

# PROGRAMMATORE DI EPROM VIA CENTRONICS

DALLA 2716 ALLA 27512

*Un programmatore dovrebbe poter lavorare sul massimo numero possibile di tipi di EPROM ed essere costruito economicamente, con componenti facili da trovare in commercio. Nel caso ideale, dovrebbe addirittura funzionare con qualsiasi computer. Lo schema è facilmente comprensibile e c'è anche un software controllato a menù, per IBM e compatibili.*

di Peter Seng

**C**i sono due categorie di programmatori di EPROM: nel settore professionale si trovano per lo più apparecchi autonomi (stand-alone)

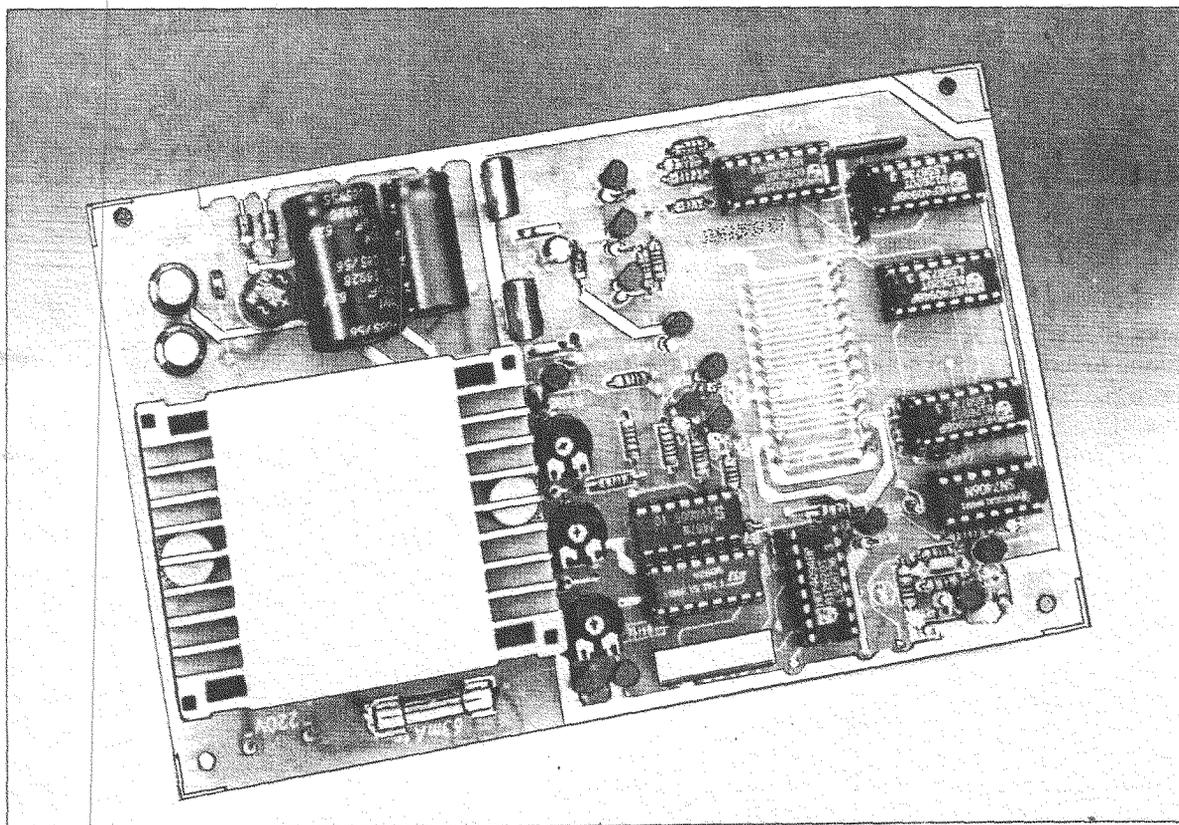
con il proprio processore ed un'interfaccia per l'inserimento dei dati. Di norma vengono utilizzati per la duplicazione da EPROM master e per questo sono

equipaggiati con almeno due (alcuni arrivano fino a nove) zoccoli di programmazione, in modo da poter produrre contemporaneamente parecchie copie.

