Architettura degli Elaboratori – Corso B
Turno di Laboratorio 2

Docente: Claudio Schifanella

Prima Lezione:
Esercitazione sulla rappresentazione dei numeri

Potenze di 2

2^0	1	2^{16}	65,536
2^1	2	2^{17}	131,072
2^2	4	2^{18}	262,144
2^3	8	2^{19}	524,288
2^4	16	2^{20}	1,048,576
2^5	32	2^{21}	2,097,152
2^6	64	2^{22}	4,194,304
2^7	128	2^{23}	8,388,608
2^8	256	2^{24}	16,777,216
2^9	512	2^{25}	33,554,432
2^{10}	1,024	2^{26}	67,108,864
2^{11}	2,048	2^{27}	134,217,728
2^{12}	4,096	2^{28}	268,435,456
2^{13}	8,192	2^{29}	536,870,912
2^{14}	16,384	2^{30}	1,073,741,824
2^{15}	32,768	2^{31}	2,147,483,648

Numeri a precisione finita

- ***Numeri naturali***
- Consideriamo la base due: con **3** cifre binarie si possono rappresentare i numeri compresi tra **0** (limite inferiore) e $\mathbf{2^3-1}$ (limite superiore).

numero	rappresentazione
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Conversione tra basi: dec → bin

Divisioni successive per 2

171	1
85	1
42	0
21	1
10	0
5	1
2	0
1	1

Per la parte intera di un numero

1	0	1	0	1	0	1	1
7	6	5	4	3	2	1	0

Esercizio 1

Un elaboratore esprime gli interi su 16 bit. Scrivere le rappresentazioni in binario puro dei numeri 256_{10}, 10_{10}, 27_{10}, 32768_{10} e 65536_{10}. Sono tutti rappresentabili su 16 bit ?

Esercizio 1

Un elaboratore esprime gli interi su 16 bit. Scrivere le rappresentazioni in binario puro dei numeri 256_{10}, 10_{10}, 27_{10}, 32768_{10} e 65536_{10}. Sono tutti rappresentabili su 16 bit ?

$\boxed{16\ \text{BIT}}$

256	
128	0
64	0
32	0
16	0
8	0
4	0
2	0
1	0
0	1

$$\begin{array}{r} 256 \\ 256 \leftarrow 2^8 \\ \hline \text{RESTO}\quad 0 \end{array}$$

$256_{10} = 0000000100000000_2$

10	
5	0
2	1
1	0
0	1

$$\begin{array}{r} 10 \\ 8 \rightarrow 2^3 \\ \hline 2 \\ 2 \rightarrow 2^1 \\ \hline 0 \end{array}$$

$10_{10} = 0000000000001010_2$

27	
13	1
6	1
3	0
1	1
0	1

$27_{10} = 0000000000011011_2$

Esercizio 1

Un elaboratore esprime gli interi su 16 bit. Scrivere le rappresentazioni in binario puro dei numeri 256_{10}, 10_{10}, 27_{10}, 32768_{10} e 65536_{10}. Sono tutti rappresentabili su 16 bit ?

$$\begin{array}{r} 32768 \\ 32768 \\ \hline 0 \end{array} \quad 32768 \to 2^{15}$$

$32768_{10} = 1000000000000000_2$

$$65536$$
$$65536 \to 2^{16}$$

NO in 16 bit, ne servono 17!

Conversione tra basi: bin → dec

1 1 0 1 0 . 1 (26.5)

$1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1}$ (26.5)

per gli interi:

1	1
1	2 x 1 + 1 = 3
0	2 x 3 = 6
1	2 x 6 + 1 = 13
0	2 x 13 = 26

Prove degli esercizi precedenti

$10_{10} \overset{?}{=} 1010_2$

1	1
0	$1 \times 2 + 0 = 2$
1	$2 \times 2 + 1 = 5$
0	$5 \times 2 + 0 = 10$

$27_{10} \overset{?}{=} 11011$

1	1
1	$1 \times 2 + 1 = 3$
0	$3 \times 2 + 0 = 6$
1	$6 \times 2 + 1 = 13$
1	$13 \times 2 + 1 = 27$

