

Conversioni decimale - esadecimale - binario

di Marco Di Martino - Peschiera Borromeo

Possiedo un personal e una HP 41 e, volendo programmare in linguaggio macchina (sì, proprio linguaggio macchina, non assembler) mi son trovato spesso a che fare con numeri binari e esadecimali e relative conversioni. Ho sviluppato così questo programma che consente di convertire numeri nelle tre basi (2, 10 o 16) fra loro e ho inoltre aggiunto una routine (LBL C) che somma un numero decimale ad uno esadecimale fornendo il risultato in esadecimale, utile quando si vogliono (o si devono) calcolare le locazioni di memoria.

Per convertire un numero binario o esadecimale in decimale ho usato il tipico algoritmo che consiste nel separare ogni singola cifra del numero da convertire (linee da 17 a 28), moltiplicarla per una potenza di 2 o di 16 e sommare i risultati (linee da 36 a 54). Esempio:

$$\begin{array}{r}
 107_{16} \text{ diventa} \quad 7 \times 16^0 = 7 + \\
 \quad \quad \quad \quad 0 \times 16^1 = 0 + \\
 \quad \quad \quad \quad 1 \times 16^2 = 256 = \\
 \hline
 263_{10}
 \end{array}$$

Ovviamente quanto detto vale per convertire un numero in base qualunque a decimale.

Per eseguire il procedimento opposto esistono, o perlomeno io conosco, due sistemi (anche se sostanzialmente sono gli stessi); uno per fare i calcoli con carta e matita, l'altro molto più adatto per un calcolatore. Il primo sistema, quello per carta e matita, per intenderci, consiste nel dividere ripetutamente il numero decimale per la base in cui deve essere convertito, arrestandosi al risultato intero e segnando il resto, fino a quando il risultato della divisione non sarà minore della base; a questo punto il risultato è dato dalla parte intera dell'ultimo quoziente seguito da tutti i resti ottenuti dalle divisioni presi però in ordine inverso a quello con cui sono stati ottenuti. Chiaro vero? Facciamo un esempio: vogliamo convertire il numero 10130₁₀ in esadecimale.

$$\begin{array}{r}
 10130 : 16 = 633 \\
 (2) \quad \quad 633 : 16 = 39 \\
 \quad \quad (9) \quad \quad 39 : 16 = 2 \\
 \quad \quad \quad (7)
 \end{array}$$

il risultato è 2792₁₆.

Il sistema usato nel programma, invece, consiste nel dividere ripetutamente il numero decimale per la base sino ad ottenere un risultato minore della base stessa (linee da 86 a 93) dopodiché la parte intera del quoziente fornirà la cifra più significativa e le cifre seguenti saranno date dalla parte intera del prodotto fra la parte frazionaria e la base.

Probabilmente un esempio numerico sarà più chiaro: convertiamo 1201₁₀ in esadecimale

$$\begin{array}{r}
 1201 : 16 = 75.0625 \\
 75.0625 : 16 = 4.6914... \quad 1^{\text{a}} \text{ cifra } 4 \\
 0.6914... \times 16 = 11.0625 \quad 2^{\text{a}} \text{ cifra } B \\
 0.0625 \times 16 = 1 \quad \quad \quad 3^{\text{a}} \text{ cifra } 1 \\
 \text{cioè } 1201_{10} = 4B1_{16}
 \end{array}$$

Entrambi i procedimenti valgono per qualsiasi base.

Il programma accetta come input massimo le seguenti cifre

7.9 E 28 ₁₀	per conversione dec. esadec.
16.777.215 ₁₀	per conversione dec. bin.
FFFFFFFF ₁₆	per conversione esadec. bin.
FFFFFFFFFFFFFF ₁₆	per conversione esadec. dec.
111111111 ₂	per conversione bin. dec. o bin. esadec.

è possibile però arrivare a 12 cifre binarie con la seguente procedura: introdurre la cifra binaria nel registro alpha, memorizzare 2 in R₂₂ e digitare XEQ 09.

Per chi possedesse il modulo estensione funzioni (HP 82180A) le linee da 07 a 57 possono essere cambiate in questo modo:

07 ALENG	20 ATOX	33 X<=Y?
08 STO 17	21 ASTO 16	34 GT0 08
09 ASTO 16	22 ASHF	35 55
10 ASHF	23 ASTO 21	36 GT0 06
11 ASTO 21	24 X=0?	37*LBL 08
12 0	25 GT0 07	38 48
13 STO 19	26 48	39*LBL 06
14*LBL 04	27 X=Y?	40 -
15 1	28 GT0 04	41 RCL 22
16 ST- 17	29*LBL 05	42 RCL 17
17 CLA	30 CLX	43 Y+X
18 ARCL 16	31 57	44 *
19 ARCL 21	32 X<>Y	45 ST+ 19

Le funzioni disponibili in questo modulo non solo consentono un risparmio di 11 passi (3 registri di programma) e di 1 registro dati, ma soprattutto portano a un risparmio di tempo che può arrivare anche al 50%.

Innanzitutto, con l'istruzione ATOX (preleva il carattere più a sinistra della stringa contenuta nel registro alpha e lascia il suo codice numerico nel registro X), il carattere è immediatamente prelevato dalla stringa senza più bisogno della lunga routine (linee da 17 a 34); inoltre non è più necessario dover confrontare il carattere estratto con tutti gli altri contenuti nei registri da 0 a 15 per calcolare il valore che rappresenta, infatti il codice dei caratteri numerici e alfabetici è uguale a quello ASCII, quindi è sufficiente sottrarre, al codice, quel numero, che lo riporta al valore rappresentato dal carattere. Per esempio nel caso dei caratteri alfabetici da A a F (codice ASCII da 65 a 70; valore esadecimale da 10 a 15) bisognerà sottrarre 55.

