



RADIOAMATORI E PERSONAL COMPUTERS  
IDEE - PROGETTI - INFORMAZIONI



SUPPLEMENTO A **EQ** ELETTRONICA N. 11

# **XELECTRON**

L. 2000

# IC-745: IL MEGLIO PER LA SSB, CW, RTTY, FM.

Qualche dato saliente è sufficiente per distinguere questo nuovissimo ricetrasmittitore della ICOM progettato in modo da fronteggiare le avverse condizioni in gamma. La sezione ricevente a copertura generale (0.1 - 30 MHz) può essere predisposta anche sulle sole gamme radiantistiche. Il segnale all'ingresso è accoppiato direttamente al miscelatore e, se richiesto, la sensibilità può essere accentuata inserendo l'apposito preamplificatore a basso rumore. Ciò si traduce in + 12.5 dB di "intercept point" ed in 103 dB di dinamica. La prima media frequenza ad un valore molto alto (70 MHz) elimina virtualmente immagini e spurie. La selettività richiesta è raggiunta nella seconda conversione con gli appositi filtri ed è accentuata per mezzo dei

controlli SHIFT e WIDTH, mentre con la tacca di assorbimento - NOTCH-vengono efficacemente eliminate le interferenze. Il fastidioso segnale del radar russo può diventare un ricordo con le doppie costanti nella risposta del N. B. e dell'AGC. È stata prevista pure la demodulazione in AM. Il TX si distingue per il tasso d'intermodulazione molto basso: - 38 dB alla piena potenza erogata di 100 W su tutte le gamme radiantistiche da 1.8 a 30 MHz. Vi è comunque la possibilità, nel caso fosse richiesto, di abilitare l'emissione del TX in modo continuo entro gli estremi accennati. La sintonia può essere fatta con incrementi di soli 10 Hz! RIT e XIT aumentano la flessibilità operativa. In 16 memorie è possibile registrare la frequenza nonché i dati concernenti la predisposizione HAM/GENERAL ed il modo operativo prescelto. La frequenza operativa letta su un grande visore con 6 cifre può essere trasferita fra i due VFO oppure da questi due in memoria.

Ovviamente è possibile la ricerca entro le memorie oppure entro dei programmi limiti dello spettro. L'alimentazione avviene in continua (13.8V), un apposito scomparto interno

accomoda l'alimentatore a commutazione PS-35 opzionale. Altre unità opzionali quali filtri (250 Hz in CW!), calibratore, manipolatore ed unità FM potenziano le prestazioni dell'apparato.

## CARATTERISTICHE DI RILIEVO

Gammae operative: 1.8 - 2 MHz, 3.45 - 4.1 MHz, 6.95 - 7.5 MHz, 9.95 - 10.5 MHz, 13.95 - 14.5 MHz, 17.95 - 18.5 MHz, 20.95 - 21.5 MHz, 24.45 - 25.1 MHz, 27.95 - 30 MHz.

Possibilità della copertura continua da 1.8 a 30 MHz.

Ricevitore: 0.1 - 30 MHz in 30 bande

Sensibilità in SSB, CW, RTTY: 0.1 - 1.6 MHz < di 3.2  $\mu$  V per 10 dB S/D

1.6 - 3 - MHz < di 0,15  $\mu$  V per 10 dB S/D

Stabilità in frequenza:

< di 500 Hz a freddo

< di 100 Hz a regime

Peso: 8 Kg (11 Kg con al. interno)

Dimensioni: 111 x 280 x 355 mm

Potenza all'ingresso del PA:

200 W PEP

Regolabile in continuità fra 10 W ed il valore max.

Microfono: 600  $\Omega$

(non in dotazione)

Configurazione Rx:

a 3 conversioni

Livello d'uscita audio: > 2 W

Impedenza d'uscita: 8  $\Omega$

## ASSISTENZA TECNICA

S.A.T. - v. Washington, 1

Milano - tel. 432704

Centri autorizzati:

A.R.T.E. - v. Mazzini, 53

Firenze - tel. 243251

RTX Radio Service -

v. Concordia, 15 Saronno -

tel. 9624543

e presso tutti i rivenditori

Marcucci S.p.A.



ICOM

MARCUCCI S.p.A.

Milano - Via f.lli Bronzetti, 37  
ang. C.so XXII Marzo Tel. 7386051





# XELECTRON

SUPPLEMENTO  ELETTRONICA

novembre 1983

## sommario

- 4 **IL MONITOR** (Cattò)
- 18 **Le nuove opportunità** (Arias)
- 21 **Voice Robot** (Michienzi)
- 28 **Interfacciamo la TI-57** (Ibridi)  
atto secondo
- 36 **Penna luminosa per il Sinclair Spectrum**
- 41 **Conosciamo il NewBrain**
- 44 **Tutto quello che avete sempre desiderato sapere sul G5 ...e che avete fin troppo osato chiedere** (Becattini)  
**QUALCOSA DI PIÙ SULLA STRUTTURA INTERNA  
IL RELEASE 3.00 del GBASIC  
G5 e il LINGUAGGIO MACCHINA**
- 59 **Una ECONOMICA TASTIERA per il vostro MICROCOMPUTER** (Becattini)
- 62 **IMPARIAMO a INTERFACCIARE** (Becattini)  
**G5 - INTERFACCIA per STAMPANTE  
G5 - INTERFACCIA per REGISTRATORE**
- 68 **Il G5 e la RTTY** (Becattini)
- 71 **Scheda acquisizione dati per Personal IBM**
- 72 **Io e il G5** (Francescangeli)
- 81 **Il Micro-Professor (MPF III) della MICROTECH**
- 82 **Imparate oggi il Personal Computing**

EDITORE  
DIRETTORE RESPONSABILE  
REDAZIONE - AMMINISTRAZIONE  
ABBONAMENTI - PUBBLICITÀ  
40121 Bologna - via C. Boldrini, 22 - (051) 552706-551202

Registrazione Tribunale di Bologna, n. 3330 del 4-3-1988  
Diritti riproduz. traduzione riservati a termine di legge  
Iscritta al Reg. Naz. Stampa di cui alla Legge n. 416 art. 11  
del 5/8/81 col n. 00653 vol. 7 foglio 417 in data 18/12/82.  
Spedizione in abbonamento postale - gruppo III  
Pubblicità inferiore al 70%

DISTRIBUZIONE PER L'ITALIA  
SODIP - 20125 Milano - via Zuretti, 25 - ☎ 6987

DISTRIBUZIONE PER L'ESTERO  
Messagerie Internazionali - via Calabria, 23  
20090 FIZZONASCO di Pieve E. - (MI)

Cambio indirizzo L. 1.000 in francobolli  
Manoscritti, disegni, fotografie,  
anche se non pubblicati, non si restituiscono

s.n.c. edizioni CD  
Giorgio Totti

ABBONAMENTO Italia annuo L. 30.000 (nuovi)  
L. 29.000 (rinnovi)  
ARRETRATI L. 2.500 cadauno  
Raccoglitori per annate L. 8.000 (abbonati L. 7.200)  
+ L. 2.000 spese spedizione.

STAMPA: Tipo-Lito Lame - Bologna - via Zanardi, 506/B

SI PUÒ PAGARE inviando assegni personali e circolari,  
vaglia postali, o a mezzo conto corrente postale 343400,  
o versare gli importi direttamente presso la nostra Sede.  
Per piccoli importi si possono inviare anche francobolli

A TUTTI gli abbonati, nuovi e rinnovi, sconto del 10% su  
tutti i volumi delle edizioni CD.

ABBONAMENTI ESTERO L. 37.000  
Mandat de Poste International  
Postanweisung für das Ausland  
payable à / zahlbar an

edizioni CD  
40121 Bologna  
via Boldrini, 22  
Italia

# IL MONITOR

ovvero  
come risparmiare modificando seriamente un  
normale televisore bianco e nero

---

*ing. Sergio Cattò*

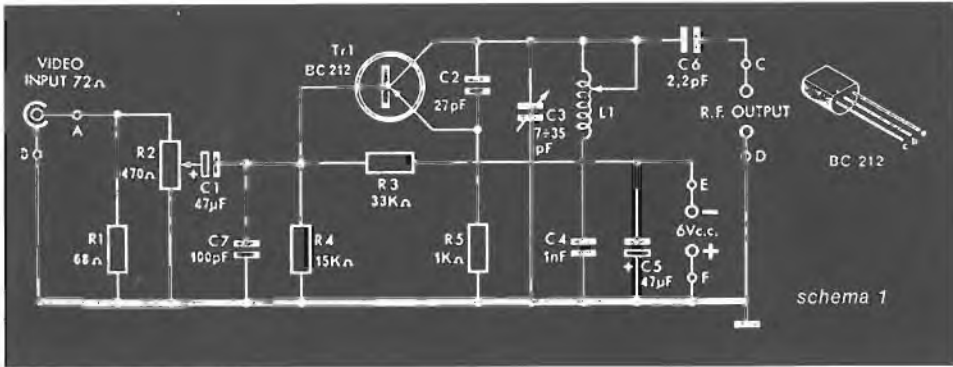
---



figura 1

---

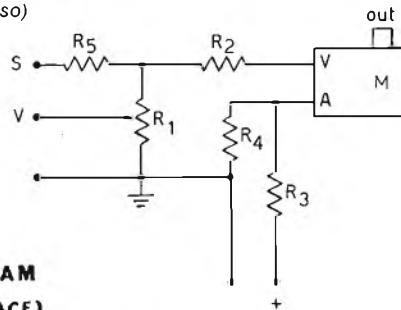
*Questa chiacchierata ha lo scopo di trarre di impaccio chi, affetto da personal computerite acuta, volesse risparmiare la spesa per l'acquisto di un monitor "professionale". Il modo più semplice è quello di utilizzare un modulatore VHF o UHF che trasforma l'uscita video del computer in una "emittente privata".*



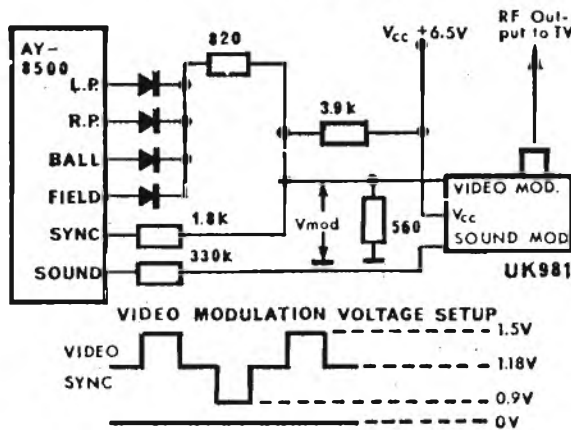
Il modulatore commerciale più comune è l'ASTEC, proveniente da Hong Kong; anzi, moltissime altre Aziende, vista la sua buona qualità ed economicità, preferiscono "ritargarlo" (vedi GBC o come in figura 5 la californiana M&R).  
 Nello schema 2 è rappresentato il tipico collegamento del moduletto ASTEC (figure 2 e 3).

schema 2

- M modulatore ASTEC e similari
- R<sub>1</sub> 200 Ω, potenziometro semifisso (attenuatore d'ingresso)
- R<sub>2</sub> 100 Ω, 1/2 W
- R<sub>3</sub> 1.000 Ω (vedi testo)
- R<sub>4</sub> 1.000 Ω
- R<sub>5</sub> 1.800 Ω (vedi testo)
- + tensione di alimentazione compresa tra 7 e 12 V
- V segnale video
- S sincronismo
- out segnale video RF



**CONNECTION DIAGRAM**  
**(AY-8500/UK981 INTERFACE)**



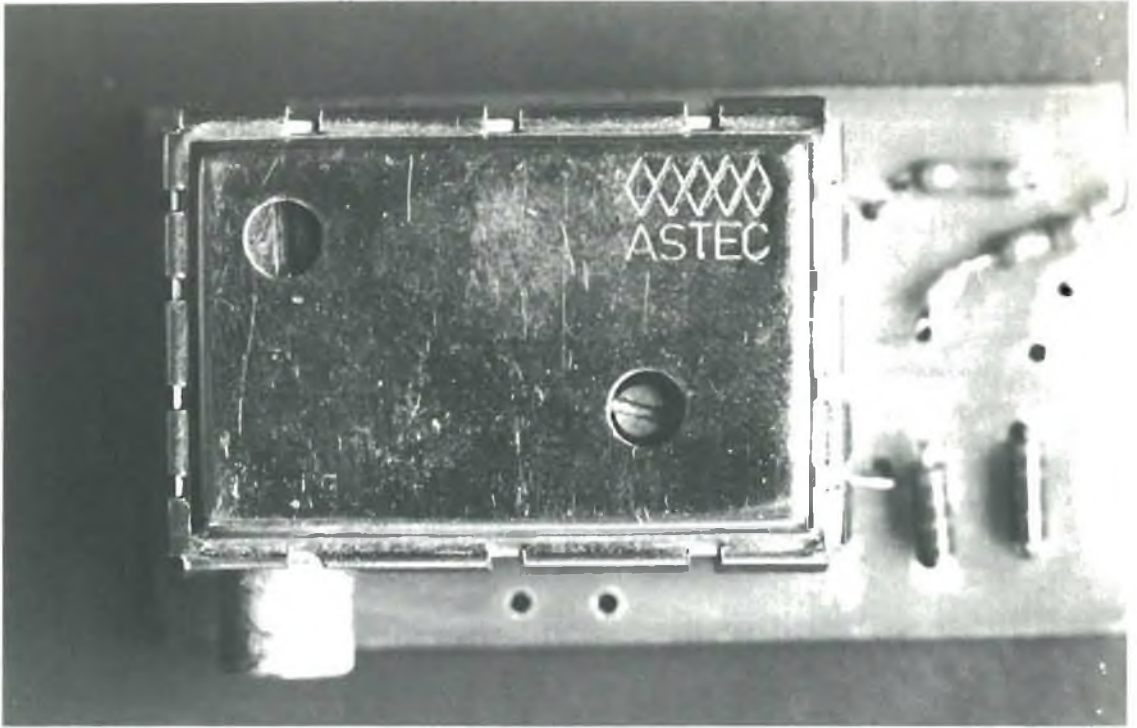


figura 2

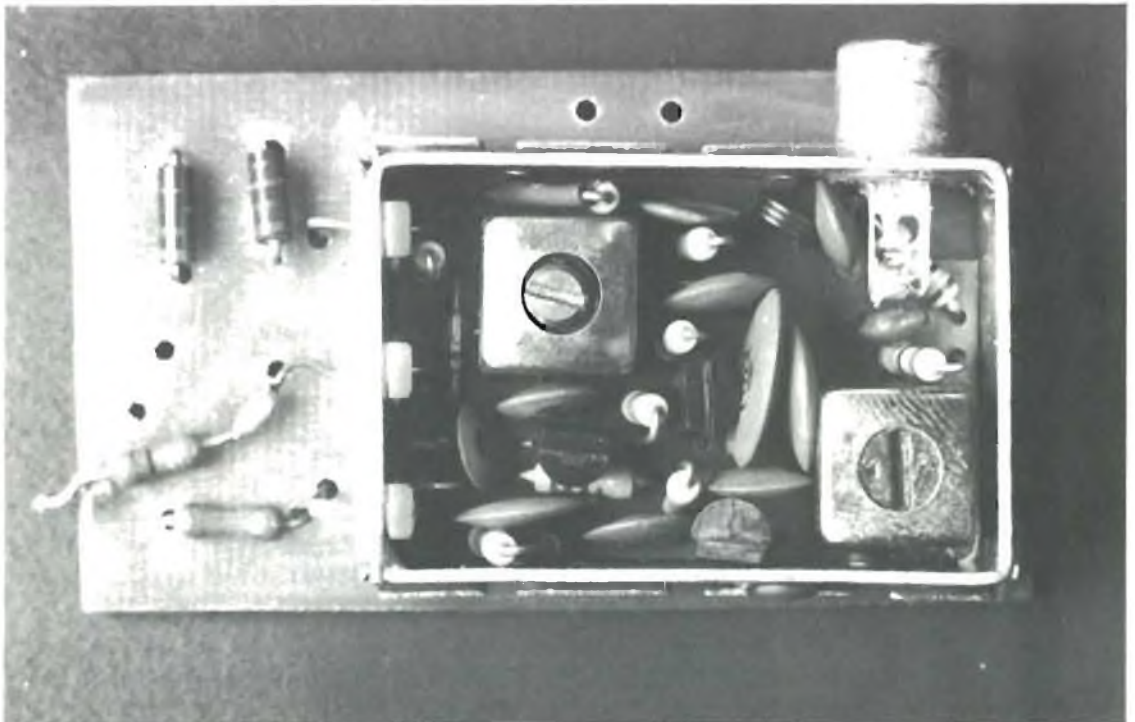


figura 3

Come commento al circuito si può dire che, cortocircuitando il resistore  $R_3$ , si può alimentare il modulatore con soli 5V e che il potenziometro  $R_1$  fa le funzioni di attenuatore qualora, caso non infrequente, si fosse in presenza di segnali tanto robusti da mandare in crisi il modulatore stesso.

Sempre nello schema 2 possiamo trovare il diagramma di connessione del modulo UK981 Amtron (che poi è il solito ASTEC con un'etichetta autoadesiva) con l'integrato AY8500, conosciutissimo da chi si diletta con giochi televisivi (video-game).

Chi preferisse la scatola di montaggio potrà utilizzare il kit KS340 della Kuriuskit: come possiamo vedere nello schema 1 la circuitazione rientra nella più classica ma efficiente normalità.

Questi modulatori vanno posti all'interno del personal computer, solitamente piuttosto spazioso.

La severissima normativa statunitense riguardante i radiodisturbi impone anche l'uso di nuclei toroidali di ferrite su cui avvolgere almeno 3 o 4 spire sia dei collegamenti d'ingresso che di quelli d'uscita.

Le figure 4 e 5 ci mostrano l'interno di un Apple con il modulatore e due nuclei toroidali: l'accorgimento è tanto semplice che vale la pena di metterlo in pratica.

Fatto ciò, basta collegare un normale televisore, sintonizzarsi sulla frequenza di emissione del modulatore, e il gioco sarebbe fatto, dico sarebbe perchè è possibile evitare la fatica di scollegare l'antenna e collegare il cavo proveniente dal computer.



*figura 4*

*Si notino i nuclei toroidali sulle connessioni di ingresso e di uscita del modulatore.*





figura 5  
Una possibile collocazione di un modulatore commerciale.

Qualche volta la non standardizzazione dei connettori ci può creare problemi e così l'immane Ditta californiana ci fornisce quello che, malgrado il roboante nome, è un semplice deviatore d'antenna magari con l'aggiunta di un adattatore d'impedenza (figure 6 e 7).

\* \* \*

Può anche capitare che non funzioni nulla: controlliamo tutto... poi leggiamo il manuale di istruzioni... e ci accorgiamo che le uscite per il monitor sono due: una per il segnale video vero e proprio, la seconda per il segnale di sincronismo (di solito negativo).

Se si manifesta questa eventualità, basta utilizzare ancora lo schema 1, aggiungere il resistore  $R_5$  e siamo a posto.

I possessori di un Pet Commodore hanno poi ulteriori difficoltà; pur essendo già inserito nel computer un valido monitor, potrebbe nascere l'esigenza di uno più grande, magari per utilizzarlo in qualche gioco di gruppo o più seriamente per sfruttare il segnale quale emissione utile a una TV privata. Infatti i segnali forniti sono tre: sincronismo orizzontale, sincronismo verticale e uscita video.



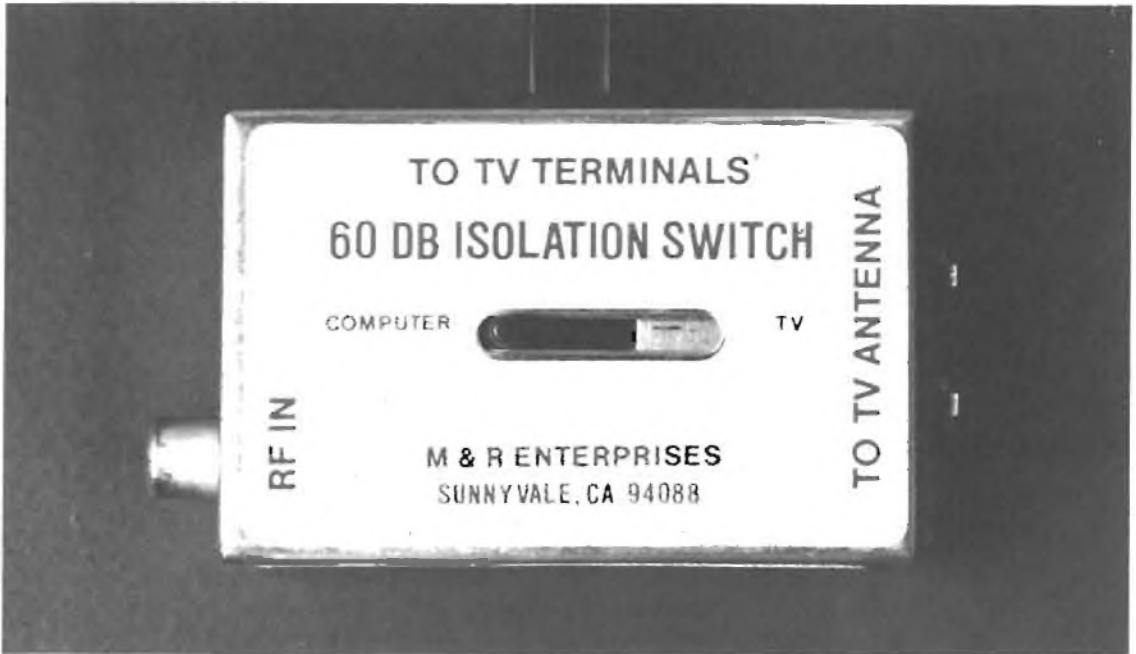


figura 6  
Deviatore d'antenna.

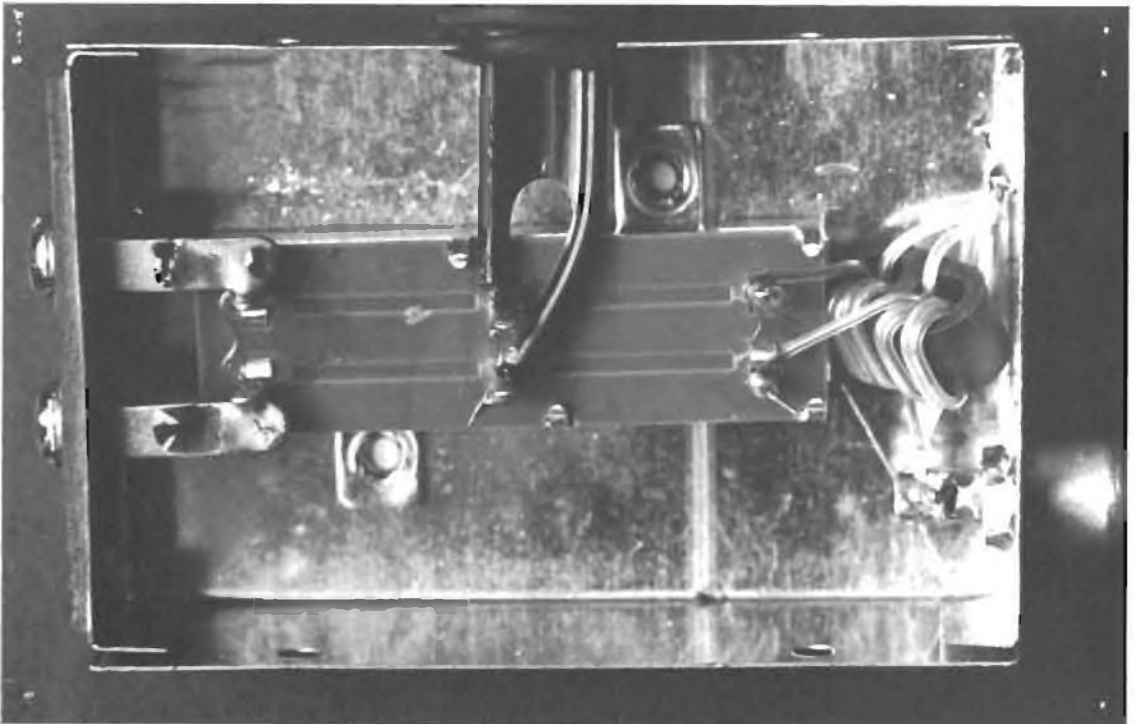
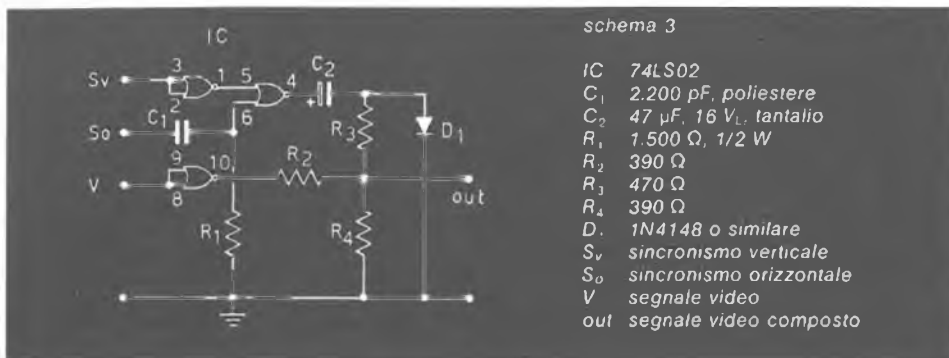


figura 7  
Interno del deviatore d'antenna; si noti l'adattatore di impedenza.

Quest'ultimo **non** è, come si potrebbe facilmente credere, l'insieme composto dei tre ma **solo** un singolo segnale che comanda l'accensione o lo spegnimento del pennello elettronico del tubo catodico.

Niente paura!

Lo schema 3 risolve il problema.



Il circuito si basa sul comunissimo TTL 74LS02, un integrato quattro porte nand a due ingressi, di cui solo tre porte saranno utilizzate.

