

Come già fatto sul n° 5 di MCmicrocomputer e visto il successo ottenuto in termini di programmi arrivati in redazione, ne pubblichiamo due, per la calcolatrice TI-57, del lettore Marcello Antonucci di Roma: il primo riguarda il problema della conversione da un numero, decimale ad un binario ed il secondo la determinazione della frazione generatrice di un numero decimale. Soprattutto quest'ultimo presenta caratteristiche abbastanza originali, ma tutto sommato, come l'esperto lettore potrà arguire, questi programmi rispecchiano la semplicità d'uso della TI-57.

Veramente notevole è invece il contributo del lettore Stefano Laporta di Bologna per l'"Angolo delle TI". Già "contribuente" per il n° 7, questo lettore torna subito sulla ribalta per presentarci in ANTEPRIMA ASSOLUTA delle notizie sulla TI-57: anche in questa calcolatrice sono state "scoperte" delle istruzioni nascoste!

Prima di passare all'analisi dei due programmi facciamo alcune precisazioni, su esplicita richiesta di un paio di lettori. I listati dei programmi presentati su questo numero, così come quelli del n° 5, NON sono ottenuti direttamente dalla TI-57, in quanto questa non è in alcun modo collegabile con la stampante PC-100C (oppure qualche lettore ... in gamba ... c'è riuscito?). Questi listati sono ottenuti invece con la TI-59 con un apposito programma, ideato da Pierluigi Panunzi, che permette l'impostazione da tastiera delle singole istruzioni del listato della TI-57 e che in uscita fornisce su stampante un listing il cui "formato" è simile a quello ottenibile con le TI-58 e 59.

In particolare, da sinistra a destra, avremo il numero del passo di programma, la codifica dell'istruzione digitata (così come si legge sul display della TI-57) ed infine la decodifica dell'istruzione stessa.

In particolare se la funzione è preceduta da una "I", vuol dire che bisogna premere il tasto "INV" (fatto confermato dal "-" nella seconda colonna).

Inoltre, seguendo le modalità di codifica della stampante, le istruzioni " $x=t$ " ed " $x \geq t$ " sono indicate rispettivamente con "EQ" e "GE".

INVIATECI I VOSTRI PROGRAMMI!

Se, qualunque sia la vostra macchina, avete realizzato programmi o routine che ritenete possano interessare altri lettori, inviateceli. Saranno esaminati e, se pubblicati, ricompensati con valutazioni approssimativamente fra le 30 e le 100.000 lire, secondo la complessità, la genialità, l'originalità e la presentazione del materiale e della documentazione (listati, diagrammi, commenti ecc.). Per ragioni organizzative non possiamo impegnarci, salvo eventuali accordi presi prima dell'invio, alla restituzione dei materiali, che resteranno di proprietà della redazione che si impegna a non divulgarli (se non tramite la rivista) senza l'autorizzazione dei rispettivi autori.

Conversione decimale-binario

di Marcello Antonucci - Roma

Questo programma serve a trasformare un numero positivo, negativo o nullo purché intero, dalla forma decimale a quella binaria: è particolarmente indirizzato a chi faccia semplici esercizi di aritmetica binaria, ad esempio quelli proposti da corsi di elettronica digitale e/o microcomputer.

L'algoritmo risolutivo è quello classico e prevede successive divisioni per 2 del numero dato, con conseguente annotazione del resto ottenuto, che può essere solo 0 o

1. Quando l'ultimo quoziente è nullo si scrivono tutti i resti di seguito, in ordine inverso rispetto a come sono stati ricavati.

Perciò il programma prevede essenzialmente la divisione per 2 del numero, l'isolamento del quoziente intero e del resto, nonché il calcolo del "peso" di tale resto. Tale peso, ricavato come esponente di 2, sarà a sua volta l'esponente da dare a 10 per avere visualizzato un numero in base 10 formalmente coincidente con il numero richiesto in base 2.

Così il primo resto vale $2^0 = 1$ (perciò sul display si avrà 10^0), mentre l'n-esimo varrà 2^{n-1} e perciò sul display si avrà 10^{n-1} . Per questo il registro pesi (R0) partirà da 8 per calare fino a 0 ed il peso di ogni cifra è $8-R0$.

