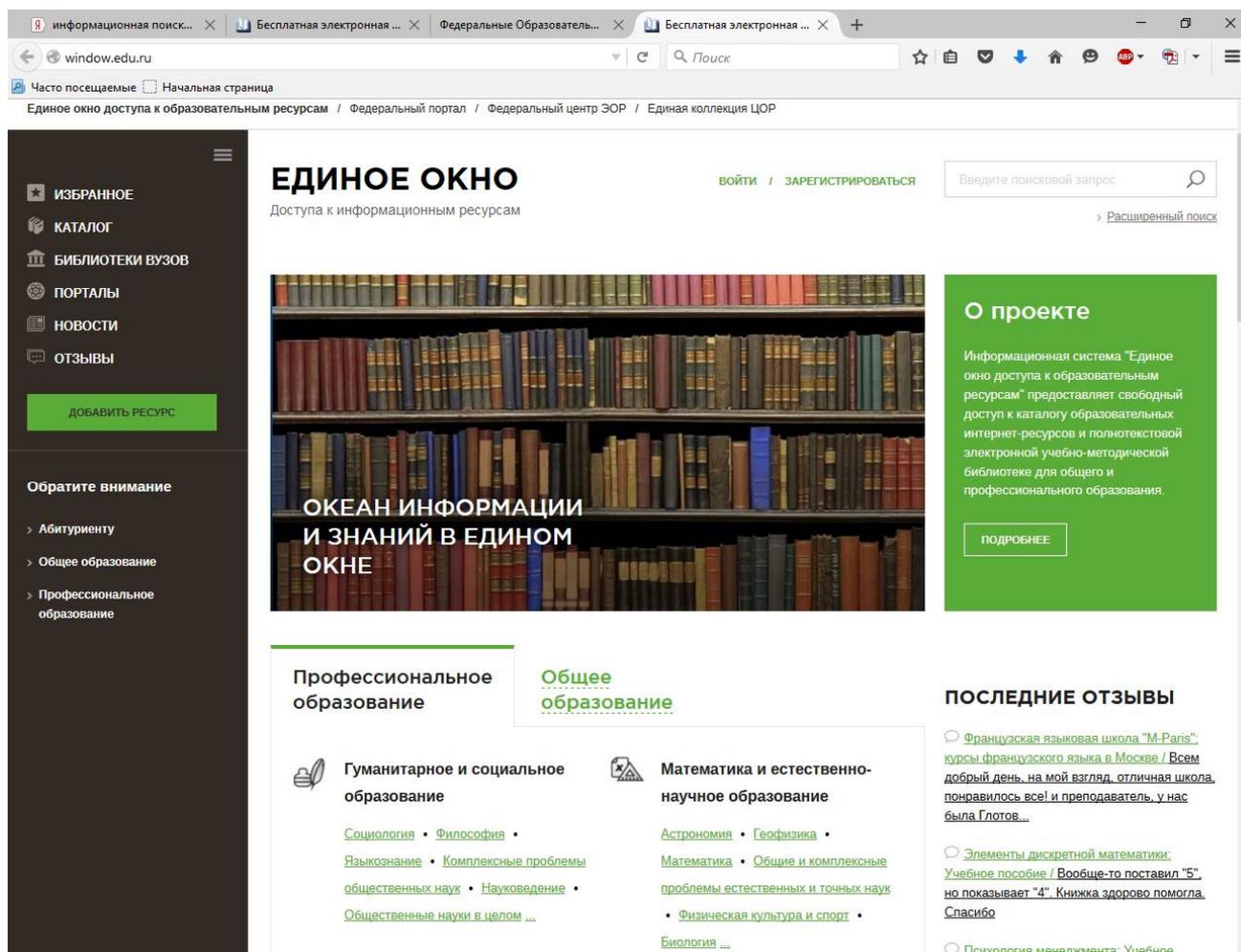


Рисунок 8

На сайте Российской национальной библиотеки (<http://www.nlr.ru/>) содержится большой каталог ссылок «Информационные ресурсы в сети Интернет» (рисунок 9), к которому можно перейти по ссылкам: **Ресурсы** → **Ресурсы Интернет**. Используя этот каталог можно найти различные документы, хранящиеся в электронном виде: книги, журнальные статьи, ГОСТы, патенты и т. д.

