

Sistemi numerici: numeri con segno

Esercizi risolti

1 Esercizio

Siano dati i seguenti numeri in base due: 11, 111, 1111, 11111, 10, 100, 1000, 10000, 01, 0011, 100001, 1010001.

Si indichi il loro valore decimale, interpretandoli come numeri in modulo e segno, complemento a uno e complemento a due (ovvero per ciascun numero binario si determinino i tre valori decimali corrispondenti alle tre rappresentazioni indicate).

Soluzione

Base 2 numero originale	Base 10		
	modulo e segno	complemento a uno	complemento a due
11	-1	-0	-1
111	-3	-0	-1
1111	-7	-0	-1
11111	-15	-0	-1
10	-0	-1	-2
100	-0	-3	-4
1000	-0	-7	-8
10000	-0	-15	-16
01	+1	+1	+1
0011	+3	+3	+3
100001	-1	-30	-31
1010001	-17	-46	-47

2 Esercizio

Siano dati i seguenti numeri in base dieci: +7, -7, +63, -63, +128, -128. Si indichi il loro valore binario nelle rappresentazioni modulo e segno, complemento a uno e complemento a due su 8 bit.

Soluzione

In binario puro, e sul numero minimo di bit, si ha che:

$$\begin{aligned}
 +7 &= 111 \\
 +63 &= 111111 \\
 +128 &= 10000000
 \end{aligned}$$

Le opportune conversioni permettono di completare la seguente tabella:

Base 10 numero originale	modulo e segno	Base 2	
		complemento a uno	complemento a due
+7	0 0000111	00000111	00000111
-7	1 0000111	11111000	11111001
+63	0 0111111	00111111	00111111
-63	1 0111111	11000000	11000001
+128	overflow	overflow	overflow
-128	overflow	overflow	10000000

3 Esercizio

Siano dati due numeri, x_1 e x_2 , nella base dieci di seguito riportati. Li si rappresentano nel formato in complemento a due su otto bit. Si effettuino quindi le operazioni necessarie per ottenere i numeri y_1 e y_2 nella stessa rappresentazione. Si indichino eventuali overflow.

1. $x_1 = +125_{10}$, $x_2 = -3_{10}$ e $y_1 = x_1 - x_2$, $y_2 = x_2 - x_1$.
2. $x_1 = +74_{10}$, $x_2 = -105_{10}$ e $y_1 = x_1 + x_2$, $y_2 = -x_1 - x_2$,
3. $x_1 = -97_{10}$, $x_2 = 31_{10}$ e $y_1 = -x_1 + x_2$, $y_2 = x_1 - x_2$

Soluzione

1. Innanzitutto si procede alla conversione dei due numeri $+x_1$ e $-x_2$ in binario puro:

$$\begin{array}{r|l}
 :2 & \\
 125 & \\
 62 & 1 \\
 31 & 0 \\
 15 & 1 \\
 7 & 1 \\
 3 & 1 \\
 1 & 1 \\
 0 & 1
 \end{array}
 \qquad
 \begin{array}{r|l}
 :2 & \\
 3 & \\
 1 & 1 \\
 0 & 1
 \end{array}$$

per cui in binario puro sul numero minimo di bit si ha:

$$\begin{array}{rclcl}
 +x_1 & = & +125_{10} & = & 1111101_2 \\
 -x_2 & = & +3_{10} & = & 11_2
 \end{array}$$

In complemento a due su 8 bit si ottiene:

$$\begin{aligned} +x_1 &= +125_{10} = 01111101_2 \\ -x_1 &= -125_{10} = 10000011_2 \\ -x_2 &= +3_{10} = 00000011_2 \\ +x_2 &= -3_{10} = 11111101_2 \end{aligned}$$

Da cui $y_1 = x_1 - x_2 = x_1 + (-x_2)$:

$$\begin{array}{r} \leftarrow \text{Riporti} \\ +x_1 + \\ -x_2 + \\ \hline +y_1 = \\ 10000000 \end{array}$$

Con risultato errato a causa di un overflow.

Analogamente $y_2 = x_2 - x_1 = x_2 + (-x_1)$:

$$\begin{array}{r} \leftarrow \text{Riporti} \\ +x_2 + \\ -x_1 + \\ \hline +y_2 = \\ 10000000 \end{array}$$

Con risultato corretto, ovvero assenza di overflow.