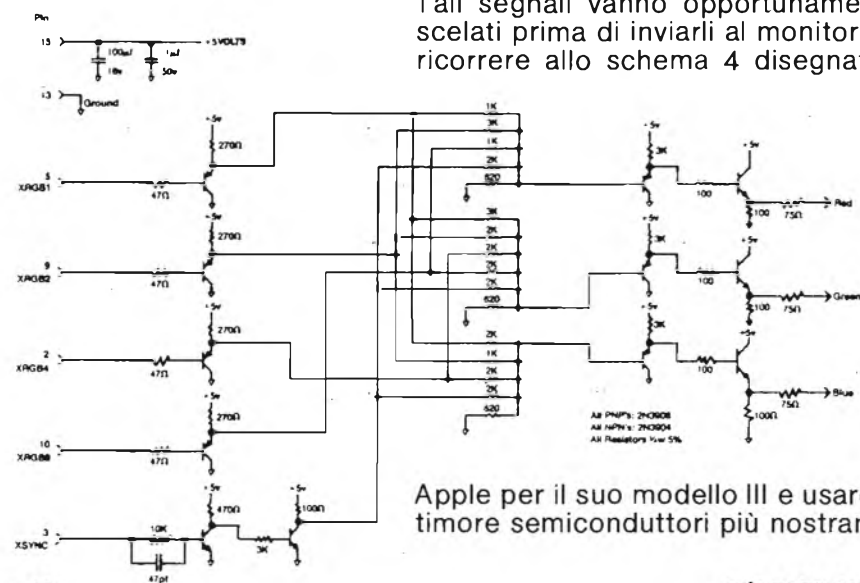
Naturalmente il circuito di schema 3 fornisce in uscita un segnale video composto che, prima di essere mandato al televisore, deve alimentare un modulatore come quelli descritti più sopra.

Possiamo trovare molte uscite monitor anche nei modelli di personal più costosi che prevedono l'uscita colore secondo lo standard RGB.

Questa sigla sta ad indicare che al monitor devono giungere tre segnali ben distinti corrispondenti ai colori rosso (Red), verde (Green), blu (Bleu). Il monitor è in questo caso piuttosto costoso dato che esistono tre circuiti distinti; l'immagine a colori è molto dettagliata e solo lontanamente paragonabile a quella di monitor a colori composti o al miglior televisore.

Normalmente i segnali che escono dalla vostra scatola delle meraviglie sono cinque: quattro uscite TTL le cui combinazioni danno 16 colori + un segnale di sincronizzazione a impulsi negativi.

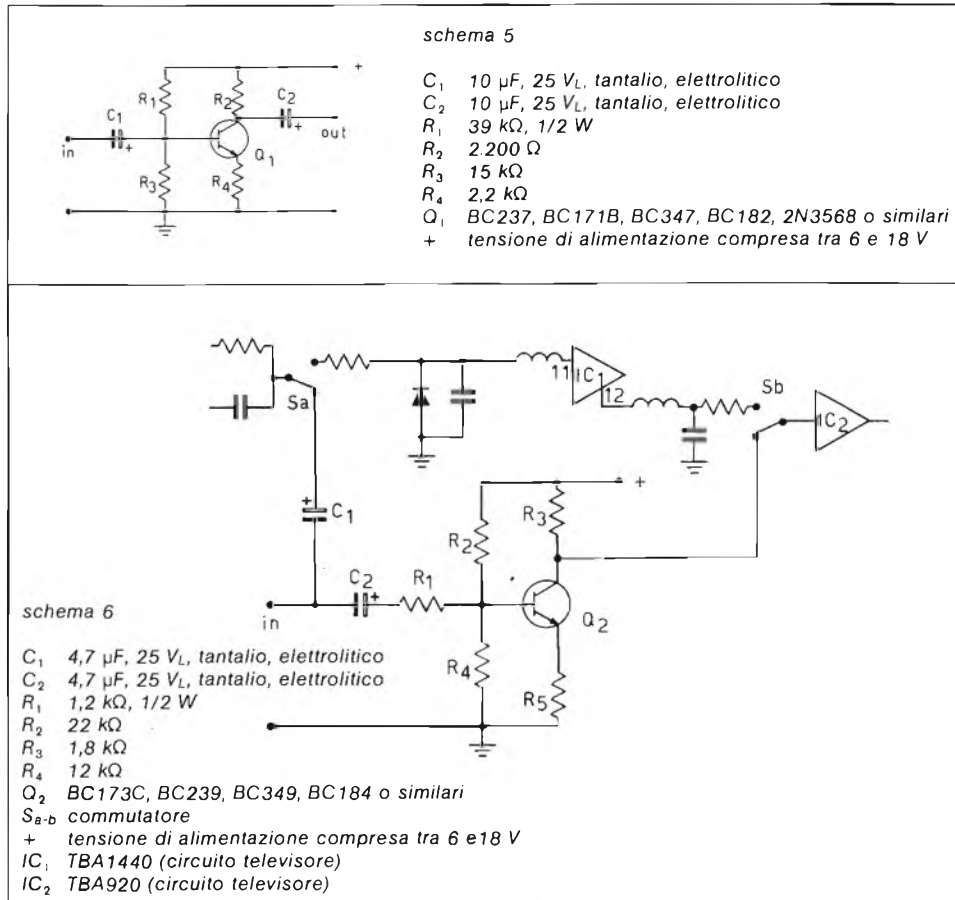
Tali segnali vanno opportunamente miscelati prima di inviarli al monitor; si può ricorrere allo schema 4 disegnato dalla



Apple per il suo modello III e usare senza timore semiconduttori più nostrani.

L'uso del modulatore ha l'indubbio vantaggio di non modificare il televisore originario, con buona pace familiare, ma presenta risoluzione piuttosto scarsa e grande sensibilità alle interferenze degli affollatissimi canali televisivi. Questo sistema è dunque sufficiente solo per usi hobbistici o dimostrativi limitati nel tempo.

Ma se riuscite a vincere la diffidenza di genitori, mogli e figli, allora siete pronti per i circuiti degli schemi 5 e 6.



La modifica prevede di intervenire sul televisore stesso e, se siete accorti, non ne comprometterete la funzione originaria.

Si tratta di introdurre il segnale proveniente dal personal computer dopo gli stadi di radiofrequenza: si risparmia il modulatore, non esistono disturbi dovuti a interferenze e la risoluzione dell'immagine è decisamente superiore.

Il costo, poi, è irrisorio, forse inferiore al più economico dei modulatori commerciali.

La parte più completa della realizzazione è quella riguardante l'individuazione del punto di introduzione del segnale primo sullo schema elettrico, poi sul "paziente".

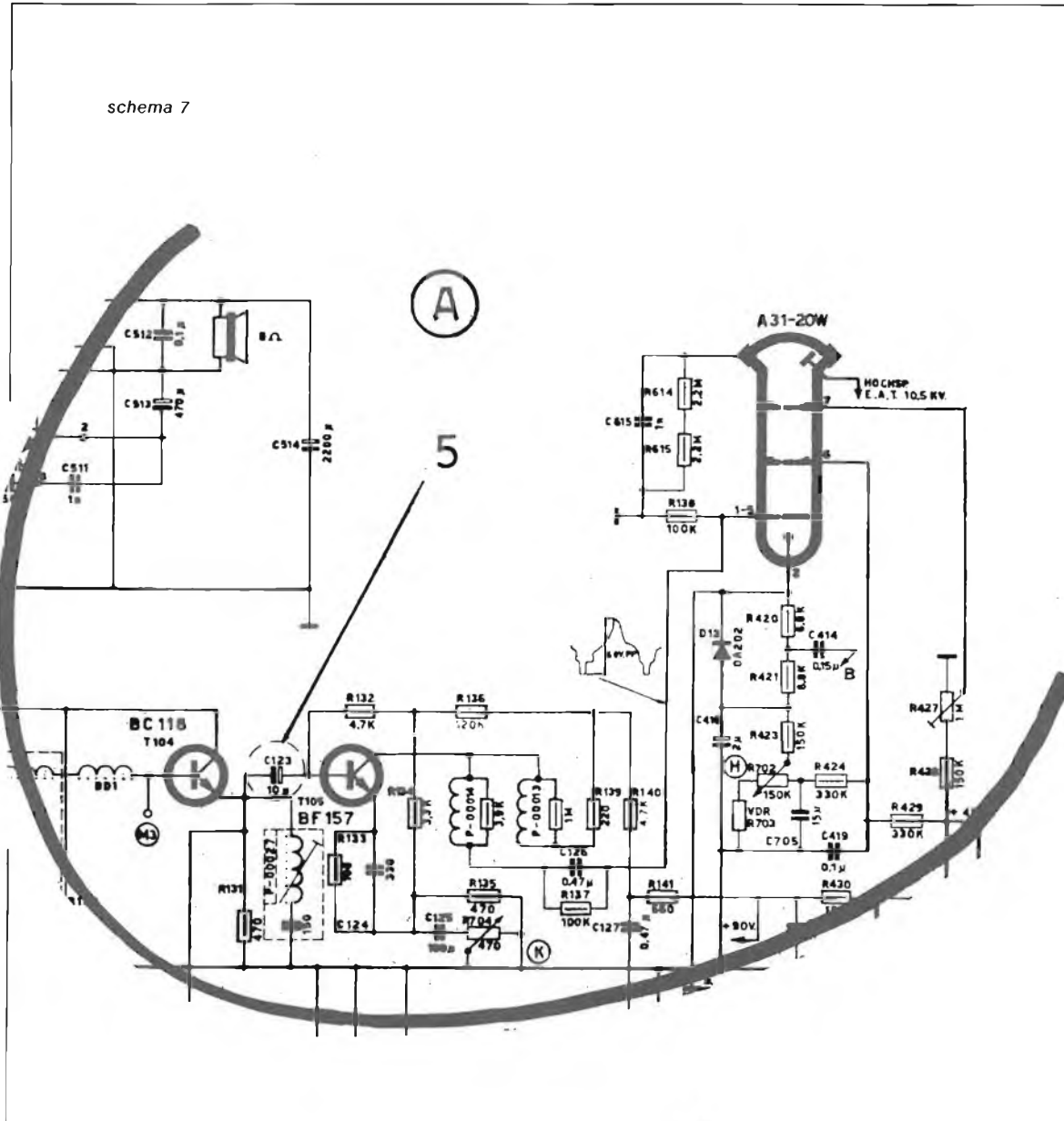
Per le modifiche si è utilizzato un anzianotto Korting mod. 671 e un moderno Grundig mod. 1221, scelta motivata dal fatto che erano già in casa.

Negli schemi 7 e 8 (A e B), quelli dei due televisori, cerchi tratteggiati fanno riferimento alle aree su cui intervenire.

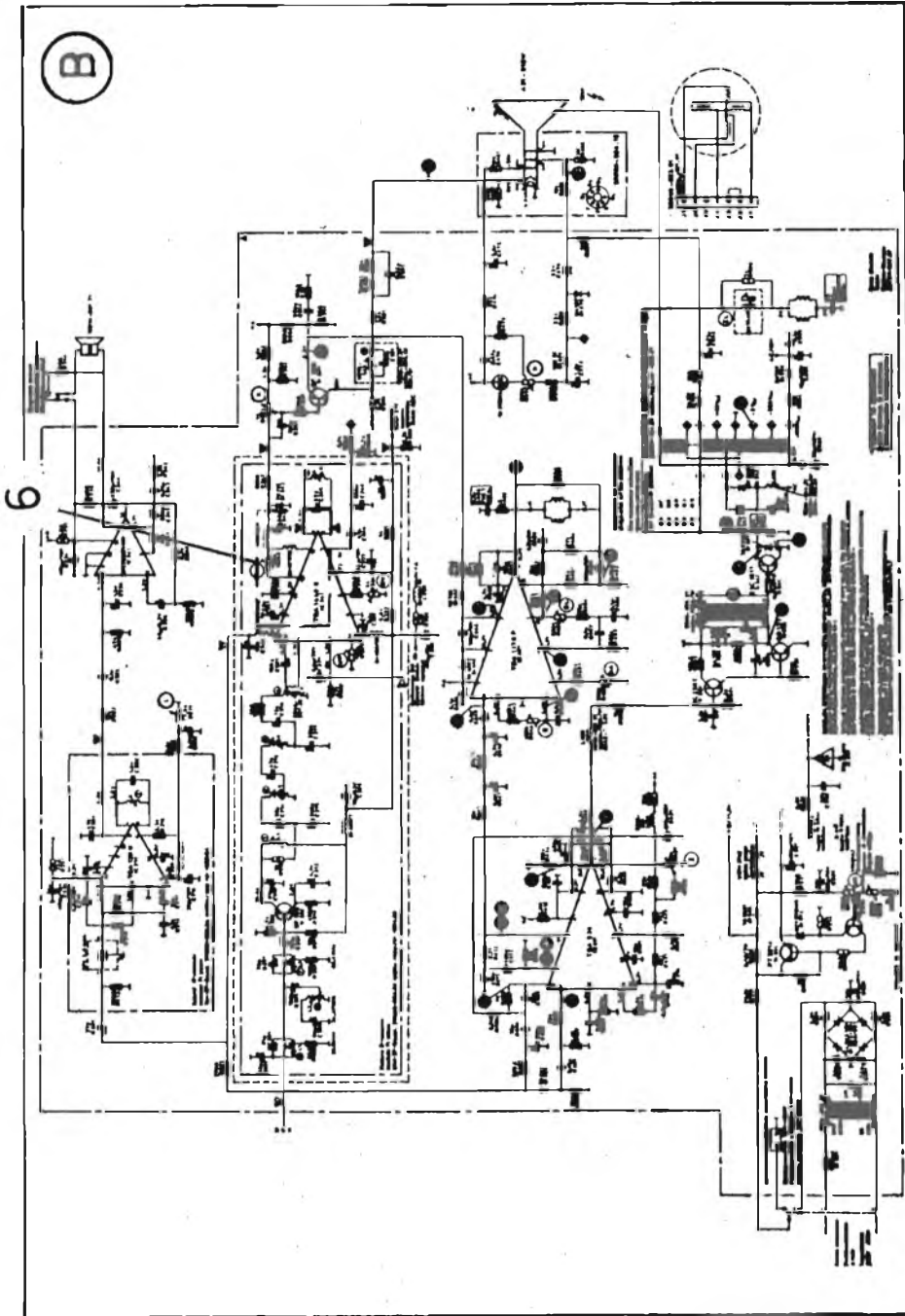
Gli schemi 5 e 6 non sono uguali poiché la moderna circuitazione del Grundig con soli tre circuiti integrati complica l'intervento.

Infatti il segnale per i circuiti di sincronizzazione esce dal terminale 12 del TBA 144OG mentre il segnale video esce dal terminale 11: siamo costretti così a intervenire in due punti del circuito.

schema 7







schema 8

## UN PROBLEMA DI SICUREZZA

Spesso nei moderni televisori, per ragioni di risparmio, si fa uso di autotrasformatori con il telaio connesso direttamente a una fase della rete elettrica.

### ATTENZIONE!

La connessione al computer può mettere in pericolo la sicurezza di chi opera sulla tastiera.

In tale eventualità è consigliabile alimentare il televisore attraverso un trasformatore separatore di rete, con rapporto di trasformazione 1:1.

Simili problemi non esistono coi televisori più vecchi, come il Korting delle fotografie, che sono sempre provvisti di trasformatore.

Per non perdere la funzione TELEVISORE è necessario fissare al mobile un deviatore (semplice o doppio a seconda dello schema usato) di buona qualità ma senza particolari specifiche.



figura 8

Ingresso monitor e deviatore d'antenna.

---

Nelle figure 8 e 9 si è fatto uso di un deviatore ITT da circuito stampato e una presa da pannello con connettore a vite di tipo microfonic per bassa frequenza: componenti più comuni di così non si poteva utilizzare. Il cavo è il solito RG59.



figura 9  
 Si noti il connettore AUDIO per l'ingresso monitor.

Un'occhiata alla figura 10 mostra la casalinga realizzazione sulla solita bassetta di bachelite millebuchi; il circuito è stato poi incollato al mobile.



figura 10  
 Esempio di montaggio dello schema 5.



figura 11  
La qualità dell'immagine è piuttosto buona.



figura 12





figura 13

Il risultato (figure 11, 12, 13) è decisamente buono e senza dubbio superiore a quello ottenibile con il costoso modulatore commerciale (i passaggi fotografici e tipografici non ne permettono il pieno apprezzamento). L'alimentazione conviene sempre prelevarla dal personal computer, i cui alimentatori sono quasi sempre abbondantemente dimensionati; inoltre il consumo dei nostri circuiti "aggiuntivi" è piuttosto limitato.

Anche se quanto sopra detto è dedicato al personal computer, gli schemi proposti, in particolare il 5 e il 6, possono essere egregiamente utilizzati per impianti televisivi a circuito chiuso o come monitor controllo per videoregistratori.

CIAO! \*\*\*\*\*

# Le nuove opportunità rilevanti potenze di calcolo disponibili per ciascuno di noi - il ruolo del "Personal" -

---

*I4AUC, ing. Marcello Arias*

---

*Negli ultimi dieci anni la tecnologia al servizio dell'informatica ha compiuto poderosi balzi in avanti, come in nessun altro settore delle scienze applicate.*

*Due sono state le linee portanti di questo sviluppo: i progressi nella tecnologia di base e la possibilità di integrazione con tecnologie diverse. La tecnologia di base, quella cioè relativa ai circuiti e ai componenti delle diverse unità hardware, ha conosciuto un'evoluzione spettacolare. Si sono ridotti enormemente i tempi di calcolo, gli ingombri fisici, i consumi energetici. La microminiaturizzazione ha consentito di ridurre le dimensioni delle memorie di ben 13.000 volte. E i costi hanno conosciuto una vera e propria caduta libera. Le memorie elettroniche a semiconduttori, per esempio, hanno visto ridurre i loro costi al ritmo del 28 per cento annuo.*

*Ma il vero salto qualitativo dell'informatica non è soltanto lo sviluppo tecnologico. È piuttosto la possibilità di una profonda integrazione tra le sue applicazioni, la sua estensione a campi fino ad ora esclusi.*

*Sono note le straordinarie potenzialità offerte dall'incontro tra informatica e comunicazioni. Le nuove tecnologie di memoria, oggi ancora in fase sperimentale, saranno determinanti nelle applicazioni che comportano l'elaborazione della voce e delle immagini. Così le nuove tecnologie delle comunicazioni, dai satelliti artificiali alle fibre ottiche, consentono già da ora un'ampia e assolutamente affidabile distribuzione delle informazioni. I microprocessori permettono, all'interno delle reti di elaborazione, di decentrare molte funzioni finora riservate ai centri di calcolo.*

*L'abbattimento dei costi della elaborazione dei dati, l'integrazione tra questa e le comunicazioni concorrono alla diffusione massiccia dell'informatica come strumento di lavoro e di conoscenza.*

*L'informazione diventa facilmente accessibile a chiunque la debba utilizzare.*

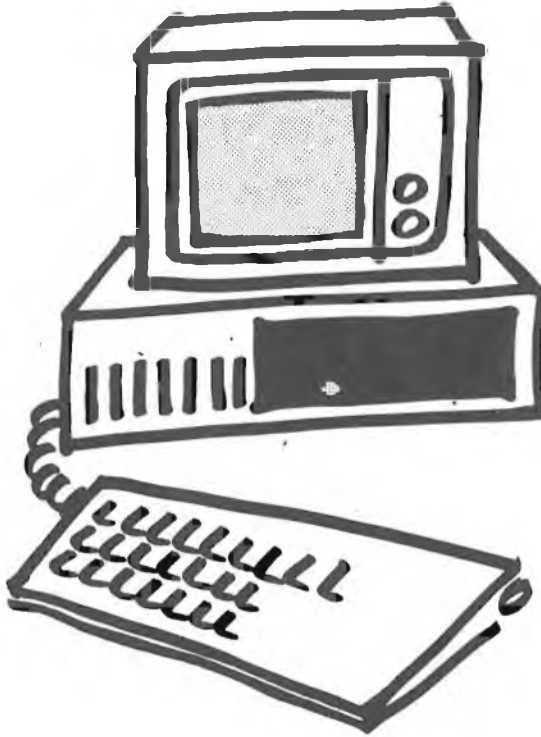
*Da qui il fenomeno delle reti di elaborazione e quello dell'informatica distribuita, in cui l'informazione e l'utente finale acquistano fisionomie diverse dal passato, più dinamiche e aperte.*

## *I nuovi protagonisti*

*Con lo sviluppo dell'informatica, a partire dagli anni '70 è venuto alla ribalta un nuovo protagonista: l'utente finale.*

*In passato, negli anni '50 e '60, l'informatica era il territorio esclusivo di pochi specialisti: i tecnici, chiusi nei centri meccanografici, operavano secondo metodologie estranee e incomprensibili ai più. Con i primi anni '70, e in particolare con l'annuncio del sistema/370 IBM, si aprì una nuova strada nella elaborazione dei dati e il calcolatore cominciò a essere strumento di lavoro abituale e familiare.*

*Oggi, da quando si è affermato il concetto di "informatica distribuita" - la possibilità cioè di raccogliere le informazioni là dove hanno origine per renderle disponibili dove servono - vedere terminali un po' dappertutto non stupisce più nessuno.*



*Si utilizza il terminale per la prenotazione dei posti in treno o in aereo; si preleva denaro contante da casse automatiche; nei supermercati, sistemi terminali sostituiscono i vecchi registratori di cassa. Nelle fabbriche, terminali raccolgono tutti i dati sulla produzione, si progetta su video. Si aggiornano le quotazioni di borsa, si vota e si battono le aste elettronicamente. Sono automatizzate le previsioni del tempo, il controllo del traffico aereo, le ricerche bibliografiche. E l'elenco potrebbe continuare a lungo.*

*Tutte queste applicazioni presuppongono il consumo diretto delle informazioni da parte di tutti o, comunque, da parte di persone prive di preparazione specifica. E proprio questo è diventato l'obiettivo primario dei produttori di informatica: l'attenzione all'utente finale, alle sue esigenze di facilità d'uso, applicative ed ergonomiche.*

## La facilità d'uso

Dalla fine degli anni '50 molta strada è stata percorsa nella semplificazione dell'utilizzo degli elaboratori. Oggi le operazioni da compiersi sono ormai molto facili. Ognuno di noi, anche se completamente digiuno d'informatica, non incontra difficoltà per esempio a prelevare denaro da sportelli bancari automatici. Lo stesso può dirsi anche per quelle attività che esigono l'uso di terminali più complessi, come nel caso del lavoro d'ufficio.

Personale di segreteria, che fino a ieri ha utilizzato la macchina per scrivere, è passato ora senza difficoltà a terminali per la gestione dei testi e degli archivi.

In molte redazioni, i giornalisti scrivono i loro pezzi direttamente al terminale. E, ancora, i manager simulano sui "personal" ipotesi di crescita aziendale.

Questa facilità d'uso è stata ottenuta a un prezzo elevato: la progettazione complessa e difficile. Far parlare e far capire alla macchina un linguaggio che molto si avvicina a quello naturale ha significato introdurre nella sua architettura tutta una serie sofisticata di funzioni, sia hardware sia software.

**Il Personal Computer:** insieme all'auto, al telefono, al televisore, è il grande protagonista del ventesimo secolo.

L'informatica di massa è il grande affare degli anni che viviamo, e -presumibilmente- del prossimo decennio.

Qualche cifra (leggetele con attenzione!): agli inizi degli anni '90 si prevede che saranno installati nel mondo circa 32.000.000 (trentaduemilioni!) di Personal Computers, e addirittura 300.000.000 (trecento milioni!!) alle soglie del 2.000.

Del resto i telefoni nel mondo oggi sono più di cinquecento milioni, le auto circa trecento milioni, per cui la stima di 300.000.000 di Personal è ragionevole.

E allora impariamo subito a familiarizzare con un Personal!

Questo numero di **XÉLECTRON** ve ne offre la opportunità. \* \* \* \* \*



IL COMPUTER È FACILE  
PROGRAMMIAMOLO  
INSIEME

GIANNI BECATTINI

## IL COMPUTER È FACILE PROGRAMMIAMOLO INSIEME

di Gianni Becattini

**INDISPENSABILE SE VOLETE  
COMINCIARE FINALMENTE A ENTRARE  
NEL MONDO DEI  
PERSONAL COMPUTERS**

Il volume è in vendita presso tutte le librerie e presso le Edizioni CD, via Boldrini, 22 - BOLOGNA - al prezzo di L. 7.500 (sconto 10% agli abbonati).



# Voice Robot

**preamplificatore che da'  
voce metallica**

---

*Francesco Michienzi*

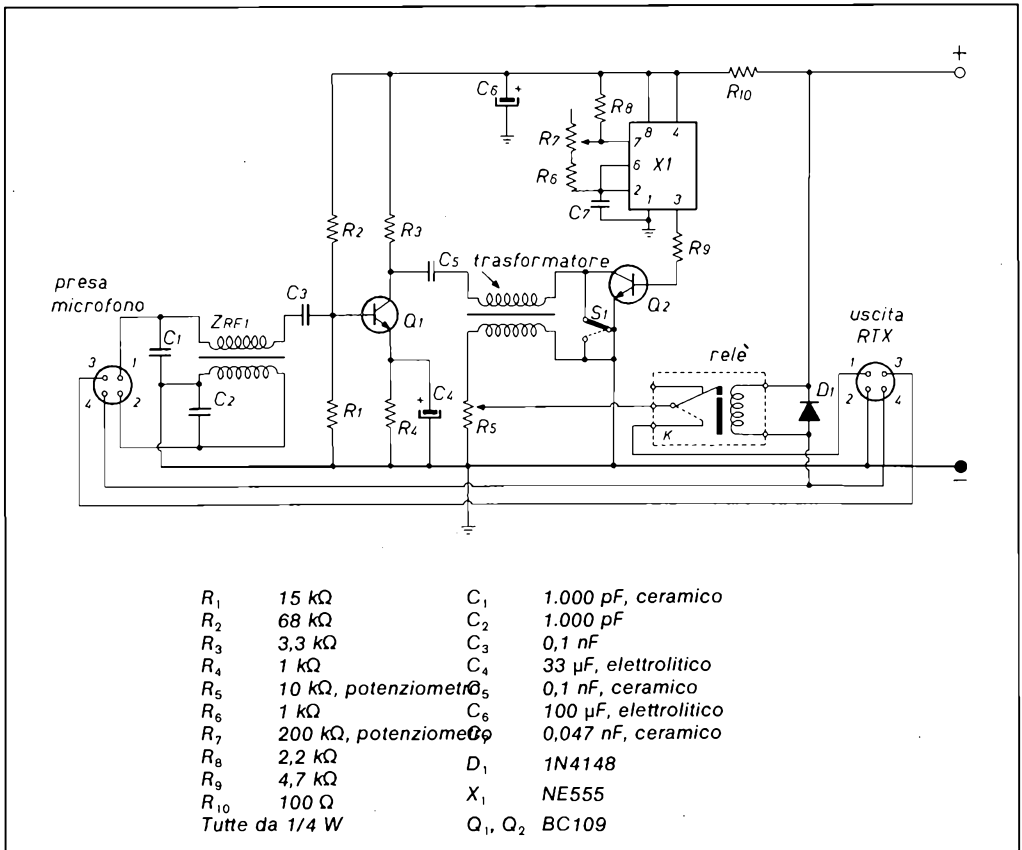
---

*In questi ultimi tempi la fantascienza, oltre a invadere il mondo della pubblicità e giochi per ragazzi con circuiti che riproducono fedelmente effetti eco, suoni spaziali e voci contraffatte, ha colpito una gran parte di radioamatori e CB che, per farsi sentire con una voce trasformata, spendono cifre che vanno dalle ottanta alle duecentomilalire, in prodotti sofisticati.*



A tale proposito ho quindi realizzato un particolare tipo di preamplificatore, che ho abbinato alla mia stazioncina, e che ho voluto presentare ai lettori di **CQ** e **XÉLECTRON** per deliziare i loro corrispondenti con una modulazione metallica, tanto per fargli credere di stare in QSO con un robot, e tutto con circa 15 milalire e un paio di ore di lavoro.

