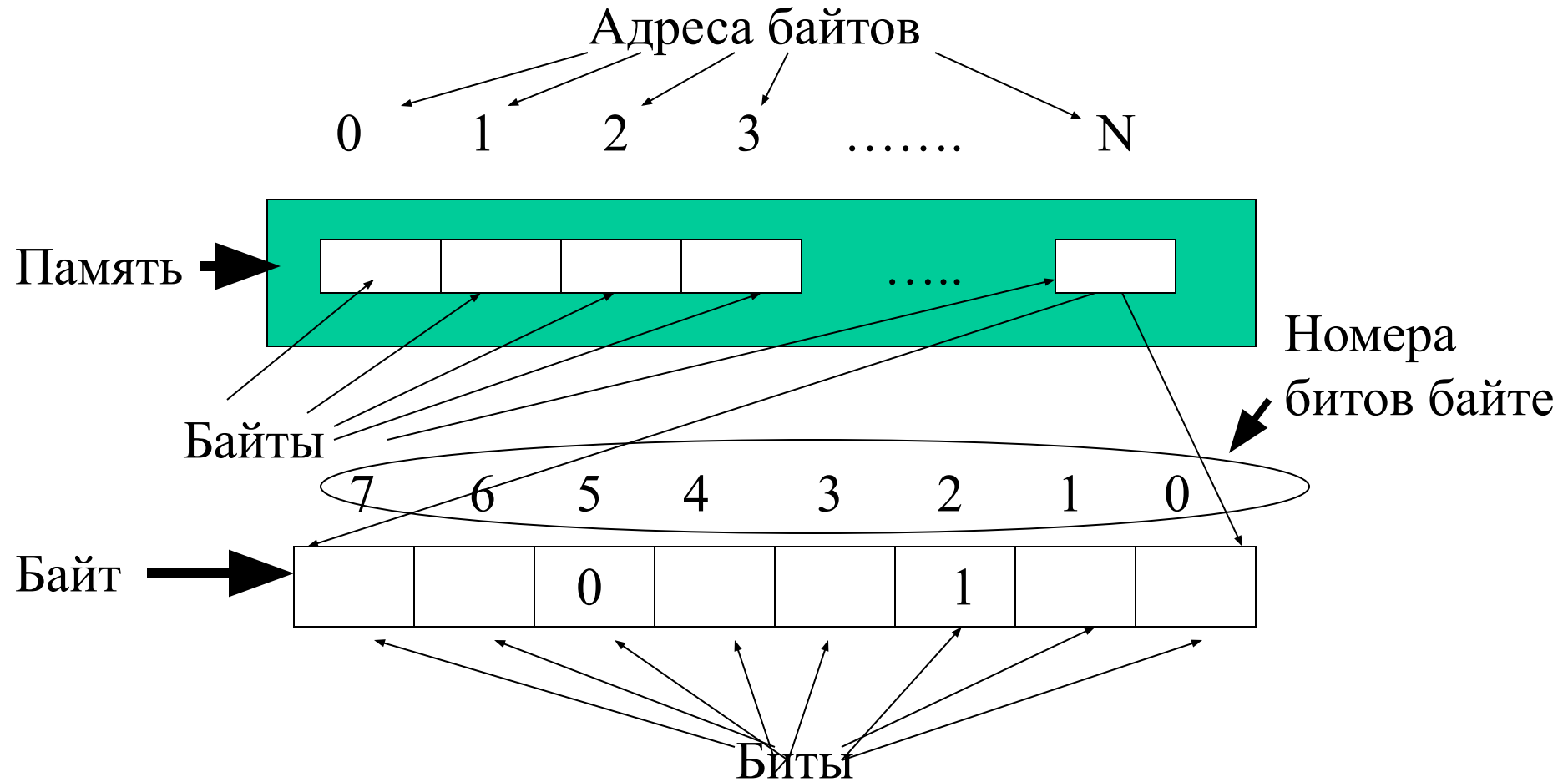


Лекция № 5.

Представление информации в ЭВМ

Структура памяти



Один байт состоит из восьми бит

Машинное слово

Последовательность битов и байтов рассматриваемых аппаратной частью ЭВМ как одно целое.

Длина машинного слова:

1 байт

2 байта

4 байта

A_1

A_2

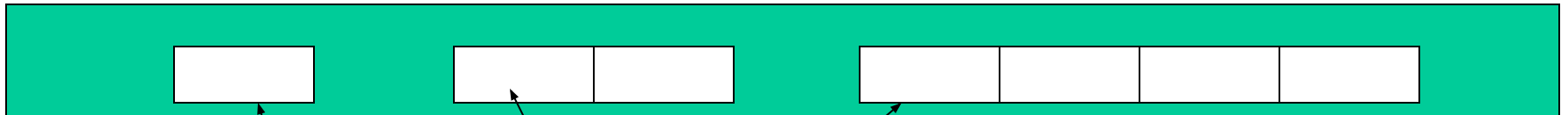
A_2+1

A_3

A_3+1

A_3+2

A_3+3



Байт определяющий адрес машинного слова

Объем (емкость) памяти

Объем памяти – наибольший объем данных, которые одновременно могут храниться в запоминающем устройстве.

