**Методическая тема: «Представление целых чисел в машинных кодах. Арифметические основы»**

**учителя информатики и ИКТ МБОУ ФМЛ Барулиной Н.Н.**

Для представления чисел в ЭВМ обычно используют битовые наборы — последовательности нулей и единиц фиксированной длины. Организовать обработку наборов фиксированной длины технически легче, чем наборов переменной длины. Позиция в битовом наборе называется разрядом. В ЭВМ разрядом называют также часть регистра (или ячейки памяти), хранящую один бит.

Целые числа являются простейшими числовыми данными, с которыми оперирует ЭВМ. Для целых чисел существуют два представления: беззнаковое (только для неотрицательных целых чисел) и со знаком. Очевидно, что отрицательные числа можно представлять только в знаковом виде. Целые числа в компьютере хранятся в формате с фиксированной запятой.

Представление целых чисел в беззнаковых целых типах.

Для беззнакового представления все разряды ячейки отводятся под представление самого числа. Например, в байте (8 бит) можно представить беззнаковые числа от 0 до 255. Поэтому, если известно, что числовая величина является неотрицательной, то выгоднее рассматривать её как беззнаковую.

Представление целых чисел в знаковых целых типах.

Для представления со знаком самый старший (левый) бит отводится под знак числа, остальные разряды - под само число. Если число положительное, то в знаковый разряд помещается 0, если отрицательное - 1. Например, в байте можно представить знаковые числа от -128 до 127.

Прямой код числа.

Представление числа в привычной форме "знак"-"величина", при которой старший разряд ячейки отводится под знак, а остальные - под запись числа в двоичной системе, называется прямым кодом двоичного числа. Например, прямой код двоичных чисел 1001 и -1001 для 8-разрядной ячейки равен 00001001 и 10001001 соответственно.

Положительные числа в ЭВМ всегда представляются с помощью прямого кода. Прямой код числа полностью совпадает с записью самого числа в ячейке машины. Прямой код отрицательного числа отличается от прямого кода соответствующего положительного числа лишь содержимым знакового разряда.

Правила записи прямого кода:

1. Перевести число в двоичную систему счисления.
2. Выставить знаковый разряд. (занести туда 0 или 1).
3. Записать число в сетку, занося i разряд числа в соответствующий разряд сетки.

Например, запишем число 43 в машинном представлении:

Переводим его в двоичную систему счисления:

1. 43=32+8+2+1=25+23+20+21=101011
2. Заносим знак

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. Заносим число, занося 0 разряд числа в 0 разряд сетки, 1 разряд числа в 1 разряд сетки и т. д.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |

Это машинное представление. На бумаге мы будем записывать следующим образом: ПК43=0000000000101011

*Целые числа без знака* обычно занимают в памяти *один* (byte) или *два (word)* байта и принимают в однобайтовом формате значения от 000000002 до 111111112, а в двухбайтовом формате — от 00000000 000000002 до 11111111 111111112.

**Диапазоны значений целых чисел без знака**

|  |  |
| --- | --- |
| Формат числа в байтах  | Диапазон  |
| Запись с порядком | Обычная запись |
| 1 | 0 ... 28–1 | 0 ... 255 |
| 2 | 0 ... 216–1 | 0 ... 65535 |

**Примеры:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |

 а) число 7210 = 10010002 в **однобайтовом** формате:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |

а) число 7210 = 10010002 в **двухбайтовом** формате:

С положительными числами разобрались, теперь рассмотрим, как хранятся отрицательные целые числа в компьютере.

*Отрицательные целые хранятся в машине в дополнительном коде.*

Правила записи дополнительного кода:

1. Записать прямой код числа, взятого по модулю.
2. Проинвертировать его. Это значит все разряды числа «перевернуть» 0 заменить на 1 и наоборот. Получим *обратный код* числа (ОК).
3. К полученному ОК прибавим 1.

Например : запишем машинное представление -43

1. ПК-43=0000000000101011
2. ОК-43=1111111111010100
3. 1111111111010100 + 1=1111111111010101

Получили дополнительный код числа -43

ДК-43=1111111111010101

Теперь самостоятельно заполним следующую таблицу:

|  |  |  |
| --- | --- | --- |
| число | Прямой код числа | Дополнительный код |
| -2 | 1000 0000 0000 0010 | 1111 1111 1111 1110 |
| -1 | 1000 0000 0000 0001 | 1111 1111 1111 1111 |
| 0 | 0000 0000 0000 0000 |  |
| 1 | 0000 0000 0000 0001 |  |
| 2 | 0000 0000 0000 0010 |  |

Теперь о главном: Зачем всё это надо?

Давайте сложим прямой и дополнительные коды 2 и -2:

0000 0000 0000 0010+1111 1111 1111 1110=0000000000000000

Таким образом, дополнительный код используется для упрощения выполнения арифметических операций. Если бы вычислительная машина работала с прямыми кодами положительных и отрицательных чисел, то при выполнении арифметических операций следовало бы выполнять ряд дополнительных действий. Например, при сложении нужно было бы проверять знаки обоих операндов и определять знак результата. Если знаки одинаковые, то вычисляется сумма операндов и ей присваивается тот же знак. Если знаки разные, то из большего по абсолютной величине числа вычитается меньшее и результату присваивается знак большего числа. То есть при таком представлении чисел (в виде только прямого кода) операция сложения реализуется через достаточно сложный алгоритм. Если же отрицательные числа представлять в виде дополнительного кода, то операция сложения, в том числе и разного знака, сводится к их поразрядному сложению.

В большинстве компьютеров операция *вычитания* *не* *используется*. Вместо нее производится сложение уменьшаемого с дополнительным кодом вычитаемого. Это позволяет существенно упростить конструкцию АЛУ.

Умножение и деление.

Во многих компьютерах умножение производится как последовательность сложений и сдвигов. Для этого в АЛУ имеется регистр, называемый накапливающим сумматором, который до начала выполнения операции содержит число ноль. В процессе выполнения операции в нем поочередно размещаются множимое и результаты промежуточных сложений, а по завершении операции — окончательный результат. Другой регистр АЛУ, участвующий в выполнении этой операции, вначале содержит множитель. Затем по мере выполнения сложений содержащееся в нем число уменьшается, пока не достигнет нулевого значения. Деление для компьютера является трудной операцией. Обычно оно реализуется путем многократного прибавления к делимому дополнительного кода делителя.

Ошибки при выполнении арифметических операций.

 При выполнении арифметических операций могут возникать ситуации, когда старшие разряды результата операции не помещаются в отведенной для него области памяти. Такая ситуация называется переполнением цифровой части (мантиссы) формата числа. Для обнаружения переполнения и оповещения о возникшей ошибке в компьютере используются специальные средства. Реакция на разные ошибки может быть разная. Так, в некоторых ЭВМ при делении на ноль вычисления прекращаются (фатальная ошибка), а при переполнении мантиссы устанавливается признак переполнения в так называемом регистре флагов и вычисления продолжаются.

Решение задач:

1. Выполнить в машинном коде 101-54

Решение:

101=64+32+4+1=1100101

54=32+16+4+2=110110

ДК-54=1111111111001010

ПК101=0000000001100101

 0000000000101111

1. Записать машинное представление чисел: 98; -102

Решение:

98=64+32+2=1100010 ПК98=0000000001100110 Для более короткой записи используют шестнадцатеричное представление: ПК98=0066

102=64+32+4+2=1100110 ДК—102=1111111110011010 (FF9A)

1. Дано машинное представление чисел. Найти их десятичный эквивалент:
	1. 0111
	2. 00АF
	3. FA10

Решение: а) 0000000100010001=1+16+256=273

 b) 0000000010101111=1+2+4+8+32+128=175

 c) 1111101000010000 – это отрицательное число, следовательно оно записано в дополнительном коде, переводим его в прямой 1000010111110000

16+32+64+128+256+1024= -1520

1. Дано машинное представление чисел FA76 и 0054. Выполнить сложение в машинном виде. В ответе записать десятичное представление результата:

FA76 + 0054=FACA. Получили отрицательное число. Переводим его в прямой код 8535. Следовательно получаем число 1333.

Задачи для самостоятельного решения

1. Для хранения целого числа со знаком в компьютере используется один байт. Сколько единиц содержит внутреннее представление числа -120?
2. Целочисленные беззнаковые переменные А, В, С занимают в памяти компьютера один байт. Если А=250, В=250, то в результате выполнения операции С:=А+В переменная С примет значение?
3. Переменные X1, X2, X3 имеют размер байт, тип – знаковый, В шестнадцатеричной системе счисления X1=A316, X2=7E, X3=B1. Чему равно значение выражения X=(X1-X2)\*X3 в десятичной системе счисления.
4. Переменные X1, X2, X3 имеют размер байт, тип – знаковый, В шестнадцатеричной системе счисления X1=1516, X2=BC, X3=A1. Чему равно значение выражения X=(X1-X2)\*X3 в десятичной системе счисления.

Используемые источники:

<http://kuzelenkov.narod.ru/mati/book/inform/inform5.html>

<http://www.titorov.ru/attachments/article/549/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F%20%D0%B0%D1%80%D0%B8%D1%84%D0%BC%D0%B5%D1%82%D0%B8%D0%BA%D0%B0.pdf>