

Correzioni (1)

- Convertire i seguenti numeri binari in formato decimale:
 - $11,01 \rightarrow 3 + 1/4 = 13/4 = 3.25$
 - $101,111 \rightarrow 5 + 7/8 = 47/8 = 5.87$
 - $10,1 \rightarrow 2.5$
- Esprimere i seguenti valori in notazione binaria:
 - $4.5 \rightarrow 100,1$
 - $2.75 \rightarrow 10,11$
- Eseguire le seguenti somme binarie:
 - $1010,001 + 1,101 \rightarrow 1011,110$
 - $111,11 + 0,01 \rightarrow 1000,00$

Correzioni (2)

- Da complemento a 2 a base 10:
 - $00011 \rightarrow 3,$
 - $01111 \rightarrow 15,$
 - $11100 \rightarrow -4,$
 - $11010 \rightarrow -6,$
 - $00000 \rightarrow 0,$
 - $10000 \rightarrow -16$
- Da base 10 a complemento a 2 su 8 bit:
 - $6, -6, 13, -1, 0$
 - $00000110, 11111010, 00001101, 11111110, 00000000$
- Numero piu' grande e piu' piccolo per la notazione in complemento a 2 su 4, 6, 8 bit
 - Numero piu' piccolo -2^{n-1} ($n=6 \rightarrow -2^5 = -32$)
 - Numero piu' grande $2^{n-1} - 1$ ($n=6 \rightarrow 2^5 - 1 = 31$)

Correzioni (3)

- Da eccesso 8 a decimale:
 - $1110 \rightarrow 14-8=6$
 - $0111 \rightarrow 7-8=-1$
 - $1000, 0010, 0000, 1001$
 $0, -6, -8, 1$
- Da decimale a eccesso 8
 - $5 \rightarrow 5+8 \rightarrow 13 \rightarrow 1101$
 - $-5 \rightarrow -5+8 \rightarrow 3 \rightarrow 0011$
 - $3, 0, 7, -8$
 $1011, 1000, 1111, 0000$

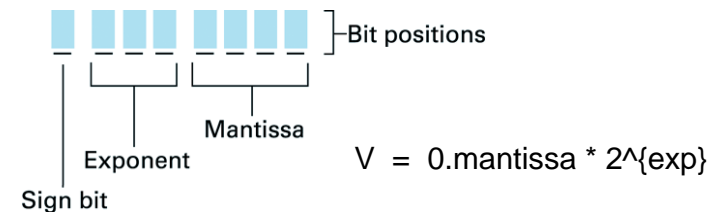
Correzioni (4)

- Numero piu' grande e piu' piccolo per la notazione in eccesso 8, 16, 32
 - **eccesso 8:** $8=2^{n-1} \rightarrow n=4$
numero piu' piccolo: $-8,$ numero piu' grande 7
 - **eccesso 16:** $16=2^{n-1} \rightarrow n=5$
numero piu' piccolo: -16 numero piu' grande 15

Rappresentazione dei numeri reali

Rappresentazione dei reali in un computer

- Bisogna rappresentare la posizione della virgola
- **Notazione in virgola mobile (floating point):** suddivisione in **tre campi**
- **Esempio con 8 bit:**
 - Partendo da sinistra: **primo bit** → **segno** (0 pos., 1 neg.)
 - **Tre bit** per **esponente**
 - **Quattro bit** per **mantissa**



Da floating point a decimale

1. **Anteporre 0**, alla **mantissa**
 $01101011 \rightarrow 0,1011$
2. Interpretare l' **esponente** come un numero in **eccesso su tre bit** (eccesso 4)
 $110 \rightarrow 6, 6-4 = 2$
3. **Spostare la virgola** della **mantissa** della quantità ottenuta dall'**esponente** a **dx** se il numero positivo a **sx** se e' negativo
 $0,1011 \rightarrow 10,11$
4. Tradurre **da binario a decimale** mettendo il segno a seconda del **bit piu' significativo** del floating point
 $10,11 \rightarrow 2,75$

Altro esempio di decodifica

1011100

- Segno: 1 → negativo
- Mantissa: 1100 → 0,1100
- Esponente: 011 → -1 in notazione in eccesso 4 → virgola a sinistra di 1 posto → 0,01100 (3/8, infatti $2 \times 2^{(-2)} + 2 \times 2^{(-3)}$)
- Numero decimale: **-3/8**

Da decimale a floating point

1. Da decimale a binario:
 $0.375 (=3/8) \rightarrow 0.011$
2. La **mantissa** si ottiene dall'**1 piu' a sinistra** completando con zeri i quattro bit
1100
3. Data la mantissa **contare di quante posizioni si deve spostare la virgola** rispetto a **0,mantissa**.
Il numero e' **negativo** se la virgola **va a sinistra**
1 bit a sinistra $\rightarrow -1$
4. Codificare il numero ottenuto in eccesso 4
 $-1 + 4 = 3 \rightarrow 011$
5. Mettere nel **bit piu' significativo** il bit di segno
00111100

Errori di troncamento

- Codifichiamo $2 + 5/8 = 2.625$ in **8 bit**
- Binario: **10,101**
- **Mantissa**: vorremmo scrivere **10101**, ma abbiamo solo 4 bit $\rightarrow 1010$, **tronco il bit meno significativo**
- **Esponente**: **110** (2)
- Risultato: **01101010**, che rappresenta 2.5 e non $2 + 5/8$
 - Infatti: $0,1010 \rightarrow 110 (2) \rightarrow 10,10 \rightarrow 2 + \frac{1}{2} = 2.5$

Esercizi

- **Decodifica**: 01001010, 01101101, 00111001
- **Codifica**: 2.75, 5.25
- **Qual e' il piu' grande** tra 01001001 e 00111101?

Correzioni (1)

Decodifica:

- **0 100 1010** $\rightarrow 5/8 = 0.625$ Infatti:
0 100 1010 \rightarrow positivo
0,1010
100 $\rightarrow 4-4=0$
0.1010
 $1/2 + 1/8 = 5/8 = 0.625 \rightarrow 0.625$

Codifica:

- **2.75** $\rightarrow 0 110 1011$ Infatti:
binario 10,11
1011 \rightarrow 2 posti a dx
2 \rightarrow 110
0 110 1011

Correzioni (2)

Decodifica:

- 0 100 1010 $\rightarrow 5/8 = 0.625$
- 0 110 1101 $\rightarrow 3 + 1/4 = 13/4 = 3.25$
- 0 011 1001 $\rightarrow 9/32$

Codifica:

- 2.75 $\rightarrow 0 110 1011$
- 5.25 $\rightarrow 0 111 1010$

Qual e' il piu' grande tra 01001001 e 00111101?

- Il primo e' 0.56, il secondo e' 0.40 \rightarrow il piu' grande e' il primo