ECE 645: Lecture 1

Number Representation

Required Reading

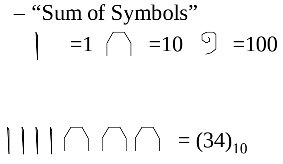
Behrooz Parhami,
Computer Arithmetic: Algorithms and Hardware Design

Chapter 1, Numbers and Arithmetic,
Sections 1.1-1.6, pp. 3-15

Chapter 2, Representing Signed Numbers,
Sections 2.1-2.6, pp. 19-31

Codes for Numbers

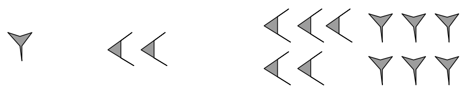
- Egyptian
  - ~4000 BC
  - "Sum of Symbols"



Positional Code for Numbers

- Babylonians
  - Positional system
  - 2000 BC
  - Radix 60
  -  = 1  = 10

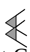
Babylonian Example



1×60^2 20×60^1 56×60^0

$= (4{,}856)_{10}$

Positional Code with Zero

- Zero Represented by Space
  - Partial solution
  - What about trailing zeros?
- Babylonians Introduced New Symbol
  -  or 
  - 4th to 1st Century BC
- Zero Allows Representation of Fractions
  - Fractions started with zero

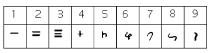
Mixed System

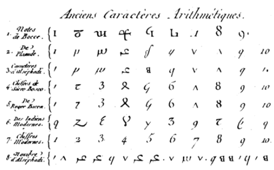
- Roman Numerals
  - Sum of all symbols
  - I=1 V=5 X=10 L=50 C=100 D=500 M=1000
  - Difficult to do arithmetic
  - *e.g.*,

$$\begin{array}{l} MCDXLVII \\ -\ \ IX \\ \hline ? \end{array}$$

Hindu-Arabic Numeral System

Brahmi numerals, India, 400 BC-400 AD





Evolution of numerals in early Europe

Positional Code Decimal System

- Documented in the 9th century
- Position of coefficient determines its value
- Coefficient in position is multiplied by radix (10) raised to the power determined by its position, *e.g.*,

$$4*10^3+8*10^2+5*10^1+6*10^0=(4{,}856)_{10}$$

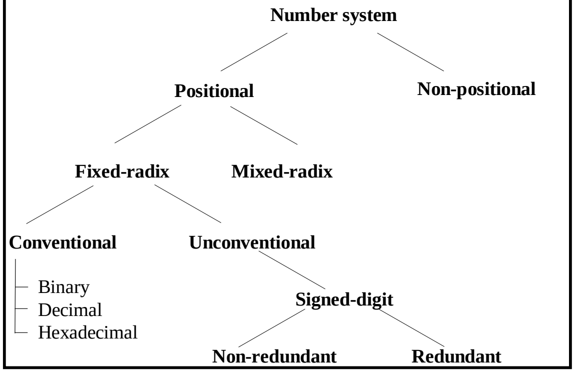
Migration of Positional Notation

- ~750 AD
  - Zero spread from India to Arabic countries
- ~1250 AD
  - Zero spread to Europe
- Importance of Zero
  - Ease of arithmetic which leads to improved commerce

Binary Number System

- Binary
  - Positional number system
  - Two symbols, B = { 0, 1 }
  - Easily implemented using switches
  - Easy to implement in electronic circuitry
  - Algebra invented by George Boole (1815-1864) allows easy manipulation of symbols

Classification of number systems (1)



Classification of number systems (2)

Positional

$$X = \sum_{i=-l}^{k-1} x_i \cdot w_i$$

w_i - weight of the digit x_i

Fixed-radix

$$X = \sum_{i=-l}^{k-1} x_i \cdot r^i$$

r - radix of the number system

Conventional fixed-radix

$$X = \sum_{i=-l}^{k-1} x_i \cdot r^i$$

r integer, $r > 0$

$x_i \in \{0, 1, \ldots, r\text{-}1\}$

Classification of number systems (3)