$256_{10} \overset{?}{=} 100000000_2$

1	1
0	$1 \times 2 + 0 = 2$
0	$2 \times 2 + 0 = 4$
0	$4 \times 2 + 0 = 8$
0	$8 \times 2 + 0 = 16$
0	$16 \times 2 + 0 = 32$
0	$32 \times 2 + 0 = 64$
0	$64 \times 2 + 0 = 128$
0	$128 \times 2 + 0 = 256$

Più facile con la notazione posizionale

256 10 27

Prove degli esercizi precedenti

$$32768_{10} \underset{?}{=} 1000000000000000_2$$

$$1 \times 2^{15} + 0 \cdot 2^{14} + \ldots + 0 \cdot 2^0 = 32768_{10}$$

32768

65536

Conversione tra basi: bin ↔ ott

Binario → Ottale

$$\underbrace{0\quad 1\quad 1}_{3}\quad \underbrace{0\quad 1\quad 0}_{2}\quad .\quad \underbrace{1\quad 0\quad 0}_{4}\ (26.5)$$

(26.5)

Ottale → Binario

(171.25)

$$\overbrace{0\quad 1\quad 0}^{2}\quad \overbrace{1\quad 0\quad 1}^{5}\quad \overbrace{0\quad 1\quad 1}^{3}\quad .\quad \overbrace{0\quad 1\quad 0}^{2}\ (171.25)$$

Conversione tra basi: bin ↔ esa

Binario ⇒ Esadecimale

$$\underbrace{0\quad 0\quad 0\quad 1}_{1}\quad \underbrace{1\quad 0\quad 1\quad 0}_{A}\ .\ \underbrace{1\quad 0\quad 0\quad 0}_{8}\qquad (26.5)$$

(26.5)

Esadecimale ⇒ Binario

$$\overbrace{1\quad 0\quad 1\quad 0}^{A}\quad \overbrace{1\quad 0\quad 1\quad 1}^{B}\ .\ \overbrace{0\quad 1\quad 0\quad 0}^{4}\qquad (171.25)$$

(171.25)

Esercizio 2

Convertire i seguente numeri binari, in esadecimale ed in ottale rispettivamente:

1. 101101100010;
2. 1011101010110111.

Esercizio 2

Convertire i seguente numeri binari, in esadecimale ed in ottale rispettivamente:

1. 101101100010;
2. 10111010101010110111.

$$\overbrace{101}^{5}\,\overbrace{101}^{5}\,\overbrace{100}^{4}\,\overbrace{010}^{2} \longrightarrow 5542_{8}$$

$$\underbrace{1011}_{B}\,\underbrace{0110}_{6}\,\underbrace{0010}_{2} \longrightarrow B62_{16}$$

ESEMPIO

5	5		B	11
5	$5\times 8+5=45$	$\leftrightarrow$	6	$11\times 16+6=182$
4	$45\times 8+4=364$		2	$182\times 16+2=2914$
2	$364\times 8+2=2914$			

Esercizio 2

Convertire i seguente numeri binari, in esadecimale ed in ottale rispettivamente:

1. 101101100010;
2. 101110101010110111.

$$00\overbrace{101}^{5}\,\overbrace{110}^{6}\,\overbrace{101}^{5}\,\overbrace{010}^{2}\,\overbrace{110}^{6}\,\overbrace{111}^{7} \longrightarrow 565267_{8}$$

$$\underbrace{0010}_{2}\,\underbrace{1110}_{E}\,\underbrace{1010}_{A}\,\underbrace{1011}_{B}\,\underbrace{0111}_{7} \longrightarrow 2EAB7_{16}$$

