

RES-RES2

di Corrado Cantelmi - Frattocchie (Roma)

Più volte mi sono proposto di imparare a memoria la scala dei valori delle resistenze (anzi, dei resistori), ma altrettante volte l'ho dimenticata.

Per avviare alla ricerca affannosa del faticoso cartellino con colori e valori corrispondenti, ho cercato di implementare il tutto sulla mia HP-41C: sono così nati i programmi "RES" e "RES2".

Il primo dei due non è altro che un mezzo per conoscere il valore di una resistenza, introducendo le prime due lettere dei colori impressi sulla resistenza stessa.

Le resistenze comunemente in commercio sono caratterizzate da tre colori (tra Nero, Marrone, Rosso, Arancione, Giallo, Verde, Bleu, Violetto, Grigio e Bianco), più il colore Oro o Argento o nessun colore (che indicano la tolleranza, rispettivamente del 5%, del 10% e del 20%). Dopo aver avviato il programma, il calcolatore fa apparire sul visore la richiesta "COLORI? 2LET", che invita l'operatore ad introdurre le prime due lettere di ciascun colore impresso sull'involucro della resistenza di cui si vuol conoscere il valore. Molto importante. I colori devono essere introdotti partendo dalla parte opposta alla tolleranza. Si voglia, ad esempio, sapere che valore ha una resistenza; dopo aver determinato i colori (per esempio Marrone, Verde, Giallo e oro), si avvia il programma e, alla richiesta, si risponde "MAVEGI", quindi si preme il tasto R/S.

Dopo circa tre secondi apparirà la scritta "150000 OHM".

Il circoletto color oro significa, come già detto, che la resistenza avrà un valore di 150000 ohm al 5%, cioè più o meno 7500 ohm. Naturalmente nel registro x verrà introdotto tale valore, nell'esempio illustrato 150000.

Mi è sembrato inutile inserire nel programma il calcolo per la tolleranza della resistenza, dal momento che i tre valori, 5%, 10%, 20%, sono facilmente ricordabili.

I valori ed i corrispondenti colori sono:

- 0 = Nero
- 1 = Marrone
- 2 = Rosso
- 3 = Arancione
- 4 = Giallo
- 5 = Verde
- 6 = Bleu
- 7 = Violetto
- 8 = Grigio
- 9 = Bianco

Se il terzo colore è l'oro (ORo), si ha una divisione del valore per dieci; se è l'argento (ARgento), per cento.

Le lettere in Maiuscolo sono quelle che devono essere introdotte.

Se al contrario, si dovesse convertire un valore in ohm nei colori corrispondenti, si dovrebbe far uso del programma "RES2".

Si debba, ad esempio, convertire il valore di 47 ohm.

Si dovrà, dapprima, introdurre il valore 47 nel registro x e quindi premere XEQ "RES2" (o il tasto a cui è stata assegnata la LBL "RES2"). In pochi secondi il calcolatore visualizzerà i colori corrispondenti (nell'esempio, "GI VI NE").

Gli algoritmi dei programmi "RES" e "RES2" sono gli stessi che si seguono facendo i calcoli a mente.

La complessità degli algoritmi è causata dai valori non interi delle resistenze, che comportano una diversa impostazione per la soluzione delle conversioni.

Vale la pena di pubblicare questi programmi, sia per l'utilità di avere un "interprete" del codice valori di quasi tutte le resistenze reperibili in commercio, sia per la buona utilizzazione, che in essi viene fatta, delle caratteristiche alfanumeriche della 41C.

I programmi funzionano bene, ovviamente i valori di resistenza trattati vengono sempre arrotondati per difetto a due cifre significative, per esempio, chiedendo il codice colori di una resistenza da 9999 ohm, la calcolatrice risponderà "BI BI RO" cioè Bianco Bianco Rosso, codice corrispondente a 9900 ohm.

L'input-output è abbastanza curato e forse esagerato in alcuni particolari come, per esempio, i passi 35 ÷ 38 del programma "RES" che servono addirittura per togliere o meno il punto di separazione decimale a seconda che si debba visualizzare un numero superiore o inferiore a 1000; una caratteristica questa piuttosto superflua, semmai sarebbe stato meglio aggiungere qualche passo in modo da rendere possibile la visualizzazione in kohm per i risultati compresi tra 10³ e 10⁶, e in Mohm per quelli oltre 10⁶.

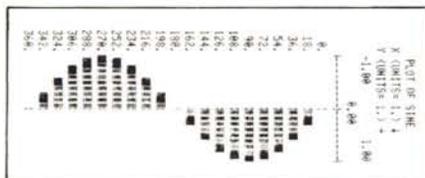
Da notare che entrambi i programmi proposti terminano con una label prima dell'END, tale label viene richiamata solo al termine dell'elaborazione; in pratica il Cantelmi ha disposto le cose in modo che il puntatore di programma, al termine dell'elaborazione, si trovi esattamente sull'ultima istruzione del listato, per cui è sufficiente premere R/S per cominciare una nuova elaborazione; ciò avviene dato che il puntatore una volta giunto all'END di un programma, proseguendo la sua corsa, ricomincia dalla prima istruzione.

BPLOT

di Corrado Cavallini - Padova

Mi permetto di inviarvi un programma di uso molto generale, e che non richiede per essere capito conoscenze molto elevate. Si tratta di un programma per il plottaggio di diagrammi a barre, che per quanto riguarda la sezione input è pressoché identico al prog. standard "PRPLOT". Il programma in questione è nato dall'idea, di sostituire all'istruzione 97 REGPLOT, del programma PRPLOT, un'opportuna subroutine che

Programma RES			Programma RES 2		
27 +	55 RTN				
28 - *	56*LBL "GI"				
01*LBL "RES"	29 ARCL Y	57 4			
02 FIX 0	30 ASHF	58 RTN			
03 CF 29	31 ASTO Z	59*LBL "VE"			
04 "COLORI? 2LET"	32 XEQ IND Z	60 5			
05 00H	33 101X	61 RTN			
06 STOP	34 *	62*LBL "BL"			
07 00FF	35 1000	63 6			
08 ASTO X	36 X=Y?	64 RTN			
09 " "	37 SF 29	65*LBL "VI"			
10 ARCL X	38 X<Y	66 7			
11 ASTO Y	39 " "	67 RTN			
12 " "	40 ARCL X	68*LBL "GR"			
13 ARCL Y	41 " OHM"	69 8			
14 ASHF	42 RVIEW	70 RTN			
15 ASTO Y	43 GTO 00	71*LBL "BI"			
16 XEQ IND Y	44*LBL "HE"	72 9			
17 10	45 0	73 RTN			
18 *	46 RTN	74*LBL "OR"			
19 " "	47*LBL "MA"	75 FIX 1			
20 ARCL Y	48 1	76 -1			
21 ASTO Z	49 RTN	77 RTN			
22 " "	50*LBL "PO"	78*LBL "AG"			
23 ARCL Z	51 2	79 FIX 2			
24 ASHF	52 RTN	80 -2			
25 ASTO Z	53*LBL "AR"	81*LBL 00			
26 XEQ IND Z	54 3	82 END			
			01*LBL "RES2"	25 LASTX	49 *
			02 10	26 GTO 16	50*LBL 16
			03 X<Y	27*LBL 14	51 " "
			04 X<Y?	28 10.1	52 XEQ IND Y
			05 GTO 13	29 10	53 "F"
			06 ENTER†	30 LASTX	54 XEQ IND X
			07 LOG	31 *	55 "F"
			08 INT	32 FRC	56 XEQ IND Z
			09 STO Z	33 X=0?	57 RVIEW
			10 101X	34 GTO 15	58 GTO 12
			11 /	35 10	59*LBL 00
			12 ENTER†	36 *	60 "HE"
			13 FRC	37 R†	61 RTN
			14 10	38 10	62*LBL 01
			15 *	39 *	63 "MA"
			16 1	40 X<Y	64 RTN
			17 ST- T	41 GTO 16	65*LBL 02
			18 RDN	42*LBL 15	66 "RO"
			19 GTO 16	43 ISG Y	67 RTN
			20*LBL 13	44 X<L	68*LBL 03
			21 FRC	45 10	69 "AR"
			22 X=0?	46 /	70 RTN
			23 GTO 14	47 R†	71*LBL 04
			24 STO Y	48 10	72 "GI"



Esempi di funzioni tracciate con "BPLOT"

eseguisse la stampa di una barra.

Una difficoltà incontrata è stata che, non contenendo il buffer della stampante più di 43 colonne speciali, per ottenere un plottaggio con la risoluzione di ± 0.5 colonne si è dovuto ricorrere all'artificio di costruire una barra come insieme di caratteri normali e speciali.

Ogni barra risulta quindi costituita da:

- 1 - La colonna Asse.
- 2 - Un certo numero di caratteri 31 (0 ÷ 18)
- 3 - Un certo numero di colonne 127 (0 ÷ 13).

In questo modo si è ottenuta la voluta precisione di ± 0.5 col. Alcune aggiunte sono state effettuate all'interno del loop LBL14... GTO14 per ottenere un corretto incolonnamento delle x-label.

La label "BP", analogamente a REGPLOT usa il valore del REGO2 come parametro, ed è quindi adatta a plottaggi di qualsiasi ampiezza. In più, però, utilizza i registri 12 e 13, ed il flag 00. Size necessario è quindi 014.

L'input dei dati del programma BPLOT è identico a quello di PRPLOT, salvo che non è possibile sopprimere la stampa dell'asse.

Per ottenere BPLOT si consiglia la seguente procedura:

- 1 - Eseguire COPY PRPLOT
- 2 - Eseguire GTO.001; cancellare 01 LBL PRPLOT; inserire 01 LBL BPLOT
- 3 - Eseguire GTO.042; cancellare 42 LBL PRPLOT; inserire 042 LBL BPLOT
- 4 - Eseguire GTO.091; inserire le segg. istruzioni:

92 ENTER↑	99 RCL 05
93 X<0?	100 ABS
94 LOG	101 -
95 INT	102 X<0?
96 CHS	103 0
97 2	104 SKPCHR
98 +	105 RDN

5 - Eseguire GTO.111; cancellare 111 REGPLOT; inserire 111 XEQ"BP"

6 - Eseguire GTO.120; eseguire DEL.179 ("PRAXIS" non viene modificato e può quindi rimanere su ROM)

7 - Eseguire GTO.176; cancellare END; inserire RTN e tutte le istruzioni segg. sino alla fine.

8 - Eseguire GTO..

Il programma risultante dovrebbe essere lungo 434 Byte.

Senza altro succederà che a qualcuno non vada bene il fatto che il programma "PRPLOT", contenuto nella ROM della

Programma BPLOT

01•LBL "BPLOT"	52 RCL 09	103 0	156 X=0?	209 1
02 AON	53 *X	104 SKPCHR	157 GTO 00	210 -
03 *NAME ?	54 XEQ 09	105 RDN	158 ACX	211 SKPCOL
04 PROMPT	55 STO 07	106 ACX	159 101X	212 LASTX
05 AOFF	56 7	107 3	160 2	213 +
06 ASTO 11	57 ACCHR	108 SKPCOL	161 STO 05	214 -
07•LBL 11	58 PRBUF	109 RCL 06	162 FIX 2	215 STO 12
08 *Y MIN ?	59 130	110 XEQ IND 11	163 RDN	216 7
09 PROMPT	60 STO 02	111 XEQ "BP"	164 GTO 01	217 /
10 STO 00	61 XROM "PRAXIS"	112 RCL 10	165•LBL 00	218 1
11 *Y MAX ?	62 RCL 10	113 ST+ 06	166 1	219 -
12 PROMPT	63 X<0?	114 RCL 09	167 ACX	220 INT
13 STO 01	64 GTO 00	115 RCL 06	168 FIX IND 05	221 X<0?
14 X<=Y?	65 RCL 09	116 X<=Y?	169•LBL 01	222 0
15 GTO 11	66 RCL 00	117 GTO 14	170 *) "	223 STO 13
16•LBL 12	67 -	118 FIX 4	171 ACA	224 7
17 *AXIS ?	68 RCL 10	119 RTN	172 RTN	225 *
18 CF 23	69 ABS	120•LBL 09	173•LBL 10	226 ST- 12
19 PROMPT	70 /	121 *+ <UNITS="	174 0	227 FS? 00
20 STO 04	71 STO 10	122 X<=Y?	175 /	228 GTO 00
21 FS? 23	72•LBL 00	123 GTO 10	176 RTN	229 XEQ 03
22 ASTO 04	73 RCL 09	124 X<Y	177•LBL "BP"	230 XEQ 05
23 RCL 01	74 RCL 08	125 ABS	178 RCL 00	231•LBL 00
24 X<Y?	75 ABS	126 X<Y?	179 -	232 119
25 GTO 12	76 X<Y?	127 X<Y	180 RCL 01	233 ACCOL
26 CLX	77 X<Y	128 LOG	181 RCL 00	234 FC? 00
27 RCL 00	78 RCL 07	129 X<0?	182 -	235 GTO 00
28 X>Y?	79 /	130 GTO 00	183 /	236 XEQ 05
29 GTO 12	80 LOG	131 INT	184 X<0?	237 XEQ 03
30•LBL 13	81 INT	132 2	185 0	238•LBL 00
31 *X MIN ?	82 2	133 X<Y	186 ENTER↑	239 PRBUF
32 PROMPT	83 -	134 X>Y?	187 1	240 RTN
33 STO 08	84 STO 05	135 GTO 01	188 X<=Y?	241•LBL 05
34 *X MAX ?	85 RCL 08	136 -	189 X<Y	242 RCL 13
35 PROMPT	86 STO 06	137 STO 05	190 RDN	243 X=0?
36 STO 09	87•LBL 14	138 0	191 RCL 02	244 RTN
37 X<=Y?	88 FIX IND 05	139 GTO 02	192 INT	245 31
38 GTO 13	89 RCL 07	140•LBL 00	193 1	246•LBL 06
39 *X INC ?	90 /	141 FRC	194 -	247 ACCHR
40 PROMPT	91 RND	142 X=0?	195 *	248 DSE Y
41 STO 10	92 ENTER↑	143 1	196 1	249 GTO 06
42•LBL "BPLOT"	93 X=0?	144 LASTX	197 +	250 RTN
43 CF 12	94 LOG	145 INT	198 FIX 0	251•LBL 03
44 ADV	95 INT	146 X<Y	199 RND	252 RCL 12
45 6	96 CHS	147 -	200 RCL 02	253 X=0?
46 SKPCHR	97 2	148•LBL 01	201 FRC	254 RTN
47 *PLOT OF "	98 +	149 * E	202 1 E3	255 127
48 ARCL 11	99 RCL 05	150•LBL 02	203 *	256•LBL 04
49 ACA	100 ABS	151 4	204 CF 00	257 ACCOL
50 PRBUF	101 -	152 SKPCHR	205 X<=Y?	258 DSE Y
51 RCL 08	102 X<0?	153 ACA	206 SF 00	259 GTO 04
		154 FIX 0	207 FC? 00	260 RTN
		155 RDN	208 X<Y	261 END

stampante 82143A, tracci i grafici delle funzioni desiderate, soltanto stampando asterischi o altri simboletti. In tal caso, il programma del sig. Cavallini può costituire una valida alternativa, tracciando i grafici per mezzo di barre di adeguata lunghezza.

Validissima è la soluzione adottata dall'autore per ottenere barre sufficientemente lunghe nonostante la scarsa capacità del buffer di riga della stampante, senza tuttavia rinunciare alla buona risoluzione che si ottiene ottenendo tali barre dall'affiancamento di un adeguato numero di righe "127" accumulate per mezzo della funzione "ACCOL".

La risoluzione ottenibile da un grafico così tracciato è identica a quella ottenibile con la funzione "PRPLOT"; forse qualcosa è peggiorato per quello che riguarda la velocità di esecuzione, che nella nuova versione è più bassa ma comunque sempre

accettabile.

Seguendo alla lettera le istruzioni fornite dall'autore della modifica, si riesce rapidamente a ottenere, dall'originale su ROM, il nuovo programma modificato; unico problema è il fatto di non poter fare a meno di trasferire il programma "PRPLOT" su RAM (memoria di programma) pena l'impossibilità di modificarlo, e ciò obbliga ad avere a disposizione una buona manciata di Byte di memoria liberi.

Posizionando l'asse delle ascisse ad un estremo del grafico, è possibile praticamente tracciare degli istogrammi.

Volendo migliorare la risoluzione del grafico, ci si può cimentare nella modifica del programma in modo che ogni colonna abbia la sommità non piatta, ma smussata secondo l'andamento della funzione nell'intorno del valore centrale preso come campione; non è facile, ma aiuterebbe molto la precisione del nostro "plotterino".