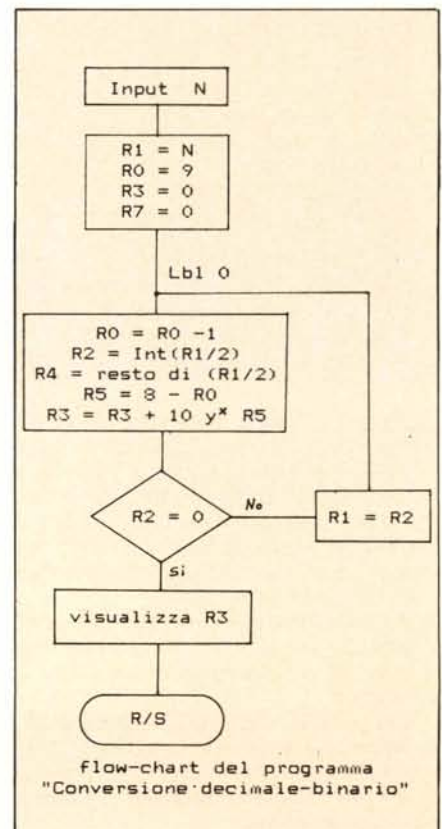
Il valore iniziale pari ad 8, unito all'uso dell'istruzione "Dsz" per decrementare questo registro, limita a 255 il valore del massimo numero impostabile e correttamente elaborato; tale limitazione è motivata anche dal fatto che 256 corrisponde a 100000000 in binario ed il display ha solo 8 cifre.

Volendo ampliare la portata (l'ultimo bit si riconosce dalla parità o meno del numero di partenza), conviene porre "Fix 8" subito dopo la label 2 e si deve sostituire il "9" presente al passo 01 con "10" o più, nonché l'"8" del passo 19 con "9" o più, mantenendolo però sempre la stessa diffe-

Conversione decimale binario

```

00 22 1   STD 1
01 09 9   ?
02 32 0   STD 0
03 15 15  CLR
04 32 3   STD 3
05 32 7   STD 7
06 39 1   RCL 1
07 36 0   LBL 0
08 45 5   =
09 02 2   =
10 32 4   STD 4
11 85 5   =
12 32 2   STD 2
13 49 4   I INT
14 39 4   PRD 4
15 33 2   RCL 2
16 96 6   DSZ
17 49 4   INT
18 32 2   STD 2
19 08 8   8
20 32 5   STD 5
21 39 9   RCL 9
22 34 5   I SUB
23 33 5   RCL 5
24 18 1   LDC
25 55 5   =
26 33 4   RCL 4
27 85 5   =
28 34 3   SUB
29 33 3   RCL 3
30 42 2   EE
31 42 2   EE
32 66 6   EQ
33 51 1   GTD 1
34 32 1   STD 1
35 51 0   LTD 0
36 96 2   LBL 2
37 83 3   RCL 3
38 81 3   R/S
39 71 1   RST
    
```



renza tra tali due numeri. Oppure, invece di queste due sostituzioni, si può cambiare la sequenza

RCL 2 Dsz Int STO 2
con la sequenza

1 INV SUM 0 RCL 2 INT STO 2
ma si perde qualcosina in velocità.

Per l'elaborazione basta introdurre il numero desiderato e premere RST R/S: dopo una decina di secondi si avrà sul display la codifica in binario del numero impostato. Se tale numero era maggiore di 255 si avrà un risultato in forma esponenziale.

Frazioni generatrici

di Marcello Antonucci - Roma

Eseguire una sia pur semplice espressione matematica con la calcolatrice presenta una grande seccatura: il risultato è in forma decimale anche quando sarebbe molto comodo sotto forma di frazione. Anche un banale $8/7$, scritto come 1.1428571... può riuscire a mettere in imbarazzo chiunque.

Questo programma risparmia la fatica di cercare a quale frazione potrebbe corrispondere quel numero, basandosi sul numero delle cifre del periodo e sulla natura di esse.

Il principio è molto semplice: supponiamo che il numero N considerato sia uguale a P/Q (ovviamente entrambi incogniti!).

Allora basterà prendere $Q = 1, 2, 3, \dots$ e vedere il valore ottenuto per P , tramite $P = N \cdot Q$.

Fissato un errore massimo ϵ su N , si avrà

$$N \pm \epsilon = P/Q \Rightarrow P = QN \pm Q\epsilon$$

Perciò si può ispezionare la differenza tra QN e l'intero immediatamente inferiore e poi quello superiore: se tale differenza risulta minore di $Q\epsilon$, allora si è arrivati alla fine, altrimenti si incrementa di un'unità Q e si ricomincia.

Ciò però non può bastare: anche frazioni facili del tipo $\sqrt{3/2}$ erano fuori della portata del programma.

