Вылиток А.А., Грацианова Т.Ю.

Сложение и вычитание, установка флагов

Определим операции сложения и вычитания на множестве битовых наборов длины k. Пусть x и y – битовые наборы. Перенумеруем биты наборов справа налево от 0 до k-1. Через z_i будем обозначать i-й бит набора z, $z_i \in \{0,1\}$. Для определения результатов (суммы и разности) воспользуемся некоторыми логическими операциями (дизъюнкция, конъюнкция, отрицание, сложение по модулю два), интерпретируя 0 как ложь, 1 как истину.

Суммой (соответственно, разностью) битовых наборов x и y назовем набор z, разряды которого определяются следующими соотношениями (попутно определим набор величин c_i , i=0,…,k , $c_i \in \{0,1\}$, где c_i означает наличие (при c_i=1) или отсутствие (при c_i=0) переноса единицы в i-й разряд, а в случае вычитания – заёма из i-го разряда):

$$z = x + y \qquad \begin{cases} c_0 = 0, \\ \text{для } i = 0,\dots,k-1 \\ z_i = x_i \oplus y_i \oplus c_i\ , \\ c_{i+1} = x_i y_i \vee x_i c_i \vee y_i c_i \end{cases} \qquad\qquad z = x - y \qquad \begin{cases} c_0 = 0, \\ \text{для } i = 0,\dots,k-1 \\ z_i = x_i \oplus y_i \oplus c_i\ , \\ c_{i+1} = y_i c_i \vee z_i c_i \vee y_i z_i \end{cases}$$

Здесь $\oplus$ означает операцию сложения по модулю два, $\vee$ – дизъюнкцию, знак конъюнкции (умножения) опущен.

Если рассматривать битовые наборы как представление беззнаковых целых чисел или знаковых в дополнительном коде, то определенные выше операции над битовыми наборами реализуют сложение и вычитание целых чисел по модулю 2^k. Во многих ЭВМ схемы сложения и вычитания чисел устроены именно так. Поскольку при подобных операциях результат может быть искажен (из-за переполнения мантиссы), для отслеживания этой ситуации одновременно с результатом устанавливаются специальные флаги.

Флаг – это бит, принимающий значение 1 ("флаг установлен"), если выполнено некоторое условие, и значение 0 ("флаг сброшен") в противном случае. Наиболее употребительны флаги: CF (carry flag, флаг переноса; устанавливается при искажении результата операции над числами без знака, например, полученная сумма "не умещается" в k-разрядную ячейку), OF (overflow flag, флаг переполнения; фиксирует искажение результата операции над знаковыми числами), ZF (zero flag, флаг нуля; фиксирует нулевой результат), SF (sign flag, флаг знака; устанавливается в 1, если в операции над знаковыми числами получился отрицательный результат).

Заметим, что сложение и вычитание битовых наборов приводит к установке значений всех флагов, независимо от того, трактуем мы эти наборы как беззнаковые числа или как числа со знаком. (Для анализа результата в знаковом случае бесполезен CF, в беззнаковом бесполезны OF и SF).

Сформулируем правила формирования значений флагов в терминах битовых наборов. В результате сложения или вычитания флаги CF, SF, ZF, OF получают следующие значения:

$$CF = c_k\ , \quad SF = z_{k-1}\ , \quad ZF = \bar{z}_{k-1}\bar{z}_{k-2}\dots\bar{z}_0\ , \quad OF = \begin{cases} \bar{x}_{k-1}\bar{y}_{k-1}z_{k-1} \vee x_{k-1}y_{k-1}\bar{z}_{k-1} & \text{при} \quad \text{сложении,} \\ \bar{x}_{k-1}y_{k-1}z_{k-1} \vee x_{k-1}\bar{y}_{k-1}\bar{z}_{k-1} & \text{при} \quad \text{вычитании} \end{cases}$$

Горизонтальная черта сверху (надчеркивание) означает логическое отрицание.