Gli apparecchi per i softwaristi sono invece costruiti in maniera fondamentalmente diversa.

Il costoso zoccolo a forza di inserzione zero (ZIF) è previsto in un solo esemplare e la parte elettronica è limitata all'adattamento di impedenza del segnale, all'indirizzamento ed alla produzione della tensione di programmazione per la EPROM. Il controllo vero e proprio è affidato ad un PC.

Il nostro programmatore di EPROM



appartiene alla seconda categoria; grazie allo schema piuttosto semplice, il costo e l'impegno costruttivo saranno limitati, ma non verrà sacrificato nulla in fatto di efficienza.

L'idea alla base del circuito programmatore si basa su un progetto per Apple II, tratto dal periodico specializzato in computer MC dell'Agosto 84. La versione originale di un programmatore di EPROM per PC dell'Autore è stata pubblicata su MC nel Gennaio '87.

Rispetto alle condizioni di allora, la versione attuale rappresenta un considerevole miglioramento per quanto riguarda alcuni importanti particolari: per esempio, l'interfacciamento con computer del tipo Laptop avviene ora senza difficoltà.

La precedente basetta, monofaccia, che doveva essere equipaggiata con un'infinità di ponticelli, è ora in versione a due facce. Questo facilita considerevolmente, tra l'altro, il montaggio dei componenti ed il fissaggio dello zoccolo di programmazione. Il software è l'attuale versione 3.0, molto più comoda e "trasparente". Con questo circuito si potranno programmare praticamente tutti i tipi di EPROM oggi disponibili, dalla 2716 alla 27512/27513.

La selezione del tipo di EPROM avviene esclusivamente via software, con l'eccezione dei chip 2716 da 2 K x 8, ormai piuttosto vecchi, per la cui programmazione è necessario azionare a mano un apposito commutatore.

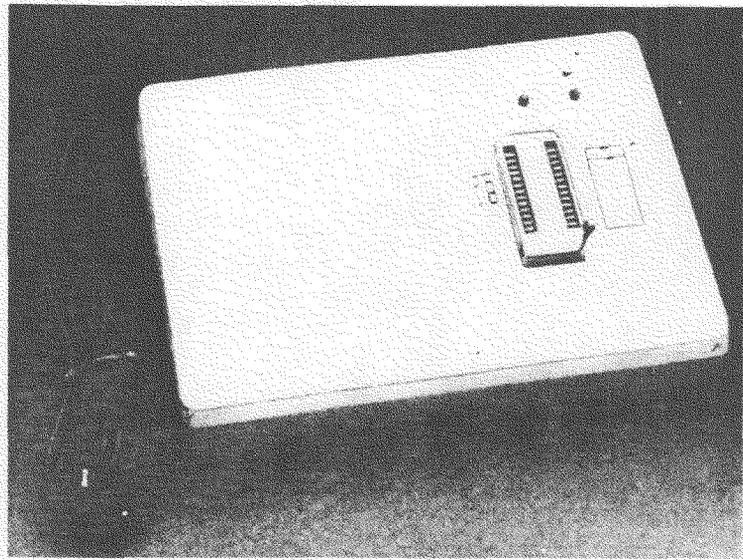
Come interfaccia tra il computer di controllo ed il programmatore abbiamo scelto quella standard parallela.

Così non solo c'è la possibilità di far gestire l'apparecchio da qualsiasi computer munito di interfaccia parallela Centronics/compatibile, ma non è nemmeno più necessario inserire una scheda di interfaccia in una slot interna. Comunque, il software qui pubblicato vale esclusivamente per IBM-compatibili.

