



COME UTILIZZARE

Anche se nelle nostre spiegazioni cerchiamo sempre di utilizzare un linguaggio semplice e facilmente comprensibile, possono egualmente capitare dei casi in cui la trattazione per chi legge non risulta esauriente al 100%.

Questa condizione sembra essersi verificata per esempio in occasione dell'articolo da noi pubblicato sul n. 73 della rivista, relativo all'interprete BASIC da 5,5 K per il nostro microcomputer.

Molti lettori infatti, dopo aver letto tale articolo, hanno tentato di caricare in memoria il Basic e a questo punto qualcuno si è trovato di fronte a difficoltà di vario genere, in taluni casi anche difficilmente prevedibili, per cui ci ha subito telefonato o scritto chiedendoci «aiuto».

I motivi di questi «insuccessi» momentanei possono essere così riassunti:

1) Qualcuno aveva male interpretato le spiegazioni fornite sulla rivista e procedendo in modo errato, non appena ha tentato di trasferire in memoria il Basic, si è subito arenato perché il computer dava ERRORE.

2) Altri hanno provato a leggerlo dalla cassetta utilizzando l'**ingresso B** dell'interfaccia LX385 e in questo caso, arrivati alla fine del 1° kapp, quando hanno provato a pigliare due volte di seguito CONTROL-4 come da noi indicato, si sono trovati automaticamente in panne, cioè è comparsa regolarmente la scritta BASIC sui display però il motorino del registratore, anziché avviarsi

come da noi specificato, è rimasto fermo.

Questo particolare non era stato indicato nel nostro articolo in quanto per noi era sottinteso che si dovesse utilizzare l'**ingresso A**, quindi ritenendolo ovvio ci siamo scordati di precisarlo.

Ripariamo comunque a tale dimenticanza dicendovi ora che PER CARICARE IL BASIC SUL MICROCOMPUTER ed anche in seguito per registrare i programmi scritti in BASIC, occorre utilizzare necessariamente il solo ingresso-uscita A di LX385.

3) Altri ancora, ritenendo che il loro computer fosse perfettamente efficiente in quanto tutti i programmi esadecimali che finora avevano provato avevano dato esito positivo, non si erano accorti che su una scheda, in particolare su quella di espansione della memoria LX386, c'erano due piste in corto oppure un piedino non staginato.

Caricando il BASIC il quale utilizza tutta l'area di memoria a disposizione, quando questo va a scrivere su tale scheda, non riuscendo più a rileggere, è ovvio che dia ERRORE.

4) In taluni casi il nastro ci è stato consegnato con un lato registrato ad un livello più basso rispetto all'altro per cui qualche volta, quando si arriva in un determinato punto, automaticamente la lettura si interrompe e sul video compare la scritta COME? perché il computer non riesce a leggerlo.

In questi frangenti per raggiungere lo scopo è sufficiente voltare il nastro dalla parte opposta e leggerlo sempre all'inizio (il BASIC è registrato su entrambi i lati) oppure alzare il volume.

5) Qualcuno non riesce a caricare il BASIC perché il suo registratore non ha le testine «ben allineate». Vale a dire che queste registrano e rileggono in una zona anormale del nastro per cui apparentemente tutto funziona alla perfezione, cioè si riescono a memorizzare e rileggere i programmi; se però in tale registratore collochiamo una cassetta con il linguaggio BASIC registrato esattamente nella posizione centrale del nastro come richiesto, il segnale che riusciremo ad ottenere con tale testina avrà sempre un livello inferiore al necessario.

Per risolvere questo problema occorre far allineare le testine oppure caricare il BASIC in memoria con un altro registratore, poi duplicarlo con il nostro su un nastro a parte in modo da poterlo leggere con la testina non allineata.

6) Abbiamo trovato 5 casi in cui risultava presente sulla scheda di espansione della memoria LX386 un cortocircuito **sotto lo zoccolo della RAM IC7**, fra il **piedino 18** di questa RAM e la pista superiore che gli passa vicino.

teva duplicare con il microcomputer utilizzando l'istruzione DUP, ha provato ad eseguire tale operazione tuttavia non essendo ancora stato spiegato esattamente come si deve utilizzare questa istruzione, è ovvio che abbia ottenuto un insuccesso totale.

Per porre rimedio a tutti questi inconvenienti ritorneremo sull'argomento, spiegandovi di nuovo come si carica il Basic in memoria e descrivendovi più dettagliatamente quelle operazioni e quelle istruzioni che nel numero precedente non sono state spiegate con sufficiente chiarezza.

COME SI CARICA IL BASIC IN MEMORIA

Precisiamo subito che per poter caricare il Basic in memoria è assolutamente indispensabile che sul BUS risultino inserite le seguenti schede:

LX382 = Scheda CPU

LX383 = Interfaccia tastiera esadecimale

LX384 = Tastiera esadecimale

LX385 = Interfaccia per registratori (utilizzare solo ingresso-uscita A)

LX386 = Espansione della memoria (sono sufficienti 8

IL linguaggio BASIC

Sull'ultimo numero della rivista, insieme all'interfaccia video e al monitor TV, vi abbiamo presentato un interprete Basic italiano-inglese da 5,5 K su cassetta che molti lettori hanno già acquistato e provato ad utilizzare con alterne fortune. In questo articolo vi forniremo tutti i consigli utili per caricare in memoria tale Basic e vi insegneremo inoltre a duplicarlo in modo da poter avere delle cassette di riserva nel caso in cui quella originale, per un'errata manovra, si cancelli.

Questo cortocircuito, finché i programmi sono brevi non crea nessun inconveniente: gli inconvenienti si manifestano solo quando noi carichiamo il Basic. Per esempio può capitare, dopo aver pigiato CONTROL-4 ed aver visto apparire BASIC sul display, che si interrompa la lettura da cassetta prima del previsto oppure che il registratore non si fermi quando il BASIC sul nastro è terminato, bensì arrivi a fondo nastro senza che sul video compaia la famosa scritta BASIC V.1.0.

Tale cortocircuito potrebbe essere presente anche sulla vostra scheda senza che voi ve ne siate accorti. Controllate quindi con un tester che queste due piste non siano in corto fra di loro, e nel caso lo siano provate ad infilare la punta di uno spillo sotto lo zoccolo per asportare il filettino di rame che crea il cortocircuito.

7) Qualcuno sapendo che la cassetta del BASIC si po-

kappa di RAM o al massimo 16 K)

LX387 = Tastiera alfanumerica

LX388 = Interfaccia video

LX380 = Alimentatore.

Se manca una sola di queste schede oppure se una di queste non funziona perché esiste un corto o un piedino di uno zoccolo che non è stagnato la lettura da cassetta si interrompe subito all'inizio o al massimo a metà strada.

È importante pilotare il motorino del registratore con il comando REMOTE di cui l'interfaccia cassette dispone diversamente tutte le operazioni, risulteranno più complicate ed arrivati alla fine del 1° kappa nella lettura del BASIC, il registratore non si fermerà come richiesto.

1°) Prendete la cassetta del Basic ed inseritela nel registratore ricordandovi di riavvolgerla completamente

in modo che il nastro parta dall'inizio.

Non ha importanza il lato su cui sistemerete la cassetta in quanto il Basic è **registrato su entrambe le parti**.

2°) Pigiare sulla tastiera esadecimale i due tasti CONTROL-6 (prima il tasto CONTROL poi tenendo questo schiacciato pigiate anche il 6) in moto da far comparire una L sul secondo display.

3°) Pigiare sul registratore il tasto ASCOLTO o PLAY.

4°) Pigiare sulla tastiera esadecimale il tasto A in modo da far comparire sui primi due display la scritta LA (nota: durante la lettura della cassetta la lettera L potrebbe trasformarsi in una S ma questo non crea nessun inconveniente pratico).

5°) Automaticamente il registratore si metterà in moto e dopo circa 10 secondi sui display della tastiera esadecimale vedremo comparire dei numeri che partendo da **0000** aumenteranno progressivamente fino ad arrivare a **03FF**.

Nota: durante questa fase sul video compariranno lettere, numeri e simboli strani del tutto casuali in quanto il video stesso non è ancora sotto il controllo del BASIC.

6°) Arrivati a **03FF** si ha una pausa di circa **5 secondi** dopodiché il registratore si ferma e a questo punto, come descritto al 10° capoverso, per proseguire occorrerà pigiare due volte di seguito CONTROL-4.

7°) Se il conteggio sui display anziché partire da **0000** parte per esempio da **1000** oppure il registratore si ferma prima di arrivare a **03FF** e sui display compare la **E** di «errore» seguita da un numero a 4 cifre significa che il nastro non si riesce a leggere.

In questi casi il difetto potrebbe essere dovuto al volume sul registratore tenuto troppo basso (controllando su un oscilloscopio l'uscita del registratore dovremmo rilevare un segnale con un'ampiezza di almeno **3-4 volt picco-picco**).

Un'altra possibilità, riscontrata in diverse riparazioni è infine quella che un filo all'interno dello spinotto jack si sia dissaldato oppure che sia presente un cortocircuito sempre all'interno dello spinotto dovuto alla calza metallica che non è stata stagnata correttamente, cioè uno o due di quei minuscoli fili della calza vanno a toccare il terminale centrale.

In pratica tutte queste cose sono da verificare. Se non riuscite a leggere il BASIC verificate quindi queste condizioni prima di effettuare un nuovo tentativo di lettura.

Importante: tutto il discorso risulta valido presupponendo sempre che l'interfaccia cassette sia già stata collaudata in precedenza con altri programmi e che abbia sempre funzionato alla perfezione.

8°) Ammesso per esempio che il difetto del mancato caricamento del Basic sia dovuto al fatto di tenere il livello del volume troppo basso, provate ad alzare questo volume e dopo aver riavvolto il nastro ripetete la prova seguendo passo passo le indicazioni fornite a partire dal punto 1 di questo paragrafo.

9°) Se tutto va bene vedrete sulla tastiera esadecimale i numeri partire da **0000** ed arrivare a **03FF** ed a questo punto, trascorsi 5 secondi, il registratore si fermerà.

10°) Quando il registratore si sarà fermato pigiate

con l'indice della mano sinistra il **tasto CONTROL** sulla tastiera esadecimale, poi con l'indice della mano destra pigiate **due volte** di seguito il **numero 4**. La prima volta sui display vi apparirà una «n» seguita da 7 trattini orizzontali proprio come se aveste pigiato il tasto RESET.

La seconda volta vi apparirà invece la scritta BASIC e sul monitor video verrà visualizzato un quadro con la scritta NUOVA ELETTRONICA, LETTURA BASIC ITALIANO-INGLESE DA CASSETTA ecc. ecc.

Contemporaneamente il registratore si metterà di nuovo in funzione (attenzione, non spengetelo quando si ferma la prima volta, bensì lasciate che sia il computer a pilotarlo tramite il comando REMOTE) e da questo punto in poi inizierà la lettura del BASIC vera e propria per la quale si richiedono in media 4-5 minuti.

Nota: non toccate le due tastiere durante la lettura del Basic diversamente la lettura stessa può interrompersi.

Se durante la lettura a un certo punto compare sul video la scritta COSA? preceduta da un numero e il registratore si ferma significa che si è verificato un errore di lettura ed il numero che compare davanti a COSA indica appunto il tipo di questo errore (vedi tabella riportata sul n. 70 a pag. 114).

Ovviamente anche in questo caso vale tutto ciò che abbiamo detto in precedenza a proposito del 1° kappa di programma cioè i motivi per cui si interrompe la lettura possono essere il **volume troppo scarso**, la testina del registratore spostata, un filo stagnato male o altre noie di questo genere.

11°) Ammesso che tutto vada per il giusto verso, trascorsi all'incirca 4 minuti il registratore si fermerà e sul video, al posto del quadro precedente, comparirà in alto a sinistra, in negativo, la scritta:

BASIC V. 1.0

per indicarci che il computer è pronto per ricevere le nostre istruzioni, cioè per essere «programmato» in linguaggio Basic.

Potremmo ancora aggiungere che in **un caso solo** ci è capitato un inconveniente perlomeno insolito, infatti arrivati alla fine della lettura del Basic da cassetta il registratore si è regolarmente fermato però sullo schermo, anziché comporre la scritta BASIC V. 1.0, è rimasto il quadro iniziale con scritto NUOVA ELETTRONICA - LETTURA BASIC DA CASSETTA.

Ebbene il motivo di tale inconveniente era dovuto ad un cortocircuito presente sull'interfaccia video LX388 in alto al centro fra le tre piste che si collegano ai terminali delle resistenze da 3.900 ohm.

Una volta asportato tale cortocircuito tutto ha funzionato alla perfezione.

In un altro caso si è verificato un inconveniente a prima vista attribuibile alla scheda di espansione LX386 ma che in realtà era dovuto alla scheda CPU: in pratica sembrava che il Basic venisse letto regolarmente tuttavia il registratore non si fermava alla fine del Basic stesso ed il nastro continuava a girare imperterrita fino ad arrivare al blocco meccanico.

Dopo aver controllato attentamente il circuito scheda per scheda ci siamo accorti che il difetto era localizzato sulla scheda CPU infatti il piedino 16 di IC1 non era sta-

gnato alla pista sottostante e questo impediva allo Z80 di indirizzare correttamente l'espansione della memoria.

Qualora vi capitasse un inconveniente di questo genere sarà quindi bene, dopo aver verificato che non esistono corti sulla scheda LX386, controllare con un tester che tutte le uscite di IC1 sulla scheda LX382 siano in collegamento elettrico con i corrispondenti terminali del connettore B che si innesta sul BUS, diversamente il Basic non può girare.

DUPLICARE il BASIC

Come vi abbiamo detto sul n. 73 il linguaggio Basic è coperto da diritti d'autore internazionali quindi ne è proibita la commercializzazione a meno di pagare una **quota fissa** per ogni unità venduta.

Il motivo per cui esistono questi diritti è piuttosto intuitivo infatti per creare un linguaggio come questo si richiedono mesi e mesi di studi approfonditi da parte di matematici e ingegneri per cui alla fine è logico che chi lo ha creato pretenda di ricavarne un frutto che lo ripaghi del tempo investito negli studi anche se poi chi lo utilizza lo perfeziona come abbiamo fatto noi per trasformarlo da inglese in bilingue.

In virtù di tali diritti ogni cassetta da noi fornita è soggetta ad un'imposta, anche se è un duplicato, per cui è inutile che ci chiediate di duplicarle a basso costo.

Un privato che già abbia acquistato un primo esemplare gravato del relativo copyright può invece duplicarla per usi personali, purché non ne faccia commercio.

Sapendo ciò e ritenendo assurdo che un lettore, nell'eventualità che il proprio nastro con il Basic si dovesse rompere o cancellare, fosse costretto per questi futili motivi a dover acquistare una nuova cassetta pagando nuovamente i diritti d'autore, abbiamo volutamente fatto inserire del nostro Basic un'istruzione (cioè il DUP) che permette appunto al lettore di duplicare la propria cassetta in modo tale da cautelarsi contro eventuali danneggiamenti.

Vi sarete già accorti che nella cassetta da noi fornita il linguaggio Basic è registrato su entrambi i lati per poter salvare il linguaggio stesso nell'eventualità che in seguito ad una manovra errata questo si cancelli su un lato, tuttavia ciò non deve farvi ritenere al «sicuro», quindi vi consigliamo, non appena sarete venuti in possesso della nostra cassetta, di effettuarne subito una o due duplicazioni.

La procedura necessaria per duplicare il Basic su un altro nastro è la seguente:

1°) prendete la cassetta con il Basic registrato e quella vergine e riavvolgetele completamente. Utilizzate cassette vergini per microcomputer, cioè con nastri di ottima qualità e senza coda iniziale (possiamo fornirvele a L. 1.500 cadauna).

2°) Collegate il registratore sull'ingresso-uscita A dell'interfaccia cassette.

3°) Inserite sul registratore la cassetta del Basic da noi fornita.

4°) Pigiare sulla tastiera esadecimale i due tasti **CONTROL-6** in modo da far comparire una L sul secondo display.

5°) Pigiare sul registratore il tasto **PLAY** o **ASCOLTO**.

6°) Pigiare sulla tastiera esadecimale il **tasto A**.

7°) Automaticamente il registratore si metterà in moto e dopo i soliti 10 secondi sui display si inizieranno a vedere dei numeri che da **0000** aumenteranno velocemente fino ad arrivare a **03FF**.

8°) Se durante questa fase la lettura si interrompe e compare il simbolo di errore sui display, dovrete riavvolgere il nastro ed iniziare una nuova lettura pigiando **CONTROL-6** e subito dopo il tasto **A**.

9°) Quando il registratore si ferma perché è arrivato alla fine del primo «kappa», cioè a **03FF**, pigiate su questo il tasto **STOP** in modo che non possa più ripartire e pigiate quindi sulla tastiera esadecimale il tasto **RESET**: sui display vi apparirà la solita «n» seguita da 7 trattini orizzontali.

10°) Estrae la cassetta del Basic dal registratore senza riavvolgerla (è importante lasciarla nella stessa identica posizione su cui si è fermata) ed al suo posto inserite la cassetta vergine completamente riavvolta.

11°) Pigiare sulla tastiera esadecimale i due tasti **CONTROL-5** per registrare (sul display comparirà una S).

12°) Pigiare sul registratore i due tasti **PLAY-REC**, cioè predisponete anch'esso per registrare.

13°) Pigiare sulla tastiera esadecimale il **tasto A**: automaticamente il nastro si metterà in movimento e dopo 10 secondi sui display compariranno i soliti numeri che da **0000** aumentano fino ad arrivare a **03FF**.

14°) Arrivati a **03FF** aspettate che il registratore si fermi poi pigiate su questo il stato di **STOP**.

15°) Togliete la cassetta appena registrata senza riavvolgerla (anche in questo caso è importante lasciarla nella stessa identica posizione in cui si è fermata) ed al suo posto inserite nuovamente quella del Basic forniti da Nuova Elettronica.

16°) Pigiare il tasto **RESET** sulla tastiera esadecimale in modo da far comparire sui display una «n» seguita da 7 trattini.

17°) Pigiare il tasto **PLAY** sul registratore e senza alcun indugio pigiate sulla tastiera esadecimale i due tasti **CONTROL-4** (tenete pigiato il tasto **CONTROL** e con l'altra mano pigiate il tasto 4).

Automaticamente vedrete comparire sul display la scritta **BASIC** mentre sul video vi apparirà il solito quadro con la scritta: **NUOVA ELETTRONICA-LETTURA BASIC DA CASSETTA**.

18°) Attendete 4-5 minuti in modo da consentire al computer di leggersi tutto il Basic e non appena vedrete che sul video compare la scritta **BASIC V 1.0** e che il registratore si ferma, pigiate sul registratore stesso il tasto di **STOP**.

19°) Estrae la cassetta con il basic dal registratore ed al suo posto inserite quella «ex-vergine» su cui avete già registrato il primo kappa (importante: non riavvolgete questa cassetta, bensì lasciatela sul punto in cui si era fermata in precedenza).

20°) Pigiare sulla tastiera alfanumerica i due tasti **BREAK-BREAK**.

21°) Pigiare sul registratore i due tasti PLAY-REC in modo da predisporlo per la registrazione.

22°) Scrivete sulla tastiera **alfanumerica** DUP quindi pigiate sempre su questa tastiera il tasto RETURN.

23°) Il registratore si metterà subito in moto per fermarsi dopo 3-4 minuti quando il Basic sarà completamente duplicato.

Nota: se avviene un errore durante questa fase occorre ripetere tutta la procedura dall'inizio.

24°) Una volta che il registratore si sarà fermato potrete pigiare il tasto di STOP sul registratore stesso e riavvolgere quindi la vostra cassetta per metterla in di sparte pronta all'uso in casi di emergenza.

IMPORTANTE: una volta terminata la fase di duplicazione è consigliabile provare a caricare il Basic in memoria con questa nuova cassetta per controllare se la registrazione è perfetta. Infatti potrebbe succedervi di dimenticare il comando di volume troppo basso o il nastro potrebbe avere dei «buchi», cioè la pista magnetica difettosa, e in tutti questi casi difficilmente riuscirete a rileggerlo.

PER MIGLIORARE LA SCHEDA DI MEMORIA LX386

Riteniamo interessante precisare che apportando alla scheda di memoria LX386 la modifica qui di seguito indicata si riescono ad eliminare fastidiosi inconvenienti che spesso abbiamo riscontrato su schede pervenute in riparazione.

Tali inconvenienti si manifestano in particolar modo quando non tutte le memorie inserite dispongono delle stesse caratteristiche, per esempio si sono utilizzate RAM di due marche diverse oppure anche della stessa marca ma con tempi di «accesso» diversi.

Queste differenze possono in qualche caso addirittura impedire la lettura del Basic infatti può capitare che arrivati ad un certo punto, il registratore si blocchi e sui display compaia la E di «errore».

Alcuni lettori già «smaliziati» a cui si era presentato questo inconveniente lo hanno eliminato invertendo di posizione le RAM sulla scheda tuttavia non essendo questo un metodo corretto di procedere in quanto non sempre da esiti positivi, abbiamo cercato di risolvere totalmente il problema apportando al circuito una modifica che in pratica ci mette al riparo da qualsiasi brutta sorpresa.

Come vedesi in fig. 1 per modificare questa scheda occorre tagliare superiormente nel punto indicato la pista che collega il piedino 5 dell'integrato SN74LS04 al piedino 6 dell'integrato SN74LS138.

Sulla parte sottostante occorre invece collegare 4 fili più una resistenza nel modo seguente:

1° filo = collega il piedino 3 dell'integrato SN74LS08 al piedino 3 dell'integrato SN74LS04

2° filo = collega il piedino 13 dell'integrato SN74LS08 al piedino 4 dell'integrato SN74LS04

3° filo = collega il piedino 11 dell'integrato SN74LS08 al piedino 5 dell'integrato SN74LS04

4° filo = collega il piedino 12 dell'integrato SN74LS08

al piedino 6 dell'integrato SN74LS138

La resistenza da 10.000 ohm da noi indicata non è strettamente necessaria comunque per chi volesse utilizzarla questa va applicata fra il piedino 5 dell'integrato SN74LS04 ed il positivo dei + 5 volt.

Come già detto, anche se molte schede LX386 funzionano perfettamente senza apportare questa modifica, **noi consigliamo** in ogni caso di effettuarla per evitare che un domani, sostituendo per un qualsiasi motivo una RAM, ci si ritrovi in panne.

Precisiamo che da oggi le schede LX386 che noi forniremo risulteranno già modificate pertanto prima di effettuare questi collegamenti esterni controllate che gli stessi non siano già presenti sul circuito stampato.

Come avrete notato ogniqualevolta riparando qualche microcomputer riscontriamo qualche imprevista anomalia che con una semplice modifica si può eliminare, non esitiamo a pubblicarla in modo tale che chiunque possa in ogni momento perfezionare il proprio montaggio.

PER UTILIZZARE IL BASIC

Alla fine della lettura del Basic da cassetta noterete che in alto sulla sinistra del video vi appare la scritta BASIC V 1.0.

A questo punto il vostro computer è già disponibile per essere programmato in ITALIANO.

Se vi interessa programmare in **inglese** dovrete scrivere sulla tastiera alfanumerica la parola:

ENG

poi pigiare il tasto RETURN.

Automaticamente la V rovesciata che compariva sotto la scritta BASIC in «negativo» verrà visualizzata in «positivo» per indicarvi che da questo punto in poi si può programmare **solo** in inglese.

Se una volta scelto l'inglese volete ritornare all'**italiano**, non dovrete fare altro che scrivere sulla tastiera:

ITA

poi pigiare il tasto RETURN.

Così facendo la V rovesciata sotto la scritta BASIC vi verrà visualizzata nuovamente in «negativo» per indicarvi che il computer comprende solo la nostra lingua nazionale.

Precisiamo che passando dall'italiano all'inglese e viceversa **automaticamente si cancella** tutto il programma scritto in precedenza quindi se vi interessa «salvare» tale programma ricordatevi di registrarlo su nastro prima di scrivere ITA o ENG.

In pratica quando noi scriviamo ITA non facciamo altro che chiamare un interprete «italiano per microcomputer» che non conosce nulla di inglese, mentre quando scriviamo ENG chiamiamo un interprete «inglese» per microcomputer che non sa una parola di italiano, pertanto una volta scelto tale interprete occorrerà sempre parlargli nella stessa lingua.

A nostro parere sarebbe consigliabile, una volta fatta esperienza a programmare in italiano, provare ad eseguire lo stesso programma in inglese per acquisire così una esperienza bilingue ed avere un domani la possibilità di lavorare con qualsiasi computer commerciale i

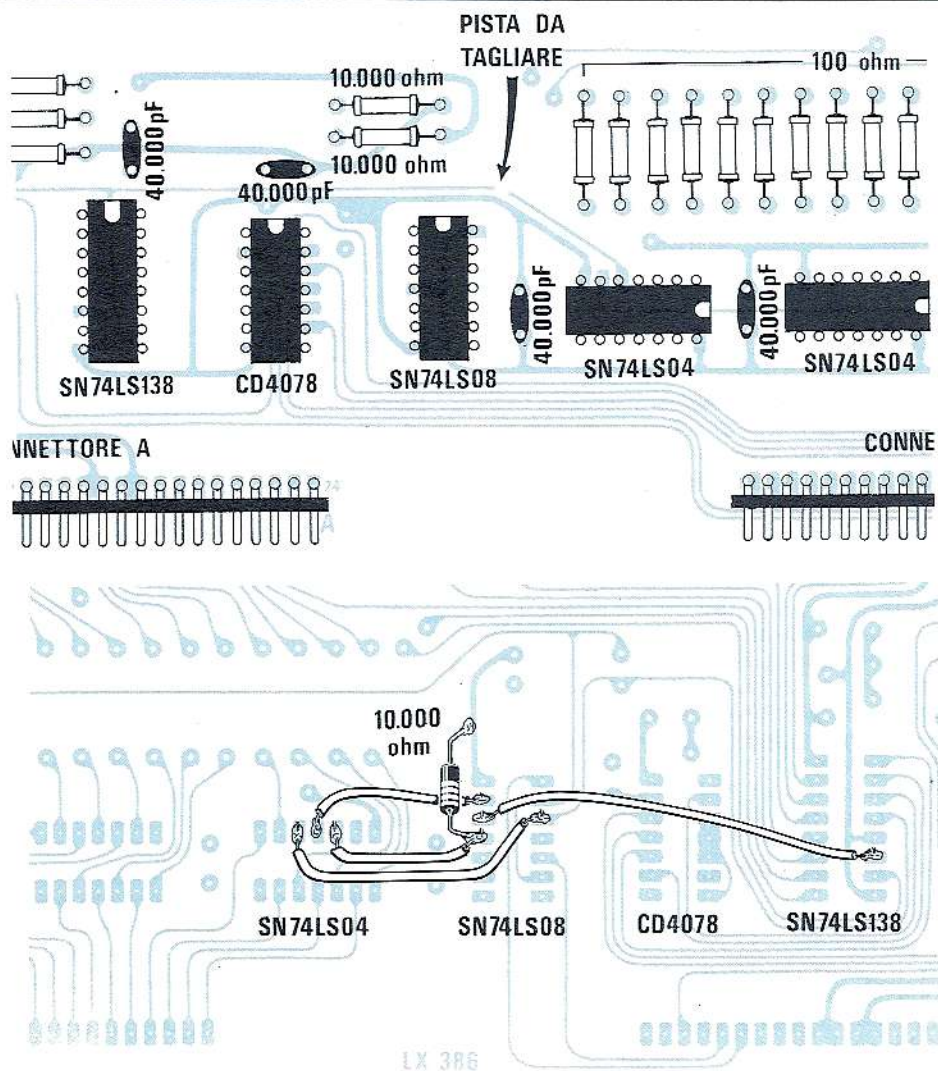


Fig. 1 Apportando questa modifica sulla scheda di memoria LX386, cioè tagliando sulla parte superiore del circuito stampato (vedi figura in alto) la pista indicata e collegando nella parte sottostante (figura sotto) i piedini degli integrati con dei fili come vedesi da disegno, si elimineranno tutti quegli inconvenienti che si manifestano quando le Ram non hanno identiche caratteristiche. Sulle schede LX386 che consegneremo da questo momento in poi tale modifica è già stata da noi effettuata.

quali parlano solo «inglese».

Quando scriverete un qualsiasi programma ricordatevi sempre di anteporre ad ogni riga di istruzioni il relativo **numero d'ordine**, cioè quel numero, esempio

10
20
30

che serve al computer per comprendere in quale sequenza le istruzioni stesse debbono essere eseguite (Nota: spiegheremo più avanti come si procede per questo numero d'ordine).

Ricordatevi inoltre, al termine di ogni riga, di pigiare il tasto RETURN per caricare la riga stessa in memoria.

Se noi non pigiamo questo tasto il computer non può sapere che la nostra riga è terminata e di sua iniziativa non si permetterà mai di trasferire questa riga in memoria bensì continuerà ad inserire in coda a questa riga i caratteri che noi pigiamo sulla tastiera.

Pigiando il tasto RETURN si vedrà invece il «cursore» ritornare a capo per indicarci che la nostra riga di programma è stata regolarmente memorizzata e sempre sulla sinistra comparirà la solita V rovesciata per indi-

carci che il computer è in attesa di una nuova istruzione.

Se questo segno per un qualsiasi motivo non appare, una condizione questa che si verifica sempre dopo che si è effettuato un «listing» del programma sul video, prima di scrivere una nuova istruzione ricordatevi di pigiare il tasto RETURN per farlo apparire diversamente la vostra istruzione, anche se scritta regolarmente sul video, non potrà essere memorizzata.

Nota: in taluni casi, invece di pigiare il tasto RETURN per far comparire la V rovesciata sulla sinistra del video, si possono anche pigiare i due tasti BREAK-BREAK però in questo modo, pur rimanendo memorizzate sulla RAM, spariranno dal video tutte le istruzioni che avevamo scritto in precedenza.

Comunque piuttosto che dilungarci a spendere tante parole senza mostrarvi nulla di concreto, pensiamo che sia molto più utile fornirvi subito un esempio pratico di programma in modo che discutendolo si possano fare con maggiore chiarezza tutte le precisazioni che molti di voi ancora non conoscono.

Tale programma serve in pratica per far apparire sul monitor video dei rettangoli concentrici, cioè uno interno all'altro, i quali ci saranno utilissimi per controllare la linearità del monitor ed eliminare, agendo sui magnetini posti sulla corona del giogo di deflessione, tutte le deformazioni sui bordi.

Il listing di questo programma è il seguente:

```
10 NOTA = PROGRAMMA TARATURA MONITOR
20 SPSC: SPEGNI (1,2)
30 A = 0: B = 63: C = 47
40 PER X = A FINO B: ACCENDI (X, A): INCREMENTA X
50 PER Y = A + 1 FINO C: ACCENDI (B, Y): INCREMENTA Y
60 PER X = B - 1 FINO A PASSO - 1: ACCENDI (X, C): INCREMENTA X
70 PER Y = C - 1 FINO A + 1 PASSO - 1: ACCENDI (A, Y): INCREMENTA Y
80 A = A + 2: B = B - 2: C = C - 2
90 SE A < 24 VA A 40
100 VA A 100
```

La prima cosa da fare per poter scrivere questo programma in memoria ed avere la certezza che possa girare correttamente è cancellare tutto ciò che si era scritto in precedenza procedendo nel modo seguente:

1) pigiate il tasto di STOP (cioè il primo in alto sulla destra della tastiera sopra RETURN) e subito dopo pigiate i due tasti BREAK-BREAK in modo da cancellare lo schermo e far apparire in alto a sinistra la scritta BASIC V 1.0 con sotto la V sdraiata in negativo.

2) Scrivete sulla tastiera alfanumerica CANCELLA (se stavate programmando in italiano) oppure ITA (se stavate programmando in inglese) poi pigiate il tasto RETURN.

Così facendo la memoria del computer si cancellerà totalmente (esclusa la parte relativa al linguaggio Basic) ed il computer stesso si appresterà a ricevere un nuovo programma facendo comparire sul video la scritta BASIC V 1.0 con sotto la solita V rovesciata in negativo.

Se non volete scrivere CANCELLA per esteso potrete

scrivere più semplicemente CA, infatti sul n. 73 vi abbiamo già anticipato che il nostro interprete è così intelligente che non ha bisogno che gli venga detta la frase per intero, bensì già dalle prime due o tre lettere è in grado di capire ciò che noi vogliamo.

In ogni caso, usando l'abbreviazione, è sempre necessario farla seguire da un «punto» diversamente l'interprete non capisce che si tratta di un'abbreviazione e non disponendo nel suo «vocabolario» di una frase come quella da voi scritta, vi chiederà COSA?

3) Una volta cancellata la memoria potremo iniziare a scrivere le istruzioni del nostro programma ricordandoci di porre davanti a ciascuna di esse il relativo numero d'ordine, cioè 10-20-30 ecc.

Come noterete passando da una riga di programma a quella successiva abbiamo sempre lasciato un salto di 10 spazi cioè 10-20-30-40 ecc. in modo tale da poter eventualmente inserire in un secondo tempo altre istruzioni nel mezzo nel caso in cui si presenti la necessità.

Per esempio, una volta arrivati alla fine del vostro programma, potreste accorgervi di aver saltato l'istruzione che andava scritta alla riga 40, cioè:

40 PER X = A FINO B: ACCENDI (X,C): INCREMENTA X

e di aver scritto in sua vece l'istruzione che andava scritta alla riga 50 numerandola 40, cioè:

40 PER Y = A + 1 FINO C: ACCENDI (B,Y): INCREMENTA Y

In tal caso, senza dover riscrivere totalmente il programma potrete scrivere la riga mancante dandogli un numero d'ordine che risulti intercalato fra 30 e 40, per esempio 35, cioè:

35 PER X=A FINO B:ACCENDI (X,C):INCREMENTA X automaticamente questa riga verrà inserita dal micro-computer fra la 30 e la 40, infatti facendo eseguire una LISTA del vostro programma sul video troverete le linee 30-35-40 una di seguito all'altra anche se la 35 è stata scritta per ultima.

Questo è il motivo valido per cui si consiglia di lasciare sempre degli spazi vuoti tra un numero d'ordine ed il successivo, infatti se in un programma fosse necessario aggiungere altre istruzioni supplementari voi potrete sempre farlo assegnando alla nuova istruzione un numero che si interponga fra quelli delle due istruzioni entro cui la nuova istruzione deve risultare intercalata.

Ad esempio se voleste aggiungere altre istruzioni prima della riga 35 potreste assegnare a queste i numeri 31-32-33-34; se invece voleste aggiungerne una fra la 35 e la 40 potreste assegnare a questa istruzione il numero 36-37-38-39.

4) Come già detto in precedenza, invece di scrivere le istruzioni per esteso come abbiamo fatto noi per maggiore comprensione, potreste utilizzare per ognuna di esse la relativa abbreviazione, per esempio alla riga 20, anziché scrivere:

20 SPSC:SPEGNI (1,2)

potreste scrivere:

20 SPSC:SP. (1,2)

ricordandovi però sempre di far seguire all'abbreviazione il «punto».

5) Alla fine di ogni riga di programma **ricordatevi sempre di pigiare** il tasto RETURN diversamente, come già detto in precedenza, la riga stessa non potrà essere caricata in memoria dal computer.

6) È molto importante rispettare la «punteggiatura» infatti se noi alla riga 20 del nostro programma scrivessimo:
20 SPSC;SPEGNI (1,2)

cioè sostituissimo i «due punti» dopo SPSC con un «punto e virgola» il computer, qualora facessimo eseguire il programma, non capirebbe che quelle sono due istruzioni diverse scritte una dopo l'altra che quelle sono due istruzioni diverse scritte una dopo l'altra e non riuscendo ad interpretare la nostra frase, farebbe apparire sul video la scritta:

20 SPSC;SP. (1,2)

COME?

Dove il ? posto subito dopo il «punto e virgola» sta ad indicare che questo; non è corretto.

7) Se ci si accorge di aver commesso un errore di questo genere nella scrittura di un programma porvi rimedio è molto semplice: basta infatti **riscrivere la stessa frase** in fondo al programma in modo corretto **con lo stesso numero d'ordine**, cioè:

20 SPSC;SP. (1,2)

e pigiare quindi il tasto RETURN perché la vecchia istruzione sbagliata venga automaticamente rimpiazzata dalla nuova corretta.

Infatti se a due istruzioni di uno stesso programma **noi assegnamo lo stesso numero d'ordine**, il computer accetta sempre come «valida» l'**ultima che abbiamo scritto** con lo stesso numero in ordine di tempo cancellando automaticamente dalla memoria la precedente.

In pratica se voi tentaste di correggere un'istruzione in questo modo vedrete che momentaneamente rimane visualizzata sul video anche la precedente, tuttavia se a questo punto farete eseguire la LISTA nel modo che vi spiegheremo più avanti, vedrete che la vecchia istruzione scompare perché sostituita dalla nuova.

8) Se mentre scrivete un'istruzione vi accorgete di aver commesso un errore prima ancora di aver pigiato il tasto RETURN, per esempio se alla riga 100 vi accorgete di aver scritto:

100 VAA 100

invece che:

100 VA A 100

(Nota: l'istruzione VA A deve sempre essere scritta lasciando uno spazio libero, cioè battendo la barra di interlinea, fra le due A), pigiando il tasto DEL vedrete il cursore, cioè il punto luminoso che indica la posizione in cui si scrive, spostarsi verso sinistra cancellando quanto già scritto e quando si saranno cancellati tutti i caratteri che avrete scritto dopo VA, li potrete riscrivere in modo corretto.

Arrivati a fine riga dovrete poi come al solito pigiare il tasto RETURN per trasferire la riga di programma in memoria.

9) Una volta scritto tutto il vostro programma potrebbe interessarvi per esempio conoscere quanta memoria avete ancora libera nel computer (particolare questo

molto utile soprattutto quando scriverete dei programmi gestionali che risultano sempre molto lunghi).

Per ottenere ciò la procedura da seguire è molto semplice infatti vi basterà scrivere:

SCRIVI MEM

o più semplicemente abbreviato:

SC.MEM

e pigiare il tasto RETURN per vedere automaticamente comparire sul video un numero, per esempio 2354.

Tale numero indica appunto i byte (cioè le locazioni di memoria RAM) ancora disponibili per i dati all'interno del microcomputer.

10) Se disponete della stampante potrebbe interessarvi pure di eseguire una lista del programma da voi scritto per catalogarla ed in tal caso, una volta accesa la stampante stessa, dovrete scrivere sulla tastiera alfanumerica:

SLISTA o più semplicemente **SL**.

Così facendo la stampante inizierà subito a scrivere il listato del vostro programma.

11) Listato il programma, se non esistono errori, potrete accingervi a farlo eseguire al computer scrivendo sulla tastiera:

ESEGUI o più semplicemente **ES**.

e pigiando quindi il tasto RETURN.

Automaticamente lo schermo del video si spegnerà dopodiché la mano di un invisibile «pittore» inizierà a disegnare sullo schermo stesso un rettangolo grande, poi un rettangolo leggermente più piccolo posto all'interno del primo, poi ancora un rettangolo più piccolo e così via fino ad arrivare al centro dello schermo.

Tutti questi rettangoli ci serviranno, come già anticipato, per poter tarare esattamente il nostro monitor video.

Una volta disegnati tutti i rettangoli il programma si ferma ed a questo punto, per poter riprendere il controllo dalla tastiera occorre pigiare il pulsante di STOP e pigiare quindi subito dopo i due tasti BREAK-BREAK i quali ci faranno comparire nell'angolo in alto a sinistra del video la solita scritta

BASIC V 1.0.

con sotto una V sdraiata in negativo.

Se per esempio dopo aver tracciato sul video la prima riga orizzontale superiore il programma si interrompe e sul video compare la scritta:

50 PERA? Y=C+1 FINO D:ACCENDI (B, Y):INCREMENTA Y

COME?

significa che alla riga 50 ci è sfuggito un errore (nel nostro esempio avevamo scritto PERA invece di PER).

In tal caso noi dovremo **riscrivere** questa riga con il relativo numero d'ordine (cioè 50) in modo corretto, poi pigiare il tasto RETURN per trasferirla in memoria e nuovamente far eseguire il programma con l'istruzione ESEGUI.

ALCUNE SPIEGAZIONI SUL PROGRAMMA

Il programma che vi abbiamo appena indicato non è altro che una semplicissima applicazione del linguaggio

Basic tuttavia per i meno esperti sarà sempre una cosa molto complicata da comprendere ed è proprio per questo che ci sentiamo in dovere di fornire qualche spiegazione in proposito.

riga 10 = l'istruzione **NOTA** (corrispondente all'inglese **REM**) viene ignorata dal computer nell'esecuzione di un programma: essa serve solo al programmatore per fornire un nome al programma stesso in modo da potersi in seguito ricordare a cosa serve questo programma.

riga 20 = con questa riga in cui sono riportate due istruzioni separate fra di loro dai «due punti» inizia il programma vero e proprio.

La prima istruzione, cioè **SPSC**, ci permette di spegnere lo schermo del video facendo comparire il cursore luminoso in alto a sinistra; questo cursore però a noi non interessa che rimanga acceso pertanto abbiamo inserito la seconda istruzione, cioè **SPEGNI** (1,2), la quale permette appunto di spegnere questo puntatore in modo da avere lo schermo totalmente buio proprio come una lavagna appena cancellata (il puntatore, dopo un **SPSC**, si trova sempre posizionato sul quadretto 1 della riga 2).

riga 30 = una volta spento lo schermo dobbiamo iniziare a disegnare i nostri rettangoli e poiché il computer non può sapere quali sono le righe che deve accendere e quali invece lasciare spente dovremo essere noi a indicargliele.

A tale proposito dobbiamo innanzitutto ricordare che nel funzionamento in «semigrafico», quello cioè che noi stiamo per utilizzare, lo schermo del video deve essere considerato come una specie di «matrice» costituita da 48 righe orizzontali (i numeri vanno da 0 a 47) e da 64 colonne verticali (i numeri vanno da 0 a 63).

Dobbiamo inoltre ricordare che, anche se alla fine sullo schermo compariranno in totale 12 rettangoli, una volta disegnato il primo di dimensioni maggiori, per disegnare gli altri di dimensioni più ridotte, rimanendo rispettate le proporzioni fra i due lati, si possono utilizzare sempre le stesse istruzioni, modificando semplicemente, di volta in volta, il punto da cui il computer deve iniziare a tracciare il rettangolo nonché la lunghezza dei due lati.

Per far questo ci sono sufficienti 3 variabili e precisamente la variabile **A**, la variabile **B** e la variabile **C**.

Alla riga 30 di programma a queste tre variabili vengono assegnati i seguenti valori:

A = 0 (dove 0 è il numero della prima riga e della prima colonna disponibili sullo schermo nel funzionamento in semigrafico)

B = 63 (dove 63 è il numero dell'ultima colonna disponibile)

C = 47 (dove 47 è il numero dell'ultima riga disponibile)

riga 40 = queste tre istruzioni servono per tracciare il primo lato in alto del rettangolo, partendo dal vertice sinistro (coordinate 0,0) ed arrivando al vertice destro (coordinate 63,0).

In pratica il vero significato di questa riga di programma è il seguente:

«Partendo con la variabile **X** uguale ad **A**, cioè uguale a 0, ed aumentando ogni volta di 1 questa variabile fino

ad arrivare a **B**, cioè a 63, accendi il punto **X** della riga **A**, cioè il punto **X** della riga 0».

In altre parole il computer viene costretto ad accendere un dopo l'altro tutti i punti della riga 0, partendo dall'angolo in alto a sinistra ed arrivando all'angolo in alto a destra.

riga 50 = questa riga di programma è molto simile alla precedente in quanto serve essa pure per tracciare un lato del rettangolo, anche se questa volta si tratta di un lato verticale.

In pratica il significato di queste istruzioni è il seguente:

«partendo con **Y** = **A** + 1, cioè uguale a 1 (infatti **A** = 0), ed aumentando progressivamente di 1 il suo valore fino ad arrivare a **C**, cioè a 47, accendi il punto **B** (cioè 63) della riga **Y**».

In altre parole il computer deve accendere il punto 63 di tutte le righe a partire dalla 1 fino alla 47 (il punto 63 della riga 0 era già stato acceso in precedenza quindi è inutile riaccenderlo) e questo lo costringe a tracciare una riga verticale sullo schermo tutta sull'estrema destra del medesimo, a partire dall'alto verso il basso.

riga 60 = questa istruzione si differenzia dalle due precedenti per il fatto che dopo il **PER ... FINO** è specificato anche il **PASSO**, cioè la quantità di cui il computer deve ogni volta «aumentare» la variabile **X** nel corso del loop (in precedenza questo **PASSO** non era stato specificato in quanto essendo l'incremento della variabile **X** o **Y** uguale a 1, il computer può farlo automaticamente).

Il significato di questa riga di programma è più o meno questo:

«Ponendo inizialmente **X** = **B** - 1, cioè a 62, e sottraendo ogni volta 1 fino ad arrivare ad **A**, cioè a 0, accendi il punto **X** della riga **C**, cioè della riga 47».

In pratica il computer, partendo dall'angolo in basso a destra, accenderà tutti i punti in orizzontale fino ad arrivare all'angolo in basso a sinistra, cioè disegnerà il lato in basso del rettangolo.

riga 70 = per completare il rettangolo ci manca ormai solo il lato verticale sinistro e per far questo noi gli diciamo:

«Ponendo inizialmente **Y** = **C** - 1, cioè a 46, e detraendo progressivamente 1 fino ad arrivare ad **A** + 1, cioè a 1, accendi il punto **A**, cioè il punto 0, della riga **Y**».

Il computer quindi accenderà per primo il punto 0 della riga 46, poi il punto 0 della riga 45, il punto 0 della riga 44 e così di seguito fino ad arrivare al punto 0 della riga 1 (il punto 0 della riga 0 è inutile riaccenderlo in quanto era già stato acceso in partenza).

riga 80 = una volta concluso il primo rettangolo, per poterne tracciare un secondo al suo interno, noi dovremmo fornire al computer tante istruzioni quante ne sono state necessarie per tracciare il primo, tuttavia seguendo questo metodo, ed essendo in totale 12 i rettangoli che debbono essere disegnati, ci ritroveremmo alla fine con qualcosa come 50 - 60 istruzioni.

Al contrario, con il metodo delle «variabili» da noi prescelto, sono sufficienti in tutto 10 righe di programma per raggiungere lo stesso fine.

Il «trucco» consiste nel ritoccare opportunamente i valori di queste variabili prima di far ripetere al compu-

ter le stesse istruzioni che abbiamo visto in precedenza alle righe 40-50-60-70.

In pratica in questa riga di programma noi diciamo al computer di porre:

$A = A + 2$ (cioè la variabile A uguale al valore che aveva in precedenza più 2, quindi $0 + 2 = 2$)

$B = B - 2$ (la variabile B uguale al valore che aveva in precedenza meno 2, quindi $63 - 2 = 61$)

$C = C - 2$ (la variabile C uguale al valore che aveva in precedenza meno 2, cioè $47 - 2 = 45$).

riga 90 = a questo punto noi diciamo al computer di controllare se il valore della variabile A è minore di 24 oppure no: se è minore di 24 lo mandiamo alla riga 40 a disegnare un nuovo rettangolo con le nuove coordinate da noi fornite; se invece A non è minore di 24 (cioè siamo già arrivati al centro dello schermo) il computer deve proseguire con la riga 100 in cui, come vedremo, vi è un'istruzione di «salto» che all'atto pratico impedisce che vengano tracciati altri rettangoli.

Il motivo di tutto questo è abbastanza ovvio infatti rimpicciolendo ogni volta il rettangolo si arriverà certamente ad un punto oltre il quale non è più possibile andare e poiché il computer da solo non saprebbe mai accorgersi di essere arrivato a questo punto e continuerebbe imperterrita a calcolarsi nuovi valori per A, B, C e a tentare di disegnare altri rettangoli, dobbiamo essere noi a dirgli «basta» al momento giusto.

riga 100 = scrivere 100 VA A 100 sembrerebbe un'istruzione assurda infatti siamo già alla riga 100, perché dunque mandiamo il computer ancora alla riga 100?

Non sarebbe meglio metterci un'istruzione di FINE?

Il motivo è molto semplice infatti a noi, per poter tarare il monitor, occorre che la figura disegnata rimanga costantemente sullo schermo.

Se a questo punto ponessimo un'istruzione FINE il computer farebbe apparire la scritta BASIC V 1.0 in alto a sinistra e questo ci rovinerebbe i rettangoli.

Con l'istruzione 100 VA A 100 esso continuerà invece ad eseguire questo salto (che in pratica è un salto sul posto) all'infinito ed impegnato com'è a «saltare» non si preoccuperà di scrivere BASIC sul video.

Una volta terminato di tarare il video, per riprendere il comando dalla tastiera e poter così scrivere altri programmi, dovremo quindi pigiare il pulsante di STOP, cioè dire al computer «smettiti di fare capriole su te stesso» e subito dopo dovremo pigiare i due tasti BREAK-BREAK.

I DUE PUNTI (:)

Vi abbiamo già precisato sulla rivista n. 73 che i «due punti» servono per poter inserire due o più istruzioni sulla stessa riga di programma in modo da rendere più veloce la scrittura del programma stesso.

Per esempio se abbiamo 3 istruzioni che occupano poco spazio e debbono necessariamente essere eseguite una di seguito all'altra in quanto «concatenate» fra di loro, è inutile che le scriviamo su tre righe successive assegnando a ciascuna un proprio numero d'ordine: meglio scriverle tutte sulla stessa riga, separandole con

i «due punti» perché in questo modo non solo agevoleremo la scrittura del programma stesso ma ci sarà anche più facile comprendere le funzioni delle varie «righe» quando andremo a rileggerci in un secondo tempo tale programma.

Nel nostro semplice «programma di taratura monitor» abbiamo diverse righe che contengono più di una istruzione.

La prima di queste è la riga 20 in cui troviamo scritto: 20 SPSC:SPEGNI (1,2)

In pratica potremmo anche modificarla come segue: 20 SPSC

25 SPEGNI (1,2)

tuttavia ciò non è conveniente per due motivi ben distinti.

1) Si fa prima a scrivere tutto su una stessa riga che non andare a capo e scriverne una seconda.

2) Le due istruzioni sono concatenate fra di loro nel senso che noi quando facciamo spegnere lo schermo con SPSC vogliamo che si spenga pure il cursore.

Scrivendole su due righe separate potrebbe accadere che nel mezzo venissero aggiunte in seguito altre istruzioni, per esempio alla riga 21-22-23-24 ed a questo punto, cercando di interpretare il programma, potrebbe anche sfuggirci il particolare che SPSC e SPEGNI (1,2) sono due operazioni l'una conseguente all'altra, quindi potrebbe risultarci difficile capire il programma stesso.