Единицы измерения объема памяти:

1 бит.

1 байт = 8 бит.

1 Кбайт (кило) = 1024 байт = 2^{10} байт, 1 кбайт = 10^3 байт.

1 Мбайт (мега) = 1024 Кбайт = 2^{20} байт, 1 мбайт = 10^6 байт.

1 Гбайт (гига) = 1024 Мбайт = 2^{30} байт, 1 гбайт = 10^9 байт.

1 Тбайт (тера) = 1024 Гбайт = 2^{40} байт, 1 тбайт = 10^{12} байт.

1 Пбайт (пета) = 1024 Тбайт = 2^{50} байт, 1 пбайт = 10^{15} байт.

Виды информации

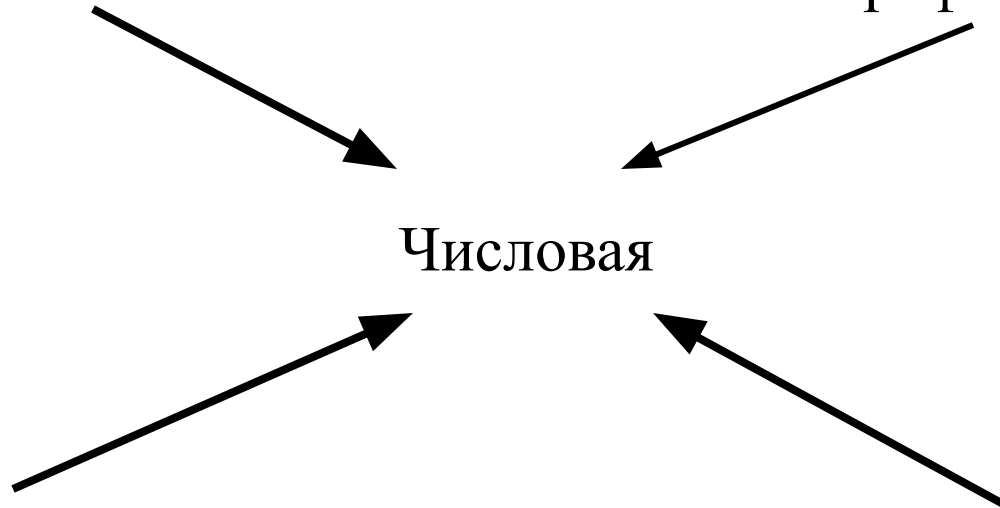
Символьная и текстовая

Графическая

Числовая

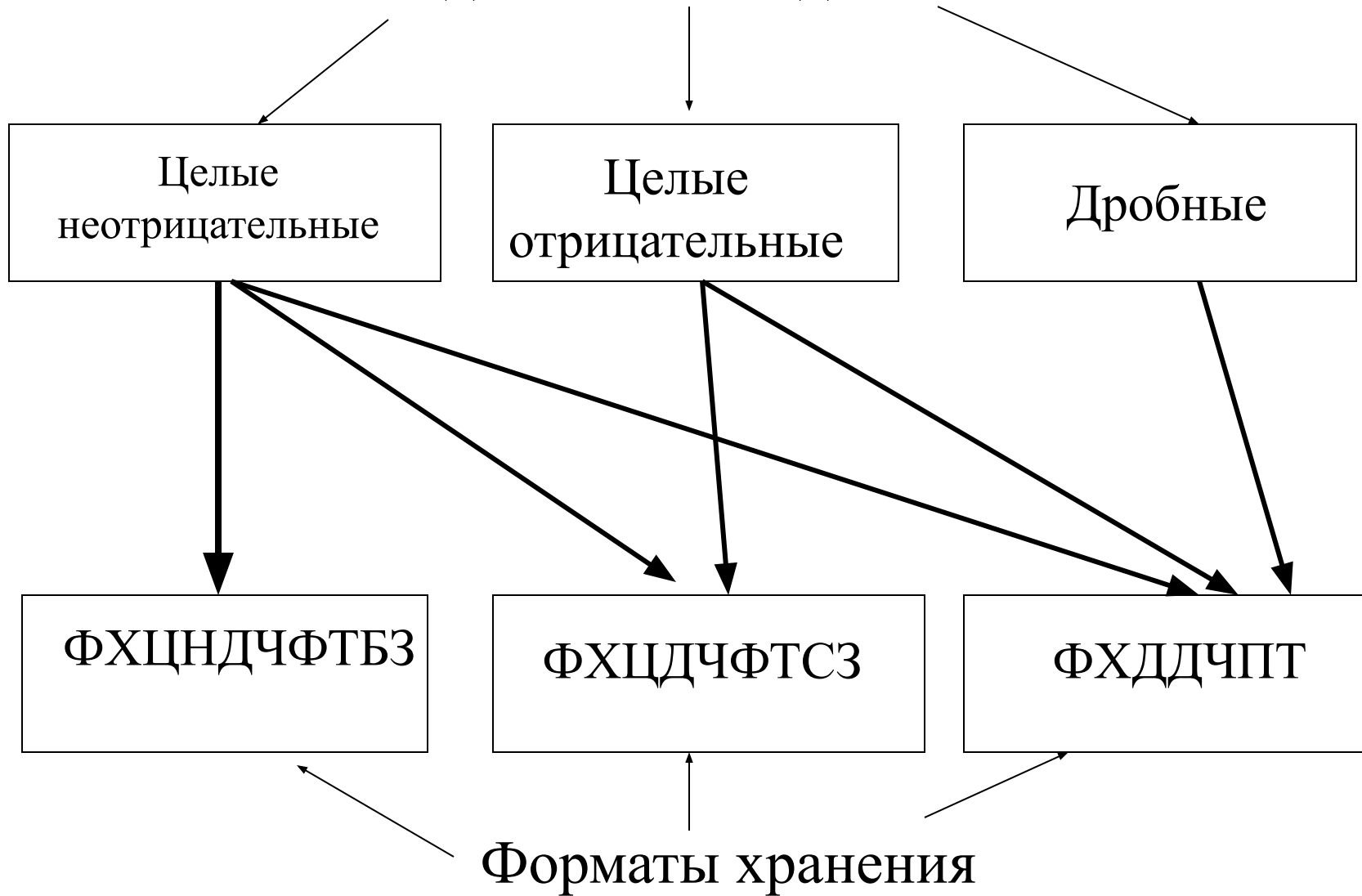
Звуковая

Видео



Форматы хранения чисел

Виды числовых данных



Формат хранения целых неотрицательных двоичных чисел с фиксированной точкой без знака

| Первый байт числа | | | | | | | | Второй байт числа | | | | | | | |
|-------------------|----------|----------|----------|----------|----------|----------|----------|-------------------|----------|----------|----------|----------|----------|----------|----------|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> | <i>b</i> |

Символ *b* обозначает двоичную цифру: 0 или 1.

Целое число над этим символом – номер ее разряда.

Цифры с номерами разрядов от 0 до 7 образуют второй байт.

Цифры с номерами разрядов от 8 до 15— первый байт числа.

Пример формата хранения числа 258_{10}

| Первый байт числа | | | | | | | | Второй байт числа | | | | | | | |
|-------------------|----|----|----|----|----|---|---|-------------------|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

Максимальное число формата

Максимальное целое неотрицательное число L определяется размером слова l в байтах, используемого для хранения числа:

$$L = 2^{8 \times l} - 1,$$

где L – максимальное целое неотрицательное число;

l – размер слова в байтах, необходимых для хранения числа.

Значение максимального числа

| Размер слова в байтах | Размер слова в битах | Значение максимального числа |
|-----------------------|----------------------|----------------------------------|
| 1 | 8 | $2^8 = 255$ |
| 2 | 16 | $2^{16} - 1 = 65535$ |
| 4 | 32 | $2^{32} - 1 =$ $=4294967295$ |
| 8 | 64 | $2^{64} - 1 > 16 \times 10^{18}$ |

Определение минимального размера слова

Найдем минимальную длину слова в байтах l_{min} , необходимого для хранения целого неотрицательного числа L :

$$\log_2(L+1) = 8 \times l;$$

$$l = \log_2(L+1) / 8.$$

Размер слова l_{min} выбирается из множества допустимых значений: 1, 2, 4, 8. Размер слова l_{min} должен быть минимальным, но не меньше l .

Пример решения задачи

Показать структуру хранения в памяти числа 505_{10} .
Использовать слово с наименьшим размером.

Переведем число 505_{10} в двоичную систему счисления.

В результате перевода получим двоичное число:

$$505_{10} = 111111001_2.$$

Минимальная количество байтов, необходимых для записи числа в память равна:

$$l = \log_2(L+1) / 8 = \log_2(505+1) / 8 = \log_2 506 / 8 \text{ (байт)}.$$

Продолжение решения примера

Очевидно: $\log_2 256 < \log_2 506 < \log_2 512$

$$\log_2 2^8 < \log_2 506 < \log_2 2^9$$

$$8 < \log_2 506 < 9$$

Откуда получаем: $1 < \log_2 506/8 < 9/8$

$$1 < \log_2 506/8 < 2$$

| Первый байт числа | | | | | | | | Второй байт числа | | | | | | | |
|-------------------|----|----|----|----|----|---|---|-------------------|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |