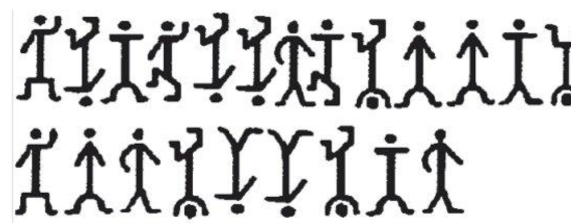
Микропроцессорная техника в приборах, системах и комплексах

Лекция 5

Кодирование информации (часть 1)



Ушаков Андрей Николаевич, старший преподаватель кафедры 303

Представление чисел в ЭВМ

- Любое число может быть представлено в виде:

$$A = \pm M_x q^{P_x}$$

где:

M_x – мантисса числа,

P_x – порядок числа,

q – основание системы счисления.

Если $q^{-1} \le |M_x| < 1$, то число называется **нормализованным**.

Формы представления чисел

В зависимости от того, как в ЭВМ представляется порядок P_x, различают 2 формы представления чисел:

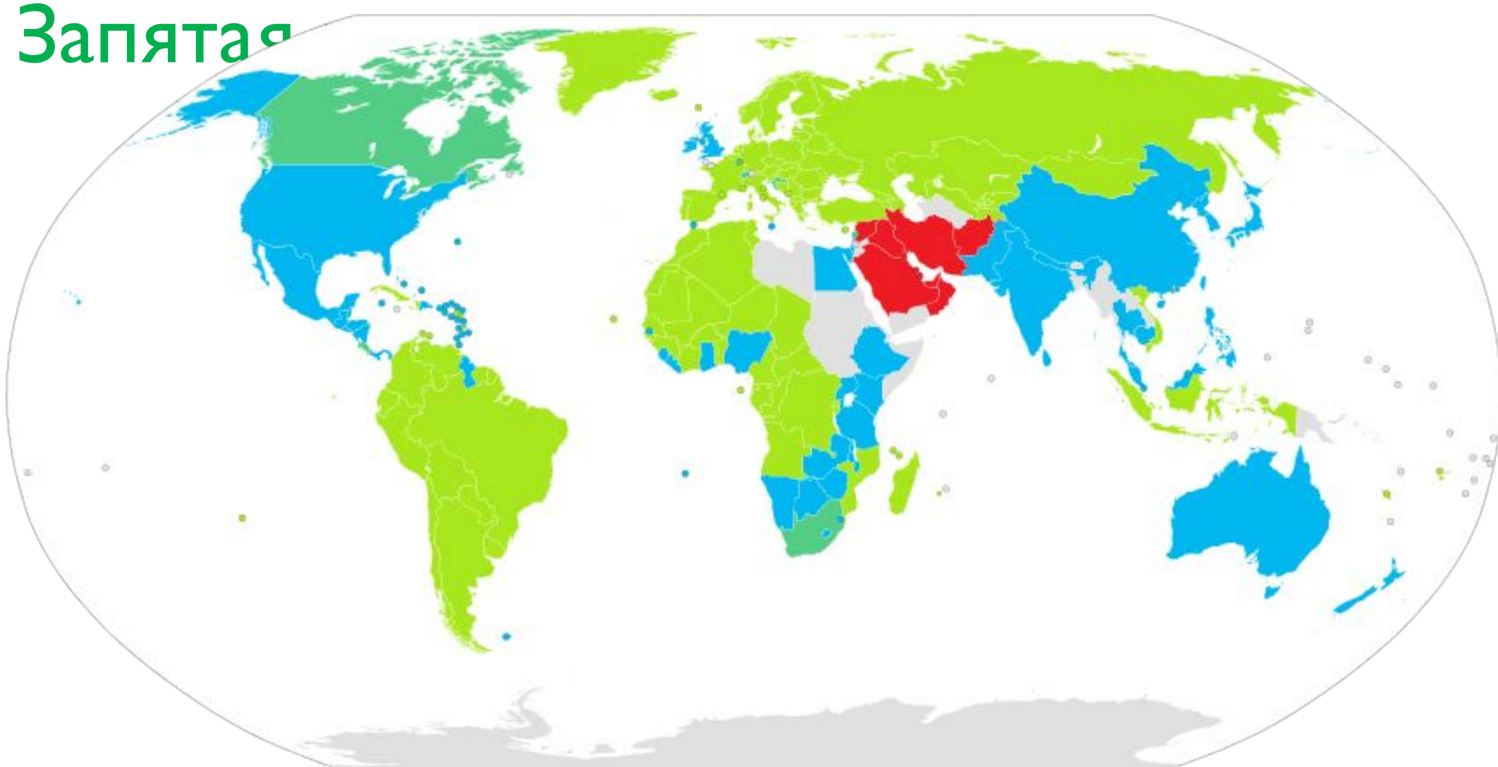
- представление числа в форме с фиксированной запятой (или точкой) (ФФЗ или ФФТ);
- представление числа в форме с плавающей запятой (или точкой) (ФПЗ или ФПТ).

Десятичный разделитель целой и дробной частей числа в мире:

Запятая Точка Мумаййез

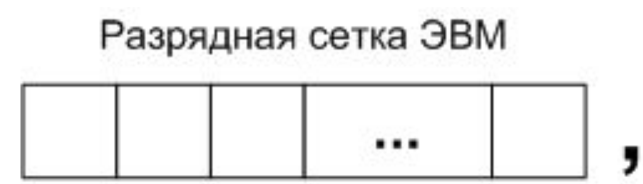
Неизвестно

Запятая

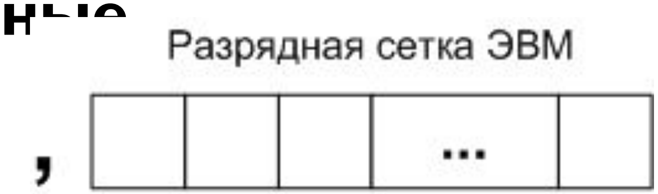


Представление чисел в ФФЗ

Если запятая фиксируется в конце числа **после последнего разряда (младший значащий разряд, МЗР)**, то все числа в ЭВМ представляются **целыми**.



Если запятая фиксируется **перед старшим значащим разрядом (СЗР) числа**, то все числа в ЭВМ **дробные**.



Диапазон представимых чисел

- Для дробных чисел:

$$|X|_{min} = 0,\underbrace{00 \ldots 01}_{\text{n}} = 2^{-n};$$

$$|X|_{max} = 0,\underbrace{11 \ldots 11}_{\text{n}} = 1 - 2^{-n}.$$

Следовательно, диапазон представимых чисел:

$$2^{-n} \le |X| \le (1 - 2^{-n}).$$

Переполнение разрядной сетки и машинный нуль

Числа, выходящие **за правую границу** диапазона, **не могут быть представлены** в ЭВМ. Говорят, что произошло **переполнение разрядной сетки** ЭВМ.

Числа, выходящие **за левую границу** диапазона, **представляются машинным нулем**. Говорят, что произошла **потеря значимости (антипереполнение)**.

Таким образом диапазон представимых в ЭВМ чисел зависит от длины разрядной сетки ЭВМ.

Ошибка представления чисел

- Считается, что **абсолютная ошибка** представления числа **составляет половину цены деления МЗР** этого числа.

Для q=2: $|\Delta| = 0{,}5 \cdot 2^{-n}$.

Тогда для относительной ошибки имеем:

$$|\delta|_{max} = \frac{|\Delta|}{X_{min}} = \frac{0{,}5 \cdot 2^{-n}}{2^{-n}} = 0{,}5 = 2^{-1};$$

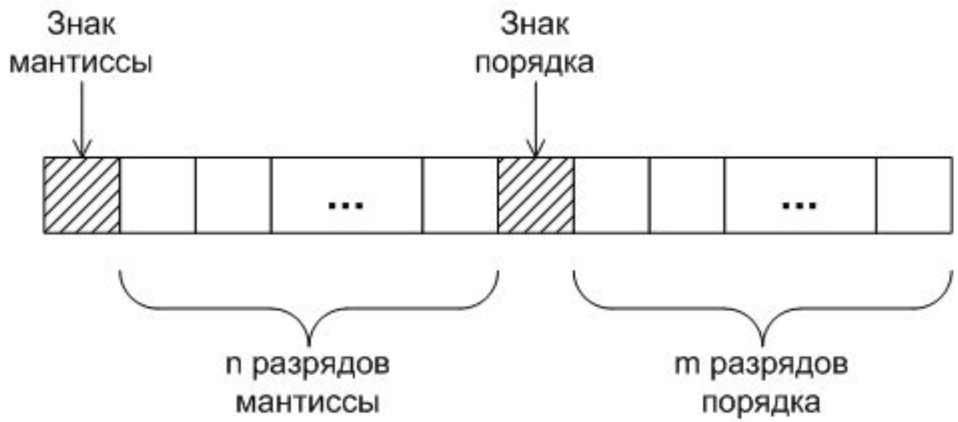
$$|\delta|_{min} = \frac{|\Delta|}{X_{max}} = \frac{0{,}5 \cdot 2^{-n}}{1 - 2^{-n}} \cong 0{,}5 \cdot 2^{-n} = 2^{-(n+1)}.$$