Osservando lo schema elettrico, ci accorgiamo che il suo funzionamento è molto semplice; il segnale BF generato dal microfono viene amplificato di circa trenta volte dalla sezione rappresentata da  $Q_1$ , e successivamente sottoposto al controllo di un interruttore elettronico costituito da un integrato NE555, denominato generatore di ritmo, e da un BC109 collegato col suo collettore al primario di un trasformatore di accoppiamento.



L'uso di tale trasformatore è da ricercarsi nel fatto che per far funzionare l'interruttore elettronico è necessario disaccoppiare la massa. Questo trasformatore reperibile ovunque (o al massimo recuperabile da qualche vecchia radiolina) ha un primario di 100  $\Omega$  che viene collegato tra l'uscita di  $Q_1$  ( $C_5$ ) e il collettore di  $Q_2$ , e il secondario di 200  $\Omega$  che invece colleghiamo da una parte a massa e dall'altra a un capo del potenziometro  $R_5$  (volume), dal quale cursore, tramite il relè, noi convogliamo il segnale all'entrata del ricetrasmittitore.

L'uso del relè è utile ad accoppiare il circuito a qualsiasi RTX, con qualsiasi tipo di commutazione.

Il piedino 3 di  $X_1$  pilota  $Q_2$ , il quale ha la funzione di interruttore elettronico, difatti quando tale interruttore rimane chiuso, cioè quando  $Q_2$  si trova in conduzione, la riproduzione sonora si verifica normalmente mentre, quando va all'interdizione, l'interruttore elettronico si apre e cessa la riproduzione.

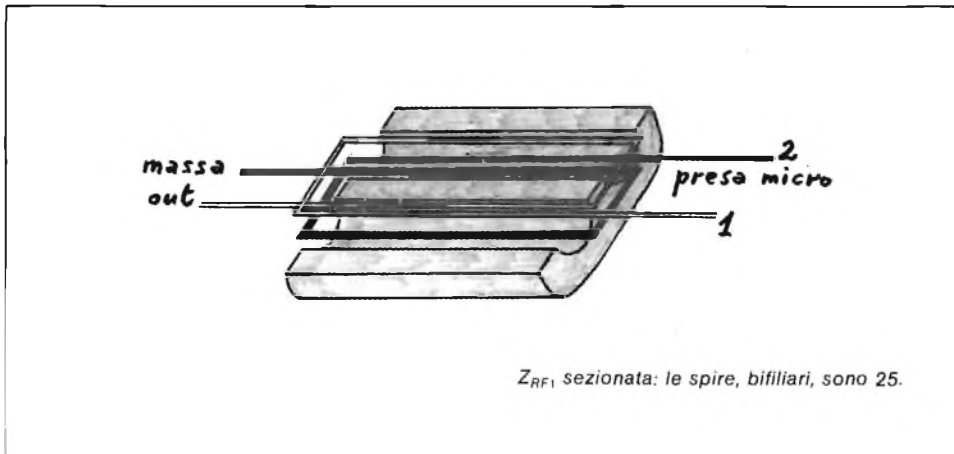
Il tempo di apertura e chiusura di tale interruttore dipende solo ed esclusivamente dal generatore di ritmo costituito da  $X_1$  (montato in configurazione astabile) il quale genera una frequenza a onda quadra che varia in rapporto alla posizione del cursore del potenziometro  $R_7$ . Pertanto, agendo su  $R_7$ , noi regoliamo la frequenza ideale che serve a modificare il timbro della voce del robot.

Per fare questa prova, una volta montato il tutto, colleghiamo il solo conduttore positivo alla boccola dell'alimentatore, perché la massa diventa unica, utilizzando il ricetrasmittitore come amplificatore (PA) e, quando siamo sicuri che l'imitazione è perfetta, parlando magari cadenzati, possiamo tranquillamente andare in QSO (CB).

Tra la massa e il collettore di  $Q_2$  è collegato un interruttore ( $S_1$ ) a levetta, che serve nel caso si voglia usare il dispositivo solo come preamplificatore BF.

Sull'entrata del circuito notiamo un filtro formato da  $C_1$ ,  $C_2$  e  $Z_{RF1}$ , che serve ad evitare che alta frequenza entri dal microfono e influenzi il preamplificatore.

L'impedenza suddetta è costituita da 25÷30 spire bifilari avvolte dentro un nucleo di ferroxcube a due fori, con i conduttori collegati in opposizione di fase, come si può vedere in figura.



Per quanto riguarda i due transistori, consiglio di non usare equivalenti perché il circuito, al momento della realizzazione, è stato provato con semiconduttori, col contenitore plastico, e dava in altoparlante oltre alla voce robot, una assordante sirena.

L'uso dei BC109 è da preferirsi perché, a mio parere, col loro contenitore metallico schermano le oscillazioni prodotte da  $X_1$  e danno una riproduzione pulita.

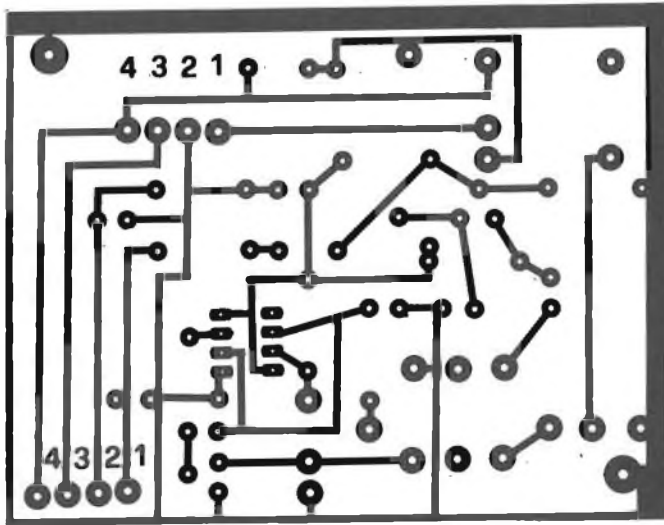


---

Per quanto riguarda il circuito stampato, consiglio chi si accingesse a costruirlo, di procurarsi prima il relè e il trasformatore e sistemare le piazzole di saldatura in corrispondenza dei terminali di tali componenti, perché non hanno tutti le stesse forme e dimensioni.

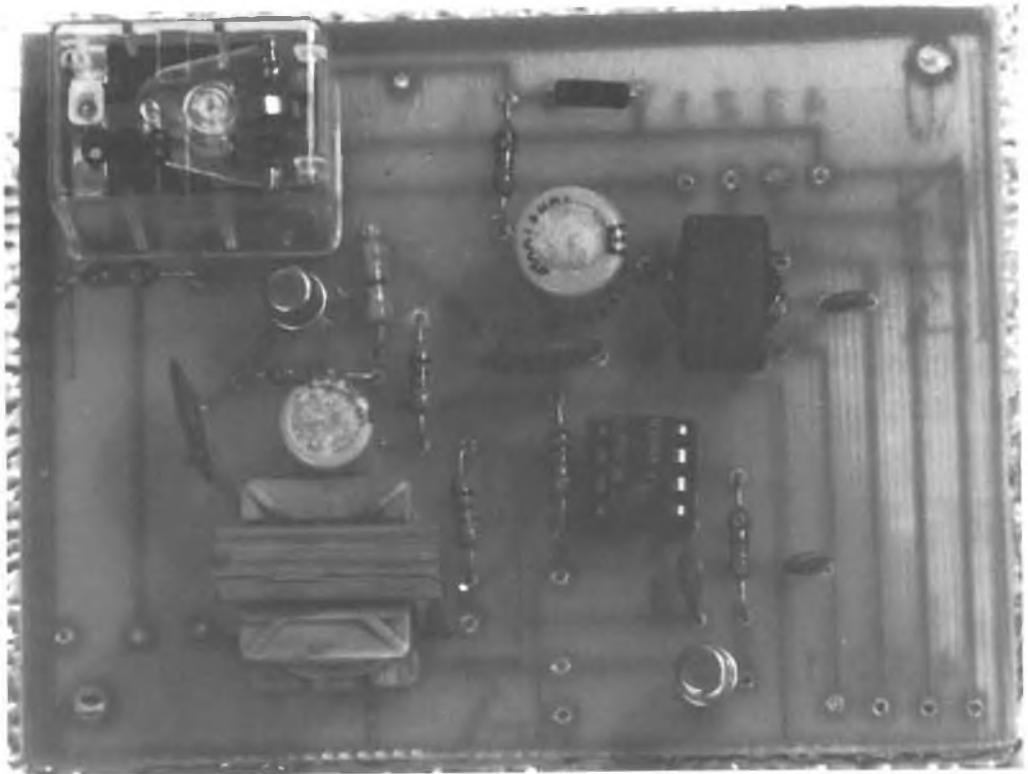


## LATO RAME



SCALA 1:1

**QUESTO STAMPATO PUÒ ESSERE RIPRODOTTO FACILMENTE.  
GIRATE PAGINA.**



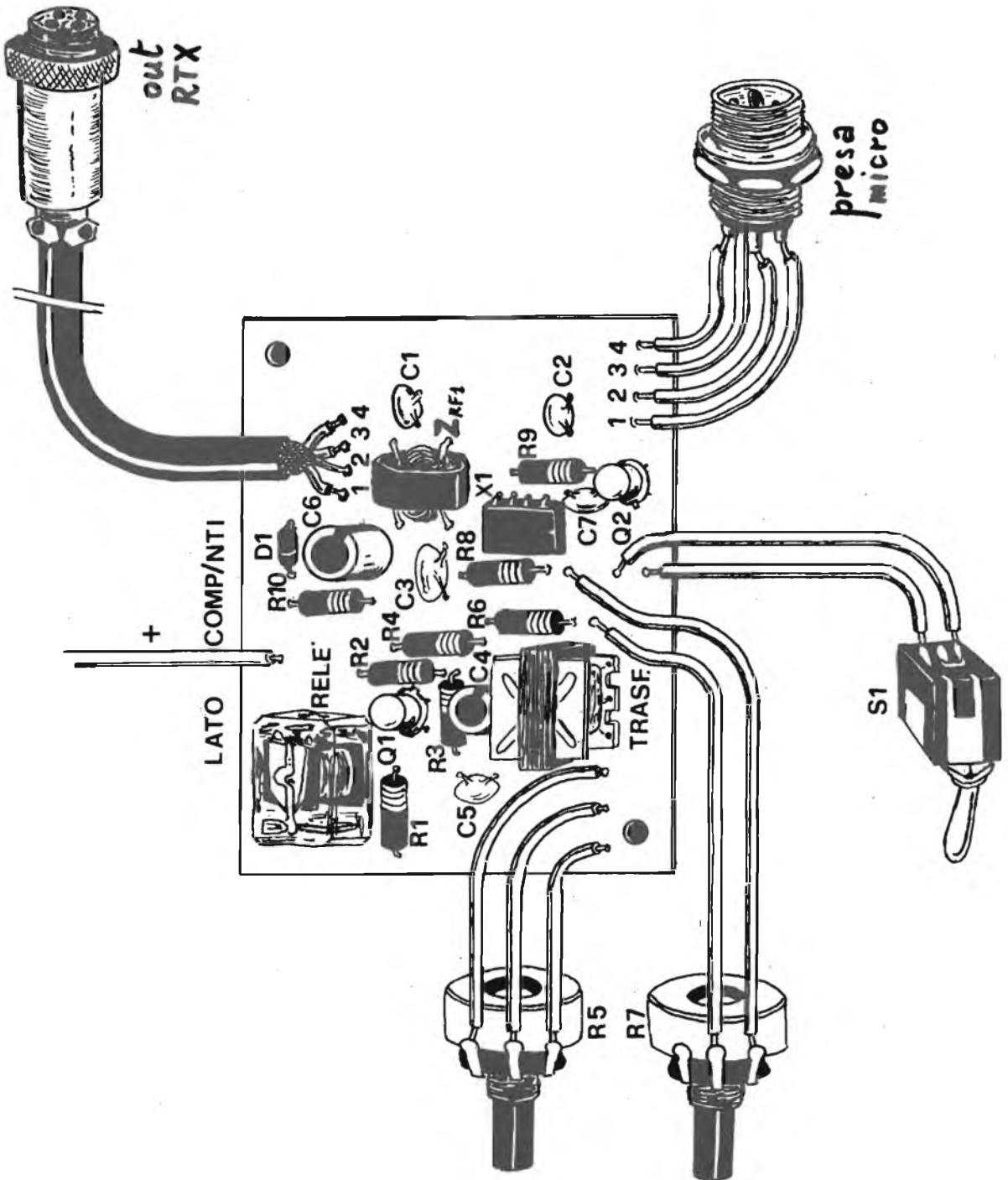
Lo spazio soprastante è lasciato bianco intenzionalmente: consente la riproduzione facile e a basso costo dello stampato pubblicato a pagina precedente (vedere CQ n. 4/83, pagina 45 e seguenti, per la procedura).

---

---

**Leggi a pagina 70 come devi fare per ricevere subito in regalo questa utilissima cassetta, + 1 set completo di 6 cacciaviti di diverse dimensioni.**





Il tutto, infine, va chiuso dentro una scatola metallica, montando sul davanti (se lo si desidera) una presa per microfono a quattro poli, e dietro un cordone a chiocciola munito di normale connettore sempre a quattro poli.

\*\*\*\*\*

# INTERFACCIAMO

## la

# TI57

-atto secondo-

---

*I4IBR, Marco Ibridi*

---

*Dopo aver presentato l'interfaccia TTL/TI57 (XÉLECTRON 3/82), il passo logico successivo è certamente quello di abilitare la '57 alla lettura delle informazioni da nastro magnetico. Non occorre nemmeno soffermarsi per un attimo nella spiegazione di come sia possibile affidare a una comune cassetta da riproduzione magnetica il compito di supportare dati binari, tanto comune è, ormai, il binomio stereocassetta/personal, da meravigliarsi nel sentire una cassetta riprodurre musica!*



Il problema di fondo, però, è che come tutti sanno (e se non lo sanno, ora lo imparano) il supporto magnetico a nastro è, per sua stessa natura, un supporto sequenziale dei dati.

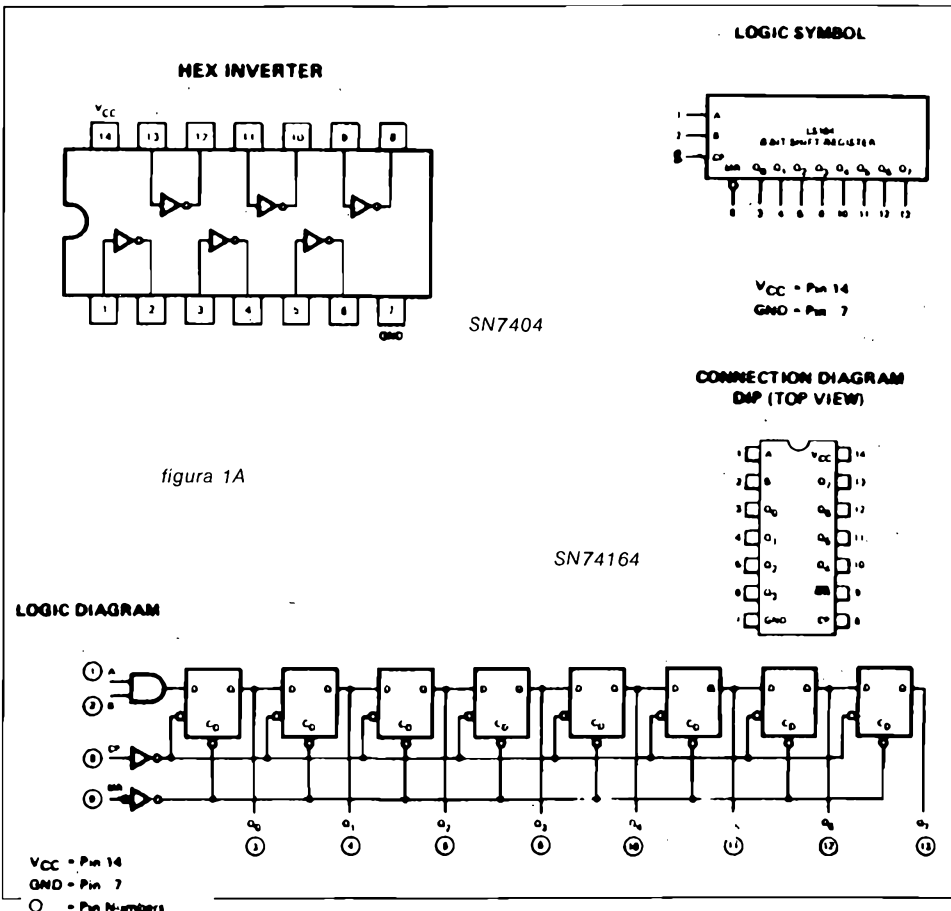
Ciò, però, vuol dire tutto e nulla.

Infatti se contrapponiamo la gestione sequenziale dei dati digitali su di un nastro a quella random di un disco, diamo sfoggio di grande cultura informatica ma non abbiamo afferrato per niente il problema della nostra adorata calcolatrice.

Questo perchè, quando parlo di "problema" mi riferisco non al fatto che su nastro non posso accedere casualmente alle informazioni, ma al fatto che, a meno non si ricorra a sistemi di riproduzione a più piste (e nel nostro caso specifico ne servirebbero ben 8), il nastro viene letto bit dopo bit e quindi la presentazione dei dati è decisamente seriale, anche se qualche Archimede Pitagorico può pensare a mostruosi lettori di minicassette con testine accavallate l'una sopra l'altra.

Compreso l'inghippo, vediamo come risolverlo.

Senza andare troppo verso il fantascientifico, il mercato delle "vecchie" e care TTL ci presenta numerosi tipi di shift-register (parolone per definire, nella maggioranza dei casi, quell'aggeggio che a ogni "botta" di clock ti sposta a destra o a sinistra il segnale di input) e tra questi l'adattissimo SN74164 che potete gustarvi "en déshabillé" alla figura 1A.



Vediamo ora il metodo adottato per la registrazione e relativa decodifica dei dati:

- 1) i sistemi autoclockanti;
- 2) i sistemi non autoclockanti.

Il sistema non autoclockante richiederà sempre un segnale a parte (normalmente il clock) che fornisca al riconoscitore di segnali un riferimento per potere interpretare esattamente la sequenza di zero e uno, e quindi nel caso di utilizzo su nastro magnetico si parlerà di una pista supplementare. Con un sistema autoclockante, ovviamente, il riferimento per l'esatta interpretazione sarà dato dal segnale stesso e ciò semplificherà di molto le cose.

Nel nostro caso specifico, ho adottato un metodo di registrazione che generalmente è utilizzato in "versione" non autoclockante ma che io ho, con opportuni artifici, adattato alla bisogna.

Si tratta della **FSK**, famosissima anche nelle comunicazioni "discrete", ovvero Frequency Shift Keying.

A tale scopo ho assegnato, affidandomi unicamente al mio libero arbitrio, un valore alla condizione logica 1 e alla condizione logica 0, rispettivamente  $\pm 2.000$  Hz e  $\pm 750$  Hz.

Ho fatto in modo però, e più avanti vedremo come, che il "riconoscitore" contasse i bit letti e che, una volta riempito lo shift-register con il byte interessato, attivasse l'interfaccia, si resettasse e continuasse la lettura.

Bello vero?

Rimane ancora un problema, infatti avrete certamente notato la mancanza di segnali del tipo "start" e "stop" che possano comunicare al riconoscitore l'inizio e la fine del byte trasmesso.

Ciò in pratica significa che, se il sistema di riconoscimento e decodifica viene attivato quando la trasmissione dei dati è già iniziata, il contatore di parola segementerà i bit ogniqualvolta il pin 7 dello SN74164 è a condizione logica 1, ottenendo così dati non coerenti, in quanto il sistema non si sarà certamente sincronizzato sul primo bit di un byte qualsiasi, rispettando quella legge che dice che se c'è qualcosa che può andar male, andrà sicuramente male.

Si è elegantemente aggirato l'ostacolo utilizzando cassette di durata minima (di quelle, per intenderci, da jingle per radio privata) in modo che su di una cassetta venga registrato un solo programma, ripescabile in qualsiasi momento riavvolgendo completamente il nastro.

E ora sotto con gli schemi!

---

## Prima parte

---

- Il convertitore seriale/parallelo
- Il "sincronizzatore" e le alimentazioni
- L'interfaccia convertitore/decodificatore

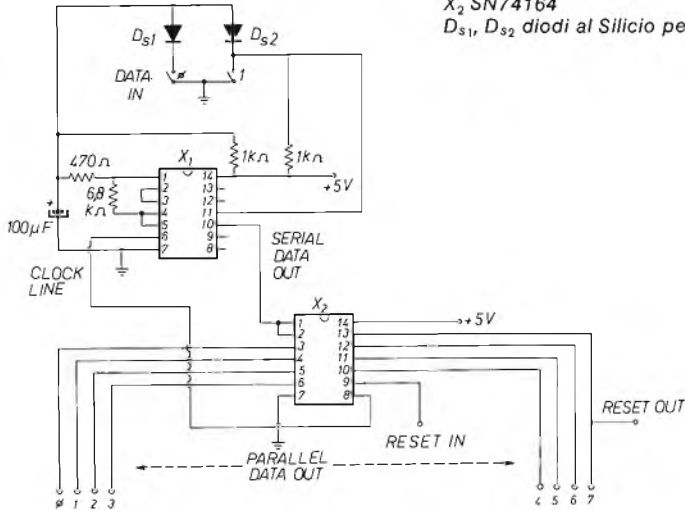
### Il convertitore seriale/parallelo

La figura 1B ci mostra il cuore del sistema, in particolare possiamo osservare l'integrato  $X_1$  (SN7404) cablato in maniera un po' insolita (schema e idea



figura 1B

$X_1$  SN7404  
 $X_2$  SN74164  
 $D_{s1}, D_{s2}$  diodi al Silicio per usi generali

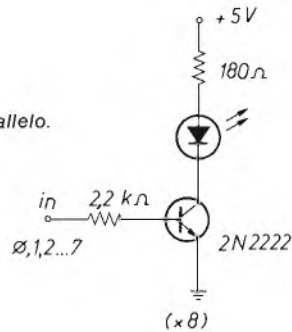


da un meraviglioso testo americano, "TTL COOKBOOK" di D. Lancaster, purtroppo non in vendita in Italia) quale "bounce eliminator", ovvero eliminatore di rimbalzi, a trigger di "Schmitt" indispensabile per il perfetto funzionamento dello SN74164 che, anche se nel progetto sarà comandata da switch elettronici, non funziona correttamente se il clock non è triggerato (provare per credere!).

Come si può vedere, sia la condizione logica 1 sia la condizione logica 0 fanno avanzare il clock mentre a condizione logica 1 il segnale è inserito nello shift-register.

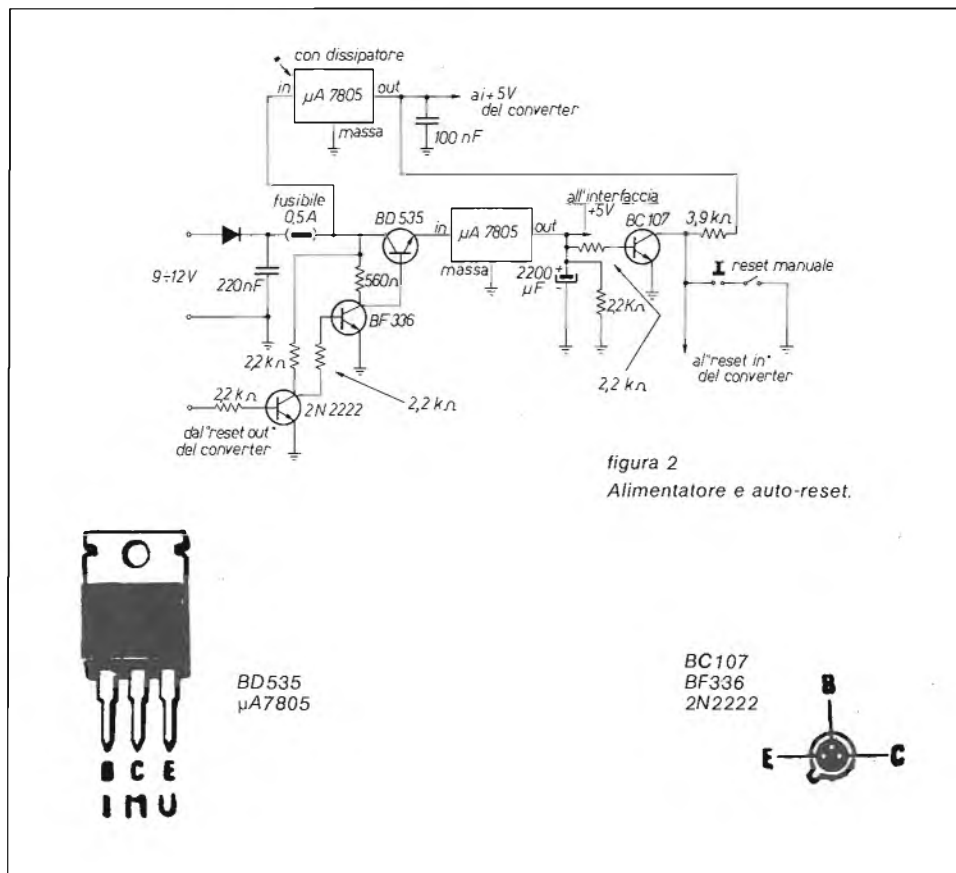
Possiamo inoltre notare in figura 1C un comodo indicatore dello stato logico presente in uscita, che andrà realizzato cablando otto circuiti identici... per la gioia dei nostri occhi.

figura 1C  
 Convertitore seriale/parallelo: display dei dati parallelo.



