В компьютерной технике для представления и обработки целых алгебраических чисел используются код со сдвигом и следующие базовые коды:

* прямой;
* обратный;
* дополнительный.

**Код со сдвигом (двоичное смещение/offset binary)**

Целочисленный отрезок от 0 до 2n сдвигается влево на 2n − 1, затем получившиеся на этом отрезке числа последовательно кодируются в порядке возрастания кодами от 000…0 до 111…1.

Двоичное смещение чаще всего используется в цифровой обработке сигналов (Digital Signal Processing).

При арифметических операциях, выполняемых в коде со сдвигом, нужно учитывать смещение, то есть проделывать на одно действие больше (например, после «обычного» сложения двух чисел у результата будет двойное смещение, одно из которых необходимо вычесть).

Из-за необходимости усложнять арифметические операции код со сдвигом для представления целых чисел используется не часто, но зато применяется для хранения порядка вещественного числа.

Достоинства кода со сдвигом:

* не требуется усложнение архитектуры процессора;
* не возникает проблема двух нулей.

Недостатки кода со сдвигом:

* при арифметических операциях нужно учитывать смещение;
* ряд положительных и отрицательных чисел несимметричен.

**Кодирование чисел в базовых кодах (прямой, обратный, дополнительный)**

Все три разновидности базового кодирования чисел (прямой, обратный и дополнительный) имеют формат представления, содержащий два поля – поле знака и поле модуля (рисунок 1.12).



Рисунок 1.12 – Формат представление алгебраических двоичных чисел

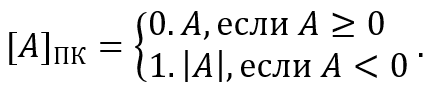
Во всех разновидностях базовой кодировки поле знака представлено одним разрядом, в котором устанавливается ноль, если число положительное, и единица, если число отрицательное. Поле модуля для каждого кода формируется по-разному. Количество разрядов поля модуля определятся диапазоном изменения отображаемых чисел (для целых чисел) или точностью их представления (для вещественных чисел).

Прямой код предназначен для хранения в памяти и представления чисел, поскольку только в нем модульное поле содержит непосредственно количественную оценку числа.

Обратный и дополнительный коды позволяют упростить арифметические операции над алгебраическими числами. Поэтому, обратный и дополнительный коды предназначены для выполнения арифметических операций над алгебраическими числами. В общем случае модульное поле в обратном/дополнительном коде не отражает количественной оценки числа.

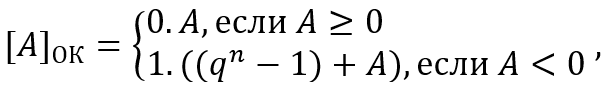
**Прямой базовый код**

При кодировании числа в прямой код запись целого числа А формируется по следующему правилу:



**Обратный базовый код (дополнение до единицы/ones' complement)**

При кодировании числа в обратный базовый код запись целого числа А формируется по следующему правилу:



где:   q – основание системы счисления;   
         n – разрядность модульного поля;  
         qn – 1 – максимальная включенная граница диапазона изменения представляемых чисел.

Достоинства обратного базового кода:

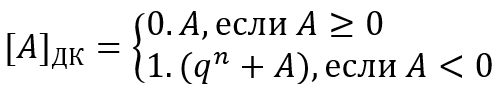
* коды положительных чисел относительно беззнакового кодирования остаются неизменными;
* количество положительных чисел равно количеству отрицательных.

Недостатки обратного базового кода:

* арифметические операции с отрицательными числами требуют усложнения архитектуры процессора;
* существует проблема двух нулей: нулю соответствуют коды 00...000 и 11...111.

**Дополнительный базовый код (дополнение до двух/twos' complement)**

При кодировании числа в дополнительный базовый код запись целого числа А формируется по следующему правилу:



где:   q – основание системы счисления;  
         n – разрядность модульного поля;   
         qn – максимальная невключенная граница диапазона изменения представляемых чисел.

Чаще всего дополнительный базовый код используется для представления отрицательных чисел.

По сути, дополнительный код представляет собой дополнение |A| до нуля.

Достоинства дополнительного базового кода:

* возможность заменить арифметическую операцию вычитания операцией сложения и сделать операции сложения одинаковыми для знаковых и беззнаковых типов данных, что существенно упрощает архитектуру процессора и увеличивает быстродействие (см. раздел 2);
* нет проблемы двух нулей.

Недостатки дополнительного базового кода:

* ряд положительных и отрицательных чисел несимметричен;
* числа в дополнительном базовом коде нельзя сравнивать как беззнаковые, или вычитать без расширения разрядности.

Несмотря на недостатки, дополнительный код в современных вычислительных системах используется при выполнении арифметических операций чаще всего.

Таким образом, из вышеописанного следует, что:

* - положительное алгебраическое число в прямом, обратном и дополнительном кодах имеет одинаковое представление (**совпадает**);
* - для того, чтобы перевести отрицательное число из обратного в прямой код (и наоборот), необходимо дополнить его модуль до включенной границы;
* - для того, чтобы перевести отрицательное число из дополнительного в прямой код (и наоборот), необходимо дополнить его модуль до невключенной границы.
* - для двоичных чисел включенной границей является значение равное 2n, а невключенной границей является значение равное 2n – 1.

*Следует обратить внимание, что обратный и дополнительный коды находятся только из прямого кода соответствующего операнда!*

**В общем виде правила формирования модуля отрицательного целого двоичного числа имеют вид:**

* чтобы сформировать модульную часть записи отрицательного числа в обратном коде, необходимо в модульной части записи этого числа в прямом коде взять обратные значения всех двоичных разрядов (т.е. проинвертировать модуль прямого кода: нули заменяются единицами, а единицы – нулями);
* для перевода отрицательного числа из обратного кода назад в прямой также необходимо проинвертировать запись модульной части числа в обратном коде;
* чтобы сформировать модульную часть записи отрицательного числа в дополнительном коде, необходимо в модульной части записи этого числа в прямом коде взять обратные значения всех двоичных разрядов (т.е. проинвертировать модуль прямого кода), и к полученному коду прибавить единицу в младший разряд;
* для перевода отрицательного числа из дополнительного кода назад в прямой также необходимо проинвертировать запись модульной части числа в дополнительном коде и к полученному коду прибавить единицу в младший разряд.

**Модифицированные коды**

Базовые коды не очень удобны при выполнении арифметических операций, поскольку при выполнении операции может произойти переполнение модульного поля, которое никак не определяется в базовом коде. Может сложиться абсурдная ситуация, когда при сложении двух отрицательных целых чисел получится положительная сумма.

Для исключения подобных ситуация существуют модификации прямого, обратного и дополнительного кода.

Поле знака в модифицированных кодах представлено двумя разрядами по следующему правилу:

* 00 – для положительных чисел;
* 11 – для отрицательных чисел.

Если в результате выполнения арифметической операции разряды в знаковом поле останутся одинаковыми (00 или 11), то переполнение модульного поля не произошло.

Если в результате выполнения арифметической операции разряды в знаковом поле окажутся разными (01 или 10) – это будет свидетельствовать о переполнении модульного поля, причем:

* старший разряд – является знакоопределяющим (0 – положительный, 1 – отрицательный);
* младший разряд не несет в себе информацию о знаке, а несет количественную оценку числа и, соответственно должен быть перемещен в модульное поле.

Таким образом, при переполнении модульного поля возможны следующие варианты:

* 01 – переполнение поля модуля, число считается положительным, единичный разряд несет количественную оценку числа и должен быть перемещен в старший разряд модульного поля;
* 10 –переполнение поля модуля, число является отрицательным, нулевой разряд несет количественную оценку числа и должен быть перемещен в старший разряд модульного поля.