Unconventional fixed-radix

$$X = \sum_{i=-l}^{k-1} x_i \cdot r^i$$

$x_i \in \{-\alpha, \ldots, \beta\}$

Signed-digit $\alpha>0 \Rightarrow$ negative digits

Non-redundant number of digits = $\alpha + \beta + 1 \leq r$

Redundant number of digits = $\alpha + \beta + 1 > r$

Fixed-point representation

Integral and fractional part

$$X = \underbrace{x_{k-1}\, x_{k-2} \ldots x_1\, x_0}_{\text{Integral part}} \,.\, \underbrace{x_{-1}\, x_{-2} \ldots x_{-l}}_{\text{Fractional part}}$$

Radix point

- NOT stored in the register
- understood to be in a fixed position

Range of numbers

Number system	X_{min}	X_{max}
Decimal $X = (x_{k-1}\, x_{k-2} \ldots x_1\, x_0 . x_{-1} \ldots x_{-l})_{10}$	0	$10^k - 10^{-l}$
Binary $X = (x_{k-1}\, x_{k-2} \ldots x_1\, x_0 . x_{-1} \ldots x_{-l})_2$	0	$2^k - 2^{-l}$
Conventional fixed-radix $X = (x_{k-1}\, x_{k-2} \ldots x_1\, x_0 . x_{-1} \ldots x_{-l})_r$	0	$r^k - r^{-l}$

Notation: $\boldsymbol{ulp} = r^{-l}$

unit in the **l**east significant **p**osition
unit in the **l**ast **p**osition

Number of digits

Number system	Number of digits in the integer part necessary to cover the range $0..X_{max}$
Decimal	$k = \lfloor \log_{10} X_{max} \rfloor + 1 = \lceil \log_{10}(X_{max}+1) \rceil$
Binary	$k = \lfloor \log_2 X_{max} \rfloor + 1 = \lceil \log_2(X_{max}+1) \rceil$
Conventional fixed-radix	$k = \lfloor \log_r X_{max} \rfloor + 1 = \lceil \log_r(X_{max}+1) \rceil$

Radix Conversion of the Integral Part

R - destination radix

$$X_I = (x_{k-1}\, x_{k-2} \ldots x_1\, x_0)_R = \sum_{i=0}^{k-1} x_i \cdot R^i =$$

$$= ((\ldots((x_{k-1} R + x_{k-2}) R + x_{k-3}) R + \ldots + x_2) R + x_1) R + x_0$$

Quotient	*Remainder*
$(\ldots((x_{k-1} R + x_{k-2}) R + x_{k-3}) R + \ldots + x_2) R + x_1$	x_0
$\ldots((x_{k-1} R + x_{k-2}) R + x_{k-3}) R + \ldots + x_2$	x_1
……….	……….
x_{k-1}	x_{k-2}
0	x_{k-1}

Radix Conversion of the Fractional Part

R - destination radix

$X_F = (.\, x_{-1}\, x_{-2} \ldots x_{-l+1}\, x_{-l})_R = \sum_{i=-l}^{-1} x_i \cdot R^i =$

$= R^{-1} (x_{-1} + R^{-1} (x_{-2} + R^{-1} (\ldots. + R^{-1} (x_{-l+1} + R^{-1} x_{-l})\ldots.)))$

Integer part	*Fractional part*
x_{-1}	$R^{-1} (x_{-2} + R^{-1} (\ldots.. + R^{-1} (x_{-l+1} + R^{-1} x_{-l})\ldots.))$
x_{-2}	$R^{-1} (\ldots.. + R^{-1} (x_{-l+1} + R^{-1} x_{-l})\ldots.)$
………………………………	
x_{-l+1}	$R^{-1} x_{-l}$
x_{-l}	...