Allora basta prevedere che ogni numeratore, che non fosse intero a meno di $Q\epsilon$, venisse elevato al quadrato e subisse ancora il test, in modo da vedere se per caso tale numeratore non fosse la radice quadrata di un ... quasi intero.

Per distinguere alla fine se il numeratore va preso così com'è o sotto radice, è stato previsto il lampeggiamento del display in quest'ultimo caso. Tra l'altro ciò è facilmente ottenibile cambiando di segno il numeratore (se è radicale) ed elevandolo alla prima potenza.

Effettuando tutte queste operazioni, il programma va "stretto" alla 57 e non permette la visualizzazione automatica dell'errore commesso. In effetti, sebbene sia comodo sapere di quanto si sbaglia, tuttavia è sufficiente imporre un valore basso per ϵ (ad esempio 10^{-6}): se il programma

trova lo stesso la frazione, ha poca importanza sapere che l'errore è di 10^{-10} quando non si può essere certi della decima cifra decimale in nessun calcolo con la TI-57.

Un'ultima nota: può sembrare strano che, quando il programma ha trovato la frazione, esca da una subroutine e trovi un R/S di fine elaborazione piuttosto che un più naturale INV SBR.

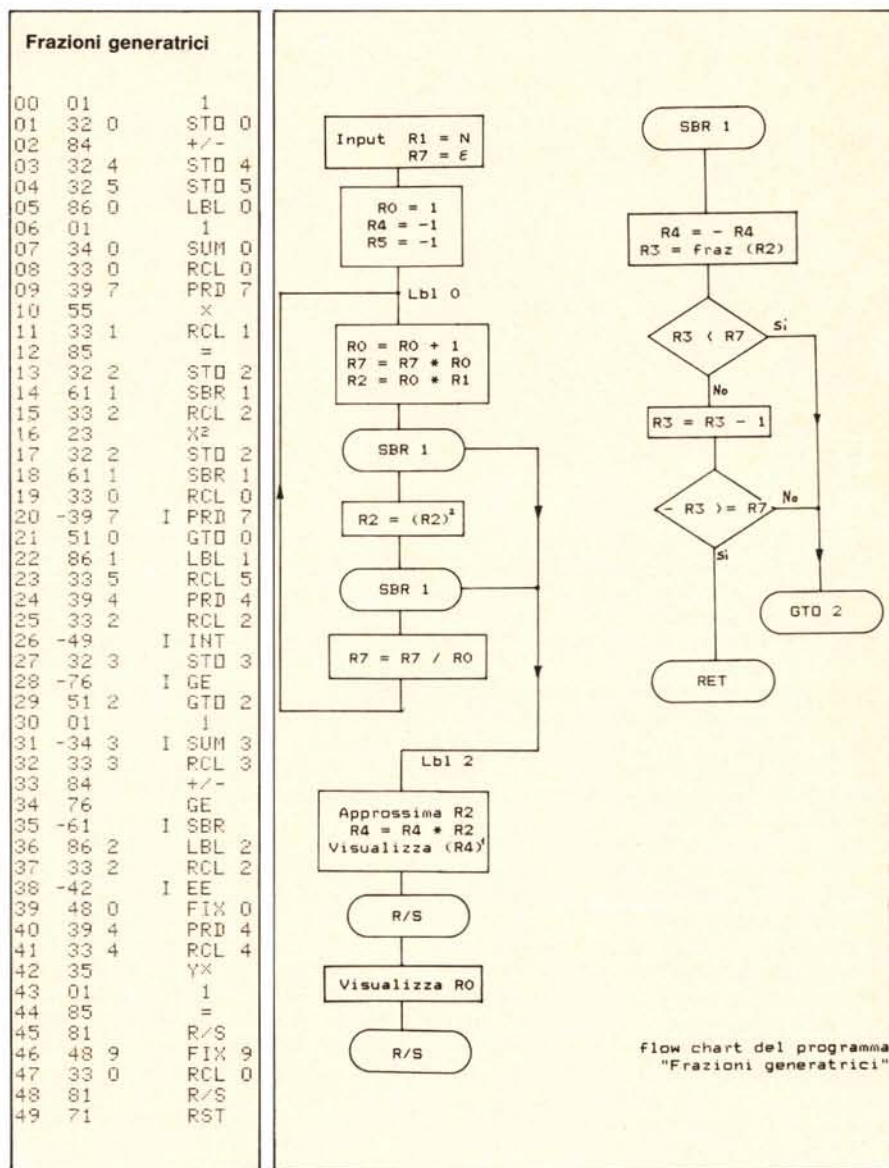
Tuttavia ciò funziona e successive elaborazioni non fanno "scoppiare" il registro di ritorno dalle subroutine, fatto questo che non trova riscontro sul manuale della calcolatrice.

