Università di Napoli Federico II – Scuola Politecnica e delle Scienze di Base
Corso di Laurea in Ingegneria Informatica

Corso di Calcolatori Elettronici I

Sistemi di numerazione posizionale

Sistemi di numerazione

- Problema: trovare un metodo per rappresentare i numeri
- Ha avuto nel tempo diverse risposte. Ad esempio:

  I II III IV V VI VII VIII IX X....L,...C...,D...,M....
- La numerazione araba è alla base della rappresentazione dei numeri anche nell'elaboratore
  - E' un sistema posizionale ordinato secondo le potenze di dieci

Sistema decimale

- I primi dieci numeri naturali sono codificati mediante le cifre 0,1,2,3,4,5,6,7,8,9
- Un **numero intero** è scomposto in una somma di potenze positive (o nulle) di dieci, ciascuna moltiplicata per uno dei primi dieci numeri naturali
- Un **numero frazionario puro** è scomposto in una somma di potenze negative di dieci, ciascuna moltiplicata per uno dei primi dieci numeri naturali
- Un **numero** è rappresentato da una stringa di cifre rappresentanti nell'ordine i fattori moltiplicativi delle diverse potenze di dieci; la parte intera e quella frazionaria sono separate (dal carattere ".”o “,”)

Esempi

- $1 \times 10^1 + 5 \times 10^0$
  - Numero rappresentato dalla stringa 15
- $1 \times 10^2 + 0 \times 10^1 + 0 \times 10^0 + 4 \times 10^{-1} + 7 \times 10^{-2}$
  - Numero rappresentato dalla stringa 100.47

Sistemi di numerazione posizionali

- Base di rappresentazione (b)
- Si usano b cifre (simboli associati ai numeri da 0 a b -1)

$$n = a_{m-1} \times b^{m-1} + \ldots + a_1 \times b^1 + a_0 \times b^0 + a_{-1} \times b^{-1} + a_{-2} \times b^{-2} \ldots + a_{-p} \times b^{-p}$$

- Rappresentazione:

$$a_{m-1} \;\ldots\; a_1 \;\; a_0 \,,\, a_{-1} \;\; a_{-2} \ldots a_{-p}$$

- In queste ipotesi la rappresentazione è unica
- La cifra con valore posizionale più elevato è chiamata cifra più significativa (e nel caso della rappresentazione binaria mediante BIT corrisponde al bit più significativo)

Esempi

- Binaria (B=2)
  - Cifre: 0,1
  - $(1100110)_2 = 1* 2^6+1* 2^5+0* 2^4+0* 2^3+1* 2^2+1* 2^1+ 0* 2^0 = 64 + 32 + 0 + 0 + 4 + 2 = (102)_{10}$
- Ottale (B=8)
  - Cifre: 0,1,2,3,4,5,6,7
  - $(146)_8 = 1*8^2 + 4*8^1 + 6*8^0 = 64 + 32 + 6 = (102)_{10}$
- Esadecimale (B=16)
  - cifre: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
  - $(66)_{16} = 6*16^1 + 6*16^0 = 96 + 6 = (102)_{10}$

Conversione da base 2 a 2^k e viceversa

- Da base 2 a 2^k
  - È sufficiente raggruppare, partendo dal bit meno significativo, gruppi di k bit ed indicare il valore corrispondente in base 2^k
  - $(11001101)_2 = (011.001.101)_2 = (315)_8$
- Da base 2^k a 2
  - È sufficiente rappresentare in base 2, utilizzando k bit, ciascuna cifra
  - $(C5)_{16} = (11000101)_2$

Conversione da base 2 a base 10 e viceversa

- Da base 2 a base 10
  - È sufficiente applicare la definizione di sistema di numerazione posizionale
  - $(1100110)_2 = 1 * 2^6 + 1 * 2^5 + 0 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 0 * 2^0 = 64 + 32 + 0 + 0 + 4 + 2 = (102)_{10}$
- Da base 10 a base 2
  - Numeri interi: algoritmo delle divisioni successive
    - 6_{10} | 2 = 3 r **0**; 3 | 2 = 1 r **1**; 1 | 2 = 0, r **1 =** 110_2 (inverso)
  - Numeri frazionari puri: algoritmo delle moltiplicazioni successive
    - 0.75_{10} x 2 = **1**.50; 0.50 x 2 = **1**.0 = 0.11_2 (diretto)

Proprietà notevoli (base 2)

- 2^k : solo il k+1-esimo bit da destra è uguale a 1
  - $4=2^2 \rightarrow (100)_2$ $32=2^5 \rightarrow (100000)_2$
- 2^k-1 : i k bit meno significativi sono uguali a 1
  - $7=2^3-1 \rightarrow (111)_2$ $63=2^6-1 \rightarrow (111111)_2$
- $x \cdot 2^k$: la rappresentazione di x “scorre” a sinistra di k bit
  - $5 \rightarrow (101)_2$ $40 = 5 \cdot 2^3 \rightarrow (101000)_2$
- $\lfloor x/2^k \rfloor$: la rappresentazione di x “scorre” a destra di k bit
  - $\lfloor 40 / 2^2 \rfloor = 10 \rightarrow (1010)_2$ $\lfloor 10 / 2^2 \rfloor = 2 \rightarrow (10)_2$
- $|x|_{2^k}$: i k bit meno significativi della rappresentazione di x
  - $|40|_{2^4} = 8 \rightarrow (1000)_2$

Somme e sottrazioni in aritmetica binaria

- Si effettuano secondo le regole del sistema decimale, ossia sommando (sottraendo) le cifre di pari peso
- Come nelle usuali operazioni su numeri decimali, si può avere un riporto sul bit di peso immediatamente superiore (carry), o un prestito dal bit di peso immediatamente superiore (borrow)
- Le somme (differenze) bit a bit sono definite come segue:

0+0=0	0-0=0
0+1=1	1-0=1
1+0=1	1-1=0
1+1=0 (carry=1)	0-1=1 (borrow=1)

- Ulteriore caso elementare:

  1 + 1 + 1 = 1 (carry=1)

Moltiplicazione in aritmetica binaria

- La moltiplicazione bit a bit può essere definita come segue:

  0x0=0

  0x1=0

  1x0=0

  1x1=1