Правила установки значений флагов можно определить и в терминах беззнаковых или знаковых (в дополнительном коде) чисел. В следующей таблице наборы x, y, и z интерпретируются как числа (со знаком или без знака), набор z — сумма или разность (по модулю 2^k) наборов x и y, знаки +, – означают обычные (не по модулю 2^k) операции сложения и вычитания над целыми числами.[1]

Флаг	Значения флага в терминах			
	беззнаковых чисел		знаковых чисел	
	0	**1**	**0**	**1**
ZF	$z \neq 0$	$z = 0$	$z \neq 0$	$z = 0$
SF	$z \in [0, 2^{k-1}-1]$	$z \in [2^{k-1}, 2^k-1]$	$z \geq 0$	$z < 0$
CF	*при сложении*: $x+y < 2^k$ *при вычитании*: $x \geq y$	*при сложении*: $x+y \geq 2^k$ *при вычитании*: $x < y$	*при сложении*: $x \geq 0, y \geq 0$ *или* $xy \leq 0, x+y < 0$ *при вычитании*: $x < 0, y \geq 0$ *или* $x \geq y \geq 0$ *или* $0 > x \geq y$	*при сложении*: $x < 0, y < 0$ *или* $xy < 0, x+y \geq 0$ *при вычитании*: $x \geq 0, y < 0$ *или* $x < y < 0$ *или* $0 \leq x < y$
OF	*при сложении*: $x, y, z < (\geq) 2^{k-1}$ *или* $x < (\geq) 2^{k-1} \leq (>) y$ *при вычитании*: $x, y < (\geq) 2^{k-1}$ *или* $x, z < (\geq) 2^{k-1}$, $y \geq (<) 2^{k-1}$	*при сложении*: $x, y < (\geq) 2^{k-1}$, $z \geq (<) 2^{k-1}$ *при вычитании*: $y, z < (\geq) 2^{k-1}$, $x \geq (<) 2^{k-1}$	*при сложении и вычитании*: $-2^{k-1} \leq x \pm y < 2^{k-1}$	*при сложении и вычитании*: $x \pm y < -2^{k-1}$ *или* $x \pm y \geq 2^{k-1}$

Проверка условий с помощью вычитания и флагов

Практически во всех типах ЭВМ существуют команды перехода по условию. Многие такие команды укладываются в схему "*Если op1$\diamond$op2, то переход по адресу A*", где *op1, op2* – операнды (битовые наборы длины k), $\diamond$ – одна из операций отношения: $=_б$, $\neq_б$, $<_б$, $>_б$, $\leq_б$, $\geq_б$, $=_{зн}$, $\neq_{зн}$, $<_{зн}$, $>_{зн}$, $\leq_{зн}$, $\geq_{зн}$ (здесь индекс "*б*" означает, что операнды трактуются как числа без знака, "*зн*" – со знаком).

Пусть для представления знаковых чисел используется дополнительный код. Приведем метод, позволяющий проверить истинность условия *op1$\diamond$op2* по значениям флагов (OF, CF, SF, ZF) , получаемым при вычитании *op1–op2* по модулю 2^k .

Поскольку разность (по модулю 2^k) двух одинаковых чисел (знаковых или беззнаковых) равна нулю, разность неравных чисел отлична от нуля и флаг ZF равен единице, если и только если результат операции – ноль, справедливы утверждения:

$$op1 = op2 \Leftrightarrow ZF = 1, \qquad op1 \neq op2 \Leftrightarrow ZF = 0.$$

Таким образом, для того чтобы установить равны или нет два числа, достаточно знать чему равен ZF.

Поведение флагов при других соотношениях между операндами (кроме ≠ и =) различно в знаковом и беззнаковом случаях. Поэтому рассмотрим эти случаи отдельно.

Вспомним, что в терминах беззнаковых чисел флаг CF равен единице, если уменьшаемое меньше вычитаемого, и равен нулю в противном случае. Тогда можно утверждать, что:

$$op1 <_б op2 \Leftrightarrow CF = 1, \qquad op1 \geq_б op2 \Leftrightarrow CF = 0.$$

[1] Доказательство эквивалентности правил, приведенных в таблице, и правил, сформулированных ранее в терминах битовых наборов, предлагается в качестве упражнения.

Условия для остальных отношений можно логически выразить через уже установленные. Например, $op1 >_{б} op2 \Leftrightarrow (op1 \geq_{б} op2) \wedge (op1 \neq op2)$, где $\wedge$ означает конъюнкцию. Следовательно, $op1 >_{б} op2 \Leftrightarrow (CF = 0) \wedge (ZF = 0)$.

Для случая знаковых чисел нам важны флаги OF и SF[2]. Посмотрим, какие значения принимают эти флаги в зависимости от знаков операндов и от соотношений[3] между операндами:

Знак *op1*	Знак *op2*	Возможное при данных знаках соотношение	Возможные значения флагов при данном соотношении [4]	
			OF	*SF*
+	+	<	0	1
		≥	0	0
+	–	≥	0	0
			1	1
–	+	<	0	1
			1	0
–	–	<	0	1
		≥	0	0

Проанализируем эту таблицу. Заметим, что множества пар значений флагов (OF,SF) для отношений < и ≥ не пересекаются, а их объединение дает все возможные пары из нулей и единиц. Следовательно, по паре значений (OF,SF) можно определить, какое из двух взаимоисключающих соотношений (< или ≥) выполняется. В случае $op1 < op2$ имеем: либо OF=0 и SF=1, либо OF=1 и SF=0, или, короче, OF≠SF. В случае $op1 \geq op2$ справедливо равенство OF=SF. Таким образом, можно утверждать:

$$op1 <_{зн} op2 \Leftrightarrow OF \neq SF, \qquad op1 \geq_{зн} op2 \Leftrightarrow OF = SF$$

Подытожим полученные результаты в виде таблицы.

Соотношение $op1 \diamond op2$	Критерий истинности данного соотношения в терминах значений флагов после операции вычитания $op1 - op2$ по модулю 2^k	
	для чисел со знаком	для чисел без знака
=	$ZF = 1$	$ZF = 1$
≠	$ZF = 0$	$ZF = 0$
<	$OF \neq SF$	$CF = 1$
≥	$OF = SF$	$CF = 0$
>	$(OF = SF) \wedge (ZF = 0)$	$(CF = 0) \wedge (ZF = 0)$
≤	$(OF \neq SF) \vee (ZF = 1)$	$(CF = 1) \vee (ZF = 1)$

[2] В "бесполезности" остальных флагов можно было бы убедиться, включив и их в рассмотрение, но это заняло бы больше места.

[3] Рассмотрим, как и в беззнаковом случае, только отношения < и ≥ – остальные можно выразить при помощи логических связок.

[4] Читателю предлагается самостоятельно доказать, что именно такие значения флагов возможны в каждом из рассматриваемых случаев.

Примеры

Пусть **к=8**, то есть числа умещаются в байт. Диапазон этих чисел **–128 до 127** для знаковых и **от 0 до 255** для беззнаковых.

1 способ. Двоичная арифметика.

Переведем числа в двоичную систему счисления и произведем арифметические действия по правилам двоичной арифметики

Сумма

50+100=150 – беззнаковые
50+100= -106 – знаковые

	0	0	1	1	0	0	1	0
	0	1	1	0	0	1	0	0
	1	0	0	1	0	1	1	0
CF	SF							

200+100=44 – беззнаковые
-56+100=44 – знаковые

	1	1	0	0	1	0	0	0
	0	1	1	0	0	1	0	0
1	0	0	1	0	1	1	0	0
CF	SF							

Сумма может уместиться в байт (как в первом случае) и не уместиться, как во втором. Этот дополнительный самый левый бит в получившемся наборе – значение CF. Напомним также, что суммой двух битовых наборов являются только 8 разрядов; получившийся во втором случае дополнительный (самый старший разряд) в число не входит (это и есть сложение по модулю 256), потому появление этого бита и свидетельствует о неправильности результата, беззнаковом переполнении. В первом примере CF=0 (результат для беззнаковых чисел правильный), во втором CF=1 (результат в байт не уместился, для беззнаковых чисел он неверный).