Per il funzionamento del programma si deve operare nel seguente modo: si imposta il numero N desiderato (attenzione al fatto che deve essere positivo!) e lo si deposita in STO 1, quindi si introduce il valore di ϵ e lo si deposita in STO 7.

Premendo ora RST R/S si avrà, dopo un certo tempo dipendente da tanti fattori non troppo predicibili, sul display un valore che rappresenta il numeratore: se è lam-

peggiante, deve essere considerato posto sotto la radice, altrimenti non si deve fare altro che annotarselo. In questo secondo caso, premuto ancora R/S, si otterrà il valore del denominatore, senza alcuna complicazione. Invece nel caso in cui il display lampeggia, bisogna azzerarlo con CE e poi bisogna premere per DUE VOLTE il tasto R/S: tra l'altro questo è un difetto "congenito" della TI-57 e purtroppo abbiamo notato che in certi casi anche le TI-58 e soprattutto la 59 presentano il difetto di richiedere l'impostazione di due R/S laddove a rigor di logica ne dovrebbe bastare solo uno.

Come esempio di applicazione consideriamo $N = \pi$, $\epsilon = 10^{-6}$: si otterrà la frazione $355/113$ con un errore di circa $3 \cdot 10^{-7}$. Con $N = \sqrt{3/2}$ ed $\epsilon = 10^{-4}$ si ha come risultato $\sqrt{3/2}$ con un errore di circa $2.5 \cdot 10^{-5}$. Come tempo ci mette una vita, però un essere umano ci metterebbe molte vite!



L'ANGOLO DELLE TI

Ecco qui, in anteprima assoluta, alcune scoperte del lettore Stefano Laporta di Bologna, il quale è riuscito addirittura a far visualizzare le lettere A, B, ..., F, sul display della sua 57! Il metodo da seguire non è semplice né tantomeno intuitivo: si potrebbe paragonare al metodo iniziale della "programmazione sintetica" per la calcolatrice HP 41C, mentre viceversa consente di effettuare operazioni abbastanza strane, che sono senz'altro da ritenersi dei "buchi" nel sistema operativo della TI-57.

Ecco dunque quanto ci scrive il nostro lettore.

Le istruzioni "segrete" si dividono in due gruppi, a seconda che si possano o meno introdurre da tastiera.

A) Funzioni introducibili da tastiera

L'unica funzione appartenente a questa classe è "SBR" senza numero di etichetta, impostabile premendo SBR SST. Essa agisce così: calcola l'esponente che avrebbe il numero contenuto nel visualizzatore se fosse in notazione esponenziale, poi salta al primo passo di programma che corrisponde all'esponente; se questo è maggiore di 9, in valore assoluto, si ha errore.

Facciamo un esempio: sul display è contenuto il numero 2500, e cioè $2.5 E 03$; la calcolatrice cerca il primo passo di programma che contiene il numero "3".

Se si imposta il programma SBR (SST) 0 1 2 3 4 5 6 7 8 9 = R/S e si preme 2500 RST R/S apparirà 456789. Se si preme

N	3 =	+/-	3 +/-	3 +/- EE	. 3 =	EE 3	. 3 +/- =
0	8	11	-11	12	-12	13	-13
1	9	21	-21	22	-22	23	-23
2	10	31	-31	32	-32	33	-33
3	11	41	-41	42	-42	43	-43
4	12	51	-51	86 7	51 7	86 8	51 8
5	13	61	-61	86 4	51 4	86 5	51 5
6	14	71	-71	86 1	51 1	86 2	51 2
7	15	81	-81	86 0	51 0	83	-83
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

Tabella - Codici ottenibili con le sequenze generatrici (s.g.) a seconda del valore di N impostato.

1000000 RST R/S apparirà 789.

Questa istruzione costituisce una specie di SBR indiretto; la sequenza $n 10^x$ SBR ... salta al primo passo di programma contenente il numero n.

Ma le stranezze non sono finite qui, anzi qui comincia il bello!

B) Funzioni non direttamente introducibili da tastiera

La maggior parte delle funzioni nascoste sulla TI-57 è di questo tipo: il fatto di non poter agevolmente introdurre un qualunque byte, come succede invece nelle TI-58 e TI-59 (ad esempio l'istruzione HIR), costringe a "salti mortali" da parte del programmatore.