2. Si procede come in precedenza alla conversione in binario puro dei due numeri:

$$\begin{array}{r|l} :2 & \\ \hline 74 & \\ 37 & 0 \\ 18 & 1 \\ 9 & 0 \\ 4 & 1 \\ 2 & 0 \\ 1 & 0 \\ 0 & 1 \end{array} \quad \begin{array}{r|l} :2 & \\ \hline 105 & \\ 52 & 1 \\ 26 & 0 \\ 13 & 0 \\ 6 & 1 \\ 3 & 0 \\ 1 & 1 \\ 0 & 1 \end{array}$$

per cui in binario puro sul numero minimo di bit si ha:

$$\begin{aligned} +x_1 &= +74_{10} = 1001010_2 \\ -x_2 &= +105_{10} = 1101001_2 \end{aligned}$$

In complemento a due su 8 bit si ottiene:

$$\begin{aligned} +x_1 &= +74_{10} = 01001010_2 \\ -x_1 &= -74_{10} = 10110110_2 \\ -x_2 &= +105_{10} = 01101001_2 \\ +x_2 &= -105_{10} = 10010111_2 \end{aligned}$$

Da cui:

$$\begin{array}{rcl}
 & 00011110 & \Leftarrow \text{Riporti} \\
 +x_1 & 01001010 & + \\
 +x_2 & 10010111 & = \\
 +y_1 & \hline
 & 11100001 &
 \end{array}
 \qquad
 \begin{array}{rcl}
 & 11100000 & \Leftarrow \text{Riporti} \\
 -x_1 & 10110110 & + \\
 -x_2 & 01101001 & = \\
 +y_2 & \hline
 & 00011111 &
 \end{array}$$

Con risultati entrambi corretti.

3. Procedendo come in precedenza si ottiene in binario puro:

$$\begin{array}{rcl}
 -x_1 & = & +97 = 1100001_2 \\
 +x_2 & = & +31 = 11111_2
 \end{array}$$

e in complemento a 2:

$$\begin{array}{rcl}
 -x_1 & = & +97_{10} = 01100001_2 \\
 +x_1 & = & -97_{10} = 10011111_2 \\
 +x_2 & = & +31_{10} = 00011111_2 \\
 -x_2 & = & -31_{10} = 11100001_2
 \end{array}$$

Da cui:

$$\begin{array}{rcl}
 & 01111111 & \Leftarrow \text{Riporti} \\
 -x_1 & 01100001 & + \\
 +x_2 & 00011111 & = \\
 +y_1 & \hline
 & 10000000 &
 \end{array}
 \qquad
 \begin{array}{rcl}
 & 11111111 & \Leftarrow \text{Riporti} \\
 +x_1 & 10011111 & + \\
 -x_2 & 11100001 & = \\
 +y_2 & \hline
 & 10000000 &
 \end{array}$$

Con y_1 errato (overflow) e y_2 corretto.

4 Esercizio

Siano $x_1 = D7_{16}$, $x_2 = B3_{16}$, $x_3 = 4B_{16}$, $x_4 = 95_{16}$ numeri di 8 bit in complemento a 2 espressi in esadecimale. Eseguire le operazioni $y_1 = -x_2 + x_1$ e $y_2 = x_3 - x_4$, indicando se si verifica overflow.

Soluzione

Passando dalla rappresentazione in base sedici a quella in base due, si ottiene:

$$\begin{array}{rcl}
 x_1 & = & D7_{16} = 1101\ 0111_2 \\
 x_2 & = & B3_{16} = 1011\ 0011_2 \\
 x_3 & = & 4B_{16} = 0100\ 1011_2 \\
 x_4 & = & 95_{16} = 1001\ 0101_2
 \end{array}$$

Effettuando una operazione di complemento a due sui numeri x_2 e x_4 si ha:

$$\begin{array}{rcl}
 -x_2 & = & 01001101_2 \\
 -x_4 & = & 01101011_2
 \end{array}$$

e quindi:

$$\begin{array}{rcl}
 & 11011111 & \Leftarrow \text{Riporti} \\
 -x_2 & 01001101 + & \\
 +x_1 & 11010111 = & \\
 +y_1 & \underline{00100100} &
 \end{array}
 \qquad
 \begin{array}{rcl}
 & 01001011 & \Leftarrow \text{Riporti} \\
 +x_3 & 01001011 + & \\
 -x_4 & 01101011 = & \\
 +y_2 & \underline{10110110} &
 \end{array}$$

Con y_1 corretto e y_2 errato a causa di un overflow.

