

pos. 7 ← MSB (10 11.01 10)₂ pos. 0 ← LSB

$0 \times 2^3 = 0 \times 8 = 0$
 $1 \times 2^2 = 1 \times 4 = 4$
 $1 \times 2^1 = 1 \times 2 = 2$
 $0 \times 2^0 = 0 \times 1 = 0$

$(1011.0110)_2 \Rightarrow (182)_{10}$
 $128 \quad 32 \quad 16 \quad 4 \quad 2$

$128 + 32 + 16 + 4 + 2 = 160 + 22 = 182$

MSB = Most Significant Bit
 LSB = Less Significant Bit

Nel linguaggio C però non è molto importante la trasformazione da numeri binari a decimali, se non per gruppi di 4 bit, come vediamo a pagina seguente

$$\begin{array}{cc}
 \begin{array}{cccc} 8 & 4 & 2 & 1 \end{array} & \begin{array}{cccc} 8 & 4 & 2 & 1 \end{array} \\
 (1011 \cdot 0110)_2 & \\
 \begin{array}{cccc} 8 & 4 & 2 & 1 \end{array} & \begin{array}{cccc} 8 & 4 & 2 & 1 \end{array} \\
 8+2+1 & 4+2 \\
 11 & 6 \\
 \downarrow & \downarrow \\
 B & 6
 \end{array}$$

$(B6)_{16}$

oppure

0x B6

significa base 16
in linguaggio C

E' molto più utile questa
trasformazione !!!
vediamo come usarla ...

0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	(10) A
1011	(11) B
1100	(12) C
1101	(13) D
1110	(14) E
1111	(15) F

MA SCHIERATURE

OR (si usa per portare a 1 un bit qualsiasi)

$(1010.0101)_2$ $0xA5$

Voglio portare a 1 questo bit

numero originale	1010.0101	$0xA5$
	OR	1
maschera	0001.0000	$0x10$
risultato	1011.0101	$0xB5$

dove la maschera vale 0 il numero originale non viene modificato

dove la maschera vale 1 viene forzato un 1

MASCHERATURE

AND

si usa per portare
un bit a ZERO

$(0111 \cdot 1100)_2$ $0x7C$

Voglio
portare questi due
bit a zero

numero
originale $0111 \cdot 1100$

maschera $1110 \cdot 0111$

risultato $0110 \cdot 0100$

dove la
maschera
vale 1 il numero
originale rimane
inalterato

dove la maschera
vale 0 il bit viene
forzato a ZERO

$0x7C$
&

$0xE7$

$0x64$

MASCHERATURE

XOR

si usa per invertire un bit

$(1000 \cdot 1001)_2$ 0x89

Voglio invertire
i 4 bit bassi

numero originale 1000 · 1001

0x89

XOR

maschera 0000 · 1111

0x0F

risultato 1000 · 0110

0x86

dove la
maschera vale
0 rimane
inalterato

dove la
maschera vale 1
il bit viene invertito

INVERSIONE DI TUTTI I BIT DI UN NUMERO

1001 • 0100	0x94
INV ↓ ↓ NEG	~ 0x94
0110 • 1011	↓ 0x6B

molto utile per creare
la maschera AND
che, a differenza della OR
e della XOR ha tutti
i bit invertiti