Un altro esempio lo abbiamo alla riga 80 dove troviamo scritto:

80 $A = A + 2; B = B - 2; C = C - 2$

Tale riga potrebbe anche essere scritta nel modo seguente:

80 $A = A + 2$

82 $B = B - 2$

84 $C = C - 2$

tuttavia anche in questo caso valgono le motivazioni che abbiamo appena esposto cioè queste tre variabili A,B,C debbono in ogni caso essere modificate tutte insieme quindi scrivendo le tre istruzioni sulla stessa riga, oltre a risparmiare tempo, si può capire anche più facilmente il motivo per cui tale riga di programma è stata adottata.

Un discorso analogo vale anche per le righe 40-50-60-70 cioè anche queste contengono ognuna 3 istruzioni che potrebbero benissimo essere scritte su 3 righe successive, tuttavia trattandosi di istruzioni «concatenate» fra di loro, o meglio finalizzate allo stesso scopo, è più comprensibile e comodo scriverle di seguito separandole fra di loro con i due punti.

IL PUNTO E VIRGOLA (;)

Anche se lo abbiamo già riportato sulla rivista n. 73 ri-peteremo che il «punto e virgola», oltre a far parte integrante di certe istruzioni tipo RICHIEDI o SCRIVI TAB, può essere utilizzato insieme all'istruzione SCRIVI per poter scrivere più parole o più numeri sulla stessa riga, sia sul video che sulla stampante, senza essere costretti ogni volta ad andare a capo.

Ad esempio se noi scriviamo il seguente programma:

```
10 SCRIVI «TRANSISTOR»
20 SCRIVI «INTEGRATO»
30 SCRIVI «DISPLAY»
40 FINE
```

facendolo eseguire con l'istruzione ESEGUI, sul video ci appariranno le scritte:

```
TRANSISTOR
INTEGRATO
DISPLAY
```

una sotto l'altra in quanto il «cursore» del video o della stampante, ogni volta che gli si dice «scrivi», ritorna automaticamente a capo.

Se noi invece facciamo seguire alle prime due istruzioni il «punto e virgola», cioè:

```
10 SCRIVI «TRANSISTOR»;
20 SCRIVI «INTEGRATO»;
30 SCRIVI «DISPLAY»
40 FINE
```

le scritte sul video o sulla stampante ci appariranno nel modo seguente:

```
TRANSISTORINTEGRATODISPLAY
```

in quanto tale «punto e virgola» impedisce al cursore di ritornare a capo.

Se poi ci interessa che le tre scritte risultino distanziate fra di loro di uno o due spazi, dovremo ricordarci, inserendo il programma in memoria, di battere alla fine di ognuna di esse, una o due volte la barra di interlinea, cioè:

```
10 SCRIVI «TRANSISTOR »;
20 SCRIVI «INTEGRATO »;
30 SCRIVI «DISPLAY »
40 FINE
```

ed in tal caso, facendo eseguire il programma, ci apparirà:

```
TRANSISTOR INTEGRATO DISPLAY
```

Nota: la barra deve essere battuta prima delle virgolette di chiusura.

Precisiamo che lo stesso risultato si potrebbe ottenere molto più semplicemente utilizzando una sola istruzione, cioè:

```
10 SCRIVI «TRANSISTOR INTEGRATO DISPLAY»
20 FINE
```

tuttavia in molti casi, soprattutto quando si utilizzano le variabili, ciò non è possibile ed è necessario ricorrere al «punto e virgola».

Lo stesso discorso vale ovviamente anche per i numeri o le variabili numeriche infatti scrivendo:

```
10 DATI 3,67
20 LEGGI X,Y
30 SCRIVI X:SCRIVI Y
40 FINE
```

e facendo eseguire tale programma, il computer ci farà apparire sul video i numeri

3

67

uno sotto l'altro.

Impiegando invece l'istruzione:

```
30 SCRIVI X;Y
```

sul video ci appariranno i numeri:

```
3 67 uno di seguito all'altro
```

L'ISTRUZIONE SCRIVI TAB

Chiunque abbia cercato di utilizzare l'istruzione SCRIVI TAB o SSCRIVI TAB seguendo le indicazioni da noi fornite sul n. 73 si sarà trovato immancabilmente in difficoltà per due motivi ben precisi:

1) La sintassi esatta dell'istruzione SCRIVI TAB non è quella da noi riportata cioè:

SCRIVI TAB n, «xx»

bensi la seguente:

SCRIVI TAB (n); «xx»

Come vedete la differenza sostanziale tra le due scritte è che la variabile n corrispondente alla «spaziatura» che si vuole ottenere deve essere racchiusa tra parentesi e seguita subito dopo da un «punto e virgola».

Esempio:

```
10 SCRIVI TAB (15); «GIACENZA»
20 SCRIVI TAB (17); 365
30 FINE
```

Facendo eseguire questo programma sullo schermo del video, distante 15 spazi dal bordo sinistro, ci verrà visualizzata la scritta GIACENZA e sotto ad essa il numero 365.

Ricordiamo che dopo il «punto e virgola» si può inserire indifferentemente una scritta alfanumerica (purché racchiusa tra virgolette), una variabile alfanumerica, un numero oppure una variabile numerica.

2) Per quanto riguarda invece l'istruzione SSCRIVI TAB il motivo per cui si sono incontrate delle difficoltà è ben diverso infatti tale istruzione, contrariamente a quanto scritto sulla rivista n. 73, nel Basic da 5,5 K non è ancora presente quindi non è possibile utilizzarla.

Per chi avesse necessità nei propri programmi di una simile istruzione possiamo comunque indicarvi una soluzione di «ripiego» in grado di fornire gli stessi risultati pratici.

Tale soluzione consiste nell'inserire nel programma una subroutine siffatta:

```
1000 PER X = 1 FINO T:SSCRIVI" ";:INCREMENTA X:RITORNA
```

(ricordatevi di battere la barra di interlinea fra le due virgolette)

e nel chiamare quindi questa subroutine con l'istruzione CHIAMA ogniquale è necessario nella stampa lasciare un certo numero di spazi vuoti, ricordandosi preventivamente di assegnare alla variabile T il valore numerico corrispondete agli spazi vuoti desiderati.

Facciamo un esempio pratico:

```
10 T = 15
20 CHIAMA 1000
30 SSCRIVI «GIACENZA»
40 FINE
1000 PER X = 1 FINO T:SSCRIVI" ";:INCREMENTA X:RITORNA
```

Tutto questo programma fa in pratica le veci di un'unica istruzione:

```
SSCRIVI TAB(15); «GIACENZA»
```

la quale, come già detto, non è purtroppo ancora disponibile nel BASIC da 5,5 K versione 1.0, cioè nel Basic che attualmente vi abbiamo fornito.

Infatti alla riga 10 noi assegnamo alla variabile T il va-

lore 15 poi alla riga 20 diciamo al computer di eseguire la subroutine riportata alla riga 1000.

Questa subroutine non fa altro che scrivere per T volte consecutive, cioè per 15 volte, uno spazio vuoto sulla stampante ed essendo presente un (.) dopo l'istruzione SSCRIVI, il cursore della stampante **non ritorna mai a capo** bensì si sposta solo in orizzontale verso destra di 15 spazi.

Eseguita la subroutine si ritorna al programma principale, cioè alla riga 30 e qui troviamo un normalissimo SSCRIVI «GIACENZA» che obbliga la stampante a scrivere GIACENZA dopo i 15 spazi vuoti lasciati in precedenza.

Poiché dopo «GIACENZA» non è presente nessun «punto e virgola», se noi dopo la riga 30 inserissimo per esempio l'istruzione:

35 SSCRIVI «PRELIEVO»

la parola PRELIEVO verrebbe stampata all'inizio della riga successiva, cioè il cursore ritornerebbe automaticamente a capo.

È importante rispettare alla lettera la punteggiatura della riga 1000 infatti se per caso non inserissimo il «punto e virgola» prima dei «due punti» che precedono INCREMENTA X, cioè scrivessimo semplicemente:

1000 PER X=1 FINO T:SSCRIVI "":INCREMENTA X:RITORNA

anziché ottenere una scrittura spostata di 15 spazi dal bordo sinistro, otterremmo l'effetto di far avanzare la carta di 15 righe in quanto per 15 volte consecutive il cursore della stampante si sposterebbe di una posizione verso destra in seguito all'istruzione SSCRIVI "": contenuta in tale riga poi automaticamente ritornerebbe a capo della riga successiva.

Se poi tra le due «virgolette» anziché lasciare un unico spazio vuoto, battessimo per 3 volte consecutive la barra di interlinea, cioè:

1000 PER X=1 FINO T:SSCRIVI "":INCREMENTA X:RITORNA

la scrittura GIACENZA, anziché spostata di 15 spazi dal margine sinistro, ci apparirebbe spostata di 45 spazi, infatti in questo caso per 15 volte consecutive la stampante scrive 3 «spazi vuoti», quelli cioè che noi abbiamo inserito tra virgolette, quindi $15 \times 3 = 45$.