5 Esercizio

Siano dati i numeri $x_1 = -15.42_{10}$ e $x_2 = +12.24_{10}$. Rappresentarli nel formato in complemento a due sul numero di bit minimo necessario per rappresentare entrambi i numeri conservandone la precisione di partenza, ed effettuare, nella stessa rappresentazione, le operazioni $y_1 = -x_1 - x_2$, $y_2 = x_1 - x_2$. Si indichi la presenza di eventuali overflow.

Soluzione

Dato che la precisione dei due numeri in base dieci (con due cifre decimali) è pari a $1/100$, si ha che:

$$\begin{aligned}
 b^{-m} &\leq \frac{1}{100} \\
 b^m &\geq 100 \\
 m &\geq 7
 \end{aligned}$$

Occorrono quindi almeno 7 cifre per la parte frazionaria. Procedendo come in precedenza si ottiene in complemento a 2:

$$\begin{aligned}
 -x_1 &= +15.42_{10} = 01111.0110101_2 \\
 +x_1 &= -15.42_{10} = 10000.1001011_2 \\
 +x_2 &= +12.24_{10} = 01100.0011110_2 \\
 -x_2 &= -12.24_{10} = 10011.1100010_2
 \end{aligned}$$

Da cui allineando le virgole e agendo come se esse non esistessero, si ha:

$$\begin{array}{rcl}
 & 11111.100000 & \Leftarrow \text{Riporti} \\
 -x_1 & 01111.0110101 + & \\
 -x_2 & 10011.1100010 = & \\
 +y_1 & \underline{00011.0010111} &
 \end{array}
 \qquad
 \begin{array}{rcl}
 & 100111.000010 & \Leftarrow \text{Riporti} \\
 +x_1 & 10000.1001011 + & \\
 -x_2 & 10011.1100010 = & \\
 +y_2 & \underline{00100.0101101} &
 \end{array}$$

Con risultato corretto per la prima somma e errato, con presenza di overflow, per la seconda.

6 Esercizio

Uno strumento di misura deve essere in grado di misurare il numero di ore in un intervallo di tempo che va da 3000 anni prima a 3000 anni dopo la nascita di Cristo. Dato che si desidera utilizzare la rappresentazione in complemento a due, si indichino quante sono le cifre necessarie allo strumento per svolgere la sua funzione.

Soluzione

Lo strumento deve essere in grado di indicare tanto valori negativi quanto positivi sino a un massimo (in valore assoluto) uguale a:

$$\begin{aligned} 3000 [\text{anni}] \cdot 365 [\text{giorni/anno}] \cdot 24 [\text{ore/giorno}] &= 26280000 [\text{ore}] \\ &\simeq 25 \cdot 2^{20} [\text{ore}] \\ &< 2^5 \cdot 2^{20} [\text{ore}] \end{aligned}$$

Dato che con n bit in complemento a due è possibile rappresentare numeri N il cui valore V_N varia nell'intervallo:

$$-2^{n-1} \leq V_N \leq +(2^{n-1} - 1)$$

occorrono un numero di bit n uguale a 26.

7 Esercizio

Siano dati i numeri $x_1 = -26.125_{10}$ e $x_2 = +13.7_{10}$. Li si rappresenti nel sistema binario in modulo e segno su 10 bit con 3 bit dedicati alla parte frazionaria. Una volta ottenuta tale rappresentazione di x_1 e x_2 , come è possibile, nella stessa rappresentazione, calcolare il valore $y = x_1 - 4 \cdot x_2$?

Soluzione

Si rappresentano innanzitutto i numeri $-x_1$ e x_2 in binario puro:

$$\begin{aligned} -x_1 &= +26.125_{10} = 11010.001_2 \\ +x_2 &= +13.7_{10} = 1101.101_2 \end{aligned}$$

Quindi in modulo e segno su 10 bit si ottiene:

$$\begin{aligned} -x_1 &= +26.125_{10} = 0 \ 011010.001_2 \\ +x_1 &= -26.125_{10} = 1 \ 011010.001_2 \\ +x_2 &= +13.7_{10} = 0 \ 001101.101_2 \\ -x_2 &= -13.7_{10} = 1 \ 001101.101_2 \end{aligned}$$

Denominando $z = -4 \cdot x_2$, il valore di z può essere ottenuto mediante una operazione di shift sul valore di $-x_2$, ovvero scalando il (solo) modulo di tale numero di due posizioni verso sinistra:

$$z = -4 \cdot x_2 = 4 \cdot (-x_2) = 1 \ 110110.100_2$$

A questo punto occorre calcolare $y = x_1 + z$. Considerato che x_1 e z hanno lo stesso segno, si può procedere mediante una operazione di somma tra i soli moduli dei due numeri:

$$\begin{array}{rcl}
 & 1111100.00 & \Leftarrow \text{ Riporti} \\
 +x_1 & 011010.001 & + \\
 z = -4 \cdot x_2 & 110110.100 & = \\
 +y & \hline
 & 010000.101 &
 \end{array}$$

Con risultato uguale a $y_1 = 1$ 010000.101 (il segno del risultato è uguale al segno dei due addendi avendo essi segno concorde), ma errato a causa di un overflow sulla somma appena effettuata.