## Il "sincronizzatore" e le alimentazioni

In figura 2 troviamo un circuito, in realtà abbastanza ridondante ma affidabilissimo, che, preso il segnale dal piedino 7 dello SN74164 (come vedremo l'inizio di ogni byte sarà sempre a logica 1) darà tensione all'interfaccia e, dopo qualche istante, la cui durata è dovuta al condensatore elettrolitico da 2.200  $\mu\text{F}$  e relativa resistenza, "resetterà" lo shift-register predisponendosi per ricevere il prossimo byte.



Abbastanza chiaro?

Per chiarirci ulteriormente le idee, diamo uno sguardo alla figura 3 dove è riportata l'intera codifica dei tasti della TI57 Texas utilizzata dall'interfaccia e noteremo l'aggiunta di un bit (sempre settato a 1) all'estrema destra del byte, utilizzato per sincronizzare lo shift-register.

Infatti il byte di suddetta codifica verrà letto da nastro partendo dal primo bit di destra e quindi dopo otto colpi di clock (corrispondenti ai bit del byte letto) si troverà sul pin 7 dello shift-register avviando così il procedimento sopra riportato.

Ovviamente in fase di "scrittura" (registrazione) il byte andrà immesso a rovescio di come indicato in figura 3, cioè prima l'ultimo bit di destra, poi il penultimo e così via.

2nd	0 0 1 0 1 0 0 1	8	1 1 1 0 1 1 0 1
LRN	0 1 0 0 1 0 0 1	5	0 0 0 1 1 1 0 1
SST	0 1 1 0 1 0 0 1	2	1 0 0 0 1 1 0 1
BST	1 0 1 0 1 0 0 1	.	1 1 0 0 1 1 0 1
GTO	1 1 1 0 1 0 0 1	CE	0 0 1 0 0 0 1 1
SBR	0 0 0 1 1 0 0 1	$\sqrt{x}$	0 1 0 0 0 0 1 1
RST	1 0 0 0 1 0 0 1	SUM	0 1 1 0 0 0 1 1
R/S	1 1 0 0 1 0 0 1	)	1 0 1 0 0 0 1 1
INV	0 0 1 0 0 1 0 1	9	1 1 1 0 0 0 1 1
x <sup>2</sup> t	0 1 0 0 0 1 0 1	6	0 0 0 1 0 0 1 1
STO	0 1 1 0 0 1 0 1	3	1 0 0 0 0 0 1 1
RE	1 0 1 0 0 1 0 1	+/-	1 1 0 0 0 0 1 1
7	1 1 1 0 0 1 0 1	CLR	0 0 1 0 1 0 1 1
4	0 0 0 1 0 1 0 1	1/x	0 1 0 0 1 0 1 1
1	1 0 0 0 0 1 0 1	y <sup>x</sup>	0 1 1 0 1 0 1 1
0	1 1 0 0 0 1 0 1	÷	1 0 1 0 1 0 1 1
lnx	0 0 1 0 1 1 0 1	X	1 1 1 0 1 0 1 1
x <sup>2</sup>	0 1 0 0 1 1 0 1	-	0 0 0 1 1 0 1 1
RCL	0 1 1 0 1 1 0 1	+	1 0 0 0 1 0 1 1
(	1 0 1 0 1 1 0 1	=	1 1 0 0 1 0 1 1

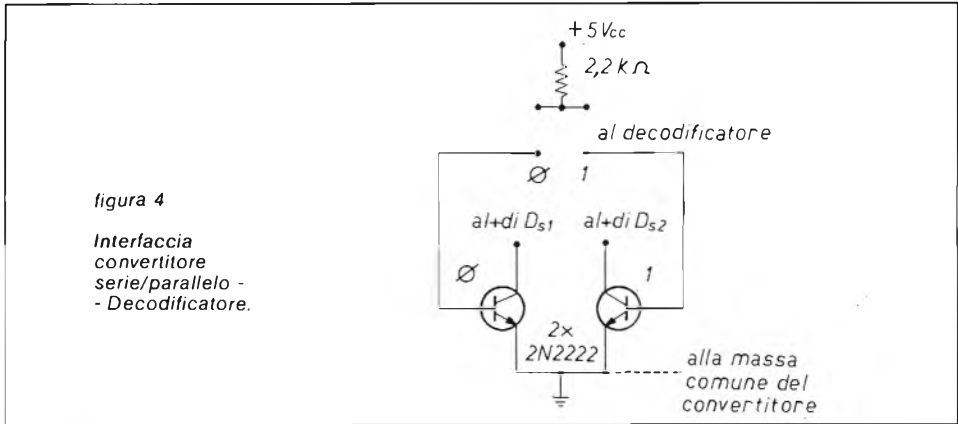
*Per le alimentazioni penso che niente di più banale possa essere disegnato, per cui non spendo nemmeno due parole per descrivere ciò che può fare un comune  $\mu$ A7805 e se ancora qualcuno non lo dovesse conoscere rimando il Lettore a trattazioni ben più approfondite apparse spesso su queste pagine.*

### *L'interfaccia convertitore/decodificatore*

*L'ingresso descritto alla figura 1B per gli stati 1 e 0, ben si presta per contatti meccanici, ma appena ci si affida, come del resto ho fatto io, a qualcosa di più sofisticato iniziano le grane.*

*Infatti il decodificatore utilizzato per questo progetto "all purpose", cioè multiuso, con l'uscita affidata a degli switch elettronici realizzati con tecnologia c-mos che garantiscono un'uscita "neutra" ma che offrono, in chiusura, una resistenza troppo alta per il circuito di ingresso adottato.*

Di qui l'esigenza della piccola interfaccia che, pur rappresentando il progetto nella sua globalità, rende però di uso più modulare i singoli blocchi logici.



## Seconda parte

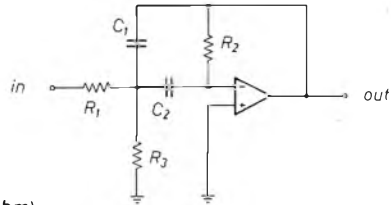
### Il decodificatore a filtri attivi

Come già spiegato in precedenza, il segnale vero e proprio registrato su nastro è costituito da gruppi di otto note che possono assumere la frequenza di  $\pm 750$  Hz (valore assegnato arbitrariamente allo stato 0) e di  $\pm 2.000$  Hz (stato logico 1).

Se ne evince che per riconoscere dette frequenze avremo bisogno di due filtri passabanda.

Detti filtri sono stati realizzati secondo lo schema di figura 5, dove è pure indicato il principio con cui essi sono stati calcolati (forse anche un po' troppo empiricamente!).

**figura 5**  
**Filtro attivo tipo Butterworth - 2° ordine.**  
**Passa-banda.**



Esempio di calcolo: (valori in hertz, farad, ohm)

$C_1$  e  $C_2$  fissati di valore uguale e rappresentati dal valore  $C$  nel calcolo

$Q$  = rapporto tra frequenza di risonanza e larghezza di banda passante a  $-3$  dB quindi

$$C = C_1 = C_2$$

$$Q = \frac{f_r}{f_b}$$

$$K = 6,28 f_r$$

$$R_1 = 10 \frac{Q}{KC}$$

$$R_2 = 2R_1$$

$$R_3 = 10 \frac{1}{2QKC}$$

(continua  
a pagina  
seguente)

Nel nostro caso per una  $f_r = 750 \text{ Hz}$

$$C = C_1 = C_2 = 10 \text{ nF} = 10^{-8}$$

$Q = 4$  (rapporto lasciato costante anche per i 2.000 Hz)

$$K = \pm 4.710$$

$$R_1 = 10 \frac{4}{4.710 \cdot 10^{-8}} = 85.000$$

$$R_2 = 2R_1 = 170.000$$

$$R_3 = 10 \frac{1}{2 \cdot 4 \cdot 4.710 \cdot 10^{-8}} = 2.650$$

Di filtri attivi un po' tutti hanno parlato e soprattutto sulle pagine di **CQ ELETTRONICA**, nomi autorevoli in questo campo hanno tenuto interessantissime disquisizioni sull'argomento per cui invito tutti gli interessati a rileggersele attentamente.

Lo schema proposto in figura 6 non è molto "a la page" ma vi posso assicurare che funziona benissimo ed è estremamente versatile dato che l'uscita sfrutta i "contatti" di un interruttore allo stato solido.

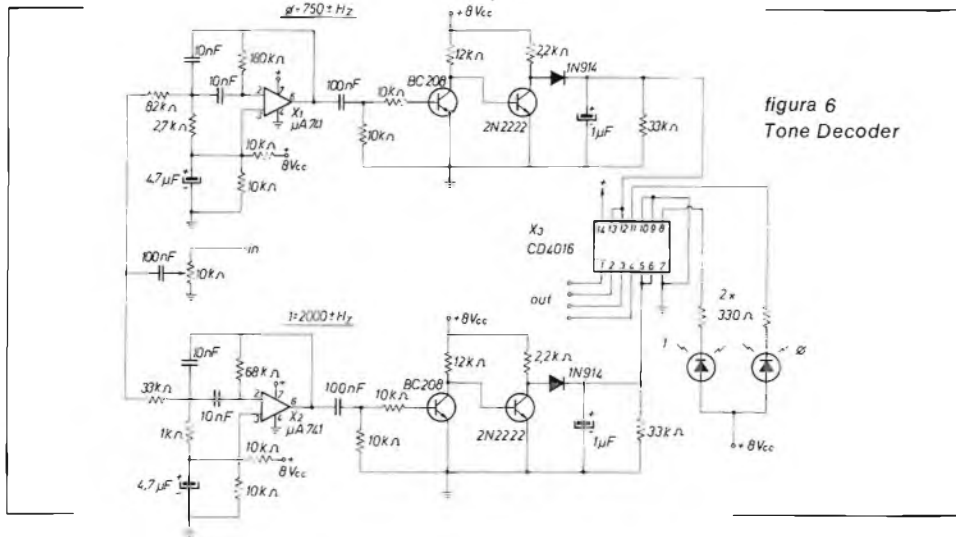


figura 6  
Tone Decoder

L'intero circuito non necessita di alcuna taratura, l'unica attenzione va alla regolazione del potenziometro da 10 kΩ all'ingresso, che deve essere regolato a un livello non troppo alto per evitare di saturare l'ingresso, e quindi attivare contemporaneamente i due switch in uscita, ma neppure troppo basso per "catturare" le eventuali variazioni di frequenza dovute al non perfetto trascinamento del nastro.

E qui siamo alla fine.

Sì, direte voi, ma questa cassetta come la registriamo?

È presto detto.

A chi ha molta pazienza posso assicurare che è in avanzata fase di progettazione il dispositivo automatico di registrazione, mentre a tutti gli altri posso consigliare (come del resto fa il sottoscritto tuttora) di munirsi di due tastini telegrafici, di un oscillografo a due tonalità (ovviamente 750 e 2.000 Hz), e di tanta pazienza ricordando che, una volta registrato, il programma non scappa più e che oltretutto non si deve lottare con la tastiera "ballerina" della '57. \*\*\*\*

---

# PENNA LUMINOSA

---

per il **sinclair**  
**Spectrum**



## 1. GRAPHIC PEN

La GRAPHIC PEN è un dispositivo che rileva segnali dal video e li trasforma in dati elaborati dallo ZX SPECTRUM.

Con un appropriato software si possono quindi produrre disegni direttamente sullo schermo video.

## 2. CONNESSIONE DELLA GRAPHIC PEN ALLO SPECTRUM

- Togliere lo spinotto di alimentazione dallo ZX SPECTRUM e inserirlo sulla scatola di controllo della GRAPHIC PEN.

- Inserire lo spinotto che esce dalla scatola di controllo all'alimentazione dello ZX SPECTRUM.

- Caricare il programma dalla cassetta contenente il programma della GRAPHIC PEN con LOAD"".

- Terminato il caricamento del programma togliere lo spinotto che dal registratore va alla presa "EAR" dello ZX SPECTRUM.

- Inserire l'apposito spinotto che esce dalla scatola di controllo nella presa "EAR" dello ZX SPECTRUM.

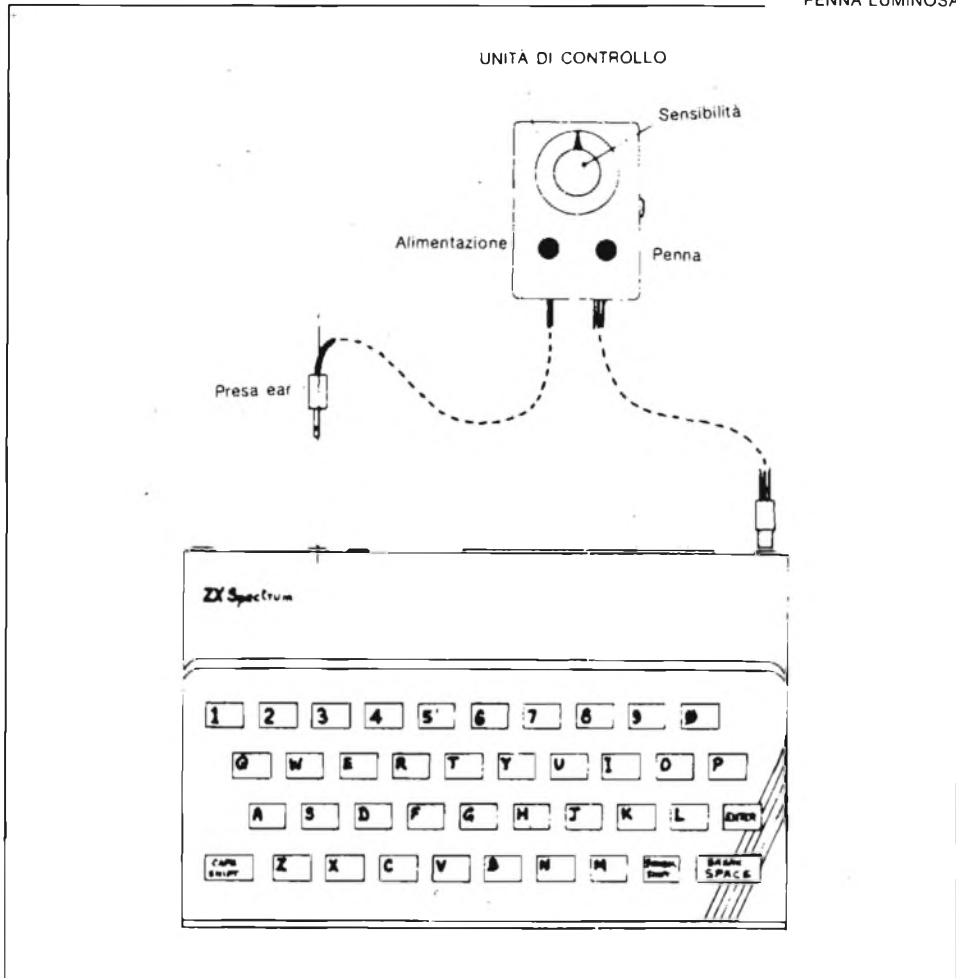
## 3. USO DELLA GRAPHIC PEN

L'uso della GRAPHIC PEN è reso molto facile dal programma caricato. Può essere usata per eseguire delle scelte (ad esempio scegliere una voce da "un menù") semplicemente puntando la GRAPHIC PEN sulla voce desiderata oppure si può usare per disegnare, semplicemente puntandola nel punto dello schermo desiderato; il computer avrà a disposizione le coordinate "x" e "y" da elaborare.

## 4. DISEGNARE CON LA GRAPHIC PEN

Nella cassetta in dotazione vi è un programma BASIC che permette di disegnare con varie funzioni sullo schermo del vostro televisore.





Nelle ultime due righe dello schermo appariranno delle lettere in "reserve" che indicano il "menù" delle varie funzioni che sono:

### **EDMCRFBIPNTKraL**

Per eseguire i disegni sullo schermo basterà puntare la GRAPHIC PEN sulla lettera che rappresenta la funzione da eseguire, quindi premere un tasto.

Le lettere presenti nel "menù" indicano le seguenti funzioni:

- |           |  |
|-----------|--|
| ERASE     | - cancella l'ultima funzione eseguita                  |
| DRAW      | - disegna una linea tra due punti                      |
| MOVE      | - permette di spostare l'origine della nostra funzione |
| CIRCLE    | - disegna un cerchio                                   |
| RECTANGLE | - disegna un rettangolo                                |
| FILL      | - riempie un'area                                      |
| HDRAW     | - disegna direttamente sullo schermo                   |
| BORDER    | - per ridefinire il colore del bordo                   |

- INK - per ridefinire il colore dell'inchiostro
- PAPER - per ridefinire il colore dello sfondo
- NEW - cancella lo schermo
- TAPE - per salvare o caricare un disegno
- KEEP - copia in una zona di memoria il tuo schermo
- RECALL - richiama lo schermo memorizzato
- ARC - disegna un arco fra tre punti
- LETTERE - mette dei testi nel video

Il punto di origine è rappresentato da una "X", questo è il punto di base per le varie funzioni.

Gli altri punti (alcune funzioni richiedono due o tre punti per essere eseguite) vengono determinati dall'incrocio degli assi (punto in cui è puntata la GRAPHIC PEN al momento in cui si spinge il tasto ENTER).

### ERASE

Questo comando cancella la linea o le forme che hai appena disegnato. Per usare questa funzione basta puntare la GRAPHIC PEN sulla **E** e premere un tasto.

### DRAW

Questa funzione disegna una linea dall'origine all'incrocio degli assi.

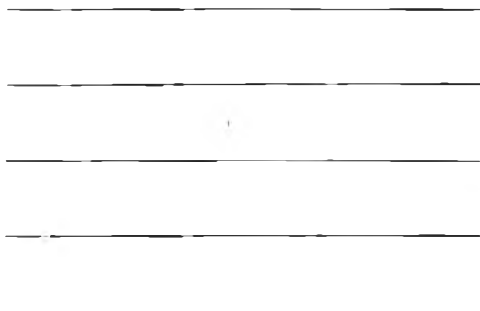
A - fissa una fine della linea posizionando l'origine (X) al punto desiderato (usa il comando MOVE).

B - fissa l'altra fine della linea puntando la penna al punto desiderato e premendo un tasto.

C - punta la GRAPHIC PEN nel **D** e premi un tasto.

### MOVE

Questo comando muove l'origine (X). Usa questo comando per muovere l'origine attraverso il video. Per usare MOVE, puntare semplicemente la GRAPHIC PEN al **M** e premere un tasto.



**CIRCLE**

Ci sono tre cose da fare per disegnare un cerchio:

- A - fissare il centro del cerchio posizionando la origine (X) al punto voluto (usare MOVE).
- B - scegliere il raggio del cerchio muovendo la GRAPHIC PEN e premere un tasto.
- C - puntare la GRAPHIC PEN nel **C** e premere un tasto.

**RECTANGLE**

Ci sono tre cose da fare per disegnare un rettangolo.

- A - muovere l'origine (X) dove sarà un angolo (usare il MOVE).
- B - determinare la posizione dell'angolo opposto del rettangolo.
- C - puntare la GRAPHIC PEN sulla **R** e premere un tasto.

**FILL**

Serve per riempire delle aree.

- A - piazzare il bersaglio dentro alla figura da riempire.
- B - se necessario usare il comando INK per selezionare il colore desiderato.
- C - puntare la GRAPHIC PEN sulla **F** e premere un tasto.

**HDRAW**

Abilita a disegnare a mano libera. La GRAPHIC PEN scriverà man mano tu attraverserai il video.

Puntare la GRAPHIC PEN sulla **H** e premere un tasto. Poi puntare la GRAPHIC PEN nel punto che vuoi per iniziare a disegnare, premere un tasto.

**BORDER INK PAPER**

Questi tre comandi sono molto simili fra loro. Ti aiutano a cambiare i colori del bordo, dell'interno e delle scritte. Per cancellare lo schermo usare il tasto "NEW".

**TAPE**

Una volta scelto TAPE puntando la GRAPHIC PEN alla **T**, sarai in grado di scegliere fra SAVE o LOAD.

In ogni caso ti sarà chiesto di sostituire il nome da usare nel SAVE o LOAD.

**KEEP**

**(è possibile usarlo solamente con 48 K)**

Questa funzione fa copiare lo schermo nella memoria. Questa poi può essere istantaneamente richiamata usando il RECALL. Prende lo schermo attraverso lo spazio accessibile del BASIC (su RAMTOP vedi pag. 179 del Manuale SPECTRUM di Basic).

Lo SPECTRUM 48K può tenere in memoria fino a 5 schermi pieni una volta che il BASIC sarà stato cancellato col NEW.

Per usarlo puntare la penna sul **K** e premere un tasto.

**RECALL**

Questo comando permette di richiamare istantaneamente una copia del video dalla memoria.

**ARC**

Questa funzione ha bisogno di tre punti per essere definita.

- A - l'origine (X) determina uno dei due estremi dell'arco
- B - determina il secondo punto per cui l'arco deve passare

- C - determina il terzo punto, oppure l'altro estremo dell'arco
- D - puntare la GRAPHIC PEN sulla lettera **A** e premere un tasto

## 5. LETTERE

Questa funzione ti permette di inserire dei testi sullo schermo.

- A - determina il punto di inizio del testo usando il comando MOVE
- B -puntare la GRAPHIC PEN sulla lettera **L** e premere un tasto
- C - scrivere il testo e premere ENTER

## 6. USO DELLA ROUTINE IN CODICE MACCHINA DELLA GRAPHIC PEN

Il programma in codice macchina della GRAPHIC PEN può essere usato per determinare dei punti sullo schermo TV e farli elaborare da qualsiasi vostro programma. Per eseguire il LOAD di queste routine occorre fare quanto segue:

```
48k CLEAR 60799
LOAD"" CODE
16k CLEAR 28199
LOAD"" CODE
```

Per eseguire il SAVE:

```
48k SAVE"" CODE 60800,3470
16k SAVE"" CODE 28200,3470
```

Per eseguire questa routine usare i seguenti comandi:

```
48k RANDOMIZE USR 60800
16k RANDOMIZE USR 28200
```

oppure:

```
48k RANDOMIZE USR 60806
16k RANDOMIZE USR 28206
```

Per non ridefinire i colori BORDER, PAPER, INK.

Prima di fare partire la routine occorre inizializzare il programma con:

```
48k CLEAR 60799
16k CLEAR 28199
```

Facendo successivamente eseguire le routine in codice macchina.

Le coordinate del punto della GRAPHIC PEN sullo schermo vengono determinate con:

```
48k LET A = USR 63582
16k LET A = USR 30982
```

Scomponendo poi nei valori X e Y:

```
LET Y = INT (A/256)
LET X = A-256*Y
```

**N.B.** – Quando useremo le sole routine in linguaggio macchina nella versione 48k, i byte e la routine del "parlato" saranno tralasciati.

*Per qualunque ulteriore informazione:*

 **NUOVA NEWEL** s.a.s.  
di Ciampitti A. & C.  
MATERIALE ELETTRONICO  
Milano - Via Duprè, 5 - tel. 32.70.226

# Conosciamo il NewBrain

## HARDWARE

### Lo Z80A

Il più potente microprocessore a 8 bit disponibile, con il vantaggio di poter utilizzare sul NewBrain tutto il software già sviluppato e facilmente reperibile.

### RAM di 32K

Tutti i modelli hanno come standard 32K di memoria RAM. Collegando uno schermo video di 80 colonne, restano 28K a disposizione per i vostri programmi e i dati. Se scegliete uno schermo di 40 colonne, lo spazio disponibile aumenta a oltre 30K.

### ROM di 29K

Serve a contenere il software di sistema, che include il BASIC, il sistema operativo, il package matematico, quello per la grafica e lo Screen Editor.

### Tastiera

La tastiera del NewBrain è simile a quella di una normale macchina da scrivere, progettata per soddisfare le esigenze di una dattilografa provetta, così come per essere facilmente utilizzata da parte del principiante.

### Visore incorporato (modello AD)

Il display fluorescente incorporato nel NewBrain visualizza lettere di notevole dimensione, facilmente leggibili. La sua posizione permette un ampio angolo di visuale e il particolare colore verde garantisce di non affaticare la vista.

### Doppio registratore di cassette

Il NewBrain può collegare due registratori di cassette, in modo da rendere semplice l'aggiornamento e la copia dei vostri archivi. Il trascinamento del nastro viene controllato da calcolatore e i dati sono trasferiti a 1200 baud.

### Presca per televisore

Tramite questa presa il NewBrain è in grado di utilizzare un normale apparecchio televisivo.

### Presca per video

Il NewBrain è dotato anche di una uscita per video a larga banda. Collegando alla stessa un normale monitor video, la qualità dell'immagine risulta particolarmente buona.

### Comunicazioni

Esistono due uscite per comunicazioni, entrambe pilotate da software.

1. RS232/V24: bidirezionale, con velocità di trasferimento selezionabile da software, per il collegamento ad altri NewBrain, a calcolatori diversi, a periferiche, modem, terminali video, ecc.
2. RS232/V24: monodirezionale, utilizzabile per il collegamento di una stampante.

### Connettore per espansioni

Permette di ampliare la configurazione del NewBrain.

## SOFTWARE


Il NewBrain è dotato di un corredo di software veramente notevole: il compilatore BASIC, ampliato rispetto allo standard ANSI; un potente sistema operativo; il Package matematico in virgola mobile (dieci cifre significative); lo Screen Editor per il controllo del cursore e la gestione dello schermo e della riga; il Package grafico, che consente di disegnare linee e archi con facilità, come pure di eseguire complesse funzioni di tracciamento.

Tutto questo software è residente nella memoria ROM fornita con entrambi i modelli.

Utilizzando il modulo di espansione ROM, è disponibile, in aggiunta o in sostituzione, altro software: l'Assembler Z80, il linguaggio COMAL, il Package statistico e il Package per il trattamento dei testi.

Passando a configurazioni con disco il vostro NewBrain è in grado di sfruttare tutto il software applicativo e i linguaggi disponibili nell'ambito del ben noto e diffuso sistema operativo CP/M®.

®CP/M è marchio registrato della Digital Research Inc.

(foto alle pagine seguenti) 

Attacchi per due registratori di cassette Per memorizzare dati e programmi.

Controlli del livello di uscita su cassetta

Attacco per monitor video standard.

Attacco per Modulo di Espansione.

Attacco RS232/V24. Per terminali video, modem, ecc.

Attacco per televisore domestico.

Display con capacità di 16 caratteri, in grado di rappresentare i 64 caratteri ASCII/ISO ottenibili con il tasto maiuscole premuto.

Attacco RS232/V24 per stampante.

Attacco Alimentatore



*NewBrain*

Contenitore in ABS

Tasto Control. Permette la generazione dei 32 codici di controllo ASCII/ISO.