Shortcut for r=b^g, R=b^G

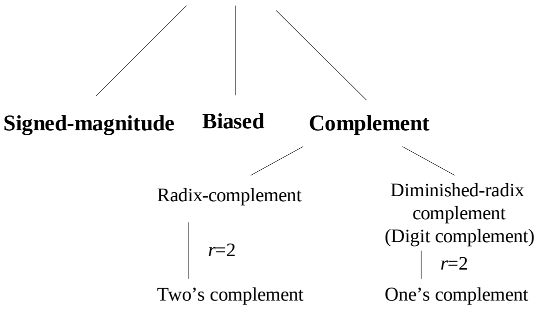
r=b^g → b → R=b^G

4=2^2 → 2 → 8=2^3

$(2301.302)_4$ = (10 | 11 0 | 0 01. 11 0 | 0 10)$_2$ = $(261.62)_8$

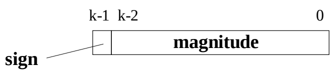
Signed Number Representations

Representations of signed numbers



	Signed-magnitude	Biased	Two's complement	One's complement
7	0111	1111	0111	0111
6	0110	1110	0110	0110
5	0101	1101	0101	0101
4	0100	1100	0100	0100
3	0011	1011	0011	0011
2	0010	1010	0010	0010
1	0001	1001	0001	0001
0	0000	1000	0000	0000
-0	1000			1111
-1	1001	0111	1111	1110
-2	1010	0110	1110	1101
-3	1011	0101	1101	1100
-4	1100	0100	1100	1011
-5	1101	0011	1011	1010
-6	1110	0010	1010	1001
-7	1111	0001	1001	1000
-8		0000	1000	

Signed-magnitude representation of signed numbers



Advantages:

- conceptual simplicity
- symmetric range: $-(2^{k-1}-1)$ .. $-(2^{k-1}-1)$
- simple negation

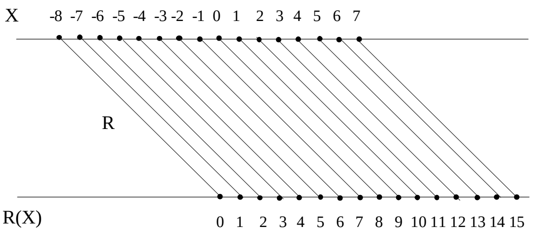
Disadvantages:

- addition of numbers with the same sign and with a different sign handled differently

Biased representation of signed integers

R(X) = X + B $B = 2^{k-1}$, k=4

$-2^{k-1} \le X \le 2^{k-1}-1$



Biased representation with radix 2

Signed number X

Representation mapping

Unsigned Representation R(X)

Binary mapping

Bit vector $(x_{k-1}x_{k-2}\ldots x_0.x_{-1}\ldots x_{-l})$

$$R(X) = \sum_{i=-l}^{k-1} x_i \cdot 2^i$$

Complement Signed Number Representations

Complement representations with radix 2

Signed number X

Representation mapping

Unsigned Representation $R(X)$

Binary mapping

Bit vector $(x_{k-1}x_{k-2}\ldots x_0.x_{-1}\ldots x_{-l})$

$$R(X) = \sum_{i=-l}^{k-1} x_i \cdot 2^i$$

Useful dependencies

$1 - x_i = \overline{x_i}$

x_i	$1 - x_i$	$\overline{x_i}$
0	1	1
1	0	0

$$|X| = \begin{cases} X \text{ when } X \geq 0 \\ -X \text{ when } X < 0 \end{cases}$$

One's complement transformation

For

$$A = \sum_{i=-l}^{k-1} A_i \cdot 2^i \geq 0$$

$$\mathbf{OC(}A\mathbf{)} \overset{\text{def}}{=} \overline{A} = 2^k - 2^{-l} - A$$

$$\begin{array}{cccccccc} k\text{-1} & k\text{-2} & \ldots & 0 & \text{-1} & \text{-2} & \ldots & -l \\ 1 & 1 & \ldots & 1\,. & 1 & 1 & \ldots & 1 \\ - \; A_{k-1} & A_{k-2} & \ldots & A_0\,. & A_{-1} & A_{-2} & \ldots & A_{-l} \\ \hline \overline{A}_{k-1} & \overline{A}_{k-2} & \ldots & \overline{A}_0\,. & \overline{A}_{-1} & \overline{A}_{-2} & \ldots & \overline{A}_{-l} \end{array}$$