Диапазон ошибок:

$$2^{-(n+1)} \leq |\delta| \leq 2^{-1}.$$

Представление чисел в ФПЗ

- $$X = \pm M_x \cdot q^{\pm P_x}$$



Мантисса числа всегда нормализована.

Число n разрядов мантиссы определяет точность представления числа.

Число m разрядов порядка задаёт диапазон представимых чисел.

Диапазон представимых чисел

- Для q=2 имеем:

$$|X|_{max} = 0,\underbrace{11\ldots11}_{n} \cdot 2^{\underbrace{11\ldots11}_{m}} = (1-2^{-n}) \cdot 2^{2^m-1}$$

$$|X|_{min} = 0,1 \cdot 2^{-\underbrace{11\ldots11}_{m}} = 2^{-1} \cdot 2^{-2^m}$$

Следовательно, диапазон представимых чисел:

$$2^{-1} \cdot 2^{-2^m} \le |X| \le (1-2^{-n}) \cdot 2^{2^m-1}$$

Ошибка представления мантиссы

- Абсолютная ошибка представления мантиссы: $|\Delta| = 0{,}5 \cdot 2^{-n}$.

Тогда для относительной ошибки имеем:

$$|\delta|_{max} = \frac{|\Delta|}{X_{min}} = \frac{0{,}5 \cdot 2^{-n}}{2^{-1}} = 2^{-n};$$

$$|\delta|_{min} = \frac{|\Delta|}{X_{max}} = \frac{0{,}5 \cdot 2^{-n}}{1 - 2^{-n}} \cong 2^{-(n+1)}.$$

Точность представления чисел и точность вычислений

Следует отличать **точность представления** чисел от **точности вычислений**.

Точность вычислений зависит от чисел верных знаков в исходных данных и от метода вычислений.

Отметим, что при работе с числами в ФФЗ при длительных вычислениях происходит накопление ошибки, чего нет при работе с числами в ФПЗ.

Стандарт IEEE 754-2008

Стандарт описывает:

- Формат ЧПЗ: мантисса, порядок, знак числа;
- Представление «+0», «–0», «+∞», «–∞», NaN (Not-a-Number, *нечисло*);
- Исключительные ситуации: деление на нуль, переполнение, потерю значимости, работу с денормализованными числами;
- Методы для преобразования числа при выполнении математических операций;
- Операции арифметические и др.

Используемые в ЭВМ форматы ЧПЗ

Стандарт IEEE 754-2008 определяет 5 основных форматов ЧПЗ:

- Двоичные:
  - Одинарной точности (binary32);
  - Двойной точности (binary64);
  - Четверной точности (binary128).
- Десятичные:
  - Decimal64;
  - Decimal128.

Сводная таблица основных форматов ЧПЗ

Название	Основание	Кол-во разрядов мантиссы	Порядок (бит)	Смещение порядка
Binary32	2	23+1*	8	$2^7 - 1=127$
Binary64		52+1*	11	$2^{10} - 1=1023$
Binary128		112+1*	15	$2^{14} - 1=16383$
Decimal64	10	16	9,58	398
Decimal128		34	13,58	6176

Наиболее употребляемыми форматами ЧПЗ являются форматы одинарной (binary32) и двойной (binary64) точности.

* – целая часть мантиссы; она всегда есть и равна 1, и поэтому её разряд не включается в состав формата в явном виде.

ЧПЗ одинарной точности

Знак	Порядок	Мантисса	
0	0 1 1 1 1 1 0 0	0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	= 0,15625

31 … 24 | 23 … 16 | 15 … 8 | 7 … 0

Порядок числа:

$$1111100_2 = 124_{10} \Rightarrow 124_{10} - 127_{10} = -3_{10}$$

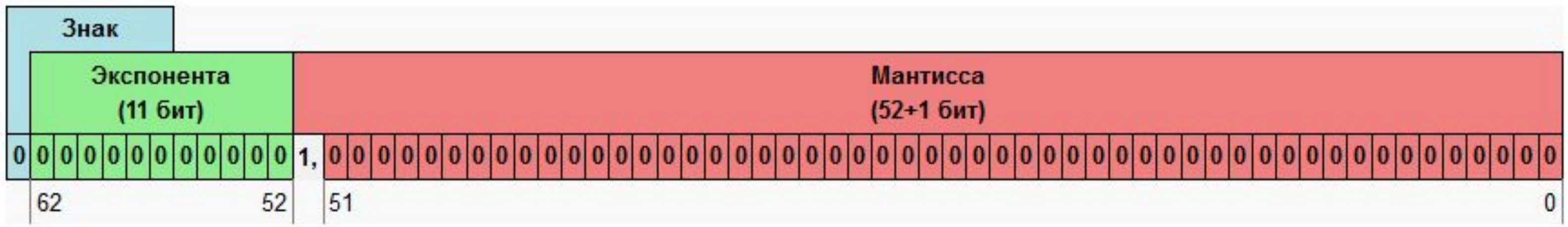
Мантисса числа:

$$1{,}01_2 = 1 \cdot 2^0 + 1 \cdot 2^{-2} = 1{,}25_{10}$$

Число:

$$1{,}25 \cdot 2^{-3} = 0{,}15625$$

ЧПЗ двойной точности



Например, если число:

0100000001011110110111010010111100011010100111111011111001110111

Порядок:

$$10000000101_2 = 1029_{10} \Rightarrow 1029_{10} - 1023_{10} = 6_{10}$$

Мантисса:

$$1{,}1110110111010010111100011010100111111011111001110111_2 = 1{,}929_{10}$$

Число:

$$1{,}929 \cdot 2^6 = 123{,}456$$

ЧПЗ формат Decimal64

знак	комбинация	порядок	мантисса
1 бит	5 бит	8 бит	50 бит
s	mmmmm	xxxxxxxx	cccccccccccccccccccccccccccccccccccccccccccccccccc

Комбинация	СЗР порядка	СЗР мантиссы	Другое
00mmm	00	0xxx	–
01mmm	01	0xxx	–
10mmm	10	0xxx	–
1100m	00	100x	–
1101m	01	100x	–
1110m	10	100x	–
11110	–	–	+∞, –∞
11111	–	–	NaN

DPD – Densely Packed Decimal (плотно упакованная десятичная дробь)

Представление числа

- Пусть X – некоторое число в q-ичной системе счисления и имеет n разрядов целой части и m разрядов дробной части:

$$X = sign(a_{n-1}a_{n-2}\dots a_1a_0a_{-1}\dots a_{-m}),$$

где a_i – цифры, а **sign** – функция знака числа.

Прямой код числа

- Прямым кодом числа X называется целое (k+n+m)-разрядное число, определяемое формулой:

$$[X]_{\text{пр}} = \begin{cases} \overbrace{00\ldots0}^{k}\,\overbrace{a_{n-1}\ldots a_1 a_0}^{n}\,\overbrace{a_{-1}\ldots a_{-m}}^{m}, X \geq 0 \\ \underbrace{(q-1)(q-1)\ldots(q-1)}_{k}\,\underbrace{a_{n-1}\ldots a_1 a_0}_{n}\,\underbrace{a_{-1}\ldots a_{-m}}_{m}, X \leq 0 \end{cases}$$

Старшие k разрядов отводятся под знак числа.

Число нуль имеет в прямом коде два представления:

$$[+0]_{\text{пр}} = \underbrace{00\ldots0}_{k}\,\underbrace{00\ldots0}_{n+m}$$

$$[-0]_{\text{пр}} = \underbrace{(q-1)(q-1)\ldots(q-1)}_{k}\,\underbrace{00\ldots0}_{n+m}$$

Применение, достоинство и недостатки прямого кода

Прямые коды применяются в устройствах ввода/вывода и в запоминающих устройствах.

