

Rappresentazione base 2

1 bit → 2 combinazioni →	0 1
2 bit → 4 combinazioni →	00 01 10 11
3 bit → 8 combinazioni →	000 001 010 011 100 101 110 111
...	
n bit → 2^n combinazioni →	0 + {combinazioni n-1 bit} 1 + {combinazioni n-1 bit}

Rappresentazione base 2

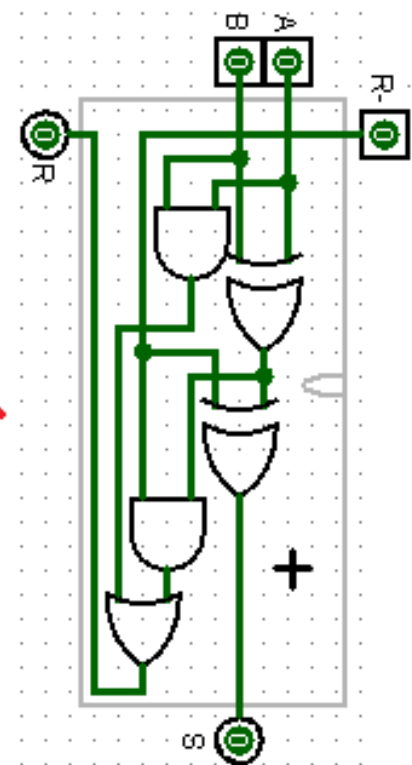
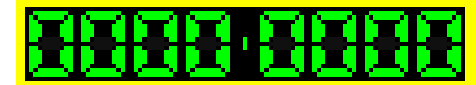
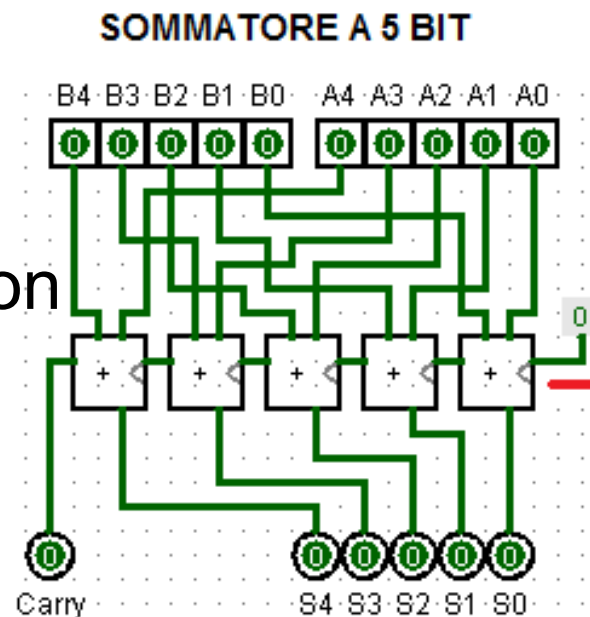
Nei computer in realtà si utilizza la notazione «**complemento a 2**». I motivi principali sono tre:

- La rappresentazione dello 0 è univoca
- La differenza tra due numeri può essere ottenuta sfruttando lo stesso circuito logico con cui si effettuano le somme.

01011+

00110=

10001



- La rappresentazione di un numero positivo in complemento a 2 coincide con la normale rappresentazione binaria

Rappresentazione base 2

Aumentando il numero di bit di una sequenza binaria aumento il numero di combinazioni associate.

Ad una sequenza binaria

$$C_{(n-1)} C_{(n-2)} \dots C_2 C_1 C_0 \cdot C_{-1} C_{-2} C_{-3} \dots C_{-m}$$

posso associare un numero razionale **X** (e questo in modo univoco!) sfruttando la seguente formula di rappresentazione:

$$X = \sum_{i=1}^{n-1} C_i \cdot b^i + \sum_{i=-1}^{-m} C_i \cdot b^i$$

dove $0 \leq C_i < b$ e $b=2$. Quindi con **n** bit posso rappresentare un numero da 0 a 2^n-1 :

$$n \text{ bit: } 0 \dots 00 \rightarrow 0 - 1 \dots 11 \rightarrow 1 \cdot 2^{n-1} + 1 \cdot 2^{n-2} + \dots + 2 \cdot 2^1 + 1 \cdot 2^0 = 2^n - 1$$

Rappresentazione base 2

Esempio (intero: da base 2 a base 10)

CAMBIO DI BASE: NUMERI INTERI

Trasforma il Numero: dalla base: alla base:

Cambia notazione

RISULTATO:

$$(10111011)_2 = 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (187)_{10}$$

187

$$(187)_{10} = 1 \times 10^2 + 8 \times 10^1 + 7 \times 10^0 = (187)_{10}$$

187 : 10 ==>

quoziente=18

resto=7

18 : 10 ==>

quoziente=1

resto=8

1 : 10 ==>

quoziente=0

resto=1

187



Rappresentazione base 2

Esempio (intero: da base 10 a base 2)