Properties:

$0 \leq OC(A) \leq 2^k - 2^{-l}$

OC(OC(*A*)) = *A*

One's Complement Representation of Signed Numbers

For $\quad -(2^{k-1} - 2^{-l}) \le X \le 2^{k-1} - 2^{-l}$

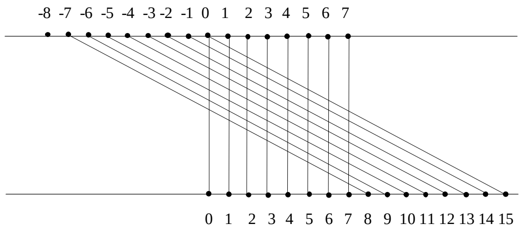
$$R(X) \overset{\text{def}}{=} \begin{cases} X & \text{for } X > 0 \\ 0 \text{ or } OC(0) & \text{for } X = 0 \\ OC(|X|) & \text{for } X < 0 \end{cases}$$

$$0 \le R(X) \le 2^k - 2^{-l}$$

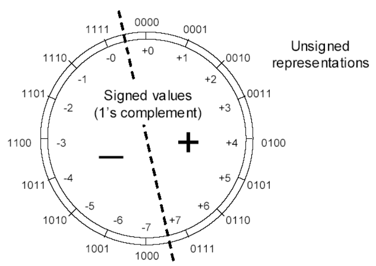
One's complement representation of signed integers

$X>0$	0	$X<0$
X	0, 2^k-1	$X+2^k-1 = 2^k-1 - \lvert X \rvert$

k=4



One's complement representation of signed numbers



Two's complement transformation (1)

For $A = \sum_{i=-l}^{k-1} A_i \cdot 2^i \geq 0$

$$\mathbf{TC}(A) \overset{\text{def}}{=} \begin{cases} \bar{A} + 2^{-l} = 2^k - A & \text{for } A > 0 \\ 0 & \text{for } A = 0 \end{cases}$$

$2^k - A = 2^k - A - 2^{-l} + 2^{-l} =$

$= (2^k - 2^{-l} - A) + 2^{-l} = \bar{A} + 2^{-l}$

Properties:

$0 \leq TC(A) \leq 2^k - 2^{-l}$

$\mathbf{TC(TC(}A\mathbf{)) = }A$

Two's complement transformation (2)

For $A = \sum_{i=-l}^{k-1} A_i \cdot 2^i \geq 0$

$$\mathbf{TC}(A) \overset{\text{def}}{=} \bar{A} + 2^{-l} \text{ mod } 2^k = 2^k - A \text{ mod } 2^k$$

Two's Complement Representation of Signed Numbers

For $-2^{k-1} \leq X \leq 2^{k-1} - 2^{-l}$

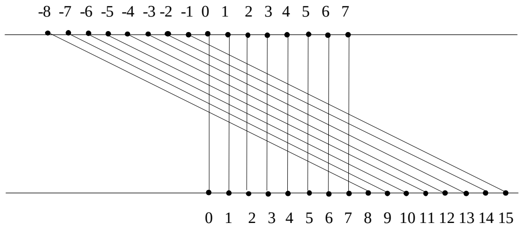
$$R(X) \overset{\text{def}}{=} \begin{cases} X & \text{for } X \geq 0 \\ TC(|X|) & \text{for } X < 0 \end{cases}$$

$0 \leq R(X) \leq 2^k - 2^{-l}$

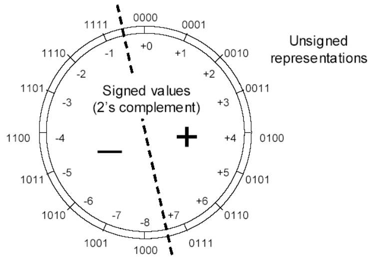
Two's complement representation of signed integers