Tasto Shift. Ha la stessa funzione del tasto maiuscole su una normale macchina da scrivere, permettendo la generazione di tutti i 96 caratteri di stampa ASCII/ISO. La funzione fissa maiuscole può essere eseguita da software.

Tasto di ripetizione.

Tasto Graphics. Permette la generazione dei 64 simboli grafici viewdata, o altri caratteri grafici, secondo il modo d'uso.

Tasto Video text, usato insieme con il modulo opzionale video text

Tasti controllo cursore. Se tenuti premuti, ripetono automaticamente la funzione.

Le dimensioni di XÉLECTRON ci hanno impedito di riprodurre il NewBrain in dimensioni reali, come indica invece il display della foto originale. La lunghezza del NewBrain è di 27 cm.





# Tutto quello che avete sempre desiderato sapere sul G5 ...e che avete fin troppo osato chiedere

---

*ing. Gianni Becattini*

---

## **QUALCOSA DI PIÙ SULLA STRUTTURA INTERNA**

*Il notevole interesse riscosso dal microcomputer G5 a suo tempo presentato su **CQ ELETTRONICA** [1] ha fatto sì che in molti insorgesse il desiderio di sapere qualcosa di più sulla sua struttura interna.*

*È quindi opportuno procedere a qualche chiarimento.*

Sull'hardware poco da dire: già abbondantemente descritto su **CQ ELETTRONICA** [1 e 2] il circuito è risultato particolarmente indovinato e gli stampati ben fatti, per cui il 90% dei kit forniti hanno funzionato alla prima senza difficoltà mentre nel restante dieci per cento si sono trovati praticamente solo quelli montati con scarsa precisione.

Sulla parte firmware, ossia sull'interprete GBASIC, invece, il discorso è ben diverso, essendo legate soprattutto ad esso le notevoli prestazioni del G5 ed essendo stati concentrati in esso la maggior parte dei miei sforzi.

### **Struttura del GBASIC**

A voler descrivere il GBASIC [3, 4] nel dettaglio dal punto di vista della struttura, anche se può sembrare una frase fatta, non sarebbero sufficienti vari volumi. Dovremo pertanto accontentarci di una descrizione abbastanza di massima limitata a quelli che possono essere gli interessi del normale sperimentatore.

Il GBASIC è contenuto su una memoria ROM (una EPROM, in pratica, per essere più facilmente aggiornabile), che risiede nei primi 16K dello spazio indirizzabile dal G5 e inizia dalla locazione 0000; non sono quindi necessari particolari circuiti per il reset iniziale.

È composto da tre moduli, denominati rispettivamente BASIC, l'interprete vero e proprio, VDR, il Video Driver che provvede al controllo della parte video e all'interfacciamento di tastiera e stampante, e l'ACAS ossia la parte destinata al controllo del registratore audio.

La separazione tra i vari moduli è molto netta: essi colloquiano tramite alcune routines, denominate a seconda del punto di vista PUBBLICHE (dall'esterno del modulo) o ESTERNE (se viste da altri moduli) e attraverso i dati contenuti nella memoria RAM.

### **Organizzazione della memoria RAM**

Il GBASIC è stato progettato in modo da potersi facilmente adattare senza interventi circuitali, come la modifica di ponticelli o sostituzione delle ROM, a varie dimensioni della memoria RAM. Questo adattamento a sfruttare al meglio la memoria disponibile avviene in modo del tutto automatico grazie a una procedura (detta LOKRAM) lanciata a ogni reset, e alla organizzazione della memoria RAM.

La procedura LOKRAM "cerca" la fine della RAM per mezzo di scritture e letture successive [3] così da identificare l'ultima locazione disponibile, detta INFO4. Qualora si desiderasse tuttavia limitare per qualsiasi motivo l'occupazione di RAM del GBASIC, basterà usare la frase BLANK n, dove n è il valore della più alta locazione che intendiamo lasciare al GBASIC. Questa limitazione resterà operativa fino al successivo reset [4].

I programmi risiedono in RAM subito dopo le prime circa mille locazioni riservate ad uso interno del GBASIC a partire dalla locazione INFO0 e vanno a salire fino alla locazione INFO1 che sale e scende in funzione della lunghezza del programma stesso.

Le variabili sono invece memorizzate a scendere da INFO4. Il limite inferiore del loro spazio è dato da INFO3. Ovviamente INFO3 si sposta verso il basso via via che vengano definite nuove variabili.

Le stringhe e le variabili numeriche con indice occupano invece lo spazio che sta sopra ai programmi, a crescere, fino alla locazione INFO2.

La condizione di memoria piena si ha quando INFO2 (o INFO1 se non vengono usate le variabili con indice) arriva a interferire con INFO3.

Con il metodo ora descritto si ha il migliore sfruttamento della RAM, in quanto avviene la suddivisione automatica programmi/dati; si possono quindi inserire programmi lunghi con poche variabili o viceversa.

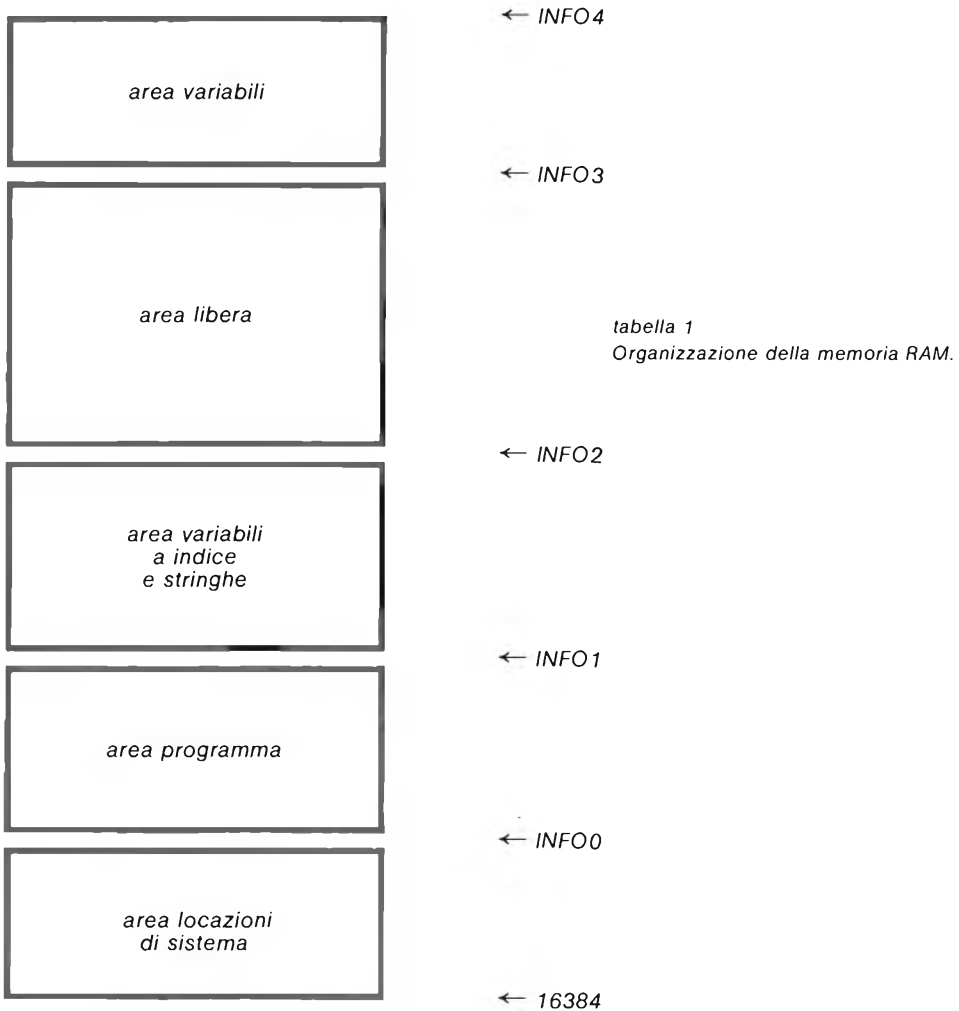
---



---

**NON PERDERE IL N. 12 DI  
"CQ ELETTRONICA" CONTIENE LA  
NUOVA IMMAGINE DELLA RIVISTA DI  
ELETTRONICA A PIÙ ALTO  
CONTENUTO DI INFORMAZIONE**

La sottostante tabella riassume quanto esposto.



Le varie locazioni INFO1, INFO2, ecc. possono essere facilmente lette dal programma mediante la funzione INFO(N) dove N è il numero della INFO desiderata.

Ad esempio, INFO(3) fornisce il valore di INFO3.

### Formato dei numeri

Il GBASIC rappresenta internamente i numeri mediante il codice BCD, ossia con due cifre per ogni byte. Questo metodo evita gli errori interni di arrotondamento durante le conversioni decimale/binario e viceversa, anche se comporta un maggiore spreco di memoria e una più bassa velocità di calcolo.

Tutti i numeri vengono rappresentati con la stessa precisione, che è di 10 cifre significative di mantissa ed esponente compreso tra -128 e +127.

Ogni numero occupa 7 bytes: 5 per la mantissa, 1 per il segno e 1 per l'esponente secondo il formato descritto dal seguente esempio:

Numero rappresentato: 123

byte	contenuto (hex)	note
N	12	<i>mantissa (0.1230000000)</i>
N+1	30	<i>mantissa (0.1230000000)</i>
N+2	00	<i>mantissa (0.1230000000)</i>
N+3	00	<i>mantissa (0.1230000000)</i>
N+4	00	<i>mantissa (0.1230000000)</i>
N+5	00	<i>segno (01 = meno)</i>
N+6	83	<i>esponente nella forma 128+E</i>

Essendo:  
 123 = . 123 x 10 alla 3  
 l'esponente vale 131 = 83H

Pur essendo disposti verso l'alto, ossia a crescere, i numeri sono sempre riferiti da tutte le routine interne a partire dalla locazione più alta. Il numero dell'esempio precedente si identifica quindi internamente come all'indirizzo N+6 e non all'indirizzo N.

### Le variabili

L'area riservata alle variabili contiene gli identificatori e i contenuti delle variabili numeriche scalari e gli identificatori con i puntatori relativi alle variabili numeriche con indice. In altre parole i contenuti variabili numeriche sono memorizzati in questa area assieme al nome (ad esempio A, C1) mentre per le variabili stringa o a indice si memorizza solo il nome e l'indirizzo di dove poi i valori numerici sono effettivamente memorizzati. Questi ultimi, come già spiegato in precedenza, si trovano nell'apposita area tra INFO1 e INFO2.

Le variabili scalari, cioè senza indice, sono memorizzate come segue:

byte	contenuto
da N a N+6	<i>il numero contenuto nella variabile</i>
N+7	<i>eventuale parte numerica del nome (ad esempio C1); se la variabile è composta di una sola lettera, allora N+7 vale 00;</i>
N+8	<i>parte alfabetica del nome</i>

Anche le variabili si identificano con l'indirizzo del byte più alto. La variabile precedente, ad esempio, ha indirizzo N+8.

Le variabili a indice occupano sempre 9 bytes nell'area variabili ma i significati dei vari bytes sono diversi.

byte	contenuto
da N a N+2	<i>non usati</i>
N+3	<i>semi-byte meno significativo dell'indirizzo a cui è memorizzata la variabile;</i>
N+4	<i>come sopra, più significativo;</i>
N+5	<i>numero massimo degli elementi che compongono il vettore (poiché è presente anche l'elemento 0, è uguale al numero di elementi stabilito nel DIM più uno);</i>
N+6	<i>riservato per usi futuri</i>
N+7 e N+8	<i>come nelle variabili scalari; l'unica differenza è data dal bit più significativo di N7, sempre a uno per indicare variabile con indice.</i>

L'indirizzo dato da  $N+3$  e  $N+4$  è detto VETTADDR, punta alla locazione di memoria ove inizia la memorizzazione del primo elemento della variabile multipla (al byte meno significativo, cioè  $N$ , non  $N+6$ ). Gli elementi successivi sono memorizzati di seguito, a salire:

indirizzo	contenuto
VETTADDR	→ elemento 0, byte $N$
.....	
VETTADDR+6	→ elemento 0, byte $N+6$
.....	
VETTADDR+7	→ elemento 1, byte $N$
.....	ecc.

Le variabili stringa sono memorizzate in forma molto simile a quella delle variabili a indice, come illustrato sotto:

byte	contenuto
da $N$ a $N+2$	non usati
$N+3$	semi-byte meno significativo dell'indirizzo a cui è memorizzata la variabile;
$N+4$	come sopra, più significativo;
$N+5$	numero massimo dei caratteri che compongono la stringa (fissati con la frase DIM) più uno (il byte in più è riservato ad uso interno);
$N+6$	riservato per usi futuri;
$N+7$ e $N+8$	come nelle variabili a indice; in questo caso anche il bit più significativo di $N+8$ è posto a uno per indicare variabile stringa.

L'indirizzo dato da  $N+3$  e  $N+4$  è detto sempre VETTADDR e punta alla locazione di memoria ove inizia la memorizzazione del primo elemento della variabile stringa.

Gli elementi successivi sono memorizzati di seguito, a salire.

Ad esempio, nella parola GATTO, VETTADDR punta alla lettera G.

indirizzo:	contenuto:
VETTADDR	→ primo carattere della stringa
.....	
VETTADDR+ $K-1$ ( $K$ è la lunghezza della stringa)	→ ultimo elemento della stringa
.....	
VETTADDR+ $K$	→ byte per uso interno (è sempre uguale a zero).

Gli ultimi bytes non usati dalle variabili stringa sono sempre posti uguali a 00 dai vari programmi. Il rilevamento di 00 nella lettura della stringa può quindi essere utilizzato come segnalazione di fine dati validi.

tabella 2

I codici di tutte le parole-chiave del GBASIC release 3.00.

parola	codice (hex)	parola	codice (hex)
INK	02	VSET	03
TEXT	04	GRAPH	05
BLANK	06	NOBLANK	07
ORG	08	RESET	09
PLOT	0A	UNPLOT	0B
VRAM	0C	SPDEP	0D
SPCOL	0E	SPMOV	0F
SPFORM	100	SPDEF	11
SPTYP	12	AT	13
SAVE	14	LOAD	15
VERIFY	16	ONERROR	17
ARG	18	LET	80
NEXT	81	IF	82
GOTO	83	GOSUB	84
RETURN	85	READ	86
DATA	87	FOR	88
LPRINT	89	INPUT	8A
DIM	8B	STOP	8C
END	8D	RESTORE	8E
REM	8F	CLEAR	90
POKE	92	OUT	93
ON	94	PRINT	97
RUN	A0	LLIST	A1
ERASE	A3	LIST	AA
(	E0	ABS	C1
*	E2	+	E3
SQR	C4	-	E5
INT	C5	/	E7
- (unario)	E9	>=	EA
<=	EB	CALL	CD
RND	CE	=>	EF
=<	F0	<>	F1
SGN	D2	SIN	D3
<	F4	=	F5
>	F5	TAN	D7
COS	DB	VAL	D9
PEEK	DA	INP	DB
SPCOIN	DE	PVRAM	DE
INFO	DF		

**Calcolo della occupazione di memoria**

Un programma occupa il seguente spazio:

1) pel programma (bello, eh, il "pel"): tanti bytes quante sono le battute, tenuto conto però che le parole chiave occupano un solo byte + numero delle linee moltiplicato 4;

2) per le variabili scalari: numero delle variabili moltiplicato 9;

3) per le variabili a indice (per ciascuna):

$$(N+1) \times 7 + 9$$

dove N è il valore dell'elemento massimo. Si ricordi che le variabili a indice occupano spazio fin dal momento del DIM e che se non viene specificata una frase DIM ma vengono utilizzate ugualmente variabili multiple, queste sono dimensionate automaticamente al valore 100 (11 elementi). Per una minore occupazione di memoria, quando servano meno elementi di 11, converrà pertanto dimensionare la variabile esattamente al valore voluto.



4) per le variabili stringa (per ciascuna):

$N+10$

dove  $N$  è il valore fissato nella frase DIM. Le stringhe non hanno dimensionamento automatico. Una volta dimensionate, occupano spazio anche se usate solo parzialmente.

### Conclusione

Spero di essere stato sufficientemente chiaro e di avere esaurito almeno la maggior parte delle vostre domande. Ulteriori informazioni potranno essere reperite nella **Bibliografia** [5, 6]

### Bibliografia

- [1] - Becattini - **G5 - CQ ELETTRONICA** n. 1 e 2/1983.
- [2] - Becattini - **Informazioni complementari sul G5 - CQ ELETTRONICA** n. 4/1983.
- [3] - Becattini - **IL COMPUTER È FACILE, PROGRAMMIAMOLO INSIEME** - edizioni CD, Bologna, aprile 1983.
- [4] - Becattini - **Il GBASIC release 3.00**, su questo stesso **XÉLECTRON**.
- [5] - Becattini - **Il G5 e il linguaggio macchina**, su questo stesso **XÉLECTRON**.
- [6] - TEXAS INSTRUMENTS, **TMS 9918/9929 User's Manual**.

---

## Attesissimo IL RELEASE 3.00 del GBASIC

*Una nuova versione del software di base per il microcomputer G5 presentato su **CQ ELETTRONICA** consente di espandere notevolmente le sue già brillanti prestazioni con la sola riprogrammazione delle memorie EPROM.*

Come ho già avuto occasione di esprimere su queste pagine, l'interesse suscitato dalla presentazione del microcomputer G5 ha superato tutte le mie stesse aspettative: ai primi di Agosto erano già stati consegnati più di cento kit e altri quaranta erano in ordine.

Molte delle prestazioni che hanno decretato il notevole successo del G5 sono dovute al GBASIC, l'interprete BASIC presente sulle memorie EPROM da me creato appositamente allo scopo. Oggi sono lieto di presentare l'ulteriore miglioramento del GBASIC 3.00 che molti attendevano già da tempo, in perfetto accordo con le previsioni a suo tempo effettuate.

Riepiloghiamo le tappe fino a qui percorse:

- 1) Gennaio 83 - Viene presentato il G5 con il GBASIC release 1.0.
- 2) Aprile 83 - Iniziano le consegne dei kit e già viene fornito il release 2.0 con notevoli miglioramenti.
- 3) Aprile 83 - Esce il libro "**IL COMPUTER È FACILE**" pubblicato dalle edizioni CD, un testo non solo specifico per il G5 ma anche di validità del tutto generale.

Sacrificando una parte delle vacanze, oltre al release 3.00 oggetto di questo articolo, ho preparato la serie di mini articoli relativi al G5 che trovate su questo stesso **XÉLECTRON**.

## Evoluzione del GBASIC

Rispetto alla versione 2.1 del GBASIC, l'unica consegnata, è stata completamente amputata la sezione relativa alla cassetta digitale, a grande richiesta, soprattutto per l'assurdo costo della cassetta digitale, lievitato per il rincaro del dollaro.

Elencherò quindi nel seguito tutte le differenze tra il release 1.0 (descritto nel libro) e il release 3.00 che è l'attuale.

## Innovazioni sostanziali

La capacità aritmetica è stata estesa da 6 a 10 cifre significative, sempre con esponente +127...-128. A seguito di ciò è stata rifatta l'aritmetica della parte grafica, non più in virgola mobile ma ad interi, in modo da accelerare assai il tracciamento di linee. Inoltre sono state aggiunte le stringhe, anche se per adesso in versione non completissima ma pur sempre sufficiente alla maggior parte degli usi. Sono state inoltre inserite delle facilità per l'uso del linguaggio macchina. Possono essere inserite più frasi per ogni riga separandole con i due punti “;”.

## Eliminati

Sono state eliminate la frase PUSH e la funzione POP, non necessarie. Sono stati inoltre eliminati gli errori dalle funzioni RND e SGN che ora operano correttamente. La funzionalità di RND() può essere visualizzata facilmente mediante l'esecuzione del programma ARZPL, descritto su **CQ ELETTRONICA** numero 2/83. È stata inoltre “sbloccata” la funzione CTRL/C che consente ora di interrompere un programma in esecuzione senza passare dal reset. CTRL/U provoca la cancellazione della linea in corso.

## Dialogo iniziale

Il dialogo iniziale è stato modificato così:

*(soliti messaggi)*

*Per programma nuovo batti N →*

Qualunque lettera si batta che non sia N si assume programma vecchio. Questo per evitare pericolose confusioni che potrebbero portare alla cancellazione accidentale del programma. Attenzione però! Almeno una volta all'inizio è necessario battere N o si incorrerà in ERRORI DI FUORI MEMORIA a seguito della non corretta inizializzazione dell'area programma.

## Tastiera

Per solito le tastiere ASCII hanno i numeri in posizione “minuscolo”, cioè per generarli non si deve premere il tasto SHIFT. Questo è molto scomodo per l'uso del GBASIC che invece accetta solo parole chiave in maiuscolo, costringendo un continuo lavoro di maiuscolo/minuscolo. Ho eliminato questo inconveniente aggiungendo un comando riconosciuto a livello di driver tastiera: al reset il GBASIC si predispose per convertire automaticamente le lettere minuscole in maiuscole (lasciando inalterati tutti gli altri codici). Battendo CTRL/D si disabilita questa caratteristica e le minuscole vengono lasciate inalterate fino al successivo CTRL/D. Sperimentate subito l'utilità di questo nuovo comando.

## Messaggi di errore

Nelle precedenti versioni, nel modo GRAPH, risultavano troppo lunghi i messaggi di errore che “strabordavano” quindi dallo schermo. Adesso i

messaggi di errore hanno la seguente forma:

*ERR tipo LN numero*

ad esempio:

*ERR DI LINGUAGGIO LN 100*

La chiarezza è sempre sufficiente e l'inconveniente è eliminato.

### **Nuovi comandi**

**SAVE** - scrive su cassetta il programma GBASIC contenuto in memoria. La cassetta, un qualsiasi registratore di qualità decente, si interfaccia in modo molto semplice.

**NOTA** - per tutto ciò che riguarda il registratore vedi, per maggiori dettagli, la "G5 - Interfaccia per registratore" su questo stesso **XÉLECTRON**.

**LOAD** - carica da cassetta il primo programma che viene trovato a partire dalla prima locazione dell'area programmi. Eventuali errori di lettura vengono segnalati.

**VERIFY** - consente di verificare un nastro, dopo la registrazione, in modo da essere certi che sia leggibile prima di spegnere il G5 (se non si ha la memoria continua). Il comando VERIFY controlla solo la leggibilità del nastro, senza riguardo alcuno per i suoi contenuti.

**SAVE exp1, exp2** - scrive su cassetta il contenuto delle locazioni di memoria assolute da exp1 a exp2.

Esempio:

*SAVE 20000,20768*

scrive su nastro il contenuto delle locazioni dalla numero 20000 alla 20768.

**LOAD exp1** - carica da cassetta il programma a partire dalla locazione specificata da exp1. Non può essere usato per programmi in GBASIC ma solo per programmi in linguaggio macchina.

**VERIFY exp1** - varia la velocità di trasferimento delle informazioni da e per il registratore. Exp1 può assumere valori tra 0 e 255: la velocità standard è 20. Valori minori corrispondono a velocità più alte e viceversa. In questo modo l'utente può sfruttare al meglio il registratore da lui posseduto.

**LIST exp1, exp2** - lista limitata: parte dalla linea exp1 e si arresta dopo exp2 linee.

Esempio:

*LIST 1000,10*

significa "lista dieci linee a partire dalla 1000".

Il comando LIST senza argomenti lista ora le prime 22 linee di programma. LIST nn, 255 provoca invece la lista di tutto il programma. Quanto detto per la frase LIST vale anche per LLIST.

### **Nuove frasi**

**ONERROR numlin** - In caso di errore, anziché aversi una segnalazione, si salta alla frase numlin. In questo modo anche la condizione di errore può essere gestita dal programma.

Esempio:

*10 ONERROR 30  
15 BUMBA  
30 PRINT "ERRORE"*

*RUN  
ERRORE*

**PLOT** - è stata aggiunta la possibilità di una indefinita ripetizione del tracciamento, a partire dal punto di arrivo fino a un nuovo punto e così di seguito. Esempio:

*PLOT 10,20,30,60,46,87,60,20*

traccia una linea dal punto 10,20 al punto 30,60; da qui al 46,87, da qui al 60,20.

**PAPER & INK** - queste due nuove istruzioni consentono di scegliere il colore del tracciamento e del fondo in modo grafico e agiscono sulle frasi PLOT, SET e sulle scritte. Basta indicare il colore scelto (dalla tabella che sta sul libro).

Esempio:

*10 PAPER 5  
20 INK 15  
30 PLOT 20,30,40,50*

In questo modo viene scelto il colore 5 per il fondo e 15 per il tratto.

Poiché PAPER e INK influiscono anche sulle scritte, è possibile l'inversione del contrasto.

Si tenga presente che lo schermo è diviso, anche ai fini della grafica, in 32x24 quadratini di 8x8 punti. In ciascuna delle 8 righe di ogni quadratino è possibile una unica definizione di colori per il fondo e per il tratto. Le combinazioni di colore non sono quindi illimitate. Questa limitazione è presente in quasi tutti i microcomputer commerciali. Anche lo Spectrum della Sinclair non fa di meglio, anzi in esso la limitazione è ancora maggiore in quanto la scelta di fondo e di colore deve avvenire per interi quadratini e non sulle linee e come nel G5 (se non avete capito niente, basta che proviate a tracciare linee in diversi colori che si intersechino tra loro).

**TEXT exp1, exp2** - anche la frase TEXT consente ora la gestione del colore. Esp1 controlla infatti il colore del fondo e del bordo dello schermo (sempre scuro nella versione precedente) mentre esp2 quello della scrittura. Il bordo dello schermo si conserva anche nel modo grafico, dando luogo a una specie di cornice. La frase TEXT seguita da esp1, esp2 ha il solo scopo di mutare i parametri indicati e può essere usata anche nel modo GRAPH al solo scopo di mutare il colore del bordo, mentre senza argomenti conserva il solito significato.

Esempio:

*TEXT 0,15*

stabilisce il colore del fondo e del bordo in 0 e quello per le scritte in 15.

**SPDEF exp1, exp2 [exp3]** - su suggerimento di molti, è stata aggiunta la possibilità di definire la forma dello sprite in modo numerico. Esp1 e esp2 hanno il solito significato, mentre esp3 è la definizione del rigo in forma decimale. Esp3 viene convertita in binario e gli uni hanno significato di punto acceso mentre gli zeri di punto spento.

Esempio:

*SPDEF 0,0,11*

definisce il rigo 0 dello sprite 0 come 00001011 (=11 decimale).