In questo modo per ogni carattere verrà eseguito solo un test condizionale e un salto, mentre senza le funzioni estese, nel caso della F il loop veniva ripetuto 14 volte, nel caso della A 9 volte, nel caso del 2 una volta e così via.

Per chi non possedesse un HP-41 e avesse a che fare ugualmente con conversioni fra numeri a base diversa da 10 esiste un sistema molto semplice per convertire numeri binari in esadecimali e viceversa; nel primo caso, bisogna dividere il numero binario in gruppi di 4 cifre e convertire ogni singolo gruppo, secondo la seguente tabella:

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

in pratica ad ogni 4 cifre binarie corrisponderà una cifra esadecimale e viceversa.

Conversioni decimale - esadecimale - binario

01*LBL "CONV"	55*LBL 03	109*LBL C
02*LBL a	56 1	110 STO 23
03*LBL 10	57 STO 18	111 AON
04 16	58 GTO 04	112 CLA
05 STO 22	59*LBL 07	113 STOP
06*LBL 09	60 RCL 19	114 AOFF
07 ASTO 16	61 RTN	115 XEQ 10
08 ASHF	62*LBL b	116 RCL 23
09 ASTO 21	63*LBL 13	117 +
10 1	64 CLA	118 GTO 11
11 STO 18	65 ARCL X	119*LBL D
12 0	66 2	120 XEQ 10
13 STO 19	67 STO 22	121 GTO 12
14*LBL 08	68 XEQ 09	122*LBL d
15 -1	69 RTN	123 XEQ 13
16 STO 17	70*LBL 8	124 GTO 11
17*LBL 04	71*LBL 12	125*LBL E
18 1	72 2	126 FIX 0
19 ST+ 17	73 STO 18	127 SF 01
20 " "	74 GTO 02	128 CF 29
21 ASTO Y	75*LBL A	129 "0"
22 ARCL 16	76*LBL 11	130 ASTO 00
23 ARCL 21	77 16	131 "1"
24 ASTO 16	78 STO 18	132 ASTO 01
25 ASHF	79*LBL 02	133 "2"
26 ASTO 21	80 .5	134 ASTO 02
27 ASHF	81 ST+ Z	135 "3"
28 ASTO X	82 1.9	136 ASTO 03
29 X=Y?	83 STO 16	137 "4"
30 GTO 07	84 RDN	138 ASTO 04
31 CLA	85 RDN	139 "5"
32 ASTO Y	86*LBL 14	140 ASTO 05
33 X=Y?	87 X<>Y	141 "6"
34 GTO 08	88 ISG 16	142 ASTO 06
35 STO 20	89 RCL 18	143 "7"
36 "0"	90 /	144 ASTO 07
37 ASTO Y	91 RCL 18	145 "8"
38 X=Y?	92 X<=Y?	146 ASTO 08
39 GTO 03	93 GTO 14	147 "9"
40*LBL 05	94 CLA	148 ASTO 09
41 RCL 20	95 .9	149 "A"
42 RCL IND 18	96 ST- 16	150 ASTO 10
43 X=Y?	97 X<> Z	151 "B"
44 GTO 06	98*LBL 00	152 ASTO 11
45 1	99 ARCL IND X	153 "C"
46 ST+ 18	100 AVIEW	154 ASTO 12
47 GTO 05	101 DSE 16	155 "D"
48*LBL 06	102 GTO 01	156 ASTO 13
49 RCL 22	103 RTN	157 "E"
50 RCL 17	104*LBL 01	158 ASTO 14
51 Y+X	105 FRC	159 "F"
52 RCL 18	106 RCL 18	160 ASTO 15
53 *	107 *	161 CLX
54 ST+ 19	108 GTO 00	162 END

Esempio:

$$0110\ 1001_2 = 69_{16} \quad 7A_{16} = 01111010$$

Questo sistema è molto utile quando si deve calcolare la cifra da introdurre in un registro per settare solo alcuni bit, come mi capita spesso di dover fare programmando il mio personal.

* * *

Il programma proposto questa volta è uno strumento di lavoro utile a chi ha a che fare con i calcolatori. L'autore ne descrive abbastanza bene il funzionamento, ma ha dimenticato di fornire istruzioni per l'uso di tale programma. Ecco, per ciascuna conversione, le operazioni da compiere:

DEC - HEX: impostare sul registro X il numero da convertire e premere XEQ "A".

HEX - DEC impostare il numero esadecimale da convertire, sotto forma di caratteri introdotti nel registro ALPHA e premere XEQ "a"

DEC - BIN: impostare sul registro X il numero da convertire e premere XEQ "B"

BIN - DEC: impostare sul registro X il numero da convertire e premere XEQ "b"

HEX - BIN: impostare il numero esadecimale da convertire, sotto forma di caratteri introdotti nel registro ALPHA e premere XEQ "D"

BIN - HEX: impostare sul registro X il numero da convertire e premere XEQ "d".

Per sommare un numero decimale a un numero esadecimale, con risultato in forma esadecimale, impostare il numero decimale in X, premere XEQ "C", impostare il numero esadecimale (caratteri ALPHA) da sommare, quindi premere R/S.

Ovviamente va ricordato che in modo USER, le label utilizzate in questo programma risultano assegnate automaticamente alle prime due righe di tasti.

MC

414*LBL 10
415 "TJ10"
416*LBL 21
417 AVIEW
418 PSE
419 PSE
420 RTN

Errata corrige

Nel programma "Battaglia navale" pubblicato nel n. 25, per un errore tipografico sono state omesse alcune righe che riportiamo qui sopra.

Chiediamo scusa ai lettori.