$X>0$	0	$X<0$
X	0	$X+2^k = 2^k - \|X\|$

k=4



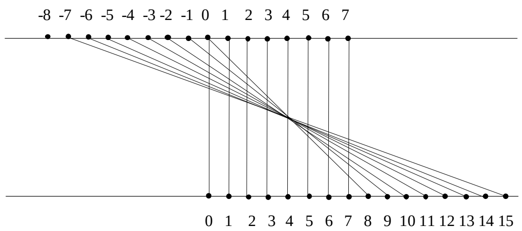
Two's complement representation of signed integers



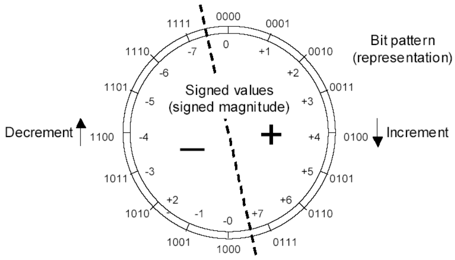
Signed-magnitude representation of signed numbers

$X>0$	0	$X<0$
X	0, 2^{k-1}	$\|X\|+2^{k-1} = -X+2^{k-1}$

k=4



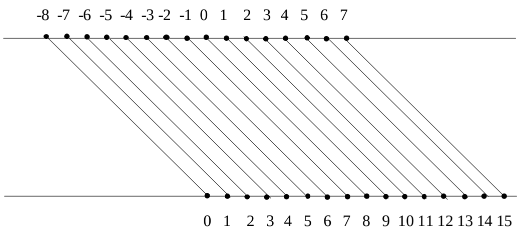
Signed-magnitude representation of signed numbers



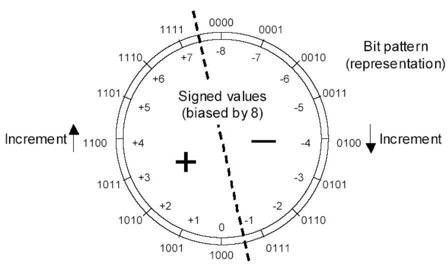
Biased representation of signed numbers

X>0	0	X<0
X+B	B	X+B

$B = 2^{k-1}$, k=4



Biased representation of signed numbers



Arithmetic Operations in Signed Number Representations

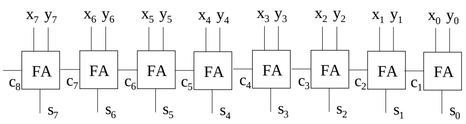
Unsigned addition vs. signed addition

Machine

	128	64	32	16	8	4	2	1
weight	128	64	32	16	8	4	2	1
carry					1	1	1	
X	0	0	0	1	0	0	1	1
+ Y	1	0	0	0	0	1	0	1
= S	1	0	0	1	1	0	0	0

Programmer

Unsigned mind	Signed mind



Out of range flags

Carry flag - C ***out-of-range for unsigned numbers***

C = 1 if result > MAX_UNSIGNED or
result < 0
0 otherwise

where MAX_UNSIGNED = 2^8-1 for 8-bit operands
2^{16}-1 for 16-bit operands

Overflow flag - V ***out-of-range for signed numbers***

V = 1 if result > MAX_SIGNED or
result < MIN_SIGNED
0 otherwise

where MAX_SIGNED = 2^7-1 for 8-bit operands
2^{15}-1 for 16-bit operands
MIN_SIGNED = -2^7 for 8-bit operands
-2^{15} for 16-bit operands

Overflow for signed numbers

Indication of overflow

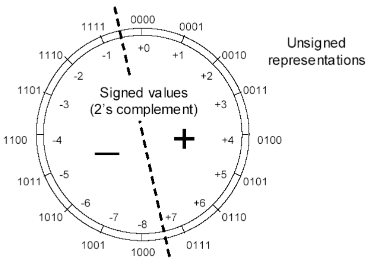
Positive	Negative
+ Positive	+ Negative
= Negative	= Positive