## Programmazione

La programmazione di una EPROM avviene fondamentalmente nel seguente modo: dapprima vengono predisposte la giusta tensione di alimentazione (5 oppure 6 V) e la giusta tensione di programmazione (12,5 V/21 V/25 V). Successivamente, tramite le linee di indirizzamento, vengono scelti gli indirizzi desiderati, applicando poi agli ingressi dei dati le configurazioni di bit (byte) da programmare.

Un impulso di durata ben definita



applicato all'ingresso di programmazione attiva il trasferimento del byte di dati nella EPROM. Al termine della programmazione, si leggono di nuovo i byte programmati, confrontandoli con i dati originali.

Se il confronto ha esito positivo, la programmazione è finita.

Uno dei criteri più importanti è stabilire esattamente la durata dell'impulso di programmazione. I precedenti algoritmi operavano in generale con un impulso della durata di 50 ms; a prima vista, questo tempo non sembra lungo ma, con l'aumento della capacità delle EPROM, il tempo totale di programmazione aumenta: per esempio, quasi 7 minuti per una 2764.

I moderni programmatori (e le moderne EPROM) si avvalgono di procedure più veloci (per la 2764, circa 35 secondi). In questi casi, la durata dell'impulso di programmazione non è fissa, ma viene ottenuta in base alla programmazione stessa (si tratta dei cosiddetti algoritmi interattivi).

La durata del primo impulso è limitata ad 1 ms; immediatamente dopo avviene la lettura.

Nel caso il contenuto della cella di memoria non sia ancora corretto, avviene una nuova programmazione da 1 ms, con successiva nuova lettura. Il numero dei tentativi fatti prima di ottenere la giusta programmazione viene registrato.

### Hardware:

- Programmazione di EPROM dal tipo 2716 al 27512/27513
- Alimentazione di programmazione 12,5/21/25 V
- Nessun componente speciale
- Immune al cortocircuito
- Funzionamento tramite interfaccia Centronics
- Non occupa slot interne
- Alimentazione autonoma

### Software:

- Per computer compatibili XT-AT (MS-DOS)
- Adattamento automatico alla scheda grafica ed al clock del computer
- Controllo completo tramite menù
- Programmazione macro
- Grafica a colori
- Riconoscimento automatico del tipo di EPROM
- Gestione file integrata
- Editor Hex/ASCII integrato
- Algoritmi da 1 ms standard e veloci
- Possibile programmazione a 16 bit (High/low byte)
- Varie utility

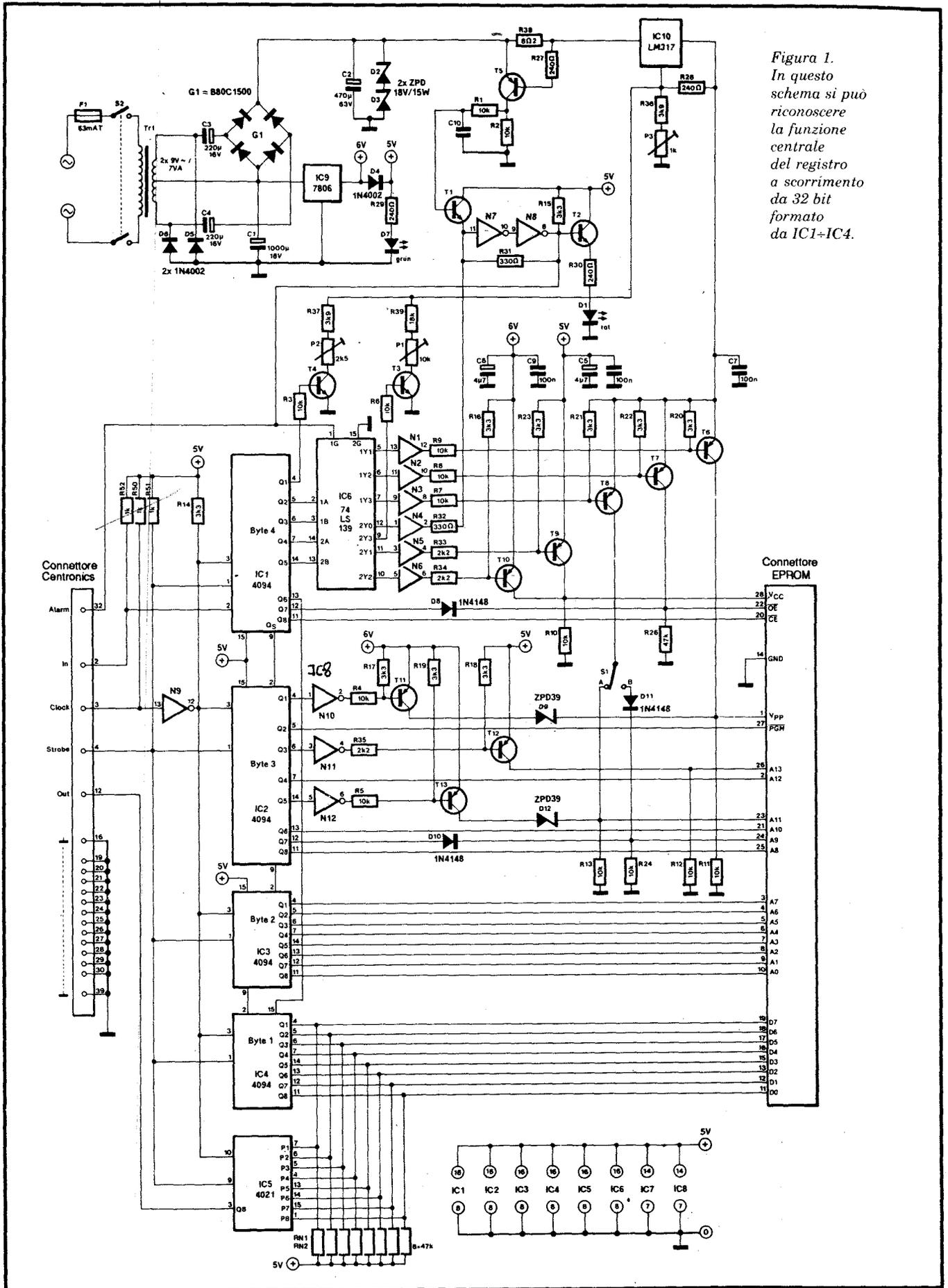


Figura 1.  
 In questo schema si può riconoscere la funzione centrale del registro a scorrimento da 32 bit formato da IC1-IC4.

Se, per esempio, sono stati necessari 3 impulsi da 1 ms, alla fine ha luogo una sovraprogrammazione con durata tripla (9 ms).

Il numero dei tentativi è di solito limitato a 15 o 25. Dopo di questi, se non si constata il successo dell'operazione, ci sono buoni motivi per supporre che la EPROM sia difettosa.

Esistono parecchi algoritmi di programmazione veloce che si distinguono soprattutto per il numero delle ripetizioni degli impulsi da 1 ms e per la durata della sovraprogrammazione. Contrariamente all'algoritmo standard, i sistemi veloci lavorano anche con una tensione di alimentazione per la EPROM aumentata a 6 V.

Ogni produttore di circuiti integrati prescrive il sistema da usare per la programmazione dei diversi tipi di EPROM.

A questo sistema è sempre necessario attenersi, non solo per garantire la stabilità a lungo termine dei dati, ma anche per non mettere in pericolo l'incolumità della EPROM. Il tentativo di programmare una EPROM di vecchio tipo mediante un algoritmo superveloce finisce quasi sempre con la distruzione totale del componente.