**BLANK exp1** - la frase BLANK usata con un argomento exp1 assume un significato del tutto diverso dall'originale. Serve infatti per limitare la memoria RAM usata dal G5 al valore specificato da exp1, ad esempio per riservare spazio a programmi in linguaggio macchina. Al reset viene comunque sempre ripristinato il valore massimo disponibile.

Esempio:

*BLANK 6000*

limita l'uso della RAM alla locazione 6000. Viene richiesta la domanda iniziale. Non vengono eseguiti controlli, quindi attenzione a non "sbassare" troppo (il limite minimo è fornito dalla funzione INFO(0) più qualcosa, vedi sotto).

**ON exp1 GOTO lista** - classica "GOTO calcolato" del BASIC. Serve per saltare a una linea scelta in una lista secondo quanto specificato da exp1.

Esempio:

```
ON N GOTO 20,30,40,50
```

se N vale 0 si salta alla linea 20, se vale 1 si salta alla 30, se 2 alla 40, se 3 alla 50. Se più di 3 si ha la segnalazione ERR DI ARGOMENTO.

La frase ON è molto comoda in particolare nei menù per selezionare una scelta tra più.

**ON exp2 GOSUB lista** - simile alla precedente ma per il salto a sottoprogrammi.

### Funzioni

**VAL(stringa)** - ritorna il valore numerico di una stringa.

Esempio:

```
PRINT VAL("123.4")
123.4
```

Grazie a VAL, è possibile utilizzare nei calcoli un valore inizialmente assegnato come stringa.

Esempio:

```
LET A=VAL(A$)/2
```

**STR\$(espressione)** - converte in stringa il risultato della espressione contenuta in parentesi. È la funzione contraria di VAL().

**CHR\$(espressione)** - ritorna una stringa di lunghezza uno contenente il carattere specificato, in codice ASCII, dalla espressione. È del tutto simile alla CHR() già descritta nel libro, ma può essere usata come normale stringa. CHR() è stata conservata solo per compatibilità con le versioni precedenti.

**LEN(stringa)** - ritorna la lunghezza effettiva della stringa specificata (vedi dopo).

**INFO(exp1)** - questa è una funzione multiuso che fornisce una serie di informazioni su parametri di sistema. A seconda del valore di exp1 fornisce:

**Exp1=0** - valore della prima locazione a partire dalla quale vengono memorizzati i programmi.

**Exp1=1** - valore della ultima locazione usata dai programmi.

**Exp1=2** - valore della più alta locazione usata dalle matrici e dalle stringhe (che si trovano a valle del programma).

**Exp1=3** - valore della più bassa locazione di memoria usata dalle variabili. Le variabili partono dalla locazione più alta disponibile e vanno a scendere. Sulla base di quanto detto è ovvio che per conoscere la memoria libera basta fare

```
PRINT INFO(3)-INFO(2)
```

**Exp1=4** - valore della più alta locazione di RAM disponibile nel sistema. Viene "cercata" automaticamente dal GBASIC al reset come spiegato sul numero 4 di **CQ ELETTRONICA**. Evita il PEEK. Questo valore può essere variato con la frase BLANK (vedi sopra).

**Exp1=5** - valore della versione di GBASIC. Nella 3.00 ritorna il valore 300. In questo modo il programma può essere informato del Release di firmware disponibile e sapere se usare o meno certe funzioni.

**Exp1=6** - indirizzo del buffer di ingresso della tastiera (quello del GBASIC) in cui vengono accumulati tutti gli input prima della pressione del ritorno carrello.

**EXP1=7** - indirizzo della locazione JBASE, a partire dalla quale sono memorizzati gli indirizzi delle routines interne ricambiabili dall'utente.

**NOTA** - per una migliore comprensione di quanto sopra, riferirsi alla nota precedente questa, e "Il G5 e il linguaggio macchina" su questo stesso **XÉLECTRON**.

**CHR(22) & CHR(23)** - (solo in modo **GRAPH**) - servono per dare origine alla scrittura con caratteri giganti (doppia altezza). Questo tipo di scrittura è particolarmente utile per titoli o frasi da porre in grande evidenza. La scrittura in questo modo avviene ovviamente a righe alterne.

### Stringhe

In un primo momento era mia intenzione rimandare l'introduzione delle stringhe alle versioni successive. Poi, dietro il gran numero di richieste, decisi di inserire almeno qualche rozza possibilità. Durante il lavoro, come spesso accade, non ho resistito alla tentazione e ho creato le principali strutture che consentiranno in futuro un utilizzo quasi normale. Già adesso le possibilità sono molte. Unica limitazione, ma non grave, visto che anche la Hewlett-Packard fa lo stesso, almeno nelle macchine di mia conoscenza, è la necessità di predimensionamento della misura massima delle stringhe. Questo avviene mediante la frase DIM.

Esempio:

```
10 DIM A$(20), B$(10)
```

10 e 20 sono rispettivamente le lunghezze massime delle stringhe A\$ e B\$ [diverse completamente dalle variabili A e B e dai vettori A( ) e B( )].

Le stringhe possono essere al momento usate nelle frasi:

LET - ad esempio LET A\$="PLUTO" oppure A\$=B\$

INPUT - ad esempio INPUT "Come ti chiami?", C\$

PRINT - ad esempio PRINT "Il tuo nome è", A\$

IF - ad esempio IF A\$="MIAO" THEN PRINT "GATTO". Oppure IF A\$=B\$... ecc. Si noti che sono ammessi solo gli operatori =, <, e >.

Per ora non sono ammesse operazioni tra stringhe: non si può fare ad esempio LET A\$=B\$+C\$.

Sono presenti le funzioni VAL(X\$), STR\$(X), CHR\$(N) e LEN(X\$) come precedentemente descritto.

A proposito di questa ultima, si noti l'importante differenza tra lo spazio assegnato a una stringa, quello cioè riservato con la frase DIM e la lunghezza della stringa, quella cioè data dalla funzione LEN(X\$).

Esempio:

```
10 DIM A$(100)
20 LET A$="GATTO"
```

La lunghezza A\$ ha spazio 20 e, dopo la 20, lunghezza 5.

### Conclusione

Credo che le innovazioni apportate al GBASIC siano tali da confermare l'impegno verso il lettore sia mio personale che dell'Editore per quanto a suo tempo promesso.

## G5 e il LINGUAGGIO MACCHINA

*Come microcomputer tipicamente amatoriale e per di più dedicato agli sperimentatori, il G5 non poteva essere privo di funzioni e di routines che ne facilitassero l'uso in linguaggio macchina, ossia non in GBASIC ma nel linguaggio tipico del microprocessore Z-80 intorno al quale il G5 è costruito. Nel release 1.0, tuttavia, preso più che altro dal desiderio di fornire uno strumento valido a tutti coloro che iniziavano a programmare, mi ero concentrato assai di più sulla parte GBASIC, pur prevedendo una serie di funzioni per il collegamento con il linguaggio macchina. Nel release 3.00 queste funzioni sono state messe meglio a punto e ridotte di numero nonché integrate con altre come descriveremo nel seguito.*

### La funzione CALL

La funzione CALL del release 3.00 è la funzione per il collegamento con il linguaggio macchina, mentre la ARG serve per il passaggio dei parametri. POP e PUSH sono state eliminate mentre ARG è ora una frase e non più una funzione. Adesso la CALL è di uso semplicissimo:

```
LET A = CALL (25000)
```

significa: chiama il programma in linguaggio macchina all'indirizzo 25.000. Il valore di ritorno assegnato alla variabile A, 25.000 e 255 possono ovviamente essere sostituite con espressioni numeriche.

Nel saltare all'indirizzo specificato, la funzione CALL lascia al programma macchina un parametro di passaggio precedentemente predisposto con la frase ARG (vedi sotto) nella coppia di registri BC interpretata come un unico registro da 16 bit. Prima del ritorno, eseguibile semplicemente tramite l'esecuzione di una istruzione RET, il programma in linguaggio macchina dovrà predisporre il valore di ritorno, cioè il valore assunto dalla funzione CALL (), nella coppia di registri HL.

Anche il passaggio del parametro è molto semplice, avendo a disposizione la frase ARG che ha lo scopo di predisporlo in una zona di memoria riservata e non alterata da altre operazioni. La forma generale della frase ARG è

```
ARG exp1
```

dove exp1 è una espressione numerica il cui risultato sia il parametro voluto (deve essere un intero nel range 0...65536). Volendo ad esempio chiamare la routine di prima e passargli il parametro 255 basterà

```
10 ARG 255
20 LET A=CALL(25000)
```

Nella chiamata l'utente può alterare ogni registro, eccezion fatta ovviamente per il valore di ritorno dello stack. In funzione delle circostanze l'utente può anche utilizzare qualche livello dello stack del GBASIC, tenendo presente però che su di esso non avvengono controlli. La tecnica migliore è quella di salvare lo stack pointer, utilizzare uno stack separato e ripristinare quello del GBASIC prima dell'uscita.

Ad esempio:

```
INGRESSO:
          LD (TEMP1),SP
          LD GP,SPUSER
          ...
          ...
          LD SP,(TEMP1)
          RET
```

### L'organizzazione della RAM

L'organizzazione della RAM del G5 è stata già completamente descritta e deve essere già stata studiata e ben digerita prima di poter tentare l'uso di programmi in linguaggio macchina.

Ricordo comunque che per fare spazio ad eventuali programmi utente in linguaggio macchina, è indispensabile ricorrere alla frase *BLANK n* dove *n* è il valore della più alta locazione di memoria che si intende assegnare al GBASIC. Questa limitazione resterà attiva fino al successivo reset, quando la routine LOKRAM cercherà la più alta locazione di memoria disponibile. Una soluzione alternativa è quella di inserire i programmi utente in un banco diverso di memoria non contiguo con quello che ha origine da 16.384. In tal maniera la LOKRAM sarà "ingannata" dal "buco" esistente tra i due blocchi e non andrà a cercare oltre. I programmi macchina potranno in ogni caso essere salvati su nastro con il comando *SAVE inizio, fine* e ricaricati con *LOAD inizio*, dove *inizio* e *fine* sono ovviamente i valori di inizio e fine del programma medesimo.

### Routines interne del GBASIC

Il GBASIC consente all'utente di accedere ad alcune delle sue routines interne. Gli indirizzi di queste routines sono variabili in funzione del particolare release in uso. È possibile però conoscere i loro indirizzi previo l'uso della funzione *INFO(7)* che ritorna l'indirizzo base (detto *JBASE*) in una tabella che contiene tutti gli indirizzi della routines accessibili dall'esterno.

Esempio: la routine di ingresso da tastiera è identificata come *JBASE+3*. Per richiamarla basta

```
10 PRINT CALL(INFO(7)+3)
```

oppure in alternativa, comoda quando le chiamate debbano essere frequenti

```
10 LET K1=INFO(7)+3
20 PRINT CALL(K1)
```

Desiderando invece soltanto conoscere l'indirizzo in questione per utilizzarlo magari in un programma macchina, si farà

```
PRINT INFO(7)+3
```

Nel seguito riporto l'elenco completo di dette routines.

*JBASE+0* - VDR - emissione di un carattere (contenuto in acc) verso lo schermo;

*JBASE+3* - KBDIN - ricezione di un carattere da tastiera. Il carattere letto è lasciato in acc.;

*JBASE+6* - PNTR - emissione di un carattere verso la stampante;

*JBASE+9* - INSYS - inizializzazione del canale video, tastiera e stampante.



Non chiamando questa routine, le due precedenti non funzioneranno. È sufficiente chiamarla una volta e le inizializzazioni permangono;

JPBASE+12 - ERROR - questa locazione deve essere acceduta con un JMP e non con un CALL. Emette un messaggio di errore (indirizzo in HL, termine con carattere doppio apice ') e provoca la reinizializzazione del GBASIC.

JPBASE+15 - INLINE - routine per l'input di una intera linea da tastiera, con controllo sui caratteri di editing. Il ritorno avviene alla pressione del tasto RETURN. Il buffer di ingresso, dove si ritrova la linea "inputtata" inizia alla locazione INFO6 data da INFO(6).

JPBASE+18 - KSTAT - routine per il controllo di eventuali tasti battuti sulla tastiera durante l'esecuzione di un programma. Se nessun carattere è presente nel buffer della tastiera, il ritorno avviene con il flag Z (zero) settato.

### Conclusione

Sulla base delle indicazioni sopra fornite, sono certo che non ci saranno problemi per alcuno nell'utilizzo di routines utente in linguaggio macchina.

\*\*\*\*\*

## Raccoglitori per CQ ELETTRONICA e XÉLECTRON

Richiedeteli a:

**edizioni CD**  
**via C. Boldrini, 22**  
**40121 BOLOGNA**



1 RACCOGLITORE L. 8.000  
(abbonati L. 7.200)  
+ L. 2.000 spese spedizione

*Pagamento con assegni propri o circolari - vaglia  
o con c./c. P.T. n. 343400 a noi indirizzati.*

una

---

---

# ECONOMICA TASTIERA

---

---

## per il vostro MICROCOMPUTER

---

*ing. Gianni Becattini*

---

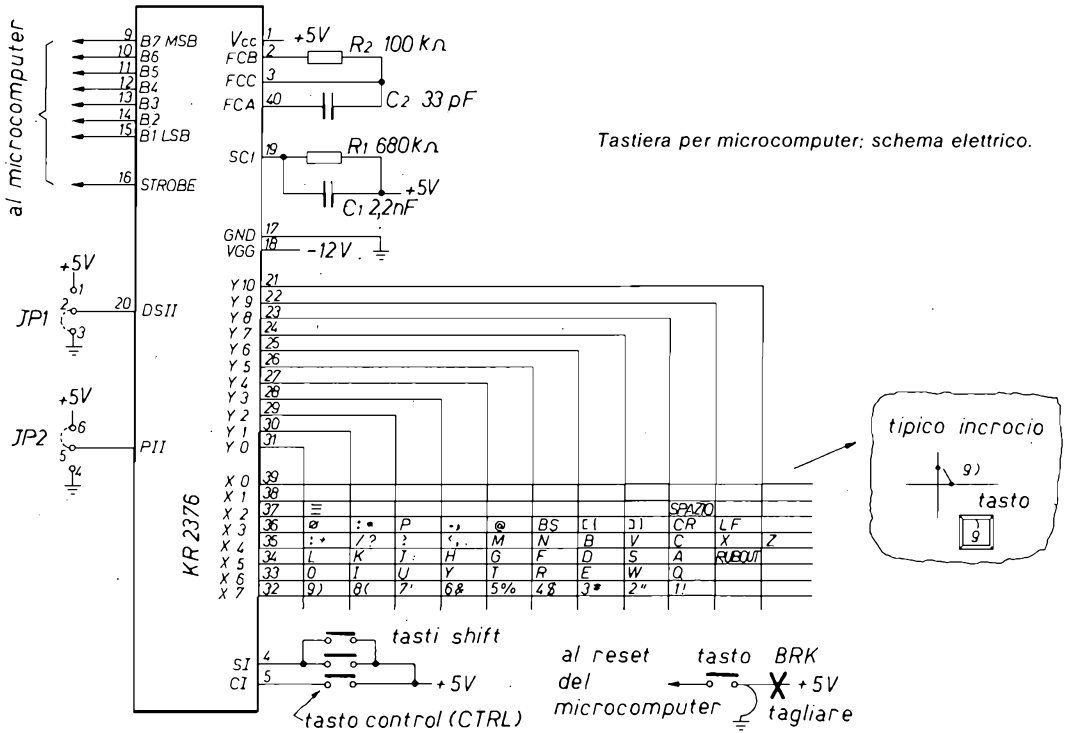
*Autocostruendo un microcomputer ci si trova spesso alle prese del "problema tastiera": autocostruirla o comprarne una già fatta?*

*In ogni caso si è spesso costretti, per ragioni di costo, a dei compromessi; una buona tastiera costa assai cara e il surplus offre spesso dei modelli di ottima qualità ma con distribuzioni strane di tasti, codifiche anomale, ecc.*

Per questo motivo molti mi hanno scritto richiedendo suggerimenti ed è in risposta a queste istanze che sono lieto di presentare un progetto veramente economico ma che consente di ottenere un prodotto di qualità elevatissima: basti dire che i tasti impiegati sono gli ottimi WEAB, svedesi, che il circuito stampato ha i fori metallizzati e che il tutto, compreso il Keyboard Encoder, un bell'integrato a 40 pin che provvede alla codifica ASCII e ad altre funzioni, costa una sessantina di migliaia di lire; si colloca quindi molto al di sotto dei prezzi attualmente correnti. È oltretutto utilizzabile anche in unione al microcomputer G5 presentato su **CQ ELETTRONICA** 1, 2 e 4/83.

### **Caratteristiche**

La tastiera presentata conta 57 tasti e consente di ottenere tutti i caratteri ASCII, incluse le lettere minuscole. Per l'alimentazione sono necessari 5 V positivi e 12 V negativi. L'unico integrato, il KR2376-ST della Standard Microsystem, provvede a convertire le pressioni dei tasti nell'equivalente codice ASCII, nonché ad eliminare eventuali disturbi o rimbalzi nella chiusura dell'interruttore. Il tocco dei tasti è estremamente piacevole, grazie alla leggerezza del meccanismo e alla curva di resistenza offerta dalla molla di richiamo, studiata per la migliore ergonomia.



Note: porre JP1 in posizione 1-2 se si desidera avere dati e STROBE attivi bassi.  
Porre JP2 in posizione 5-4 per invertire la parità.

### Costruzione

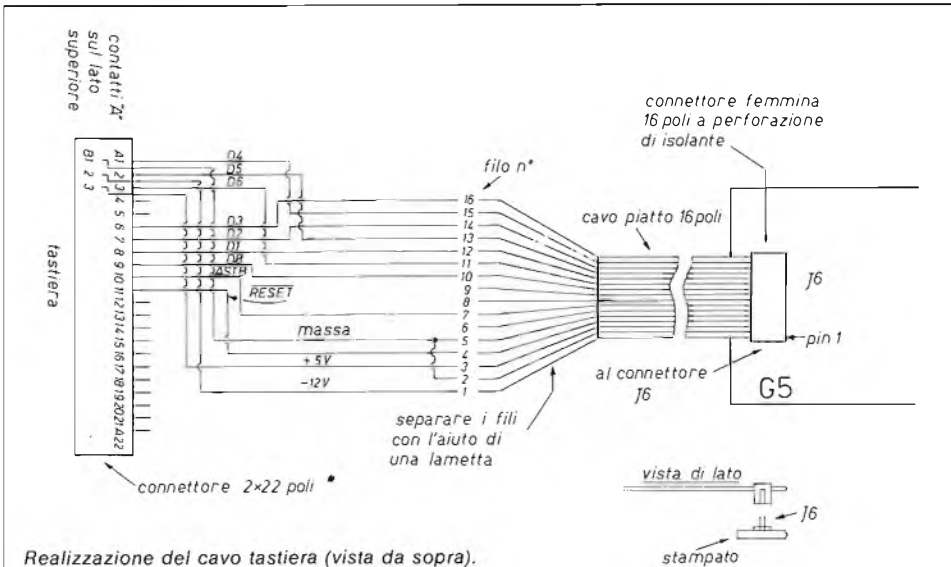
Con l'aiuto degli schizzi che riporto, la costruzione è semplicissima anche se richiede due fondamentali attenzioni:

- 1) Montare i tasti dal lato giusto dello stampato, riconoscibile per la presenza di scritte ottenute dal rame stesso.
- 2) Montare i cappucci solo dopo essere perfettamente sicuri della loro posizione, spingendoli bene a fondo. Questo è importantissimo, poichè nell'eventuale smontaggio si potrebbero rompere i gambi di supporto.

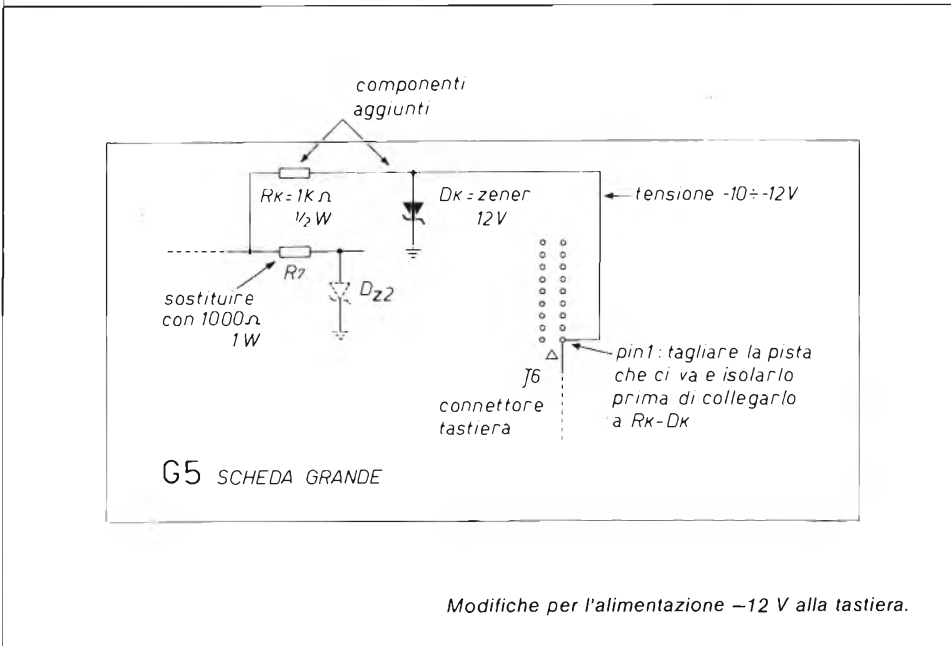
Si noterà facilmente che tra tutti i tasti ce n'è uno con la molla più dura; dovrà essere usato per la barra spaziatrice che, avendo un semplice meccanismo collegato, richiede una maggiore forza per il ritorno nella posizione di riposo. Assieme al kit vengono comunque fornite due fotocopie che illustrano tutti i particolari del montaggio.

### Collegamento al G5

Questa tastiera si presta ottimamente al collegamento con il G5. Gli schemi riportati sono estremamente dettagliati e mostrano anche come ricavare il -12 per il KR (questa tensione non è per niente critica, basta sia compresa tra -12 e -10V circa). Il tasto BRK, che è in pratica un interruttore, potrà essere impiegato per il RESET, avendo però l'accortezza di scollegarne il lato che va al +5V mandandolo invece a massa (vedi figure). Grazie alla funzione CTRL/D del G5 release 3.00 non è necessario un tasto TTY LOCK.



\* I contatti B1, B2, B3 sono i corrispondenti, sul lato inferiore, di A1, A2, A3.



### Conclusione

Ritengo che finora non siano stati presentati progetti di tastiere così economiche, almeno a parità di qualità dei prodotti impiegati. Sulla stessa scia prenderemo presto in esame anche il problema del monitor, tentando anche per esso una soluzione "di rottura" con i costi tradizionali.

\*\*\*\*\*

# IMPARIAMO a INTERFACCIARE

---

*ing. Gianni Becattini*

---

## **G5 - INTERFACCIA per STAMPANTE**

*Su qualunque microcomputer si operi, nasce prima o poi prepotente il bisogno di un organo di stampa, soprattutto per poter esaminare comodamente le liste dei propri programmi in fase di messa a punto.*

Ecco pertanto sottoposta ai lettori questa semplicissima interfaccia che consente di collegare al G5 la quasi totalità delle stampanti che operino con protocollo Centronics (detto anche "parallelo").

Questo standard, adottato non solo dalla nota Casa che lo introdusse ma da moltissime altre, presenta una serie di vantaggi non indifferente:

- 1) è certo il più diffuso;
- 2) è uno degli standard più "standard". Non ci sono infatti da spostare ponticelli né da modificare i cavi di collegamento, come spesso accade con lo RS-232. Ho provato con il risultato positivo questa interfaccia con una miriade di stampanti di Marche diverse senza alcun problema;
- 3) è quello che consente di spendere meno. Infatti anche i modelli di stampanti che possono disporre sia dell'interfaccia Centronics che della RS-232, hanno questa ultima come opzionale con relativo sovrapprezzo.

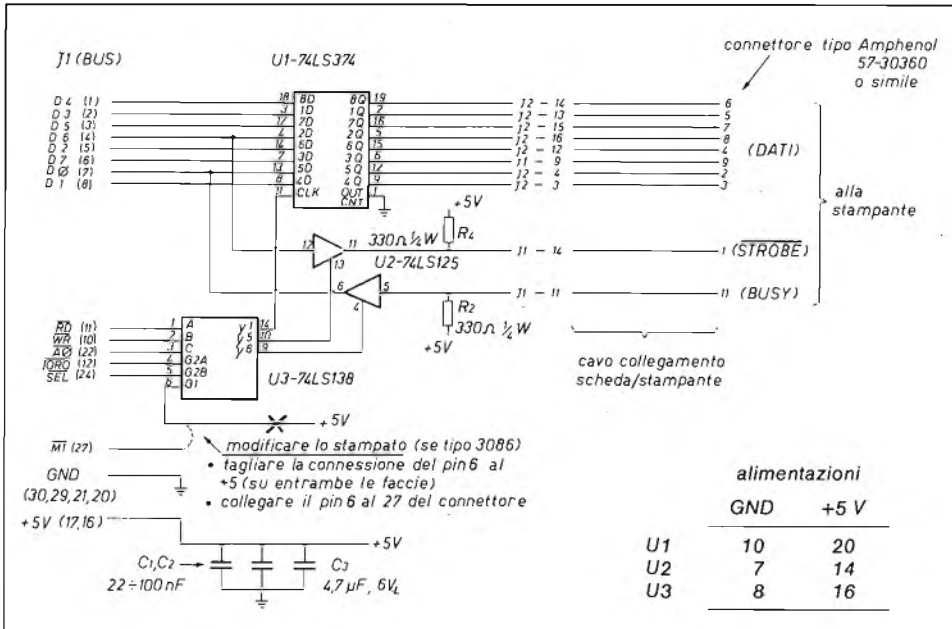
### **Il circuito**

Il circuito è molto semplice: in pratica si tratta di una porta di uscita a 8 bit realizzata con un 74LS374 e due "miniporte", una di ingresso e l'altra di uscita, ottenute con un quarto di 74LS125 cadauna per il controllo dello scambio dati. Un 74LS138 provvede a decodificare l'indirizzo e le linee RD, WR, IORQ e M1.

Per ridurre la complessità, si è rinunciato all'uso di qualche linea del protocollo Centronics, lasciando solo ciò che è indispensabile; lo scambio dati avviene così (vedi figura):

- 1) G5 emette un dato sul 374;
- 2) G5 rende vera l'uscita STROBE (quindi bassa);
- 3) G5 attende finché rimane vero (alto) il segnale BUSY emesso dalla stampante.