CAMBIO DI BASE: NUMERI INTERI

Trasforma il Numero: dalla base: alla base:

RISULTATO:

$$(187)_{10} = 1 \times 10^2 + 8 \times 10^1 + 7 \times 10^0 = (187)_{10}$$

10111011

$$(187)_{10} = 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (10111011)_2$$

187 : 2 ==>	quoziente=93	resto=1
93 : 2 ==>	quoziente=46	resto=1
46 : 2 ==>	quoziente=23	resto=0
23 : 2 ==>	quoziente=11	resto=1
11 : 2 ==>	quoziente=5	resto=1
5 : 2 ==>	quoziente=2	resto=1
2 : 2 ==>	quoziente=1	resto=0
1 : 2 ==>	quoziente=0	resto=1

10111011



Rappresentazione base 2

Esempio (razionale: da base 2 a base 10)

CAMBIO DI BASE: NUMERI CON DECIMALI

Converti il numero: (0, 1011) ₂ dalla base 2 alla base 10 con una precisione <= (0,00001) ₁₀

Mostrami la rappresentazione nella base indicata del numero con la virgola

RISULTATO:

$$(0,1011)_2 = 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} = (0,6875)_{10}$$

$$0,6875 \times 10 ==>$$

$$\text{parte decimale}=0,875$$

$$\text{parte intera}=6$$

$$0,875 \times 10 ==>$$

$$\text{parte decimale}=0,75$$

$$\text{parte intera}=8$$

$$0,75 \times 10 ==>$$

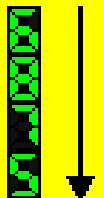
$$\text{parte decimale}=0,5$$

$$\text{parte intera}=7$$

$$0,5 \times 10 ==>$$

$$\text{parte decimale}=0,0$$

$$\text{parte intera}=5$$



$$(0,6875)_{10} = (0,6875)_{10} + \text{ERRORE}$$

ERRORE = 0 infatti:

$$(0,6875)_{10} = 6 \times 10^{-1} + 8 \times 10^{-2} + 7 \times 10^{-3} + 5 \times 10^{-4} = (0,6875)_{10}$$

Rappresentazione base 2

Esempio (razionale: da base 10 a base 2)

CAMBIO DI BASE: NUMERI CON DECIMALI

Converti il numero: (0,6875) ₁₀ dalla base 10 alla base 2 con una precisione <= (0,00001) ₁₀

Mostrami la rappresentazione nella base indicata del numero con la virgola

RISULTATO:

$$(0,6875)_{10} = 6 \times 10^{-1} + 8 \times 10^{-2} + 7 \times 10^{-3} + 5 \times 10^{-4}$$

$$0,6875 \times 2 ==>$$

$$\text{parte decimale}=0,375$$

$$\text{parte intera}=1$$

$$0,375 \times 2 ==>$$

$$\text{parte decimale}=0,75$$

$$\text{parte intera}=0$$

$$0,75 \times 2 ==>$$

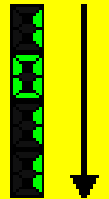
$$\text{parte decimale}=0,5$$

$$\text{parte intera}=1$$

$$0,5 \times 2 ==>$$

$$\text{parte decimale}=0,0$$

$$\text{parte intera}=1$$



$$(0,6875)_{10} = (0,1011)_2 + \text{ERRORE}$$

ERRORE = 0 infatti:

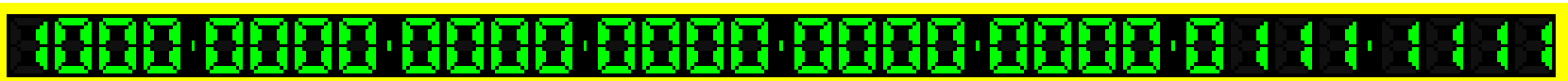
$$(0,1011)_2 = 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} = (0,6875)_{10}$$

Rappresentazione base 2

Per rappresentare i numeri negativi posso usare il 1° bit della sequenza impostandolo a 1 se il segno è meno

Esempio (uso 32 bit):

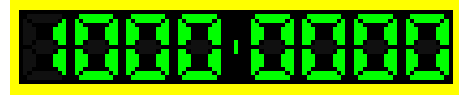
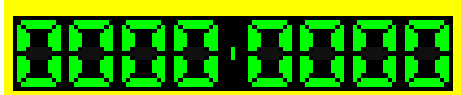
+127: 

-127: 

Perdendo un bit per il segno significa che con n bit il range (intervallo) di valori rappresentabili risulta essere inferiore:

$$- 2^{n-1} \dots + 2^{n-1}$$

Si noti che tale rappresentazione dei numeri relativi (nota con il nome **modulo e segno**) ammette per lo 0 due rappresentazioni