8 Esercizio

Siano dati i numeri x_1 e x_2 nella base dieci. Li si rappresentati nel formato in complemento a due sul numero di bit indicato. Si effettuino quindi le operazioni necessarie per ottenere i numeri y_1 e y_2 nella stessa rappresentazione. Si indichino eventuali overflow.

1. $x_1 = 255_{10}$, $x_2 = -127_{10}$ su 9 bit, e $y_1 = x_1 + x_2$, $y_2 = x_1 - x_2$
2. $x_1 = -9.9_{10}$, $x_2 = +5.7_{10}$ su 9 bit mantenendo la precisione, e $y_1 = -x_1 + x_2$, $y_2 = x_1 - x_2$
3. $x_1 = +7_{10}$, $x_2 = +121_{10}$ sul numero minimo di bit per rappresentarli entrambi, e $y_1 = x_1 + x_2$, $y_2 = -x_1 - x_2$

Soluzione

Procedendo come in precedenza, ovvero riconducendoci sempre a operazioni di somma tra numeri di segno opportuno, si ottiene quanto segue.

1.

$$\begin{array}{rcl}
 +x_1 & = & +255_{10} = 01111111_2 \\
 -x_2 & = & +127_{10} = 00111111_2 \\
 +x_2 & = & -127_{10} = 11000001_2
 \end{array}$$

Da cui:

$$\begin{array}{rcl}
 & 11111111 & \Leftarrow \text{ Riporti} \\
 +x_1 & 01111111 & + \\
 +x_2 & 11000001 & = \\
 +y_1 & \hline
 & 01000000 &
 \end{array}
 \qquad
 \begin{array}{rcl}
 & 01111111 & \Leftarrow \text{ Riporti} \\
 +x_1 & 01111111 & + \\
 -x_2 & 00111111 & = \\
 +y_2 & \hline
 & 10111110 &
 \end{array}$$

Con risultato corretto per y_1 e presenza di overflow, e risultato errato, per y_2 .

2. Mantenere la precisione richiede l'utilizzo di 4 cifre per la parte frazionaria. Segue che in complemento a due si ha:

$$\begin{aligned} -x_1 &= +9.9_{10} = 01001.1110_2 \\ +x_1 &= -9.9_{10} = 10110.0010_2 \\ +x_2 &= +5.7_{10} = 00101.1011_2 \\ -x_2 &= -5.7_{10} = 11010.0101_2 \end{aligned}$$

Da cui:

$$\begin{array}{rcl} & 000011\ 110 & \Leftarrow \text{ Riporti} \\ -x_1 & 01001.1110 + & \\ +x_2 & 00101.1011 = & \\ +y_1 & \hline 01111.1001 & \end{array} \qquad \begin{array}{rcl} & 111100\ 000 & \Leftarrow \text{ Riporti} \\ +x_1 & 10110.0010 + & \\ -x_2 & 11010.0101 = & \\ +y_2 & \hline 10000.0111 & \end{array}$$

Con risultati entrambi corretti.

3. Per rappresentare +121 in complemento a due sono necessari 8 bit ($2^{n-1} = 2^7 = 128$). Quindi:

$$\begin{aligned} +x_1 &= +7_{10} = 00000111_2 \\ -x_1 &= -7_{10} = 11111001_2 \\ +x_2 &= +121_{10} = 01111001_2 \\ -x_2 &= -121_{10} = 10000111_2 \end{aligned}$$

Da cui:

$$\begin{array}{rcl} & 01111111 & \Leftarrow \text{ Riporti} \\ +x_1 & 00000111 + & \\ +x_2 & 01111001 = & \\ +y_1 & \hline 10000000 & \end{array} \qquad \begin{array}{rcl} & 11111111 & \Leftarrow \text{ Riporti} \\ -x_1 & 11111001 + & \\ -x_2 & 10000111 = & \\ +y_2 & \hline 10000000 & \end{array}$$

Con presenza di overflow, e risultato errato, per y_1 e corretto per y_2 .