Кроме перечисленных сайтов существует огромное множество и других, позволяющих находить не только учебную информацию, но и информацию других видов (художественную литературу, музыкальные произведения и т. п.). Большой по-

пулярностью пользуются сайты сетевых энциклопедий (например, свободная энциклопедия «Википедия») и справочники с определениями понятий и справочными данными.

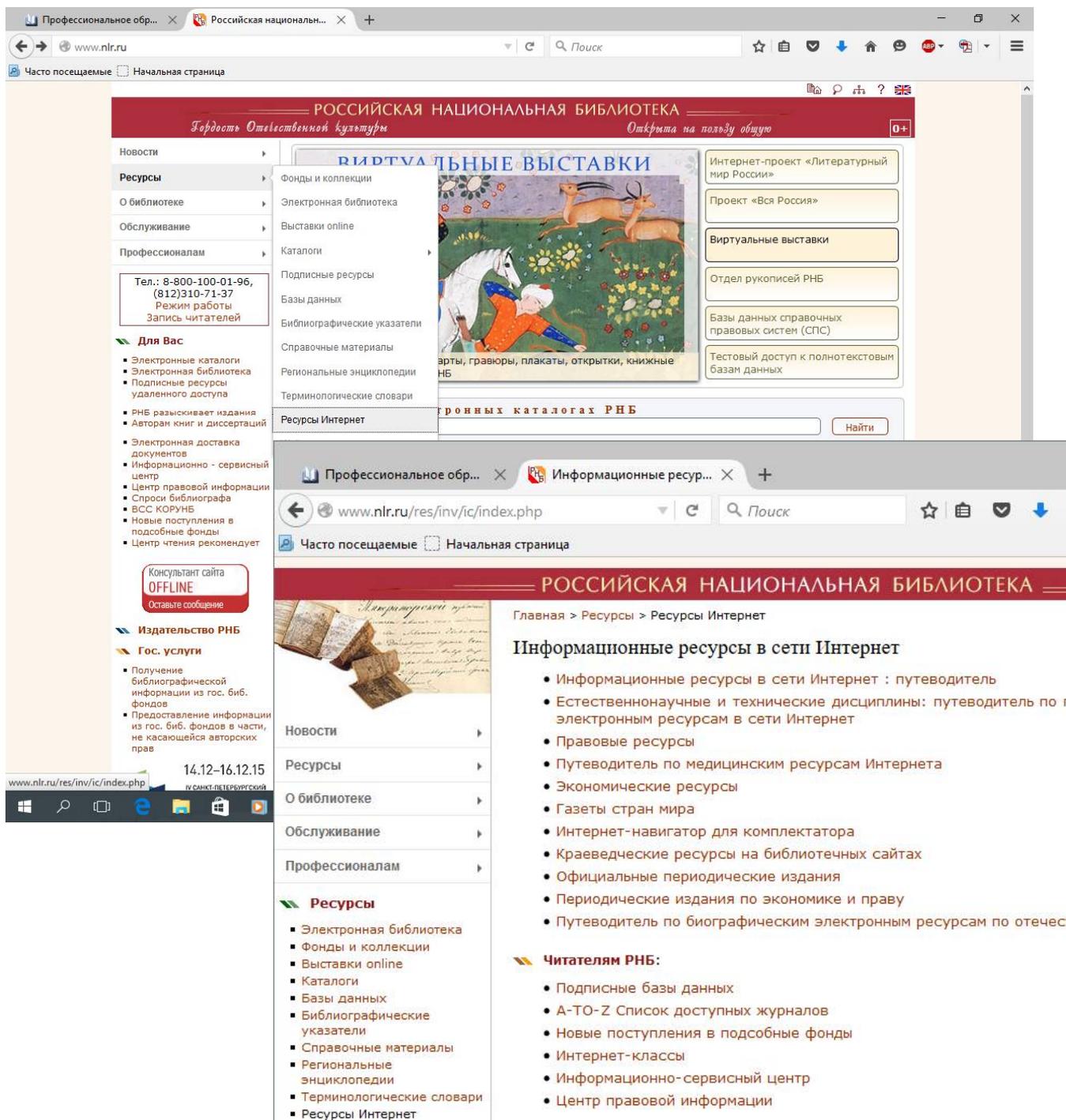


Рисунок 9

2.2 Поиск информации с помощью информационных поисковых систем

Если источник (или его местонахождение), содержащий требуемый информационный ресурс, заранее не известен, то при поиске информации в сети Интернет используют информационные поисковые системы.

Информационно-поисковая система (ИПС) – это компьютерная система, обеспечивающая поиск и отбор необходимых данных в специальной базе с описаниями источников информации на основе информационно-поискового языка и соответствующих правил поиска. Главной задачей любой ИПС является поиск информации, наиболее полно отвечающей информационным потребностям пользователя.

Различают несколько разновидностей ИПС, используемых для поиска информации в сети Интернет:

- поисковые и метапоисковые системы (Google, Яндекс, Рамблер, MetaCrawler, Nigma и др.);

- индексированные каталоги (Yahoo, Magellan, @Rus, Weblist, Улитка).

Поисковая система – это специальный web-сайт, на котором можно сделать *запрос* и получить ссылки на документы и сайты, соответствующие запросу. В состав поисковой системы, как правило, входит несколько мощных серверов.

Запрос, составляемый пользователем, состоит из ключевого слова (набора ключевых слов) или фразы, заключенной в кавычки.

Наиболее известными являются web-сайты ИПС Google (www.google.ru), Яндекс (www.yandex.ru) и Рамблер (www.rambler.ru).

Метапоисковая система – это система, способная послать запросы пользователя одновременно **нескольким поисковым серверам**, затем объединить полученные результаты и представить их пользователю в виде документа со ссылками.

Индексированный каталог – это поисковая система с классифицированным по темам списком аннотаций со ссылками на web-ресурсы.

Поиск в каталоге проводится посредством последовательного уточнения тем. Результат поиска в каталоге представляется в виде списка, состоящего из краткого описания документов с гипертекстовой ссылкой на первоисточник.

Процедуру поиска информации в глобальной сети Интернет с использованием ИПС можно представить в виде схемы, изображенной на рисунке 10 [2.7.1]. Поиск начинается с *определения информационной потребности и выбора ИПС* (держателя информационного массива). На следующем этапе формулируется *информационный запрос* – входное сообщение в ИПС, содержащее требование на поиск и выдачу информации. Обычно запрос представляет собой слово или набор слов, характеризующих искомую информацию.

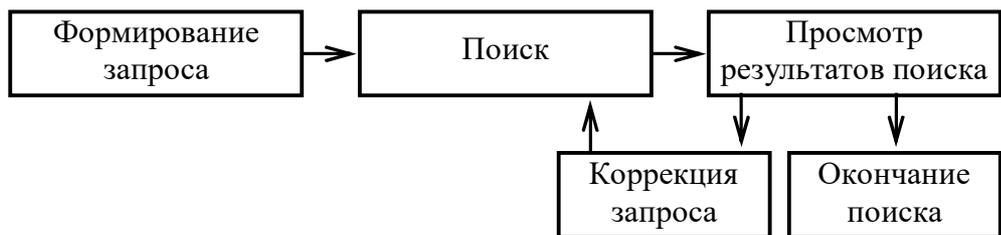


Рисунок 10

Информационный запрос записывают в поле поисковой строки ИПС (рисунок 11). После нажатия на кнопку *Найти* система осуществляет поиск затребованной информации.

