Corso di Calcolatori Elettronici I
A.A. 2010-2011

Rappresentazione dei numeri: sistemi di numerazione posizionale

Lezione 3

Prof. Roberto Canonico

Università degli Studi di Napoli Federico II
Facoltà di Ingegneria
Corso di Laurea in Ingegneria Informatica (allievi A-DE+Q-Z)
Corso di Laurea in Ingegneria dell'Automazione

Rappresentazione dei numeri

- 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 (R)
- 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

Il sistema di numerazione posizionale

- 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)

Rappresentazione mediante basi diverse

- 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

  $(66)_{16} = 6 * 16^1 + 6 * 16^0 = 96 + 6 = (102)_{10}$

Basi diverse

- Quanto più è piccola la base tanto più lunga sarà la rappresentazione di una stessa quantità.
  Es. $(109)_{10} = (1101101)_2 = (155)_8 = (6D)_{16}$
- Notazioni pratiche:
  - Esadecimale: 0x6D oppure 6DH

Codice per la rappresentazione binaria delle cifre ottali ed esadecimali

- Base ottale |D|=8 -> $\log_2 8$

Sono necessarie tre cifre binarie

- Base esadecimale |D|=16 -> $\log_2 16$

Sono necessarie 4 cifre binarie

- Per convenzione per ciascuna cifra si adotta il codice corrispondente alla rappresentazione in bimario del numero associato

Codice per le numerazioni ottale ed esadecimale

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

ottale

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

esadecimale

Teorema

- **La rappresentazione in BIT di un numero è la stessa per qualsiasi numerazione con $b=2^k$ se per la codifica delle cifre si adopera la numerazione binaria pura**

- Esempio: consideriamo la stringa 00010110
  - $(00010110)_2 = 2+4+16=22$
  - 000 010 110 = $(026)_8 = 16+6 = 22$
  - 0001 0110 = $(16)_{16} = 16+6 = 22$

- Pertanto:
  le numerazioni ottale ed esadecimale vengono impiegate come metodo abbreviato per indicare una stringa di BIT
- Esempio:
  2BE15C rappresenta 0010 1011 1110 0001 0101 1100

Conversione di base

- Numeri interi: algoritmo delle divisioni successive
- Numeri frazionari puri: algoritmo delle moltiplicazioni successive

Conversione di un numero intero da base 10 a base b

212 | 2
0 | 106 | 2
0 | 53 | 2
1 | 26 | 2
0 | 13 | 2
1 | 6 | 2
0 | 3 | 2
1 | 1 | 2
1 | 0

212 | 8
4 | 26 | 8
2 | 3 | 8
3 | 0

$212_{10} = 11010100_2$

$212_{10} = 324_8$

divisioni successive per b fino a un quoto uguale a 0
i resti (dall'ultimo al primo) danno la sequenza di cifre

Conversione di un numero frazionario puro da base 10 a base b

0.6 x 2 = 1.2 = 1+0.2
0.2 x 2 = 0.4 = 0+0.4
0.4 x 2 = 0.8 = 0+0.8
0.8 x 2 = 1.6 = 1+0.6
0.6 x 2 = 1.2
....

$(0.6)_{10} = (0.\overline{1001})_2$

0.375 x 2 = 0.75 = 0+0.75
0.75 x 2 = 1.5 = 1+0.5
0.5 x 2 = 1.0 = 1+0.0

$(0.375)_{10} = (0.011)_2$

- Il numero può risultare un numero periodico nella nuova base b o dover essere approssimato
- Moltiplicazioni successive della parte decimale per la base b fino ad ottenere un numero intero oppure fino ad identificare il periodo

Rappresentazione dei numeri (in un elaboratore)

- La rappresentazione avviene in genere trasformando il numero v in un altro numero x ed operando su quest'ultimo

- Definizione dell'alfabeto origine
- Adozione di un sistema di numerazione
- Codifica a lunghezza fissa

Rappresentazione

- Bisogna tener conto dei seguenti fattori:
  – L'insieme V dei numeri da rappresentare
  – L'insieme X dei numeri rappresentanti. Tra i due insiemi si stabilisce una corrispondenza che trasforma un elemento v di V in uno x di X. Si dice allora che x è la rappresentazione di v
  – La decomposizione in cifre del numero x
  – La codifica in bit delle cifre

Overflow

- Sia la dimensione che il numero dei registri in un calcolatore sono finiti
- La cardinalità degli insiemi numerici che si rappresentano è, invece, infinita
- È inevitabile dunque che in un insieme di cardinalità infinita solo un sotto-insieme finito di elementi possa essere rappresentato
- Gli operatori aritmetici, pur essendo talvolta chiusi rispetto all'intero insieme, quasi certamente non lo sono rispetto al sotto-insieme di cardinalità finita
- Quando accade che, per effetto di operazioni, si tenta di rappresentare un numero non contenuto nel sotto-insieme si parla di *overflow*