Formulas

$$\text{Overflow}_{2\text{'s complement}} = \overline{x_{k-1}}\,\overline{y_{k-1}}\,s_{k-1} + x_{k-1}\,y_{k-1}\,\overline{s_{k-1}} =$$

$$= c_k \oplus c_{k-1}$$

Two's complement representation of signed integers



Addition and subtraction

Two's complement

Numbers of the same sign

```
 -16 8 4 2 1
   1 1 0 1 1     -5
   1 0 1 1 0    -10
 -------------------
 1 1 0 0 0 1    -15
```

carry but not overflow

```
 -16 8 4 2 1
  |0|0 1 1 1      7
  |0|1 0 1 0     10
 -------------------
  |1|0 0 0 1    -15
```

no carry but overflow

Numbers of the opposite sign

```
 -16 8 4 2 1
   0 0 1 0 1      5
   1 0 1 1 0    -10
 -------------------
   1 1 0 1 1     -5
```

```
 -16 8 4 2 1
   0 1 0 1 0     10
   1 1 0 1 1     -5
 -------------------
 1 0 0 1 0 1      5
```

carry but not overflow

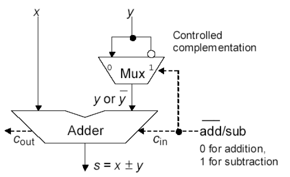


Fig. 2.7 Adder/subtractor architecture for two's-complement numbers.

Arithmetic Shift

Two's complement

Sh.L {00101_2 = +5} = 01010_2 = +10

Sh.L {11011_2 = -5} = 10110_2 = -10

Sh.L {01010_2 = +10} = 10100_2 = - 12

overflow

Sh.R {00101_2 = +5} = 00010_2 = +2 rem 1

Sh.R {11011_2 = -5} = 11101_2 = -3 rem 1

Addition and subtraction

One's complement

Numbers of the same sign

```
 -15 8 4 2 1
   1 1 0 1 0    -5
   1 0 1 0 1   -10
 ------------
 1 0 1 1 1 1
+ └──────── 1
 ------------
   1 0 0 0 0   -15
```

end-arround carry

Numbers of the opposite sign

```
 -15 8 4 2 1
   0 1 0 1 0    10
   1 1 0 1 0    -5
 ------------
 1 0 0 1 0 0
+ └──────── 1
 ------------
   0 0 1 0 1     5
```

Arithmetic Shift

One's complement

Sh.L {00101_2 = +5} = 01010_2 = +10

Sh.L {11010_2 = -5} = 10101_2 = -10

Sh.L {01010_2 = +10} = 10100_2 = - 11

overflow

Sh.R {00101_2 = +5} = 00010_2 = +2 rem 1

Sh.R {11011_2 = -5} = 11101_2 = -2 rem -1

Addition and subtraction

Signed-magnitude

Numbers of the same sign

	sign bit	magnitude	
	0	1 0 1 1	11
+	0	0 1 1 0	6
	0	1 0 0 0 1	17

carry = overflow

Numbers of the opposite sign

	sign bit	magnitude	
	1	1 0 1 1	-11
+	0	0 1 1 0	6

11 > 6

		1 0 1 1	11
−		0 1 1 0	6
	1	0 1 0 1	5

Addition/subtraction of biased numbers

$x + y + bias = (x + bias) + (y + bias) - bias$
$x - y + bias = (x + bias) - (y + bias) + bias$

A power-of-2 (or $2^a - 1$) bias simplifies the above

Comparison of biased numbers:
compare like ordinary unsigned numbers
find true difference by ordinary subtraction