### Questioni di principio

Il circuito utilizza l'interfaccia parallela Centronics per il trasferimento seriale bidirezionale dei dati; questo è possibile soltanto perché l'uscita per stampante Centronics di un computer IBM-compatibile dispone anche di alcune linee di ingresso, per esempio PE (mancanza carta), tramite la quale la stampante collegata può mandare un segnale al computer quando la carta è terminata.

Delle 8 linee di uscita (D0-D7) dell'interfaccia, sono utilizzate soltanto le prime tre. Una come uscita dati (Out), una come segnale di Clock e la terza come segnale di Strobe.

Tramite l'ingresso PE, ritorna l'informazione seriale dal programmatore al computer. Il protocollo di trasferimento è stato mantenuto più semplice possibile. In corrispondenza a ciascun fronte positivo del segnale di clock, viene rilevato e trasferito il livello istantaneo della linea IN di un registro a scorrimento da 32 bit (conversione da seriale a parallelo).

Dopo la trasmissione di 4 byte (32 bit), l'impulso di Strobe attua il trasferimento alle 32 uscite dei dati memorizzati nel registro a scorrimento.

**Tabella 1. Il cavo di connessione tra computer e programmatore deve essere almeno a 6 poli.**

Interfaccia Centronics	Programmatore di EPROM
PE Pin 12	Pin 12 Out (O)
D2 Pin 04	Pin 04 Strobe (S)
D0 Pin 02	Pin 02 In (I)
D1 Pin 03	Pin 03 Clock (C)
Gnd Pin 18-25	"Pin 16, 19-30, 33 Gnd"
Error Pin 15	Pin 32 Alarm* (A)

\* La linea di errore può essere tralasciata.

**Tabella 2. Con 4 byte vengono controllati tutti i processi del programmatore di EPROM.**

Byte 1	Pin	Byte 2	Pin	Byte 3	Pin
Q1	19	Q1	3	Q1	1
Q2	18	Q2	4	Q2	27
Q3	17	Q3	5	Q3	26
Q4	16	Q4	6	Q4	2
Q5	15	Q5	7	Q5	23
Q6	13	Q6	8	Q6	21
Q7	12	Q7	9	Q7	24
Q8	11	Q8	10	Q8	25

Byte 4	Pin
Q1	High: 12,5 V <sub>pp</sub> /Low: 21 V <sub>pp</sub>
Q2	
Q3	
Q4	
Q5	
Q6	High: lettura/basso: scrittura
Q7	
Q8	

\* Con S1 e Q1-Q5 viene stabilito il modo di funzionamento. L'assegnazione è fornita separatamente nella tabella sottostante.

S1	Q1	Q2	Q3	Q4	Q5	Funzione
B	H	H	H	X	L	12 V AN A (Identifier-Mode)
A	X	H	H	X	X	Pin 23 V <sub>pp</sub>
X	X	H	L	X	X	Pin 1 V <sub>pp</sub>
X	X	L	H	X	X	Pin 22 V <sub>pp</sub>
X	X	L	L	X	X	V <sub>pp</sub> esclusa
X	L	X	X	H	H	25 V <sub>pp</sub>
X	X	X	X	H	L	Pin 28 5 V V <sub>cc</sub>
X	X	X	X	L	H	Pin 28 6 V V <sub>cc</sub>
X	X	X	X	L	L	Pin 28 0 V/reset circuito di protezione

H: livello alto; L: livello basso; X: livello indifferente.