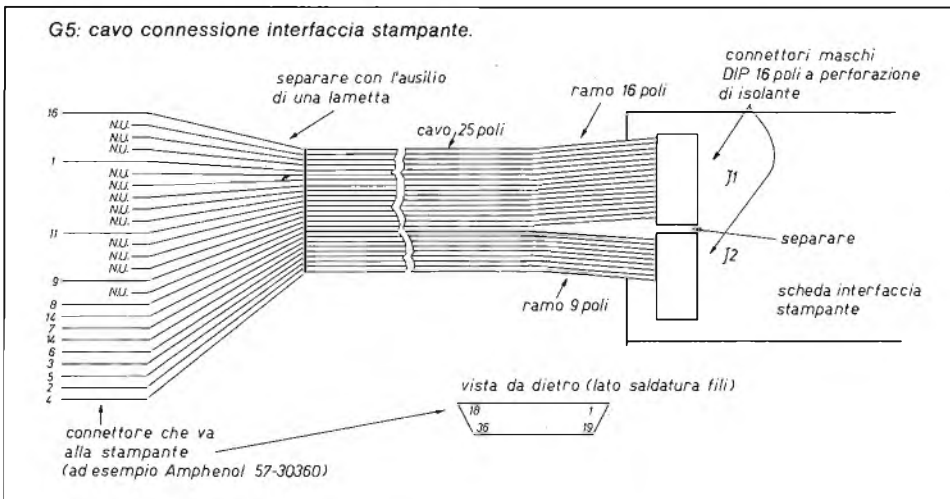
Anche in questa maniera si conserva comunque la massima efficienza.



### La costruzione

La costruzione è molto semplificata dal fatto che ho predisposto il circuito stampato. Data la semplicità, non ho ritenuto opportuno usare il foro metallizzato. Si dovrà provvedere quindi, ove necessario, a installare un ponticellino tra le due facce e a fare saldature su ambo i lati (occhio in particolare alle piste che vanno al connettore J1 (NOTA: i primi stampati, siglati 3086, hanno i fori metallizzati ma contengono un piccolo errore, come spiegato sullo schema, che deve essere corretto prima dell'uso; gli stampati senza errore sono marcati DIGINEL 30868).

Il cavo di collegamento potrà essere realizzato in vari modi; suggerisco la forma riportata in figura perché evita, almeno da un lato, le saldature. Data la grande ridondanza di informazioni degli schemi, non si troveranno in ogni caso difficoltà anche con costruzioni diverse.



Il connettore indicato va bene per il 99,9% delle stampanti oggi esistenti; solo i modelli Centronics 730 e derivati hanno un connettore di tipo "card edge" differente da quello standard. I segnali, in posizione diversa, sono comunque gli stessi e potranno essere ricavati dal rispettivo manuale.

### Installazione

La scheda interfaccia stampante, per essere utilizzabile dal GBASIC, deve essere inserita nel connettore J4 (quello più vicino al TMS9929). Essendo più corta del connettore, essa dovrà essere montata in modo da occupare i pin da 1 a 30 (cioè tutta dalla parte del dissipatore di alimentazione). In questo modo si potranno usare la frase LPRINT e il comando LLIST già contenuti nelle EPROM.

### Conclusione

Come promesso, tutto molto facile e non dispendioso. Se non si sono fatti errori di montaggio, l'interfaccia funzionerà alla prima e doterà il nostro G5 di una uscita su carta che ci sarà certo utilissima in mille occasioni.

## G5-INTERFACCIA per REGISTRATORE

*L'interfaccia per registratore audio non è certo il dispositivo che sta in testa alle classifiche delle mie simpatie personali. Purtroppo, come ho dovuto constatare anche dalle grandi richieste dei lettori, è un "male inevitabile" e quindi dobbiamo cercare di affrontarlo nel modo migliore possibile.*

Vediamo di fare alcune considerazioni preliminari:

- 1) In generale, quando a un microcomputer venga a mancare l'alimentazione, questo perde tutto ciò che la sua memoria contiene: programmi e dati.
- 2) Il G5, se dotato di memoria continua, elimina questo inconveniente; infatti appositi accumulatori provvedono alla conservazione dei contenuti delle RAM che, di tipo CMOS, hanno un consumo bassissimo e possono quindi restare attive per un tempo molto lungo.
- 3) L'inalterabilità della memoria posseduta dal G5 non risolve però del tutto il problema. Infatti rimane la necessità di battere manualmente ogni programma, quindi con grande perdita di tempo quando si decida di passare da una procedura a un'altra diversa. L'unico modo di scambiare i dati resta quindi quello delle liste su carta.

I microcomputer più "seri" possiedono, come i grandi computer, dei supporti di memoria di tipo magnetico, come dischi o floppy disk. La memoria magnetica consente una conservazione virtualmente illimitata dei dati e un facile scambio dei medesimi. I dischi hanno il difetto però di costare piuttosto cari.

Noi hobbysti che, per motivi di costo, dobbiamo spesso accontentarci di "surrogare" le unità di tipo professionale, abbiamo tuttavia sotto mano una memoria magnetica di tipo molto economico: il registratore a cassetta. Questa memoria è però in origine di tipo "analogico" in quanto è destinata a memorizzare segnali elettrici variabili di tipo audio, la musica o il parlato. Dobbiamo quindi utilizzare dei circuiti di interfaccia che consentano di impiegare il nostro mangiacassette come memoria digitale, ossia capace di registrare "1" e "0".

## I difetti dei registratori

In linea di principio non è difficile provvedere all'interfacciamento di un microcomputer con un registratore.

In pratica l'operazione non è molto semplice perchè:

- 1) i comuni mangiacassette sono per solito veramente scadenti; il loro progetto è fatto nella massima economia e ogni eventuale margine in più viene dedicato dal Costruttore ad allestire un contenitore con l'aspetto più professionale possibile, senza che questo abbia alcuna relazione con la parte funzionale;
- 2) le differenze tra registratore e registratore sono enormi, per cui è difficile progettare dei circuiti che operino indifferentemente con l'uno o l'altro tipo. Non parliamo poi del problema di interscambio dati, anche a seguito del differente allineamento delle testine.

## Tecniche di registrazione

Le tecniche di registrazione possibili sono molte. Una delle più affidabili è quella di registrare i dati nello stesso modo con cui vengono trasmessi via radio per le telescriventi (RTTY) facendo corrispondere agli stati "0" e "1" due toni audio di frequenza diversa e riconoscendoli poi tramite opportuni filtri.

Questo metodo però male si presta ai nostri scopi perchè:

- 1) consente di ottenere velocità di interscambio relativamente basse;
- 2) richiede circuiti alquanto complessi e un po' di strumentazione per la loro messa a punto.

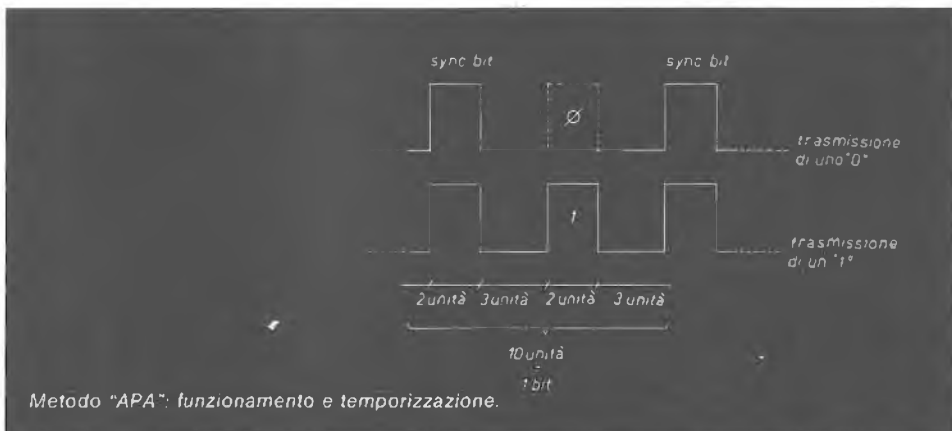
Il metodo che utilizzeremo lo chiameremo pomposamente "autosincronizzato a presenza/assenza di impulso" (APA) ed è stato messo a punto empiricamente.

Presenta i seguenti vantaggi:

- 1) richiede circuiti estremamente semplici e che non necessitano di alcuna messa a punto;
- 2) consente di ottenere buone prestazioni.

## Il metodo "APA"

Il metodo APA è semplicissimo e parte dalla considerazione che anche il peggior mangiacassette riesce sempre in definitiva a registrare un "trump",





intendendo per "trump" una qualunque perturbazione di tipo rumoroso (impulso). Il metodo consiste nel registrare due "trump" per ogni bit "1" e un solo "trump" per ogni "0". Si ha cioè (vedi figura) sempre la presenza del primo impulso (detto di sincronismo) e la presenza o la assenza del bit di dato. La presenza dell'impulso di sincronismo a ogni bit di informazione garantisce una buona tolleranza verso le variazioni di velocità della cassetta, in quanto il programma "va a cercare" il bit dato dopo un tempo fisso dall'impulso di sincronismo stesso. È così sufficiente che la velocità resti stabile almeno per piccolissimi periodi di tempo.

## Il formato di registrazione

All'inizio della registrazione è necessario poter stabilire con sicurezza quale è il bit di sincronismo e quale quello del dato. Infatti, a seguito degli inevitabili disturbi che si hanno quando si preme il tasto RECORD, il punto di inizio dell'incisione è sempre un po' incerto. Il flusso delle informazioni è quindi preceduta da una serie di dati non significativi che hanno la sola funzione di consentire l'avvio del sincronismo. In pratica si tratta di 256 "zeri" seguiti da un "uno". Il programma è fatto in modo da sincronizzarsi in lettura in modo sicuro sugli zeri dopo la lettura di pochi bit. Dopo aver letto 64 zeri consecutivi, e quindi avendo la certezza del sincronismo, cerca il primo "uno" che indica che tutti i dati che seguono hanno invece un significato e che possono essere letti. I dati sono poi registrati secondo certe regole che ne consentono il controllo di validità.

La tabella sottostante descrive esattamente questi "formati".

tabella 1

### Il formato di registrazione

1) Coda iniziale - 256 zeri - 1 uno

2) Blocco di testa: composto da:

	bytes
tipo dei dati	1
lunghezza dei dati	2
campo nome file	6
campo ausiliario	1
fine blocco	1
checksum	1

Il tipo dei dati è 30H per i programmi GBASIC. Il campo lunghezza dati, di due bytes, è inviato con il byte più significativo per primo. I campi nome e ausiliario non sono per ora utilizzati.

3) Blocchi dati (tanti quanti necessari) il 256 bytes cadauno seguiti dal checksum.

NOTA: nella serializzazione sono inviati prima i bit meno significativi.

Il checksum è calcolato semplicemente per somme successive con troncamento a 8 bit.

## La cassetta e il GBASIC

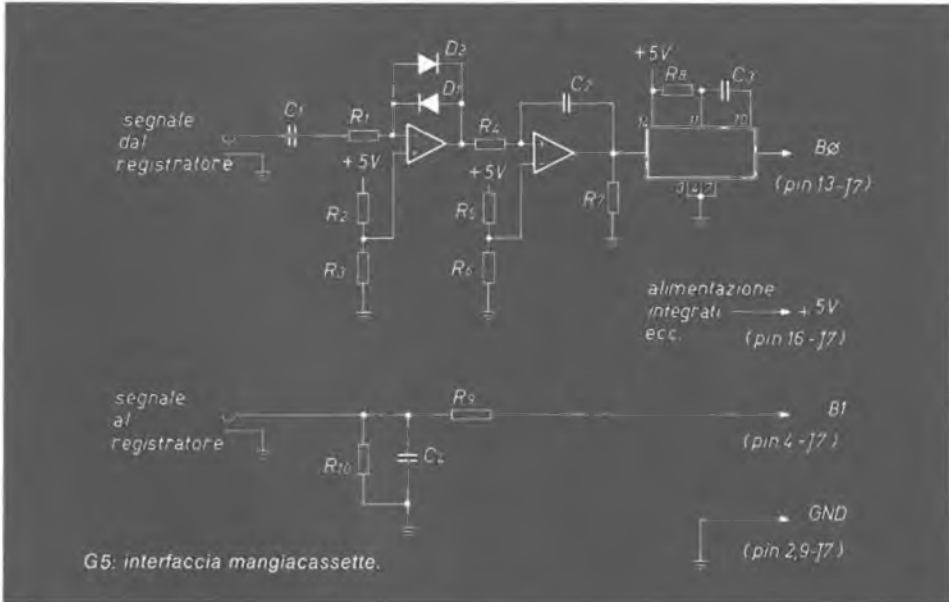
I comandi SAVE, LOAD e VERIFY sono utilizzati dal GBASIC (release 3.00 e seguenti) per scrivere, leggere e verificare i programmi dalla cassetta. Questi comandi sono già stati ampiamente descritti su questo **XÉLECTRON** (vedi paragrafo GBASIC release 3.00). Vorrei però soffermarmi un attimo sul comando VERIFY seguito da un numero, che serve a variare la velocità alla quale i dati sono registrati su cassetta e vengono rilette. Normalmente la velocità è 20. Impartendo il comando VERIFY 25, la velocità passa a 25. Numeri più alti corrispondono a velocità più basse e viceversa.

Questa facoltà di variare la velocità permette a chi possiede registratori migliori di sfruttare al meglio le caratteristiche della propria unità, così come consente l'uso di un magnetofono scadente.

Variazioni sensibili della velocità richiedono la modifica di un condensatore nel circuito (vedi sotto). Chi possiede registratori migliori, dopo avere rilevato la più alta velocità ottenibile, potrà inserire un deviatore per utilizzare ancora la velocità 20 nello scambio dati con altri amatori.

## Il circuito

Il circuito è semplicissimo ed è composto solo da due integrati e qualche passivo.



Il segnale a onda rettangolare diretto al registratore passa attraverso una resistenza e viene un po' addolcito dal gruppo RC. Il 74121 è un monostabile destinato a generare un impulso di durata fissa a ogni "trump" di ingresso. Il condensatore  $C_3$  è quello che ne determina la lunghezza. Deve quindi essere, al pari della resistenza  $R_9$ , preciso e stabile. Variando la velocità, come spiegato sopra, si dovrà variare anche il condensatore. Il valore migliore si determinerà per esperimento.

Altra soluzione è quella di lasciare fisso il condensatore e variare invece  $R_9$ . In questo modo si ha il vantaggio di poter inserire dei trimmer per trovare i valori migliori con piccole approssimazioni. L'alimentazione viene ricavata dalla stessa scheda madre del G5. Lo schema elettrico riporta anche le connessioni al G5.

## Conclusione

Come ho detto all'inizio, la cassetta audio è un male inevitabile ma ritengo che la soluzione proposta sia per così dire il male minore: il circuito è facile e sufficientemente affidabile.

Sarò particolarmente grato a tutti coloro che vorranno parteciparmi, in forma positiva e sintetica, eventuali suggerimenti derivanti dalla esperienza, ricordando di citare Marca e tipo del registratore usato. \* \* \* \* \*

# il G5 e la RTTY

ing. Gianni Becattini

*A giudicare dal numero delle richieste che mi pervengono, sono moltissimi che desiderano ricevere le emissioni RTTY con il G5; molti addirittura mi hanno rivelato di averne iniziato la costruzione solo con questo obiettivo in mente.*

Le mie promesse di occuparmi dell'argomento sono riuscite a mala pena a trattenere questa folla per qualche mese; la marea è ormai dilagante e devo passare all'azione, pena la necessità di emigrare agli antipodi. La cosa neppure mi dispiace: la mia passione per i computers nacque molti anni orsono proprio dal desiderio di veder scrivere in modo automatico la mia vetusta e adorata TG7 che conservo ancora con cura religiosa. Questi esperimenti sono poi stati l'occasione di adoperare gli apparecchi surplus della mia collezione congiungendo così due passioni apparentemente lontane (per la cronaca: i migliori ricevitori sono risultati, come prevedibile, il 390 e il 392, per la loro eccezionale stabilità; una grossa delusione invece dallo SP-600 che è risultato estremamente instabile, peggio addirittura del BC348 che grazie all'elevato rapporto di demoltiplica della sintonia consente almeno di inseguire l'emittente con una certa facilità; solo mediocre il G4/216).

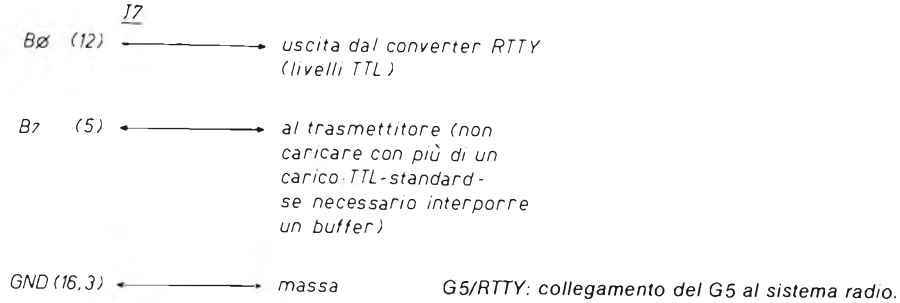
### Un primo approccio

Il problema della RTTY è più complesso di quello che può sembrare a prima vista: la semplice ricezione o trasmissione rappresentano solo una metà della mela. Non meno importanti sono le varie funzioni accessorie (memorie e altre facilità, ricezione del CW, ecc.). Per non ritardare ancora l'uscita di questa nota, ho pensato di introdurre intanto una versione preliminare del programma, denominato con poca fantasia G5/RTTY, capace della sola funzione basilare: la ricezione. Compatibilmente come sempre con le mie possibilità di tempo, G5/RTTY crescerà e sarà dotato di tutte le facilities anche sulla base dei vostri suggerimenti futuri. Vi posso fornire la lista del programma gratis previo rimborso delle pure spese di copia e spedizione: mandatemi 2.000 lire e ve lo spedisco a casa. Il mio indirizzo: G. Becattini - viale della Repubblica 84 - 50019 Sesto fiorentino (FI).

### G5/RTTY

G5/RTTY non richiede alcun hardware esterno, eccezion fatta ovviamente del converter RTTY. Consente di ricevere in Baudot a varie velocità. G5/RTTY può essere adoperato, con pochissime modifiche, su qualsiasi micro-computer basato su microprocessore Z-80. Il programma è stato assemblato per il release 3.00 del GBASIC. Utilizzandolo con altre versioni o con altri

micro, si dovranno aggiustare gli indirizzi, se necessario. La velocità di ricezione/trasmisione si regola variando il contenuto della locazione SPEED con la frase POKE. Numeri più grandi corrisponderanno a velocità più basse e viceversa. L'interfacciamento elettrico viene effettuato dalla porta del registratore come mostrato nella figura.



Essendo alcune linee poste in comune, può essere montato un deviatore per scegliere il registratore o l'uso RTTY. Il programma viene lanciato, per la ricezione in Baudot, con una semplice chiamata al suo indirizzo iniziale; ad esempio facendo

```
PRINT CALL(17402)
```

### Routines del GBASIC utilizzate

Nel corso del programma è stata richiamata la routine VDR per l'uscita di un carattere sul video. Per l'uso con release di GBASIC successivo al 3.00 il suddetto indirizzo dovrà essere modificato e il nuovo sarà ricavabile con l'aiuto della INFO(6) come spiegato nel paragrafo "G5 e il linguaggio macchina", a pagina 56 e seguenti di questo stesso XELECTRON.

### Conclusioni

Il programma può essere caricato byte per byte tramite la frase POKE, convertendo previamente tutti i codici, presenti nella lista come esadecimali, in decimali e aggiustando gli indirizzi come illustrato. Spero di essere stato chiaro e di avere soddisfatto così anche gli amici radioamatori che invito comunque a non fermarsi qui ma ad usare il G5 per quello che è, ossia per un piccolo computer personale che può dare delle notevoli soddisfazioni e contribuire sensibilmente all'allargamento della propria base di conoscenze. \*\*\*\*\*

**ABBONATI A**

**CQ ELETTRONICA**

**XELECTRON**

**RICEVERAI:**

**SUBITO IN REGALO**

1 utilissima cassetta porta-tutto (dimensioni cm 31x13 h 15), dove potrai riporre microprocessori, transistori, circuiti integrati, resistenze, diodi, pinze, viti ecc. + 1 set completo di 6 cacciaviti di diverse dimensioni.

**OPPURE IN ALTERNATIVA**

1 libro a tua scelta della collana  
"I LIBRI DELL'ELETTRONICA"

---

La tua rivista all'uscita di ogni numero direttamente a casa tua.

---

**RISPARMI:**

su eventuali aumenti di prezzo di copertina.

**ABBONAMENTO** annuale a:

**CQ ELETTRONICA + XELECTRON L. 30.000**

**RINNOVI L. 29.000**

---

**MODALITÀ DI PAGAMENTO**

Il pagamento potrà essere effettuato a mezzo: C/C postale intestato a "EDIZIONI CD" n. 343400 - assegno proprio o circolare - vaglia postale.

---

**CAMPAGNA ABBONAMENTI 1984  
VALIDA FINO AL 31/3/1984**

# Scheda acquisizione dati per Personal IBM

La Data Translation Inc., distribuita in Italia dalla Eledra 3S, ha annunciato un nuovo e interessante prodotto: il **DT 2781: scheda acquisizione dati per Personal Computer IBM**. Fino ad oggi la possibilità di utilizzo di Personal Computers in laboratorio, per acquisizione dati e per applicazioni di controllo industriale, è stata limitata dall'assenza sul mercato di sofisticate piastre di input/output analogiche e in alcuni casi dal limite di disponibilità di CPU locali. La situazione è ora mutata.

Il binomio IBM Personal Computer e Data Translation DT 2801 (sistema di I/O Analogico Digitale su singola scheda) offre all'utilizzatore la possibilità di operare nel campo dei controlli e dell'acquisizione dati, con un sistema ad alte prestazioni e basso costo.

DT 2781 è un sistema di controllo e acquisizione dati su scheda singola: A/D, D/A, I/O digitali e clock programmabile.

La DT 2781 sopporta il DMA e il trasferimento di dati I/O programmati.

Un microcalcolatore montato sulla scheda e un microcode evitano al computer principale la gestione dell'hardware di acquisizione e permettono all'utilizzatore di accedere a tutte le funzioni tramite i comandi dello IBM P.C. sotto linguaggio BASIC.

Gli ingressi e le uscite analogiche hanno 12 bit di risoluzione con una velocità di 13.000 campionamenti al secondo.

Le principali caratteristiche della scheda sono: 12 canali a 12 bit di risoluzione in ingresso con guadagno 1, 2, 4, 8, programmabile a software; 2 canali analogici a 12 bit in uscita, 16 linee di I/O digitali e clock programmabile; possibilità di utilizzo di clock esterno e di trigger esterno per inibizione di comandi.

Questo permette alla DT 2801 di operare in sincronismo con eventi esterni.

Per ulteriori informazioni rivolgersi a: ELEDRA 3S S.p.A. - viale Elvezia 18, Milano - Tel. 349751.



# Io e il G5

---

*15YJI, Francesco Francescangeli*

---

*Questo articolo si rivolge a chi, come me, ha avuto il suo primo incontro con il mondo dei microcomputer tramite il G5 dell'Ing. Becattini.*

Lungi da me l'idea di insegnare qualcosa a qualcuno!

I primi rudimenti di programmazione, infatti, li ho acquisiti dal libro **"IL COMPUTER È FACILE, PROGRAMMIAMOLO INSIEME"**, edizioni CD, che peraltro si è dimostrato un ottimo insegnante; è mia intenzione cominciare a smuovere le acque "G5" e vedere se qualcun'altro, seguendo il mio esempio, vorrà far partecipi gli altri possessori del G5 delle sue esperienze, dei suoi insuccessi (a me è capitato) e, perché no, presentare idee, listati, progetti di hardware, modifiche, ecc.

Dopo aver acquistato i vari componenti necessari e averli assemblati, come è mio vizio, definitivamente in un contenitore, solita routine: prova tensioni a vuoto, scarica condensatori, inserzione integrati, ricontrollo tensioni, ecc; finalmente collego il buon vecchio TV 12" casalingo (uscita del 9929 direttamente in base del rivelatore video) e premo il BREAK: bomba mentale!; cursore lampeggiante e video nero! Non sto ad annoiarvi con il resoconto delle ore spese alla ricerca di un guasto superiore alle mie possibilità; però, essendo di carattere tenace, non ho gettato tutto in cantina, ma, grazie alle informazioni desunte dal Nanobook volumi 1 e 2 e alla gentilezza e pazienza della ditta SUMUS, sono arrivato dopo due giorni infernali alla scoperta del guasto: sulla prima scheda memorie (1-4Kbyte) ho scoperto con la lente da orologiaio una interruzione tra una pista e un foro metallizzato, proprio vicino a quest'ultimo, delle dimensioni di un micron! Pieno di rinnovata speranza, faccio un ponticello, reinserisco la scheda nel bus, accendo e pigio il BREAK: finalmente appare "G5 - GBASIC ecc.". Consiglio: controllare sempre attentamente e con la lente da orologiaio **tutte** le piste, e prima di montare i componenti.

Dopo aver tempestato sui tasti per ore e ore, e aver scoperto che la programmazione non è poi una bestia nera ma solo una questione di logica (quella del computer, mica quella mia), mi sono buttato nella stesura dei primi timidi programmini, più grafica che matematica, naturalmente.

Ecco il primogenito:

```
10 REM PROVA RIMBALZO
20 GRAPH
25 REM TRACCIATURA PARETI
30 PLOT 99,0,99,128
40 PLOT 117,0,117,128
50 REM DEFINIZIONE PALLA
```

```

60 GOSUB 1000
70 SPTYP 1,0,1
80 SPCOL 0,11
85 REM LIMITI MOVIMENTO SPRITE
90 FOR I=0 TO 3.14159*2 STEP 0.1
95 REM MOVIMENTO SIMILE A UN PISTONE
100 SPMOV 0,100,SIN(I)*80+95
110 NEXT
120 GOTO 90
1000 SPDEF 0,0,00XXXX00F
1010 SPDEF 0,1,0X0000X0F
1020 SPDEF 0,2,X000000XF
1030 SPDEF 0,3,X000000XF
1040 SPDEF 0,4,X000000XF
1050 SPDEF 0,5,X000000XF
1060 SPDEF 0,6,0X0000X0F
1070 SPDEF 0,7,00XXXX00F
1080 RETURN

```

Provate a modificare alcuni valori delle righe 90 e 100 e vedrete quanto sia facile ottenere dallo sprite i movimenti voluti. Ad esempio alla riga 90 il valore assegnato allo STEP varia la velocità dello sprite; alla riga 100 variando il +95 fate rimbalzare lo sprite più in alto o più in basso, mentre con il \*80 fate coincidere il rimbalzo con il bordo inferiore dello schermo.