Signed Number Representations

Summary

Representing k-bit signed binary numbers

Representation	Representation for $X>0$	Representation for 0	Representation for $X<0$
Signed-magnitude	X	0, 2^{k-1}	$2^{k-1}+\|X\|$
Biased	$X+B$	B typical $B=2^{k-1}$ or 2^{k-1}-ulp	$X+B$
Complement	X	0, M mod 2^k	$M-\|X\|=M+X$
Two's complement	X	0	$2^k-\|X\|=\overline{X}+ulp$
One's complement	X	0, 2^k-ulp	$2^k-ulp-\|X\|=\overline{X}$

Value of a number in the signed representations

Representation	Value of $(x_{k-1}\,x_{k-2}\,\ldots\,x_1\,x_0.x_{-1}\,\ldots\,x_{-l})$
Signed-magnitude	$X=(-1)^{x_{k-1}}\sum_{i=-l}^{k-2} x_i\cdot 2^i$
Biased	$X=\sum_{i=-l}^{k-1} x_i\cdot 2^i - B$
Two's complement	$X=-x_{k-1}2^{k-1}+\sum_{i=-l}^{k-2} x_i\cdot 2^i$
One's complement	$X=-x_{k-1}(2^{k-1}-ulp)+\sum_{i=-l}^{k-2} x_i\cdot 2^i$

Extending the number of bits of a signed number

X $\quad x_{k-1}\, x_{k-2} \ldots x_1\, x_0\, .\, x_{-1}\, x_{-2} \ldots x_{-l}$

⇩

Y $\quad y_{k'-1}\, y_{k'-2} \ldots y_k\, y_{k-1}\, y_{k-2} \ldots y_1\, y_0\, .\, y_{-1}\, y_{-2} \ldots y_{-l}\, y_{-(l+1)} \ldots y_{-l'}$

signed-magnitude

$x_{k-1}\, 0\,0\,0\,0\,0\,0\,0\;\; x_{k-2} \ldots x_1\, x_0\, .\, x_{-1}\, x_{-2} \ldots x_{-l}\;\; 0\,0\,0\,0\,0\,0$

two's complement

$x_{k-1}\, x_{k-1}\, x_{k-1} \ldots x_{k-1}\, x_{k-2} \ldots x_1\, x_0\, .\, x_{-1}\, x_{-2} \ldots x_{-l}\;\; 0\,0\,0\,0\,0\,0$

one's complement

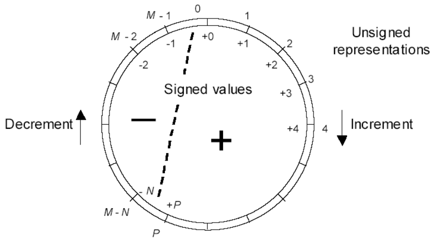
$x_{k-1}\, x_{k-1}\, x_{k-1} \ldots x_{k-1}\, x_{k-2} \ldots x_1\, x_0\, .\, x_{-1}\, x_{-2} \ldots x_{-l}\;\; x_{k-1} \ldots x_{k-1}$

biased

$x_{k-1}\; \bar{x}_{k-1} \ldots \bar{x}_{k-1}\, x_{k-2} \ldots x_1\, x_0\, .\, x_{-1}\, x_{-2} \ldots x_{-l}\;\; 0\,0\,0\,0\,0\,0$

Generalized Complement Representation

Generalized complement representation of signed integers



Generalized complement representation of signed integers

Table 2.1 Addition in a complement number system with complementation constant *M* and range [–*N*, +*P*]

Desired operation	Computation to be performed mod *M*	Correct result with no overflow	Overflow condition
$(+x) + (+y)$	$x + y$	$x + y$	$x + y > P$
$(+x) + (-y)$	$x + (M - y)$	$x - y$ if $y \le x$ $M - (y - x)$ if $y > x$	N/A
$(-x) + (+y)$	$(M - x) + y$	$y - x$ if $x \le y$ $M - (x - y)$ if $x > y$	N/A
$(-x) + (-y)$	$(M - x) + (M - y)$	$M - (x + y)$	$x + y > N$