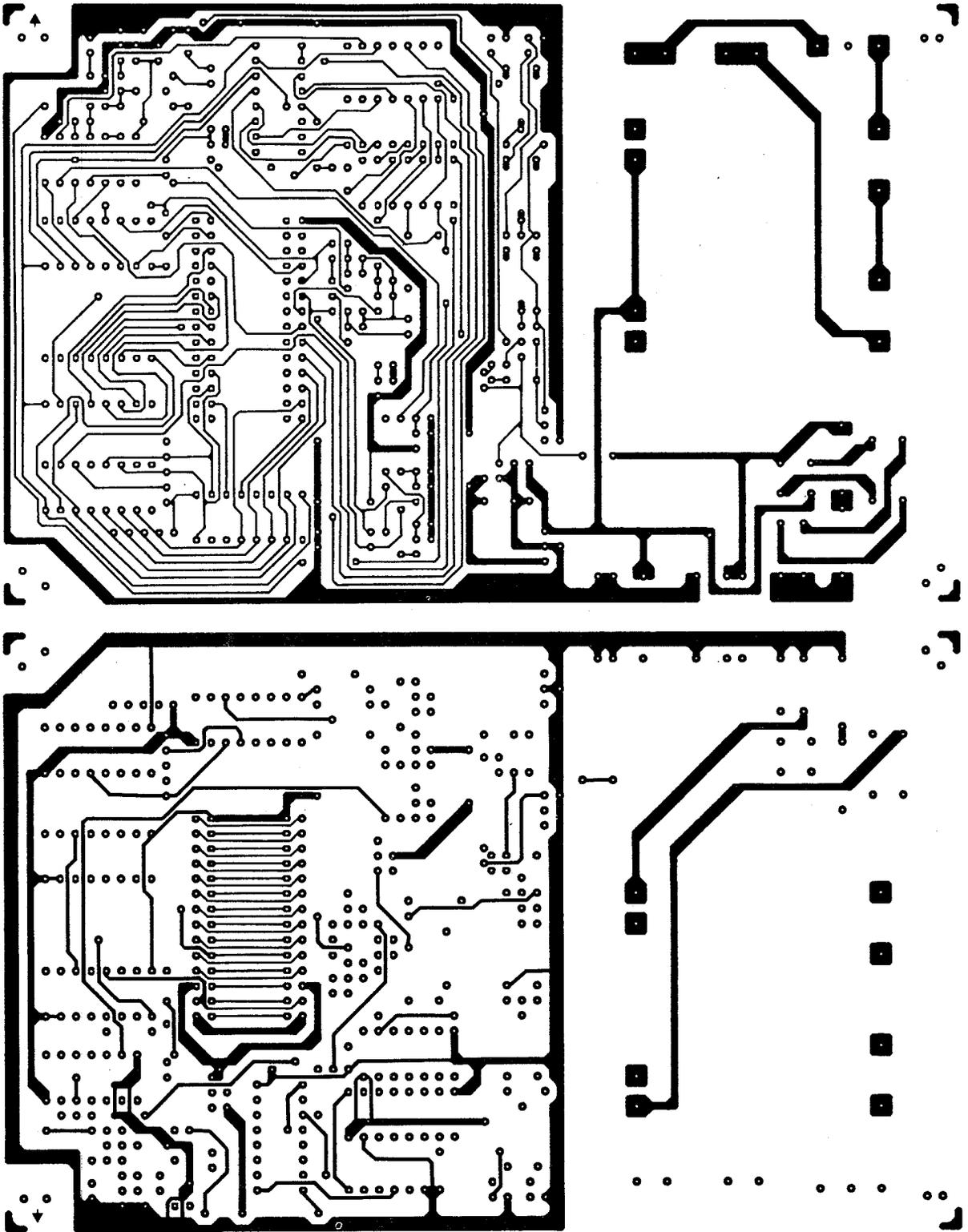


Figura 2a. Circuito stampato in scala 1:1 - Lato rame e lato componenti.