Questa modalità non è utilizzata nei calcolatori poiché non è adatta ai circuiti combinatori

Rappresentazione base 2

La rappresentazione in complemento a 2 di un numero X può essere ottenuta in questo modo:

- Converto il numero X (esempio -120) senza segno in binario (a n bit, esempio 8)

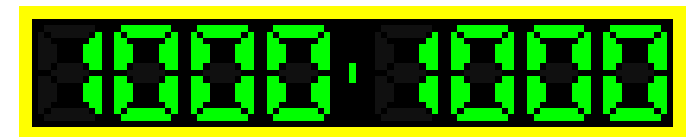
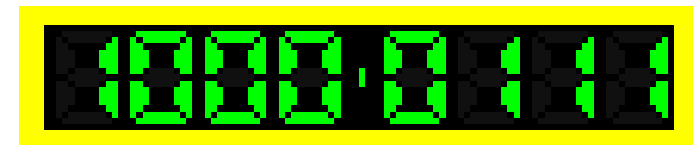
$(120)_{10} = 1 \times 10^2 + 2 \times 10^1 + 0 \times 10^0 = (120)_{10}$

$(120)_{10} = 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = (1111000)_2$

120 : 2 ==>	quoziente=60	resto=0
60 : 2 ==>	quoziente=30	resto=0
30 : 2 ==>	quoziente=15	resto=0
15 : 2 ==>	quoziente=7	resto=1
7 : 2 ==>	quoziente=3	resto=1
3 : 2 ==>	quoziente=1	resto=1
1 : 2 ==>	quoziente=0	resto=1



- Inverto nella rappresentazione ottenuta gli 0 con 1 e viceversa
- Sommo alla rappresentazione 1 e ottengo la rappresentazione in complemento a 2



Rappresentazione base 2

La rappresentazione di un numero con decimali nel computer avviene utilizzando una particolare notazione identificata con il nome di IEEE754:

Rappresentare il numero decimale : , ₁₀
utilizzando lo standard in virgola mobile in

BIT Segno:



Rappresentazione Binaria:

$$(1.25)_{10} = (1.010000000000000000000000)_2 + \text{Eventuale Errore}$$

Forma normalizzata:

$$N = 1.010000000000000000000000 \times 2^0$$

Esponente: 0 + l'eccesso 127 = 127



Il bit + significativo verrà omissso dalla rappresentazione poichè è implicito (hidden bit)

Mantissa:



$$\text{IEEE754} = (\text{1FR0 0000})_{16} = (1 \text{ 111 1111 } 010 \text{ 0000 0000 0000 0000 0000})_2$$

Rappresentazione base 2

Si noti che la sequenza di 32 bit seguente:

01111111010000000000000000000000

Interpretata come intero corrisponde a 2.141.192.192.

La stessa sequenza, vista come numero in notazione IEEE754, corrisponde a tutt'altro valore come si vede in figura:



Segno: +

Esponente: $(11111111)_2 = (255)_{10} - \text{Eccesso } [127] = 128$

Mantissa: 10000000000000000000000000000000

Forma normalizzata: $N = 1.01 \times 2^{128}$

Valore: $N = 1.25 \times 2^{128} = 4.253529586511731e+38$

Questo esempio dimostra l'importanza di come deve essere letta una sequenza binaria (definizione del tipo di dato) poiché a secondo del metodo ottengo risultati diversi.

Tipi di dato usati

TIPO	Nr. Bit	Interpretazione sequenza binaria
BOOL	1 Bit	0 false – 1 True
BYTE	8 Bit	Intero senza segno da 0 a 255.
CHAR	8 Bit	La sequenza viene interpretata come codifica ascii di un carattere
WORD	16 Bit	Intero senza segno da 0 a $2^{16}-1$
INT	16 Bit	Intero con segno – 2^{15} a $+2^{15}-1$
DWORD	32 Bit	Intero senza segno da 0 a $2^{32}-1$
DINT	32 Bit	Intero con segno – 2^{31} a $+2^{31}-1$
REAL	32 Bit	Numero reale codificato con IEEE754

Rappresentazione base 2

Ho inserito in questa pagina l'elenco dei vari convertitori utilizzati per queste spiegazioni:

❖ **NUMERI INTERI**

<http://www.brescianet.com/appunti/teoria/esercizi/CambiDiBase.htm>

❖ **NUMERI CON DECIMALI**

<http://www.brescianet.com/appunti/teoria/esercizi/frazionari.htm>

❖ **NUMERI INTERI RELATIVI**

<http://www.brescianet.com/appunti/teoria/esercizi/relativi.htm>

❖ **RAPPRESENTAZIONE FLOATING POINT**

<http://www.brescianet.com/appunti/teoria/esercizi/IEEE754.htm>

❖ **REVERSE IEEE754**

<http://www.brescianet.com/appunti/teoria/esercizi/ReverseIEEE754.htm>

FINE 3° LEZIONE