### **RELAX dopo lo stress psicologico del primogenito, e alcune curiosità**

Il movimento del cursore sul video viene di norma effettuato con una serie di istruzioni tipo:

```

PRINT CHR( 8) passo indietro cursore (Back Space)
PRINT CHR(11) cursore in alto a sinistra (File Separator)
PRINT CHR(15) passo in alto del cursore (Vertical Tab)
PRINT CHR(22) inizio sottolineatura
PRINT CHR(23) fine sottolineatura

```

Giocherellando con la tastiera, ho scoperto che esistono, almeno con la mia tastiera, anche se penso che sia lo stesso con tutte, soluzioni più rapide per ottenere lo stesso effetto. Pigiando il tasto CONTROL e il tasto H si ha l'equivalente del CHR(8) cioè un backspace con cancellazione del carattere scritto prima.

Quindi, chiamando C+H l'operazione sopra descritta, vediamo le altre:

```

C+O il cursore passa alla riga superiore ( ↑ ) e non cancella
C+J come sopra ma verso la riga inferiore
C+K il cursore si posiziona in alto a sinistra
C+L come sopra, ma cancella tutta la pagina
C+N il cursore si sposta a destra ( → ) come la barra spazio
C+I il cursore si sposta a destra di otto spazi e non cancella
C+V abilita la scrittura sottolineata (vedi CHR(22))
C+W disabilita la scrittura sottolineata (vedi CHR(23))

```

Chiamato il tasto SHIFT solo S, avremo:

```

S+ @ stampa un accento grave ( ` )
S+ ← stampa un trattino ( - )

```

S+ ↑ stampa un accento usato in spagnolo (̃)

S+ \ stampa un carattere così: ¡

Facendo C+V e poi C+L appare la pagina tipo quaderno a righe, con cursore nero, e ci si scrive di conseguenza.

### **FINE DEL RELAX e programma serio (si fa per dire)**

Indovinate qual'è il programma più diffuso tra i neofiti del microcomputer? Sorpresa: i Bioritmi!

No! direte voi.

Sì! dico io, e come mancare a questo ineluttabile appuntamento? Quindi inorridite: BIORITMI versione n° 9867.

Non credo sia il caso di spiegare cosa sono e come funzionano!

Confesso che all'inizio, per calcolare i giorni trascorsi dalla mia nascita al giorno voluto con il G5, più caparbio di me, ho esaurito tre alfabeti inglesi e un blocco notes di miserandi tentativi. Poi, un giorno, parlando con la mia amica Luisa (programmatrice, guarda caso), esposi il problema e lei, con disarmante candore, disse le fatidiche parole: "e cosa ci vuole?".

Un Fantozzi le sarebbe saltato alla gola.

Invece con espressione sfingea le dissi: "fai un po' vederel!".

Effettivamente è molto semplice, bastava pensarci.

Con una semplice subroutine si calcolano i giorni trascorsi dall'inizio del secolo alla data di nascita, si rifà il calcolo dall'inizio secolo alla data attuale e poi con una semplice sottrazione il gioco è fatto.

Ripreso pigolo, detto alla marenmana, mi sono rimesso a stendere il programma e, visto che funzionava bene con uscita numerica, ho voluto strafare e aggiungerci la stampa dei tre grafici in sinusoidale.

Ed ecco il nuovo parto:

```

10 REM CALCOLO BIORITMI CON GRAFICI
15 PRINT CHR(12)
16 PRINT TAB(12), "CALCOLO BIORITMI"
17 PRINT
18 PRINT
19 PRINT
20 INPUT "DATA DI NASCITA?", G,M,A;GOSUB 200
25 PRINT
26 PRINT
30 LET P1=D
40 INPUT "DATA ATTUALE?", G,M,A;GOSUB 200
50 LET P2=D
60 LET P3=P2-P1
65 PRINT CHR(12)
70 LET F=P3/23-INT(P3/23)
80 LET E=P3/28-INT(P3/28)
90 LET I=P3/33-INT(P3/33)
100 GRAPH
105 PLOT 0,0,0,191
110 PLOT 0,30,255,30
115 ORG 0,30
120 FOR T=0 TO F*6.28 STEP 0.01
125 SET T*40,30*SIN(T)
130 NEXT
132 REM IL PLOT È SEMPRE RIFERITO ALL'ORG PRECEDENTE
133 REM MENTRE L'ORG È SEMPRE IN ASSOLUTO

```

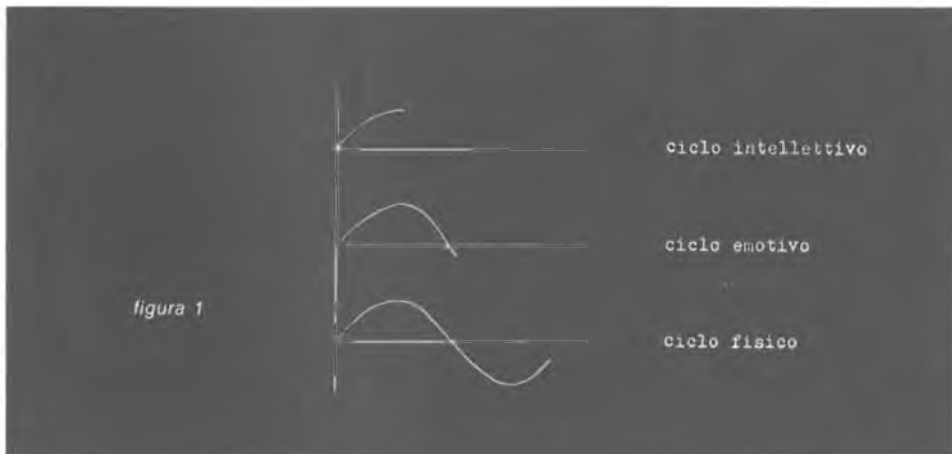


```

135 PLOT 0,65,255,65
140 ORG 0,95
145 FOR T=0 TO E*6.28 STEP 0.01
150 SET T*40,30*SIN(T)
150 NEXT
160 PLOT 0,65,255,65
165 ORG 0,160
170 FOR T=0 TO I*6.28 STEP 0.01
175 SET T*40,30*SIN(T)
180 NEXT
185 AT 23,0
190 END
200 IF A>1900 THEN LET A=A-1900
210 LET D=A*365+INT(A/4)+G
220 IF M=1 THEN GOTO 300
230 IF M=2 THEN LET D=D+1; GOTO 300
240 LET D=D-2; IF A/4=INT(A/4) THEN LET D=D+1
250 IF M>3 THEN LET D=D+1
260 IF M>5 THEN LET D=D+1
270 IF M>6 THEN LET D=D+1
280 IF M>8 THEN LET D=D+1
290 IF M>10 THEN LET D=D+1
300 LET D=D+30*(M-1)
310 RETURN

```

Il risultato grafico è circa il seguente:



È un po' lunga la tracciatura dei grafici, ma a me i disegni a pallini non sono mai piaciuti. Tanto per fare un po' d'esercizio, provate a inserire delle didascalie nei tre grafici.

Io sono un po' pigro e non l'ho ancora fatto.

Suggerimento: tenete presente che siete in modo GRAPH.

Senza altro il tutto è molto suscettibile a migliorie.

Proponetele e vediamo insieme se riesco a migliorare il mio stile piuttosto balbettante.

**ALTRO RELAX e altre curiosità**

Colto da raptus "frettoloso", dovevo battere per prova circa 100 linee di programma, ho fatto un tentativo: barando, tolsi gli spazi tra le frasi, istruzioni ecc. in questo modo:

```
10 FOR I=0 TO 100 STEP 10
20 PRINT I
30 NEXT
```

l'ho battuto così:

```
10FORI=0TO100STEP10;PRINTI;NEXT
```

Impassibile il G5 vi sfornerà la lista dei numeri da 1 a 100!

Ora non è che voglia dire di avere necessità di battere velocemente un listato, ma provate un po' a fare il conto di quanti spazi ci sono in un programma e vedrete che così si risparmia tempo, dita e forse (non lo so ancora) spazio in memoria.


Un altro esempio di compattazione di istruzioni può essere il seguente. Come suggerito dall'Ing. Becattini, si possono modificare i caratteri del sistema, memorizzati all'accensione nelle RAM video, tramite software con un programmino tipo:

```
10 LET I=8456
20 VRAM I,0
30 VRAM I+1,0
40 VRAM I+2,48
50 VRAM I+3,120
60 VRAM I+4,120
70 VRAM I+5,48
80 VRAM I+6,0
90 VRAM I+7,0
```

Il che corrisponde a sostituire il carattere "●" al carattere "!", e ciò finché non si spegne il computer.

Il tutto può essere abbreviato come segue:

```
5 REM LOCAZIONE CARATTERE
10 FOR I=8456 TO 8463
15 REM INTRODUZIONE VALORI DECIMALI DELLE 8 RIGHE
20 INPUT A
30 VRAM I,A
40 NEXT
```

All'INPUT comparirà il "?", chiedendo il valore in decimale della prima riga del carattere, poi della seconda e così via per otto volte, quante sono le righe del carattere a matrice. Finito il calcolo, ogni volta che si preme il tasto  si visualizzerà il nuovo carattere da noi impostato.

È possibile scrivere anche in cinese!

Provate a inserire i dati 0,0,25,37,255,82,93,128; non è cinese?

Allo stesso modo si possono inserire caratteri grafici di nostra invenzione. Bravo, dirà qualcuno a questo punto, e io come faccio a sapere gli indirizzi del set di caratteri? Niente paura, ho già speso tempo io per ricavarmeli. Come?

```
10 FOR I=X TO Y
20 PRINT PVRAM(I)
30 NEXT
```

o meglio, per evitare di riscrivere la riga 10 ogni volta:

```
10 INPUT A,B
25 REM STAMPA VALORI DECIMALI DELLE 8 RIGHE CARATTERE
30 PRINT PVRAM(I)
40 NEXT
50 GOTO 10
```

oppure, ancora meglio, per introdurre solo l'indirizzo di partenza:

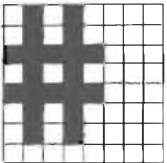
```
10 INPUT A
20 FOR I=A TO A+7
30 PRINT PVRAM(I)
40 NEXT; GOTO 10
```

Comunque, vi scrivo qui di seguito gli indirizzi assoluti dei caratteri di uso comune, esistendo anche un altro set di caratteri completo, ma con sottolineatura e altro ancora.

! 8456-8463	5 8616	I 8776	] 8936	q 9096
" 8464	6 8624	J 8784	↑ 8944	r 9104
# 8472	7 8632	K 8792	_ 8952	s 9112
\$ 8480	8 8640	L 8800	` 8960	t 9120
% 8488	9 8648	M 8808	a 8968	u 9128
& 8496	: 8656	N 8816	b 8976	v 9136
' 8504	; 8664	O 8824	c 8984	w 9144
( 8512	< 8672	P 8832	d 8992	x 9152
) 8520	= 8680	Q 8840	e 9000	y 9160
* 8528	> 8688	R 8848	f 9008	z 9168
+ 8536	? 8696	S 8856	g 9016	{ 9176
, 8544	@ 8704	T 8864	h 9024	9184
- 8552	A 8712	U 8872	i 9032	} 9192
. 8560	B 8720	V 8880	j 9040	~ 9200
/ 8568	C 8728	W 8888	k 9048	
0 8576	D 8736	X 8896	l 9056	
1 8584	E 8744	Y 8904	m 9064	
2 8592	F 8752	Z 8912	n 9072	
3 8600	G 8760	[ 8920	o 9080	
4 8608	H 8768	\ 8928	p 9088	

Esempio pratico:

figura 2

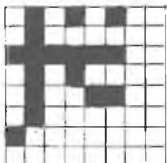


Binario	Deco.
01010000	80
01010000	80
11111000	248
01010000	80
11111000	248
01010000	80
01010000	80
01010000	80
00000000	0

#

Modificando:

figura 4



Binario	Deco.
01010100	84
01000000	64
11111100	252
01010000	80
01001100	76
01000000	64
10000000	128
00000000	0

ja

Una mancanza che ho notato nel G5 sono gli operatori logici AND e OR. Trovandomi nella necessità di dover fare una comparazione del tipo "se  $A > 0$  e se  $B < 100$  allora scrivi C" e mancandomi il logico AND, ho trovato che battendo

`"IF A > 0 THEN IF B < 100 THEN PRINT C"`

il conto tornava.

In pratica, se A non è maggiore di 0, il G5 ignora il seguito e va alla riga successiva; alla stessa maniera se trova che B non è minore di 100, pur essendo  $A > 0$ , esso ignora l'istruzione successiva.

Morale della favola, è un utile artificio per fare l'AND logico mancante.

Per l'OR il discorso è più semplice.

Basta battere:

```
110 IF A > 0 THEN GOTO 1000
120 IF B < 100 THEN GOTO 1000
```

Cioè basta che uno dei due, A o B, sia vero che si ha l'uscita (GOTO 1000).

**NUOVA FINE DEL RELAX e nuovo programmino (l'ultimo se Dio vuole)**

Una delle frasi più ricorrenti della nostra lingua è "Che giorno è il 15 Agosto?".

Si trovasse mai nessuno che rispondesse subito "Giovedì".

Mai.

E il G5 che sta a fare?

Ormai ho preso il via e quindi, forte della subroutine dei Bioritmi, mi cimento.

```
5 REM CANCELLARE IL VIDEO PER ESTETICA
10 PRINT CHR(12)
```

```

12 PRINT
14 PRINT
16 PRINT
20 PRINT
30 INPUT "QUALE GIORNO VUOI SAPERE?";G,M,A
31 REM NON BARATE COI 29 FEBBRAIO!
32 IF M=2 THEN IF G=29 THEN IF A/4=INT(A/4) THEN GOTO 40
35 PRINT "NON È UN ANNO BISESTILE!";GOTO 30
40 PRINT
45 PRINT
50 IF A>1900 THEN LET A=A-1900
60 LET D=A*365+INT(A/4)+G
70 IF M=1 THEN GOTO 150
80 IF M=2 THEN LET D=D+1;GOTO 150
90 LET D=D-2;IF A/4=INT(A/4) THEN LET D=D+1
100 IF M> 3 THEN LET D=D+1
110 IF M> 5 THEN LET D=D+1
120 IF M> 6 THEN LET D=D+1
130 IF M> 8 THEN LET D=D+1
140 IF M>10 THEN LET D=D+1
150 LET D=D+30*(M-1)
160 LET X=INT((D/7-INT(D/7))*7)
170 IF X=0 THEN PRINT "È MARTEDÌ";GOTO 30
180 IF X=1 THEN PRINT "È MERCOLEDÌ";GOTO 30
190 IF X=2 THEN PRINT "È GIOVEDÌ";GOTO 30
200 IF X=3 THEN PRINT "È VENERDÌ";GOTO 30
210 IF X=4 THEN PRINT "È SABATO";GOTO 30
220 IF X=5 THEN PRINT "È DOMENICA, SE DIO VUOLE!";GOTO 30
230 IF X=6 THEN PRINT "È LUNEDÌ";GOTO 30

```

Per uscire dal programma basta dare una lettera anziché tre numeri in risposta all'INPUT.

A proposito, le date vanno inserite così: 23,2,83 oppure 23,2,1983, non con barre, trattini o all'inglese, altrimenti si dovrebbe fare il programma notevolmente più complicato.

## ULTIMO RELAX

Il mio grande vizio (o pregio) è la curiosità. C'è un tasto, denominato CONTROL (CTRL) che mi ha sempre affascinato, fin da quando terminai la costruzione della tastiera. Avrei, secondo voi, potuto lasciarlo in pace?

Mai e poi mai.

Risultato: la folgorante scoperta che battendo due soli tasti potevo dare istruzioni complete.

Vediamo come.

Innanzitutto quanto segue vale solo se preceduto da un numero di linea, cioè nel corso di battitura di un programma. Non meravigliatevi se il risultato immediato è un quadratino bianco; se effettuate un LIST, vi verranno stampate le istruzioni complete. Quindi, definendo al solito CTRL con C+, avremo:

```

C+E GRAPH
C+F BLANK
C+G NOBLANK
C+P SPFORM

```

C+Q SPDEF

C+R SPTYP

C+S AT

C+D TEZT (ci deve essere un errore nelle mie ROM; equivale a TEXT, che infatti non lavora sul mio G5)

Ci sono anche istruzioni riferite evidentemente a futuri usi, tipo registratore o dischi, quali:

C+T SAVE

C+ LABEL

C+X MAP

C+Y FORMAT

C+Z BOOT

C+↑ FILES

C+J RW

C+\  
RO

Infine, il rebus:

C+B GATTO

Si, proprio GATTO. Non chiedetemi altro! Come vedete il G5 non finisce mai di stupire!

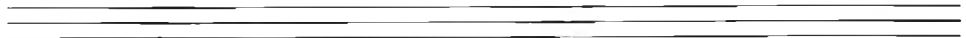
*Qui termina la mia fatica, spesa per stimolare qualcun'altro, senz'altro più pratico di me di programmazione, a scrivere qualcosa di più sul G5, per invogliare quelli che, come me, non amano comperare a scatola chiusa e trovano soddisfazione nel farsi da sé i propri strumenti; oltretutto se poi si guastano sanno dove mettere le mani e il saldatore, invece che inviare tutto alla Ditta X.*



È doveroso comunque un plauso all'Ing. Becattini per questa pregevole realizzazione, che sono sicuro avrà stuzzicato l'appetito di quanti sono in bilico tra il "lo compro" e il "non so".

Spero vivamente che l'Autore, stimolato da future nostre numerose richieste, voglia derogare dalla sua affermazione che il G5 "è un giocattolo" ed estendere, magari un po' per volta, il GBASIC con alcune funzioni che, senza portarlo a livelli IBM, esaltino le sue capacità, tipo variabili alfanumeriche, variabili stringhe e poche altre, di cui personalmente ho sentito il bisogno tentando di tradurre alcuni programmi di altri computer.

Suvvia, ingegnere, ci faccia contenti! \*\*\*\*\*



**NON PERDERE IL N. 12 DI  
"CQ ELETTRONICA", CONTIENE  
L'INDICE ANALITICO 1983**

# il Micro-Professor (MPF III) della MULTITECH

## A) Caratteristiche generali

Calcolatore di design estremamente avanzato e perfettamente ergonomico riassume nella sua configurazione caratteristiche uniche nel suo genere.

Tastiera in grado di gestire, in modo immediato, maiuscole, minuscole, 40 o 80 colonne, editing, inserzione/cancellazione caratteri, tasti funzione, tastierino numerico e funzioni Basic one key.

Il calcolatore distaccato dalla stessa, è in grado di lavorare su grafica ad alta (6 colori) o bassa risoluzione (16 colori), su TV e/o monitor, generare suoni via software con istruzioni Basic specializzate.

È già completo, nella configurazione base, di interfaccia per stampante (Centronic) con 4 selezioni dirette (per Epson/C.itoh/termica/altre stampanti) con grafica predefinita, interfaccia per registratore, interfaccia analogica per paddle, slot (interni) di espansione per schede tipo Z80 (x CP/M) e drive-disk, ecc.

All'esterno è presente uno slot compatibile Apple II e per interfacce speciali (slot 2).

La memoria RAM è di 64 k più 2 kbyte sta-

lici per le 80 colonne di testo. La memoria ROM è di 24 kbyte (con interfaccia Basic Apple Soft).

## B) Caratteristiche tecniche

- 1) CPU 6502, 1 MHz
- 2) ROM: 24 k con interprete Basic Apple Soft
- 3) RAM: 64 k dinamiche più 2 k statiche per le 80 colonne di testo (espandibili a 128 k con scheda supplementare)
- 4) Video
  - testo: 40x24 / 80x24 (2 pagine)
  - grafica GR: 40x48 16 colori (2 pagine)
  - grafica HGR: 280x192 6 colori (2 pagine)
- 5) Caratteri: 96 ASCII maiuscole e minuscole più funzioni Basic one key.
- 6) Tastiera: distaccata con 90 tasti multifunzioni
- 7) Interfacce residenti
  - per registratore (input/output)
  - per monitor
  - per TV (color, e sistema PAL)
  - per paddle o joy-stick (a 9 pin)

-per stampante in standard Centronic con software grafico residente

-per Epson e C.itoh selezionabile attraverso switch.

-per 3 slot interni disponibili per interfacce originali Multitech (Z80 per CP/M - FDI ecc.)

8) Slot esterno per espansioni (corrispondente allo slot 2 dell'Apple II E) e con lo stesso compatibile.

9) Generatore di suoni a 36 toni programmabile anche da Basic con istruzioni specializzate.

10) Alimentatore entrocontenuto tipo switching

11) Dimensioni CPU: 40x6x25 cm  
dimensioni tastiera: 40x3x18,4 cm

## C) Opzionali

- Interfaccia per 2 drive-disk
- Drive disk slim-line (come MPF II)
- Interfaccia seriale RS 232 C
- Schede Z80 per CP/M
- Generatore caratteri cinesi
- Monitor fosfori bianchi/verdi/arancio e color
- Stampanti 40 colonne (termiche) 80/132 colonne (carta comune)
- Registratore
- Paddle o joy-stick



Distributore esclusivo  
DIGITEK COMPUTER  
via Marmolada, 9/11  
43058 SORBOLO (PR)  
tel. 0521/69635

# Imparate oggi il Personal Computing

Al Centro Informatica Personale, il primo "Personal Computer Learning Center" in Italia, indipendente dalle case costruttrici, potete trovare:

**1** Corsi sul personal computer, per imparare ad usarlo, per capire cos'è e cosa può fare. Sono disponibili i seguenti corsi:

## INTRODUZIONE AL PERSONAL COMPUTER

Cos'è e cosa si può fare con un personal  
Durata: 8 ore, concentrate o distribuite

## INTRODUZIONE AL VISICALC

Corso sul programma applicativo più diffuso;  
utile per previsioni, analisi economiche, ecc.  
Durata: 4 ore, anche di sabato

## INTRODUZIONE AL BASIC

Corso introduttivo alla soluzione di problemi  
su personal computer in linguaggio BASIC  
Durata: 12 ore, concentrate o distribuite

Questi corsi non richiedono alcuna precedente esperienza di calcolatori o di matematica. Sono possibili corsi individuali, per piccoli gruppi e corsi per organizzazioni.

**2** Possibilità di usare, da soli o con assistenza, personal computer di diversi tipi.

**3** Servizio di gestione dati contabili, indirizzari, archivi, per attività di piccole dimensioni; gestione testi, consulenza per applicazioni, sviluppo di programmi applicativi.

Per ulteriori informazioni:

Centro Informatica Personale Srl  
Personal Computer Learning Center  
corso Vercelli, 9 - 20144 MILANO  
tel. 02-4390367



### DIGIMODEM IIA: IL PRIMO DEMODULATORE CON FILTRI DIGITALI

Soluzioni circuitali innovative hanno reso possibile la realizzazione di un mod./demodulatore economico, facile da usare, e dalle prestazioni altamente professionali.



- # filtri digitali separati su mark e space;
- # mark selezionabile 2125 ÷ 1275 Hz con regolazione fine;
- # shift selezionabile 170, 425, 850 Hz con possibilità variazione;
- # regolazione della larghezza di banda dei filtri: 5-100 Hz;
- # possibilità di ricezione anche su di un solo tono;
- # circuito con decisione di soglia completamente digitale;
- # originale ed efficiente sistema di sintonia a VU meter + due led;
- # possibilità di collegamento oscilloscopio esterno per sintonia ad ellissi;
- # possibilità di misurazione con frequenzimetro esterno delle frequenze di sintonia dei filtri e dello shift;
- # input audio con dinamico di ~ 60 dB;
- # out standard AFSK 2125-2295 Hz per TX in SSB o altri a richiesta;
- # out per modulazione FSK con fotoaccoppiatore;
- # in-out digitale TTL, e courant loop con alimentazione isolata.

### RY-83: DECODIFICATORE VISUALIZZATORE TTY-CW



UN APPARECCHIO CHE NON PUÒ  
MANCARE IN UNA MODERNA STA-  
ZIONE D'ASCOLTO

Gestito a microprocessore, decodifica e visualizza su monitor video e/o televisore un segnale tipo TTY (codice ASCII e BAUDOT) o CW.

- # Input 20 mA courant loop; isolamento con fotoaccoppiatore;
- # Codici ASCII & BAUDOT, 45.5, 50, 75, 100, 110, 150, 200 bits/sec con commutatore di selezione;
- # Codice Morse esteso, inseguimento automatico di velocità;
- # Output video per monitor e UHF per televisore;
- # Formato 512 caratteri 32 colonne x 16 righe con scrolling;
- # Memoria testo di 1024 caratteri: richiamo della pagina precedente con pulsante monostabile;
- # Pulsante «letter» in baudot;
- # Line Feed automatico;
- # Possibilità di correzione ortografica.

A richiesta versione con microdemodulatore incorporato per input audio.

#### • CONDIZIONI DI VENDITA:

I prezzi si intendono I.V.A. esclusa. Spedizioni in contrassegno con spese postali a carico del destinatario. Imballo gratis. Per richiesta cataloghi e informazioni scritte inviare L. 1.000 in francobolli a titolo di parziale rimborso spese. Per quantità, per rivendita, per esecuzioni particolari o per applicazioni personalizzate, richiedere offerta scritta.

#### • PREZZI:

DECODIFICATORE RY-83 .....	L. 270.000
DECODIFICATORE RY-83 CON MICRODEM .....	L. 310.000
DEMODULATORE DIGIMODEM II A .....	L. 240.000



# HAM CENTER

di PIZZIRANI P. & C. s.a.s.

VIA CARTIERA, 23 - ☎ (051) 84.66.52 - 84.28.58  
40044 BORGONUOVO DI PONTECCHIO MARCONI  
(BOLOGNA) ITALY

## Amanti della Radio. Elettronici. Hobbysti.

TRANSISTOR B.F  
TRANSISTOR R.F  
RESISTENZE  
CONDENSATORI  
COMPONENTI PASSIVI  
CIRCUITI INTEGRATI  
TOROIDI "AMIDON"  
FERRITI  
SUPPORTI PER BOBINE  
CUFFIE PER HI-FI ED S.S.B  
MINUTERIE VARIE

**RICHIEDETE IL NOSTRO LISTINO**

PER NECESSITÀ RIVOLGETEVI AL NS. INDIRIZZO E SAREMO LIETI DI SODDISFARE LE VS. ESIGENZE.

SPEDIZIONI IN CONTRASSEGNO + SPESE POSTALI

...Ricordate **HAM CENTER** è sinonimo di garanzia e qualità!!!