Esercizio 3
Convertire il seguente numero esadecimale in binario:

$$AE8F.$$

Esercizio 3

Convertire il seguente numero esadecimale in binario:

$$AE8F.$$

$$\overbrace{1010}^{A}\ \overbrace{1110}^{E}\ \overbrace{1000}^{8}\ \overbrace{1111}^{F}$$

A 10

E $10 \times 16 + 14 = 174$

8 $174 \times 16 + 8 = 2792$

F $2792 \times 16 + 15 = 44687$

Conversione tra basi: dec → bin

0.25		***0***.50
0.50		***1***.00

Per la parte decimale di un numero

Moltiplicazioni successive per 2

1	0	1	0	1	0	1	1	.	0	1
7	6	5	4	3	2	1	0		-1	-2

Esempio 35,59375

Per la parte intera

35	1	0.59375	***1***.1875
17	1	0.1875	***0***.375
8	0	0.375	***0***.75
4	0	0.75	***1***.5
2	0	0.5	***1***.0
1	1		

1	0	0	0	1	1	.	1	0	0	1	1
5	4	3	2	1	0		-1	-2	-3	-4	-5

Esercizio 4

Convertire il seguente numero decimale, in binario: $234,2$. È un numero finito ?

Esercizio 4

Convertire il seguente numero decimale, in binario: $234,2$. È un numero finito ?

234	
117	0
58	1
29	0
14	1
7	0
3	1
1	1
0	1

	0,2	
	0,4	0
	0,8	0
(1,6)	0,6	1
(1,2)	0,2	1
	0,4	0
	0,8	0
(1,6)	0,6	1

$11101010,\overline{0011}$

Esercizio 5

Sia dato il numero binario frazionario 101110000,101. Convertirlo in base 8, in base 16 e in base 10.

Esercizio 5

Sia dato il numero binario frazionario 101110000,101. Convertirlo in base 8, in base 16 e in base 10.

$$000\overbrace{1\,01}^{5}\,\overbrace{1\,11}^{6}\,\overbrace{000}^{0},\overbrace{101}^{5}0 \longrightarrow 560,5_{8}$$

$$\underbrace{0001}_{1}\,\underbrace{0111}_{7}\,\underbrace{0000}_{0},\underbrace{1010}_{A} \longrightarrow 170,A_{16}$$

$$2^{8}+2^{6}+2^{5}+2^{4} = 2^{4}\left(2^{4}+2^{2}+2^{1}+1\right) = 16 \times 23 = 368_{10}$$

$$0,101 \Rightarrow 2^{-1}+2^{-3} = \frac{1}{2}+\frac{1}{8} = \frac{5}{8} = 0,625_{10}$$

$$368,625_{10}$$

Un altro esempio 35,9

35	1	Per la parte intera	0.9	***1***.8
17	1		0.8	***1***.6
8	0		0.6	***1***.2
4	0		0.2	***0***.4
2	0		0.4	***0***.8
1	1		0.8	***1***.6

1 0 0 0 1 1 . 1 1 1 0 0 1 ...

5 4 3 2 1 0 -1 -2 -3 -4 -5 -6 ...

Conversione tra basi: ott ↔ esa

Conviene passare attraverso la *rappresentazione binaria*, es. 32.4 ->

- Conversione da ottale a binario -> 011 010 . 100
- Espansione in quaterne 0001 1010 . 1000
- Conversione da binario in esadecimale -> 1A.4
- Viceversa, conversione in binario -> 0001 1010 . 1000
- Raggruppamento in terne -> 011 010 . 100
- Conversione da binario in ottale -> 32.4

Addizione binaria

ESEMPIO:

$$\begin{array}{rrrrrl} & 1 & 0 & 1 & 1 & + \\ & 0 & 1 & 1 & 1 & \\ \hline 1 & 0 & 0 & 1 & 0 & \end{array}$$

Addizione binaria

x_i	y_i	c_{i-1}	s_i	c_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Esercizio 7

Una calcolatrice esprime gli interi su 8 bit utilizzando la rappresentazione in binario puro. Scrivere le rappresentazioni di $A = 102$ e $B = 76$ ed eseguire in binario la somma $A + B$, segnalando l'eventuale overflow.