Достоинство прямого кода – удобство представления чисел.

Недостатки прямого кода:

- Необходимо различать знаковые и числовые разряды, так как они по разному участвуют в арифметических операциях.
- Операции «+» и «–» производятся по разным алгоритмам.
- 2 представления числа нуль.

Обратный (инверсный) код числа

- Обратным (инверсным) кодом числа X называется целое (k+n+m)-разрядное число, определяемое формулой:

$$[X]_{об} = \begin{cases} \overbrace{00\ldots0}^{k}\,\overbrace{a_{n-1}\ldots a_1 a_0}^{n}\,\overbrace{a_{-1}\ldots a_{-m}}^{m}, X \geq 0 \\ \underbrace{(q-1)(q-1)\ldots(q-1)}_{k}\,\underbrace{\bar{a}_{n-1}\ldots \bar{a}_1 \bar{a}_0}_{n}\,\underbrace{\bar{a}_{-1}\ldots \bar{a}_{-m}}_{m}, X \leq 0 \end{cases}$$

$$\bar{a}_i = (q-1) - a_i$$

Старшие k разрядов отводятся под знак числа.

Число нуль имеет в обратном коде два представления:

$$[+0]_{об} = \underbrace{00\ldots0}_{k}\,\underbrace{00\ldots0}_{n+m}$$

$$[-0]_{об} = \underbrace{(q-1)(q-1)\ldots(q-1)}_{k}\,\underbrace{(q-1)(q-1)\ldots(q-1)}_{n+m}$$

Выполнение операций в обратном коде

При алгебраическом сложении чисел в обратных кодах знаковые разряды числа участвуют в операции наравне с цифровыми. Если возникает **перенос** из старшего знакового разряда, то он **суммируется к младшему цифровому разряду**.

Дополнительный код

- Дополнительным кодом числа X называется целое (k+n+m)-разрядное число, определяемое формулой:

$$[X]_{\text{доп}} = \begin{cases} \overbrace{00\ldots0}^{k}\ \overbrace{a_{n-1}\ldots a_1 a_0}^{n}\overbrace{a_{-1}\ldots a_{-m}}^{m}, X > 0 \\ \underbrace{(q-1)(q-1)\ldots(q-1)}_{k}\ \underbrace{\bar{a}_{n-1}\ldots \bar{a}_1 \bar{a}_0}_{n}\underbrace{\bar{a}_{-1}\ldots(\bar{a}_{-m}+1)}_{m}, X \le 0 \end{cases}$$

Старшие k разрядов отводятся под знак числа.

Число нуль имеет в дополнительном коде одно представление.

Выполнение операций в дополнительном коде

При алгебраическом сложении в дополнительном коде знаковые и цифровые разряды числа участвуют одинаково. Если возникает **перенос** из старшего знакового разряда, то он **отбрасывается**.

Модифицированные коды

Рассмотренные коды позволяют выполнять операции «+» и «–» по единому алгоритму, как операцию сложения. При этом не требуется разделение числа на знаковые и цифровые разряды.

Если число разрядов, отводимое под знак числа, >1, то коды называются ***модифицированными***.

Переполнение разрядной сетки ЭВМ

В ЭВМ для записи машинного слова отводится определенное число разрядов, называемое длиной разрядной сетки.

При сложении чисел одного знака возможно появление результата, превышающего длину разрядной сетки. При этом старшая цифра числа попадает в знаковый разряд.

Использование модифицированных кодов позволяет сохранить знак результата.

Признак переполнения разрядной сетки ЭВМ

Неодинаковое содержимое старшего знакового разряда, сохраняющего знак результата, и младшего знакового разряда, содержащего вышедшую за разрядную сетку цифру, служит признаком переполнения разрядной сетки ЭВМ.

С целью экономии оборудования в большинстве ЭВМ принято отводить под знак числа $k=2$ разряда.

При этом старший разряд называется **знаковым**, а младший – разрядом **переполнения**.

Ситуации при выполнении вычислений

Разряд знака	Разряд переполнения	Ситуация
0	0	Результат >0, переполнения нет
0	1	Результат >0, переполнение есть
1	0	Результат <0, переполнение есть
1	1	Результат <0, переполнения нет

Пример

Инверсный код

$4_{10}+(-2_{10})=2_{10}$

$4_{10}=00100_2$

$-2_{10}=11101_2$

$2_{10}=00010_2$

$$\begin{array}{r} {}^{1\ 1} \\ +\ 00100 \\ 11101 \\ \hline 1\ 00001 \\ +\ 1 \\ \hline 00010 \end{array}$$

Дополнительный код

$4_{10}+(-2_{10})=2_{10}$

$4_{10}=00100_2$

$-2_{10}=11110_2$

$2_{10}=00010_2$

$$\begin{array}{r} {}^{1\ 1} \\ +\ 00100 \\ 11110 \\ \hline \cancel{1}\ 00010 \end{array}$$

Знак «+»

Разряды равны, значит <u>переполнения нет</u>

Двоично-десятичный код (ДДК, BCD – Binary-Coded Decimal)

В ДДК каждая десятичная цифра записывается 4-разрядным двоичным кодом.

Поскольку используются только 10 из 16 возможных двоичных комбинаций, ДДК не является экономичным.

10-ич. СС	ДДК			
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Преимущества и недостатки ДДК

Преимущества:

- Упрощенный вывод на индикацию;
- Для дробных чисел не теряется точность при переводе в десятичный формат и наоборот;
- Упрощенные операции «*» и «/» на 10 и округление.

Недостатки:

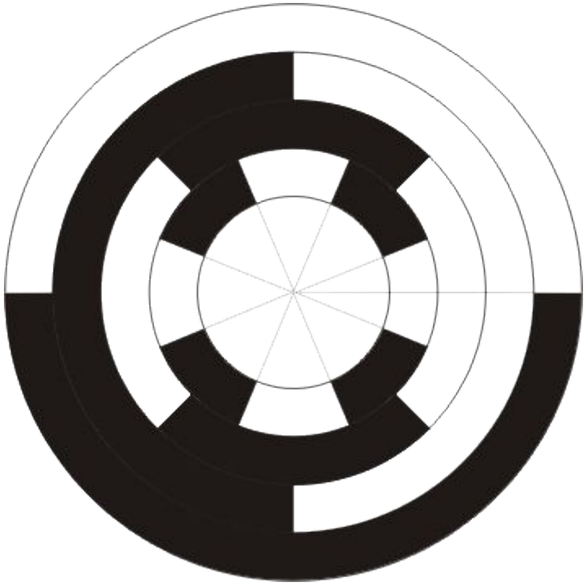
- Повышенный расход ресурсов памяти;
- Усложнены операции «+» и «–».

Код Грея

Код Грея – такой код, каждое следующее значение которого получается из предыдущего изменением только одного разряда.