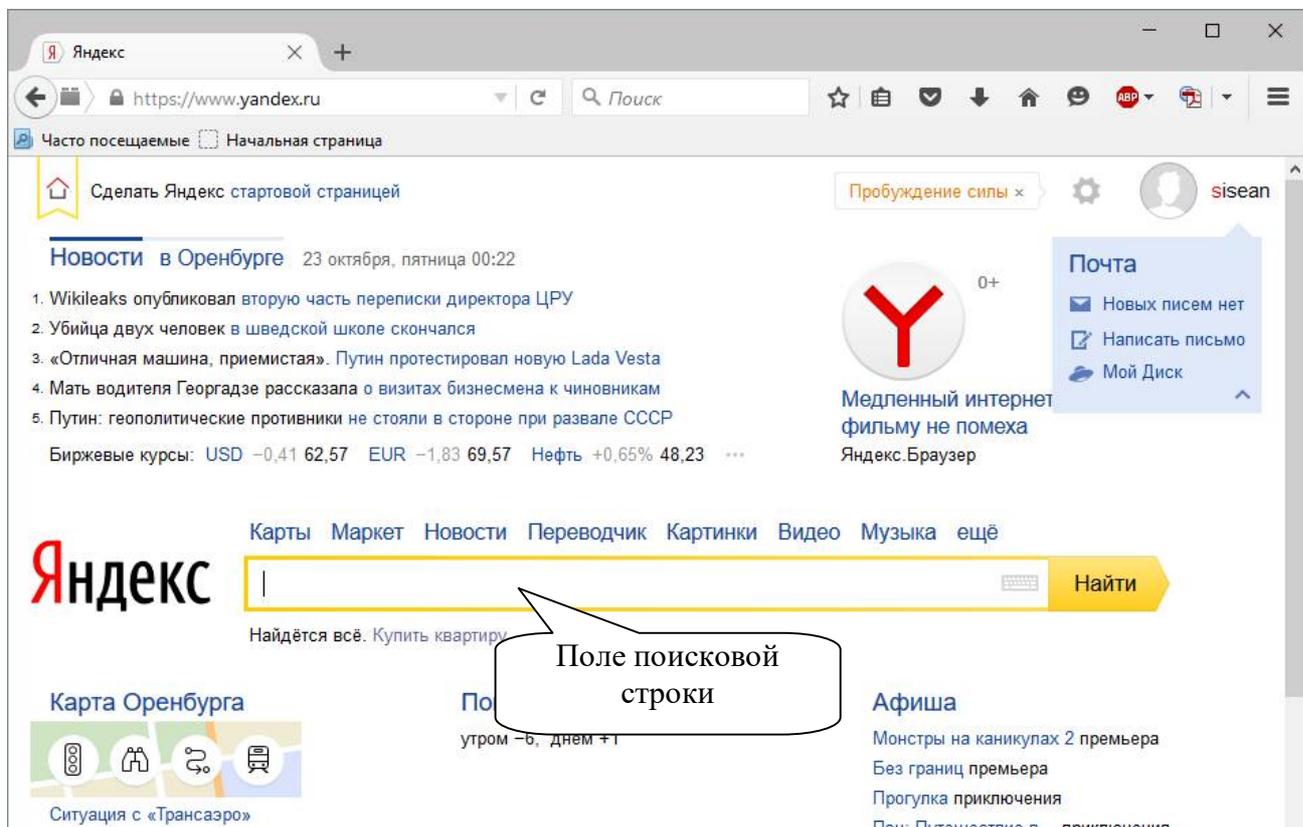


Рисунок 11

После завершения поисковой операции откроется страница с результатами поиска (на ней отображаются ссылки на сайты, на которых имеются документы, содержащие слова, включенные в поисковый запрос). После ознакомления с полученной информацией и оценки результатов поиска пользователь принимает решение на окончание поиска или на его продолжение после коррекции информационного запроса.

В общем случае процедура поиска с использованием ИПС – процедура итеративная, то есть за этапом выдачи результатов поиска следует коррекция запроса (при необходимости), поиск по скорректированному запросу и т. д. Коррекция запроса происходит в зависимости от количества полученных документов и степени их соответствия запросу пользователя.

В зависимости от полноты соответствия найденных документов запросу пользователя он может *сузить* или *расширить область поиска*, перейдя к более общим или, наоборот, более специфичным терминам. С этой целью можно применить *расширенный поиск* (рисунок 12), используя дополнительные признаки искомой информации.