Esercizio 7

Una calcolatrice esprime gli interi su 8 bit utilizzando la rappresentazione in binario puro. Scrivere le rappresentazioni di $A = 102$ e $B = 76$ ed eseguire in binario la somma $A + B$, segnalando l'eventuale overflow.

102	
51	0
25	1
12	1
6	0
3	0
1	1
0	1

76	
38	0
19	0
9	1
4	1
2	0
1	0
0	1

$$
\begin{array}{lr}
A & 01100110\ + \\
B & 01001100\ = \\
\hline
 & 10110010_2
\end{array}
$$

NO OVERFLOW

Esercizio 8

Considerare la rappresentazione di numeri interi in binario puro su 9 bit. Scrivere le rappresentazioni di $A = 328$ e $B = 202$ ed eseguire in binario la somma $A + B$, segnalando l'eventuale overflow.

Esercizio 8
Considerare la rappresentazione di numeri interi in binario puro su 9 bit. Scrivere le rappresentazioni di $A = 328$ e $B = 202$ ed eseguire in binario la somma $A + B$, segnalando l'eventuale overflow.

328	
164	0
82	0
41	0
20	1
10	0
5	0
2	1
1	0
0	1

202	
101	0
50	1
25	0
12	1
6	0
3	0
1	1
0	1

$$
\begin{array}{c|ccccccccc|l}
 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & A \\
 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & B \\
\hline
1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 &
\end{array}
$$

9 BIT

OVERFLOW!

Esercizio 9
Indicare quali delle seguenti affermazioni sono vere. Con 8 bit è possibile rappresentare:

1. tutti gli interi non negativi minori o uguali a 255 in binario puro;
2. tutti gli interi non negativi minori o uguali a 255 in modulo e segno;
3. tutti gli interi compresi nell'intervallo $[-256, +255]$ in complemento a due;
4. tutti gli interi compresi nell'intervallo $[-127, +127]$ in complemento a uno.

Esercizio 9

Indicare quali delle seguenti affermazioni sono vere. Con 8 bit è possibile rappresentare:

1. tutti gli interi non negativi minori o uguali a 255 in binario puro;
2. tutti gli interi non negativi minori o uguali a 255 in modulo e segno;
3. tutti gli interi compresi nell'intervallo $[-256, +255]$ in complemento a due;
4. tutti gli interi compresi nell'intervallo $[-127, +127]$ in complemento a uno.

1) VERO $\rightarrow [0, (2^N-1)] \rightarrow [0, 2^8-1] \rightarrow [0, 255]$

2) FALSO

↳ Modulo e Segno: $[-(2^{N-1}-1), (2^{N-1}-1)] \rightarrow [-127, +127]$

3) FALSO

↳ $[-2^{N-1}, 2^{N-1}-1] \rightarrow [-128, +127]$

4) VERO

↳ $[-(2^{N-1}-1), (2^{N-1}-1)] \rightarrow [-127, +127]$

Esercizio 10

Definire gli intervalli di rappresentazione, il min e max numero relativo rappresentabile su 16 bit considerando le seguenti codifiche:

- in modulo e segno
- in complemento a uno
- in complemento a due
- in eccesso 2^{15}

Esercizio 10

Definire gli intervalli di rappresentazione, il min e max numero relativo rappresentabile su 16 bit considerando le seguenti codifiche:

- in modulo e segno
- in complemento a uno
- in complemento a due
- in eccesso 2^{15}

1) $[-32767, +32767]$

2) $[-32767, +32767]$

3) $[-32768, +32767]$

4) $[-32768, +32767]$

Esercizio 11

Indicare quanti sono i bit necessari per rappresentare in complemento a due i numeri $A = +129_{10}$ e $B = (-271_{10})$. Riportare la codifica in binario dei due numeri utilizzando lo stesso numero minimo di bit.