Infatti è in generale necessario introdurre un piccolo programma generatore, da inserire a partire dal passo 00, programma che contiene una sequenza alquanto anomala di istruzioni.

Il programmino è

Exc (SST) Lbl 5 ... sequenza generatrice ... R/S
dove il tasto SST è premuto per generare un "Exc" senza indice e la "sequenza generatrice" è una di quelle riportate in tabella. Successivamente si preme LRN (per uscire dal modo di apprendimento), RST R/S (si elabora il programmino), LRN (si ritorna alla programmazione della 57) ed un qualsiasi tasto numerico: si sarà così generato il codice indicato sulla tabella.

Facciamo un esempio: impostiamo il programma

Exc (SST) Lbl 5 3 = R/S

dove "3 =" è la prima sequenza generatrice; premiamo poi LRN. Abbiamo in questo modo introdotto il programma generatore e a seconda di ciò che faremo da ora in poi avremo risultati differenti.

Premiamo ora RST R/S LRN: sul display vedremo 05 00 0, come se dovessimo inserire un codice composto. Premiamo ora il numero 3 e successivamente BST per andare a leggere che cosa abbiamo impostato in realtà. Come per incanto comparirà il codice "11": per vedere cosa fa premiamo LRN per uscire dall'apprendimento e premiamo SST. Sul display apparirà una "b" minuscola...

Infatti il numero 11 in esadecimale è proprio "b"! Ripetiamo la stessa cosa con i numeri 2 3 4 5 6 7: ogni volta comparirà il codice che si riferisce alle lettere A b C d E F esadecimali!

Questi codici, una volta generati, possono essere spostati usando le funzioni Ins e Del, e possono essere usati per scritte sul visualizzatore, oppure opportunamente inseriti in un programma di conversioni decimale - esadecimale (tanto per fare un esempio). Tra l'altro, far scrivere alla calcolatrice la parola "CIAO" è ora diventata una bazzecola!

C'è però una stranezza (tanto per cambiare!) se il codice viene eseguito mentre il display non è azzerato e si tratta della prima "cifra", viene impostata la notazione esponenziale. Ad esempio il codice "14" dà E se il contenuto del display era zero, "4 E 00" altrimenti.

Esempio n° 2

Se si usa la sequenza generatrice "+/-", si ottengono vari codici, tra cui quelli anomali sono 11, 21, 31, 41 ottenuti premendo i tasti numerici 0, 1, 2, 3.

Il codice "11" è apparentemente uguale al codice 11 = B esadecimale, mentre esplica una funzione diversa (evidentemente esiste una differenza non visualizzabile). Esso esegue operazioni misteriose: il display passa apparentemente in modo LRN, ma provando invece a premere EE 23. 123 +/- si vedranno cose ben strane (quasi come sulla TI-59!). Inoltre, provando ad eseguire un'operazione (ad esempio $14 \times 2 = 28$) e premendo CLR per cancellare il display, se si esegue il codice "11" e l'istruzione "=" nel visualizzatore torna il risultato che era stato cancellato, cioè 28!! Ovvero, l'istruzione CLR cancella il display, ma il numero rimane "da qualche parte" ed il codice "11" in qualche maniera lo recupera, analogamente all'istruzione "Last x" delle HP.

Il codice "21": quando viene incontrato nel corso dell'esecuzione del programma (cioè con R/S e non SST) il display passa in LRN ed il contatore di programma comincia ad avanzare velocemente ... distruggendo il programma; che sia utile per lavori spionistici???

I codici 31 e 41: si comportano rispettivamente come R/S e SST.

Esempio n° 3

Usando la sequenza generatrice "3 +/- EE" si genera un solo codice anomalo, il "12", ancora una volta da non confondere con l'analogo codice "12" che equivale alla C esadecimale.

La sua esecuzione moltiplica per una potenza di 10 il numero contenuto nel display, in maniera alquanto bizzarra.

Gli altri codici ottenibili con altre sequenze riportate in tabella sono o gli inversi dei comandi già visti, oppure codici del tutto normali, tranquillamente impostabili da tastiera.

Rimangono ancora "misteriosi" molti codici: 16, 17, 37, 47 e quelli corrispondenti alla posizione dei tasti numerici.

È peraltro possibile che con altre "sequenze generatrici" o trucchi analoghi si possano scoprire altre stranezze della TI-57: ora che i lettori conoscono queste notizie, potranno sbizzarrirsi a cercare chissà quale altra caratteristica.

P.P.