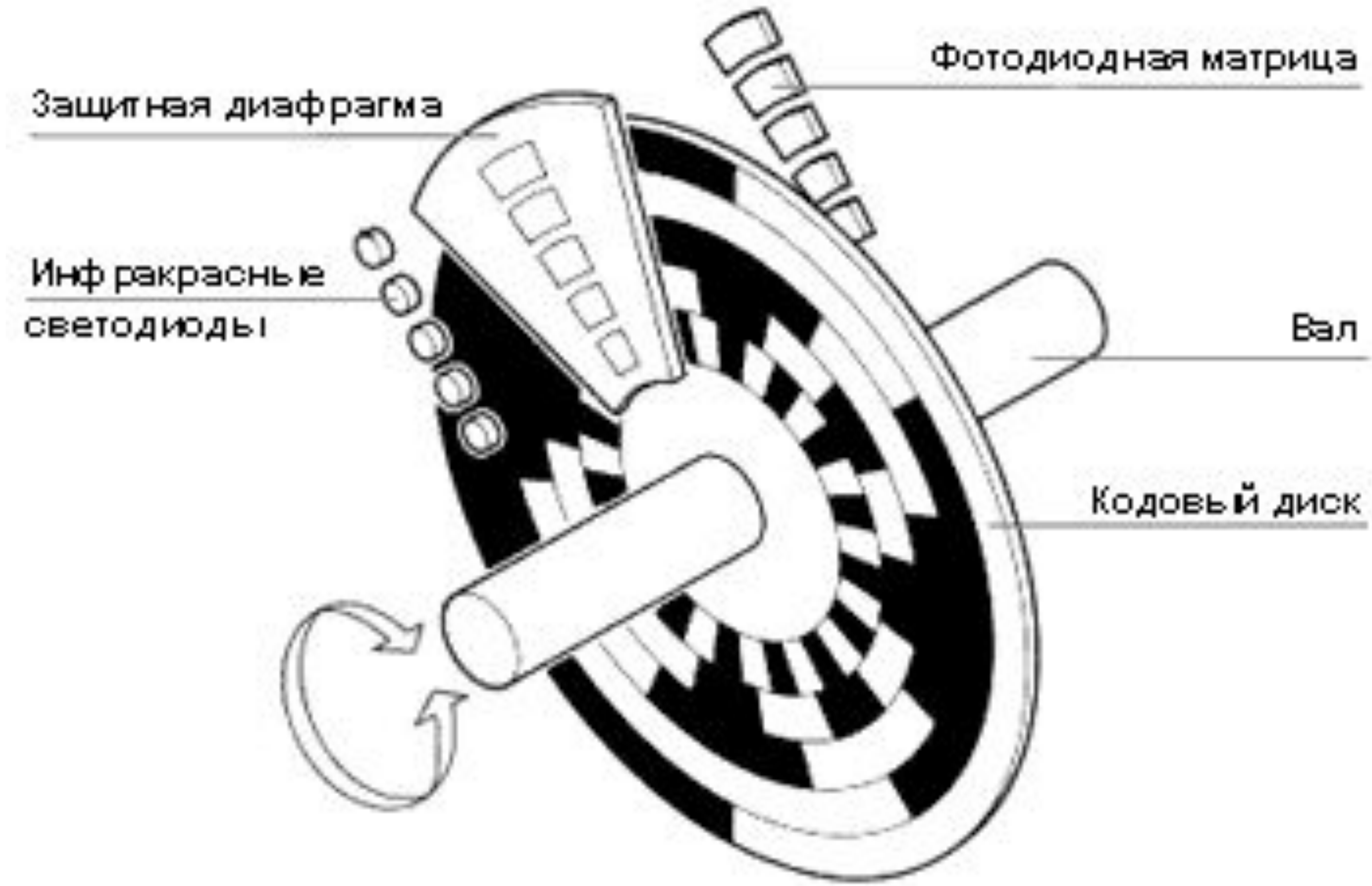
Имеет повышенную помехозащищенность. Используется в датчиках-энкодерах.

1887 – 1969



Диск кругового 4-разрядного энкодера

Принцип действия энкодера



ДДК, код Грея и двоичный код

Десятичная система счисления	Двоичная система счисления	Двоичный код	Код Грея	ДДК
		8 4 2 1		10 1
0	0	0 0 0 0	0000	0000 0000
1	1	0 0 0 1	0001	0000 0001
2	2	0 0 1 0	0011	0000 0010
3	3	0 0 1 1	0010	0000 0011
4	4	0 1 0 0	0110	0000 0100
5	5	0 1 0 1	0111	0000 0101
6	6	0 1 1 0	0101	0000 0110
7	7	0 1 1 1	0100	0000 0111
8	8	1 0 0 0	1100	0000 1000
9	9	1 0 0 1	1101	0000 1001
10	A	1 0 1 0	1111	0001 0000
11	B	1 0 1 1	1110	0001 0001
12	C	1 1 0 0	1010	0001 0010
13	D	1 1 0 1	1011	0001 0011
14	E	1 1 1 0	1001	0001 0100
15	F	1 1 1 1	1000	0001 0101

Приложение А. Нечисло (1)

Нечисло (англ. Not-a-Number, NaN) – особое состояние ЧПЗ. Может возникнуть, например, когда математическая операция завершилась с неопределённым результатом, или если в ячейку памяти попало неудовлетворяющее условию число.

NaN бывает «тихий» и «сигнальный». Последний вызывает исключение, тогда как первый лишь возвращает NaN в каче-стве результата операции.

Приложение А. Нечисло (2)

Операции приводящие к ответу NaN:

- Все математические операции, где один из операндов это NaN;
- $0/0$;
- ∞/∞;
- $0\cdot\infty$;
- $\infty+(-\infty)$;
- $\sqrt{-1}$;
- $\log_a$(отрицательное число).