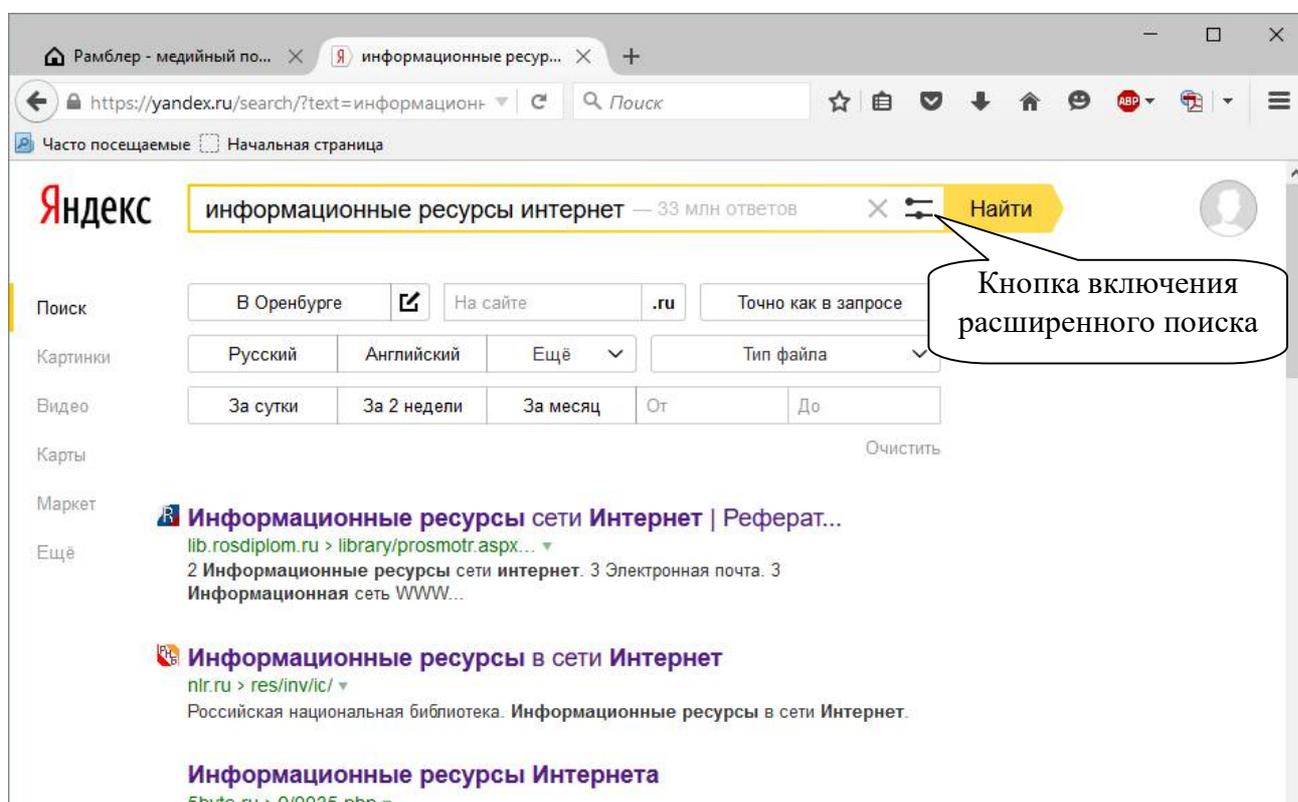


Рисунок 12

В некоторых случаях существенно упростить поиск интересующей информации позволяет использование так называемых *топов (tops)* – списков наиболее посещаемых сайтов по конкретной тематике. На рисунке 13 в качестве примера показан список ссылок ТОП100 ИПС Рамблер. Ссылки в списке сгруппированы по темам. Перемещаясь от ссылки к ссылке (например, **Наука** → **Технические** → **Электроника**) можно выйти непосредственно к адресам сайтов, содержащих публикации по выбранной тематике (рисунок 13).

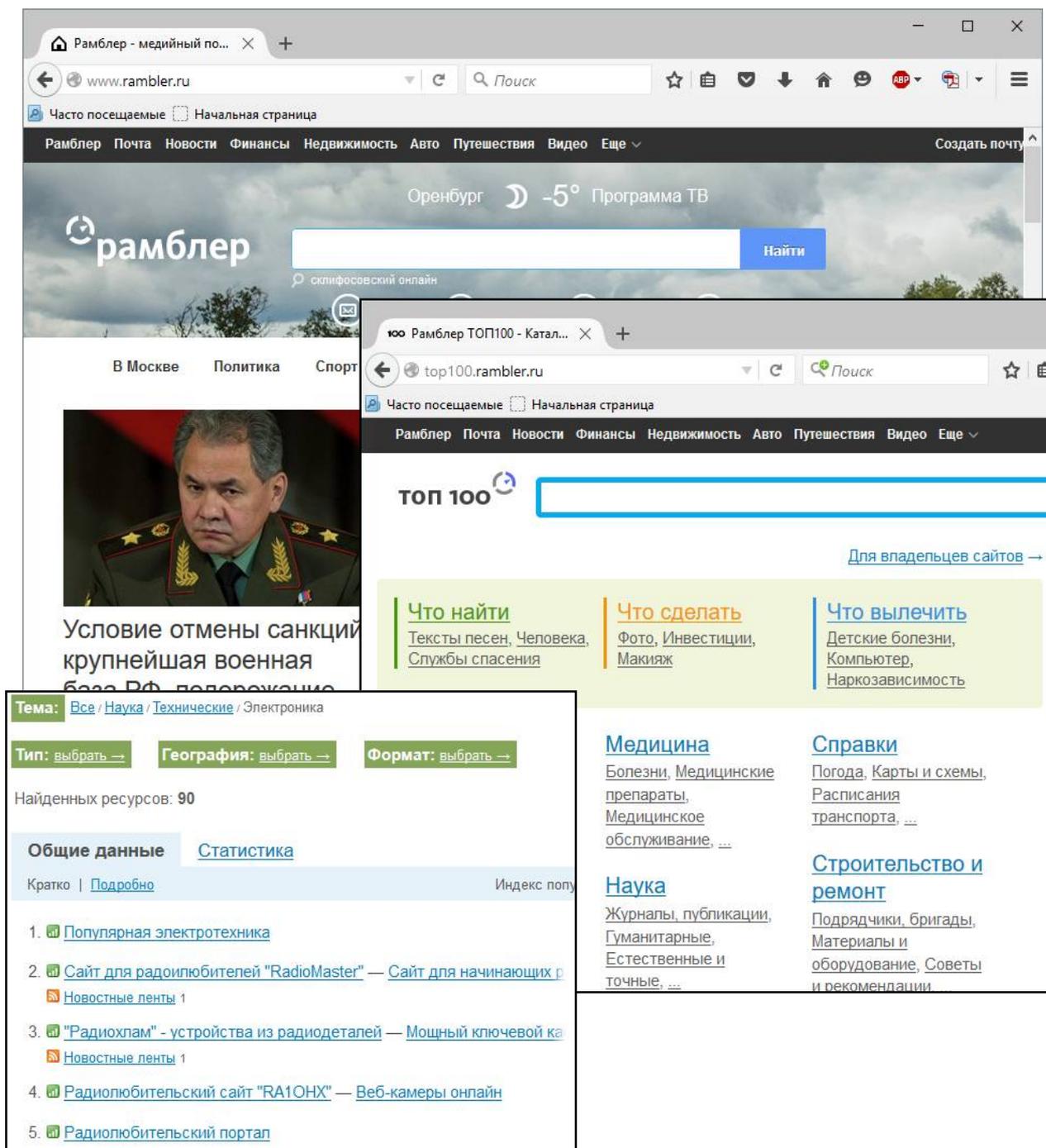


Рисунок 13

2.3 Поиск учебной информации с использованием электронных библиотечных систем

При поиске образовательных электронных информационных ресурсов по заданной тематике можно воспользоваться электронными библиотечными системами (ЭБС). *Электронная библиотечная система* – это предусмотренный федеральными государственными образовательными стандартами высшего образования (ФГОС ВО) России обязательный элемент библиотечно-информационного обеспечения студентов вузов, представляющий собой базу данных, содержащую издания учебной, учебно-методической и иной литературы, используемой в образовательном процессе. ЭБС обеспечивают широкий доступ **зарегистрированным** студентам вузов и специалистам, занятым в учебном процессе, к учебным материалам.

Для соответствия требованиям ФГОС ВО образовательные учреждения могут воспользоваться как **внешней** ЭБС, так и создать собственную электронную библиотечную систему. В Оренбургском государственном университете с сайта научной библиотеки университета (<http://artlib.osu.ru/>) обеспечивается доступ к нескольким ЭБС (рисунок 14). Поиск образовательных электронных информационных ресурсов на сайтах ЭБС выполняется таким же образом, как и на сайтах ИПС (с использованием каталогов или поисковой строки).

2.4 Задание на выполнение расчетно-графической работы

2.4.1 Используя информационную поисковую систему, найти определение (пояснение смыслового значения) термина в соответствии с вариантом задания (таблица 2). Информацию в отчете представить в виде текста, отформатированного с учетом требований СТО 02069024.101-2014. Указать адрес сайта – источника информации.

Примечание – При выполнении задания сайт свободной энциклопедии «Википедия» не использовать.

2.4.2 Используя доступные электронные библиотечные системы, найти учебник (учебное пособие), указанный в таблице 3 (в соответствии с вариантом индиви-

дуального задания), и скопировать первую страницу текста из заданного раздела. Материал представить в отчете в виде рисунка.

Оформить библиографическую запись на использованный учебник (учебное пособие).