L'ISTRUZIONE UT

Anche l'istruzione UT deve essere impiegata in modo leggermente diverso rispetto a quanto da noi riportato sulla rivista n. 73.

In particolare vi diciamo subito che di «subroutine» utente se ne può utilizzare una sola la quale peraltro deve essere scritta in uno spazio ben delimitato che va dalla riga di memoria 1680 alla riga 16FF compresa.

Per utilizzare questa subroutine utente nel corso di un programma si scrive semplicemente UT ed automaticamente la subroutine stessa viene eseguita.

Facciamo un esempio pratico.

Supponiamo di voler realizzare una subroutine utente che faccia comparire nell'angolo in alto a sinistra la scritta BRAVO in negativo (come saprete i caratteri ne-

gativi non si riescono ad ottenere col Basic agendo sulla tastiera alfanumerica, quindi se si vuole scrivere una frase in questo modo occorre utilizzare la subroutine utente scrivendola sulla tastiera esadecimale).

Un programmino che ci permetta di ottenere ciò potrebbe essere ad esempio il seguente:

Riga	Codice	Mnemonico	Descrizione
1680	C5	PUSH BC	Salva il contenuto dei registri che
1681	d5	PUSH DE	vuole utilizzare ponendolo nello
1682	E5	PUSH HL	Stack.
1683	21		Carica in HL il numero 1692 corri-
1684	92	LD HL,1692	spondente alla prima riga di dati
1685	16		da trasferire.
1686	11		Carica in DE il numero EC00 corri-
1687	00	LD DE,EC00	spondente alla 1ª cella di me-
1688	EC		memoria dell'interfaccia video.
1689	01		Carica in BC il numero 0005 corri-
168A	05	LD BC,0005	spondente al numero di dati da
168b	00		trasferire, cioè le 5 lettere che
			compongono la parola BRAVO.
168C	Ed	LDIR	Trasferisce tutti i dati come da
168d	B0		noi indicato.
168E	E1	POP HL	Ripesca dallo stack il vecchio
168F	d1	POP DE	contenuto dei registri.
1690	C1	POP BC	
1691	C9	RET	Ritorna al Basic
1692	C2	B	
1693	d2	R	
1694	C1	A	Codici ASCII relativi alla parola
1695	d6	V	BRAVO in negativo
1696	CF	O	

Per inserire questa subroutine utente in memoria, dopo aver regolarmente caricato il BASIC (non prima se no si cancella) dovete pigiare sulla tastiera esadecimale il tasto RESET in modo da far comparire sul display la n seguita da 7 trattini orizzontali, dopodiché proseguirete come segue:

1) impostate sulla tastiera esadecimale il numero 1680 quindi pigiate i due tasti CONTROL-0 per vedere il contenuto di questa riga di memoria.

2) senza preoccuparvi del numero che comparirà sulla destra in quanto trattasi di un numero del tutto casuale, impostate sulla tastiera esadecimale il codice C5, quello cioè che deve andare nella riga 1680, poi pigiate i due tasti CONTROL-0.

3) Sul display comparirà il numero 1681 ed in questa cella voi dovete memorizzare un D5 poi pigiate ancora CONTROL-0.

4) Nella cella 1682 dovete memorizzare un E5, poi nella 1683 dovete inserire un 21 e così di seguito fino ad arrivare alla 1696 in cui inserirerete un CF.

5) Una volta scritto tutto questo programma in esadecimale provate eventualmente a ricontrollarlo come vi abbiamo già insegnato tante volte in modo tale da avere la certezza che non vi siano errori.

6) Eseguito tale controllo pigiate ora i tasti CONTROL-2 per accedere ai registri poi pigiate tante volte di seguito CONTROL-0 fino ad arrivare al registro SP.

7) Nel registro SP scrivete 1600 poi pigiate CONTROL-0 per passare al registro PC.

8) Nel registro **PC** scrivete **1000** poi pigiate **CONTROL-0** per trasferire questo numero in memoria.

9) Pigiate a questo punto i due tasti **CONTROL-4** ed automaticamente sul video tornerà a comparire la scritta **BASIC V 1.0** che si era interrotta quando avevate pigiato **RESET**, per confermarvi che potete programmare in Basic.

A questo punto per collaudare la vostra subroutine utente potreste realizzare un programma di questo genere:

10 SPSC

20 UT

30 VA A 30

Facendo eseguire questo programma con l'istruzione **ESEGUI** vedrete lo schermo spegnersi totalmente e comparire nell'angolo in alto a sinistra la scritta **BRAVO** In negativo come appunto desideravate.

Ovviamente ci sarà qualcuno che invece della scritta **BRAVO** vorrà far comparire una scritta diversa di propria ideazione, quindi dobbiamo necessariamente fornire anche a questi lettori tutte le delucidazioni del caso affinché possano riuscire facilmente nell'impresa.

Diremo pertanto che i caratteri **ASCII** corrispondenti alla scritta che si vuol far comparire debbono essere inseriti in memoria a partire dalla riga 1692 e seguenti e che tale scritta al massimo può arrivare come lunghezza fino alla riga 16FF in quanto dalla 1700 in poi inizia il programma **BASIC**.

Se la frase, invece di 5 lettere come la nostra, fosse composta per esempio da 10 lettere, alla riga 168A invece di 05 dovremmo scrivere 0A infatti 10 in esadecimale si scrive 0A.

Se invece la nostra frase si componesse di 18 lettere (che in esadecimale si scrive 12), alla riga 168A dovremmo scrivere un 12).

Come vedete il «trucchetto» è molto semplice da capire.

Se poi, invece che nell'angolo a sinistra del video, volesse far apparire la vostra frase in un altro punto, non dovrete fare altro che modificare opportunamente il numero contenuto nelle locazioni di memoria 1687-1688 tenendo presente che gli indirizzi assegnati al video vanno da EC00 a EDDF.

Ponendo per esempio in queste due locazioni il numero ED00 la scritta **BRAVO** In negativo comparirà all'inizio della riga centrale sul video.

PER ESEGUIRE UN'OPERAZIONE MATEMATICA

Supponiamo che qualcuno desideri far eseguire al computer un'operazione matematica anche molto complessa, per esempio:

$$15 \times 273 + 5987 : 22$$

In tal caso gli basterà scrivere:

SCRIVI 15 * 273 + 5987 / 22

poi pigiare il tasto **RETURN** ed automaticamente sul video, sotto la nostra scrittura, comparirà il risultato, cioè: 458.27

Nota: nei numeri decimali invece della «virgola» il computer pone sempre un «punto» ed anche noi per farci comprendere dovremo scriverli con il punto.

Facciamo notare che se il risultato di una qualsiasi operazione supera il tetto massimo di 999.999, il computer ci mostra il risultato sotto forma esponenziale, per esempio il numero 1.500.000 ci verrà visualizzato in questa forma:

1.5E6

Questo E6 che compare dopo 1.5 non è altro che un «moltiplicatore» corrispondente a «10 elevato alla 6».

In pratica per ottenere il nostro numero ci basterà spostare il punto verso destra, aggiungendo degli zero, di tante posizioni quante ne indica il numero posto dopo E.

Nel nostro caso ovviamente il numero in questione sarebbe 1.500.000.

Se invece il numero che appare dopo la E è preceduto da un segno «meno», significa che occorre spostare il punto decimale verso sinistra, sempre aggiungendo degli zero quando la situazione lo richiede, di tanti spazi quanti ne indica tale numero.

Per esempio: 1.5E-4

equivale a: 0,00015

CONCLUSIONE

Con questo noi oggi chiudiamo la nostra «chiacchierata» sul Basic sperando di esservi stati utili per chiarire qualche dubbio e soprattutto di avervi fatto comprendere come si possa iniziare ad impostare qualche programma.

Non riteniamo comunque chiuso l'argomento, anzi già dal prossimo numero pensiamo di spingerci più oltre iniziando a presentarvi dei semplicissimi programmi «gestionali» realizzati appunto in linguaggio Basic.

ELETTRONICA BUTTAZZO

VOLTASTRASSE 96 9056 BASEL SVIZZERA

La ditta **ELETTRONICA BUTTAZZO** concessionaria di **NUOVA ELETTRONICA** per la **SVIZZERA** invita tutti i lettori, a visitare il nuovo centro vendita situato alla **VOLTASTRASSE 96 4056 BASILEA** Tel. (061) 57 47 80.

Un attrezzato laboratorio è a disposizione per quei lettori che si trovassero in difficoltà nel montaggio dei Kits.