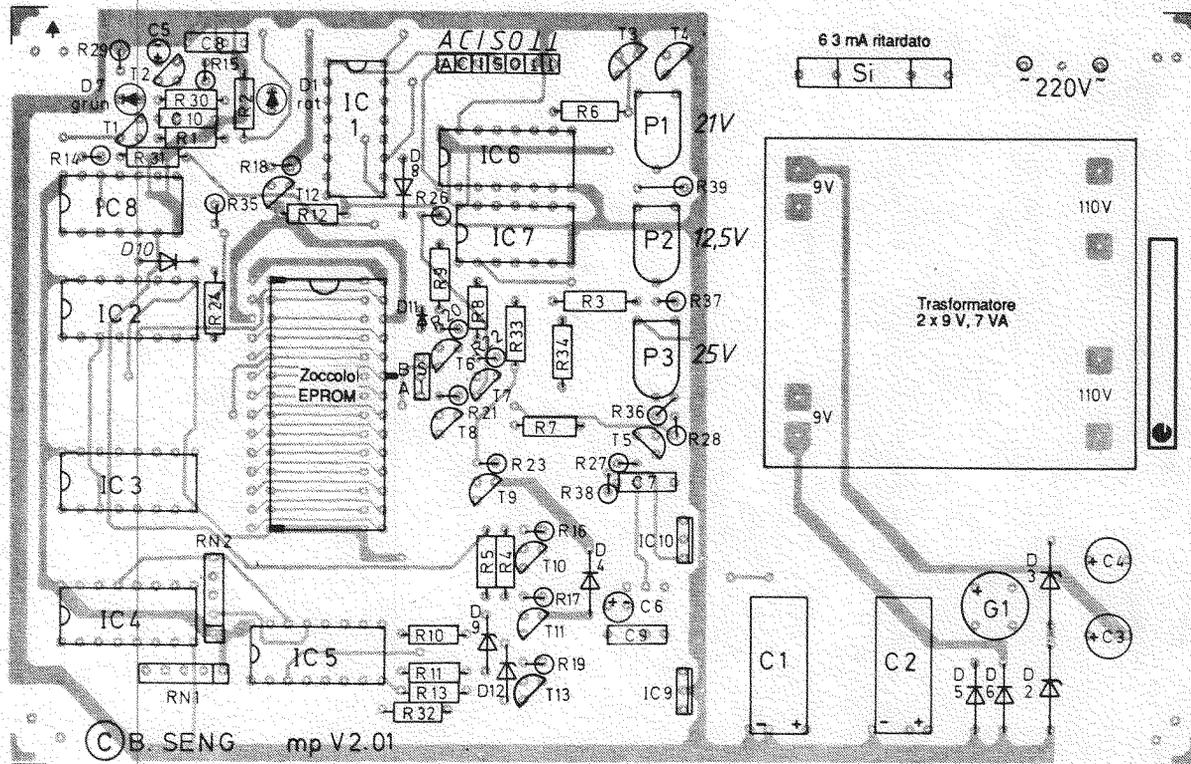


Figura 2b. La disposizione dei componenti non evidenzia nessuna particolare difficoltà; solo lo zoccolo della EPROM da programmare dovrà essere montato sul lato saldature.

I primi 16 bit (byte 1, byte 2) sono disponibili direttamente allo zoccolo della EPROM, dove svolgono la funzione delle linee di dati D0-D8 e rispettivamente dei segnali di indirizzamento A0-A7. Dai successivi 16 bit (byte 3, byte 4) vengono ricavati i restanti segnali di indirizzamento A8-A13 e tutti i segnali di controllo per la regolazione e la commutazione delle tensioni, nonché per la selezione della EPROM. Per poter rileggere il contenuto della EPROM anche in modo seriale, è necessario un altro registro a scorrimento da 8 bit (conversione da parallelo a seriale), collegato ai terminali dei dati D0-D8 dello zoccolo per la EPROM. Il byte di dati viene convertito in una configurazione di bit seriali, che raggiungono il computer tramite il piedino 12 dell'interfaccia Centronics. Clock e Strobe saranno utilizzati come segnali di controllo anche per la procedura di lettura. Per essere precisi, volendo collegare il programmatore non occorre un collegamento Centronics completo, ma soltanto una porta a 4 bit con 3 uscite ed 1 ingresso.