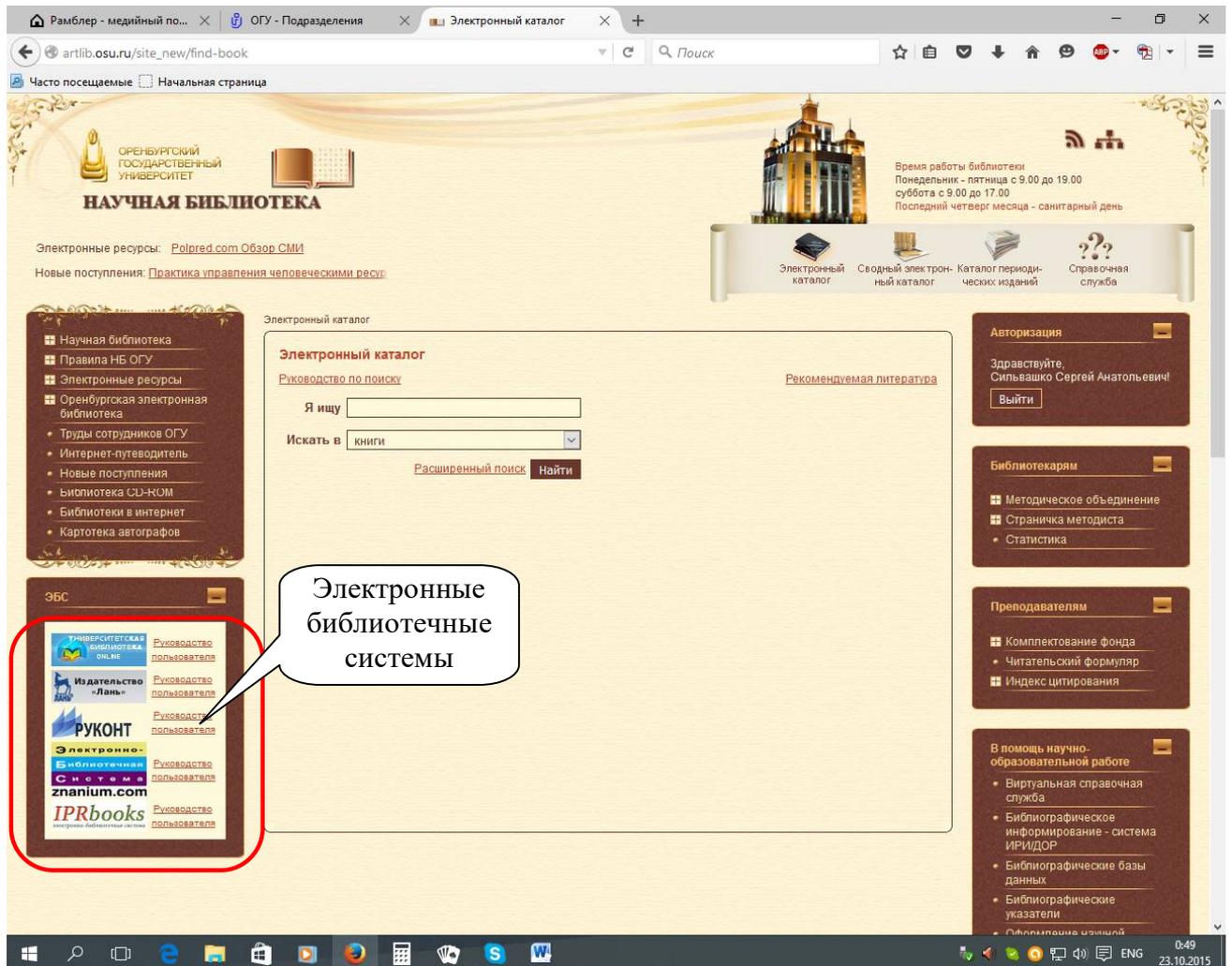


Рисунок 14

2.5 Варианты индивидуальных заданий

Таблица 2 – Варианты индивидуальных заданий к п. 2.4.1

Номера вариантов	Термин	Номера вариантов	Термин
1	Контроллер внешнего устройства	11	Принтер
2	Информация	12	Видеокарта
3	Информационная технология	13	Сетевая карта
4	Винчестер	14	Сканер
5	Концентратор	15	Информационный процесс
6	Витая пара	16	Плоттер
7	База данных	17	Коаксиальный кабель
8	База знаний	18	Компьютерный вирус
9	Информационное общество	19	Сообщение
10	Система программирования	20	Нейронная сеть

Таблица 3 – Варианты индивидуальных заданий к п. 2.4.2

Номера вариантов	Учебник (учебное пособие)	Название раздела
1	Чекмарев Ю.В. Локальные вычислительные сети	Архитектура компьютерных сетей
2	Боресков А.В. Компьютерная графика: динамика, реалистические изображения	Работа с отдельными точками
3	Березин Б.И. Начальный курс С и С++	Базовые типы данных
4	Кремень Ю.А. Основы работы в Word	Использование стилей
5	Бойченко Г.Н. Информационные сервисы Интернет в профессиональной деятельности педагога	Настройка аккаунта Google
6	Шустова Л.И. Базы данных	История развития СУБД
7	Партыка Т.Л. Информационная безопасность	Методы и системы защиты информации
8	Чекмарев Ю.В. Локальные вычислительные сети	Топологии ЛВС
9	Боресков А.В. Компьютерная графика: динамика, реалистические изображения	Рисование окружностей
10	Березин Б.И. Начальный курс С и С++	Массивы и указатели
11	Кремень Ю.А. Основы работы в Word	Создание оглавления документа

Продолжение таблицы 3

Номера вариантов	Учебник (учебное пособие)	Название раздела
12	Бойченко Г.Н. Информационные сервисы Интернет в профессиональной деятельности педагога	Редактирование таблицы
13	Шустова Л.И. Базы данных	Системы баз данных
14	Партыка Т.Л. Информационная безопасность	Информационная безопасность
15	Чекмарев Ю.В. Локальные вычислительные сети	Сетевые операционные системы ЛВС
16	Боресков А.В. Компьютерная графика: динамика, реалистические изображения	Рисование дуг эллипса
17	Березин Б.И. Начальный курс С и С++	Управляющие операторы
18	Кремень Ю.А. Основы работы в Word	Создание закладки
19	Бойченко Г.Н. Информационные сервисы Интернет в профессиональной деятельности педагога	Загрузка документа
20	Шустова Л.И. Базы данных	Файловые системы

2.6 Примерный перечень контрольных вопросов

2.6.1 Что понимают под информационными ресурсами сети Интернет?

2.6.2 Что называется информационной поисковой системой?

2.6.3 Какими способами можно осуществлять поиск информации на сайте ИПС?

2.6.4 Пояснить основные этапы процедуры поиска информации в глобальной сети Интернет.

2.6.5 Какие разновидности ИПС существуют?

2.6.6 Привести примеры названий информационных поисковых систем.

2.6.7 Что собой представляет поисковый запрос?

2.6.8 В чем различие между простым и расширенным поиском?

2.6.9 В чем различия между ИПС и электронной библиотечной системой?

2.6.10 Привести названия электронных библиотечных систем, доступ к которым обеспечивается в Оренбургском государственном университете. На сайтах каких доступных ЭБС можно найти учебные пособия по техническим дисциплинам?

2.7 Литература, рекомендованная для изучения

2.7.1 Гаврилов, М. В. Информатика и информационные технологии : учебник для бакалавров / М. В. Гаврилов, В. А. Климов. – 3-е изд., перераб. и доп. – М. : Издательство Юрайт, 2013. – 378 с. – Серия : Бакалавр. Базовый курс. – ISBN 978-5-9916-2576-0.

2.7.2 Михеева, Е. В. Практикум по информатике : учеб. пособие для студ. учреждений сред. проф. образования / Е. В. Михеева. – 9-е изд., стер. – М. : Издательский центр «Академия», 2011. – 192 с. – ISBN 978-5-7695-8117-5.

2.7.3 Гусева, Е. Н. Информатика [Электронный ресурс] : учебное пособие / Е. Н. Гусева, И. Ю. Ефимова, Р. И. Коробков и др. – 3-е изд., стереотип. – М. : Флинта, 2011. – 260 с. – ISBN 978-5-9765-1194-1. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=83542>. – ЭБС «Университетская библиотека онлайн».

2.7.4 Кадырова, Г. Р. Информатика [Электронный ресурс] : учебно-практическое пособие / Г. Р. Кадырова; Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Ульяновский государственный технический университет», д. и. Институт. – 2-е изд., дополн. и перераб. – Ульяновск : УлГТУ, 2013. – 228 с. – ISBN 978-5-9795-1151-1. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=363404>. – ЭБС «Университетская библиотека онлайн».