Самый левый разряд 8-битового набора является флагом SF. В первом примере SF=1, во втором SF=0

Чтобы установить флаг ZF, надо просмотреть все 8 битов суммы. ZF=1 только если все они равны нулю. У нас в обоих случаях ZF=0.

Чуть сложнее вычисляется флаг OF. Для его установки надо посмотреть старшие (8-е, «знаковые») биты слагаемых и суммы. Если знакомые биты слагаемых разные, то OF=0 (при сложении чисел разных знаков знакового переполнения быть не может, результат всегда правильный). Если знаковые биты слагаемых одинаковые, то OF=0 если у суммы знаковый бит такой же (то есть при сложении чисел с одинаковым знаком результат имеет тот же знак, знаковый результат в этом случае верный). Если же знак у суммы другой, значит, произошло знаковое переполнение, результат знакового сложения неверный OF=1.

В первом примере OF=1, у слагаемых на 8 месте – нули, а у результата – 1, значит, результат неверный. Во втором случае у слагаемых знаки разные, OF=0, знаковый результат верный.

Разность

50-100=206 – беззнаковые
50-100= -50 – знаковые

1	0	0	1	1	0	0	1	0
	0	1	1	0	0	1	0	0
	1	1	0	0	1	1	1	0
CF	SF							

200-100=100 – беззнаковые
-56-100=100 – знаковые

	1	1	0	0	1	0	0	0
	0	1	1	0	0	1	0	0
	0	1	1	0	0	1	0	0
CF	SF							

Флаги SF и ZF определяются точно так же, как при сложении. У нас в обоих примерах ZF=0. В первом SF=1, во втором SF=0.

Флаг CF – это дополнительный левый бит, который приходится подставить к уменьшаемому, чтобы «занять» единичку. У нас в первом примере CF=1, вычитание производится не из заданного числа, а из числа, которое его на 256 больше, беззнаковый результат, соответственно, получается неправильный. Во втором примере CF=0, результат верный.

Для вычисления OF смотрят на знаковые биты заданных чисел и результата. Если знаки у уменьшаемого и вычитаемого одинаковые, знакового переполнения произойти не может, OF=0, результат знакового вычитания правильный. Это мы можем видеть в первом примере. Если же знаки разные, знаковый результат будет верным, только если он совпадает по знаку с уменьшаемым (с первым числом). Во втором примере это не так и, следовательно, OF=1, знаковый результат неверный.

Примерно так флаги устанавливаются компьютером: вычисляются исходя из значений некоторых битов в двоичном представлении чисел. А мы уже по этим флагам можем судить о правильности результата при данной операции для знаковых или беззнаковых чисел. Для нас же этот способ не всегда удобен, мы не особенно привыкли к двоичной арифметике. Для быстрого вычисления флагов можно пользоваться способом «обратным» компьютерному: посмотреть на результат, правильный ли он для каждого представления чисел.

2 способ. Десятичная арифметика.

Представим числа двумя способами: в знаковом и беззнаковом виде, для каждого вида произведем арифметические операции по модулю 256 и оценим результат.

Сумма

В беззнаковом виде **50+100=150** получаем: ZF=0 так как результат ненулевой, SF=1 так как он больше 127. Результат верный, значит CF=0 (этот флаг «отвечает» за беззнаковые числа). По знаковому виду **50+100=-106** еще раз проверяем ZF и SF (число отрицательное) и обнаруживает, что результат неверный, значит OF=1 (флаг знаковых чисел).

200+100=44 (сложение производится по модулю 256, 300-256=44). ZF=0, SF=0, а вот CF=1 так как результат неверный. В знаковом виде это выражение выглядит так: **-56+100=44,** что является совершенно верным выражением, поэтому OF=0

Разность

Для первого примера имеем **50-100=206** (беззнаковые) или **–50** (знаковое). Отсюда ZF=0, SF=1, для беззнаковых результат неверный, а для знаковых – верный, следовательно CF=1, OF=0 .

Второй пример в беззнаковом виде выглядит так: **200-100=100** (результат верный), и так в знаковом: **-56-100=100** (результат неверный). Следовательно, ZF=0, SF=0, CF=0, OF=1