### Schema elettrico

Da quanto detto finora, è facile interpretare lo schema elettrico. All'estrema sinistra (Figura 1) si trova il connettore per l'interfaccia Centronics; subito accanto si trova il registro a scorrimento da 32 bit formato da IC1-IC4.

I dati seriali provenienti dal computer pervengono direttamente al piedino 2 (ingresso dati) di IC1. Tutti i quattro 4094 sono collegati in cascata tramite le uscite dati (Qs, piedino 9) e gli ingressi dati (piedino 2). I primi otto bit che il computer invia al programmatore di EPROM vengono memorizzati in IC1. Il secondo byte di dati sostituisce il contenuto di IC1, che inoltra la prima sequenza di bit verso IC2. Il tutto prosegue fino a quando sono stati trasmessi quattro byte. Allora il byte 1 è memorizzato in IC1, il byte 2 in IC2, e così via.

La trasmissione dei dati dal computer al programmatore avviene pertanto in forma di pacchetti da 4 byte, anche quando deve essere variato lo stato di un unico bit del primo byte.

È facile riconoscere nello schema il significato dei byte 1 e 2 e dei bit Q8-Q6/Q4 (byte 3): sono, rispettivamente, i segnali di dati e di indirizzamento. Le cose sono un po' meno chiare per quanto riguarda i restanti 12 bit. I numeri dei piedini e le designazioni sullo zoccolo si riferiscono tutti normalmente ad EPROM con 28 piedini.

Passiamo ora all'alimentazione. Il trasformatore eroga due volte 9 V, dai quali vengono ricavate cinque diverse tensioni. IC9 emette una tensione stabilizzata di 6 V, ridotta poi a circa 5 V da D4. La linea a 5 V alimenta tutti gli integrati logici, la EPROM ed il segnalatore luminoso di accensione R29/D7. IC10 stabilizza a 25 V la tensione di programmazione regolabile con P3.

In parallelo ad R36 e P3 ci sono R37/P2/T4 ed R39/P1/T3. Se uno dei due transistor T3 o T4 è in conduzione, il relativo potenziometro risulta collegato all'ingresso di controllo dell'LM317.

La tensione d'uscita di IC10 dipende anche dalla regolazione rispettivamente di P2 e di P1.

P1 serve a regolare la tensione di programmazione di 21 V, mentre P2 fa lo stesso per quella di 12,5 V. I transistor T6-T8 commutano la tensione regolata ai giusti piedini dello zoccolo porta EPROM, mentre il percorso del segnale d'uscita di T8 dipende dalla posizione del commutatore S1.

Quest'ultimo permette di commutare la tensione di programmazione (per le EPROM a 24 piedini) dal piedino 23 al piedino 24 (A9) dello zoccolo ZIF.

Quando la tensione di programmazione è presente sulla linea di indirizzamento A9 e contemporaneamente A1-A7 sono a livello basso, potranno essere indirizzate con A1 due speciali locazioni di memoria della EPROM, che contengono il contrassegno del produttore e del tipo.

Da questa "firma", contenuta solo negli integrati più recenti, si potrà ricavare l'algoritmo di programmazione ottimale.

Quando S1 è in posizione B, le EPROM tipo 2716 non potranno più essere programmate ma soltanto lette, perché il piedino 23 funziona da ingresso per gli impulsi di programmazione. Dato il notevole sforzo necessario per sostituire l'azionamento manuale del commutatore con una soluzione software, abbiamo lasciato questo piccolo svantaggio perché le 2716 stanno diventando sempre più rare.

I transistor T9 e T10 (T12 per gli integrati da 24 piedini) commutano la tensione di alimentazione  $V_{cc}$  applicata allo zoccolo della EPROM: c'è da scegliere tra +5 V con T9, o +6 V quando viene mandato in conduzione T10.

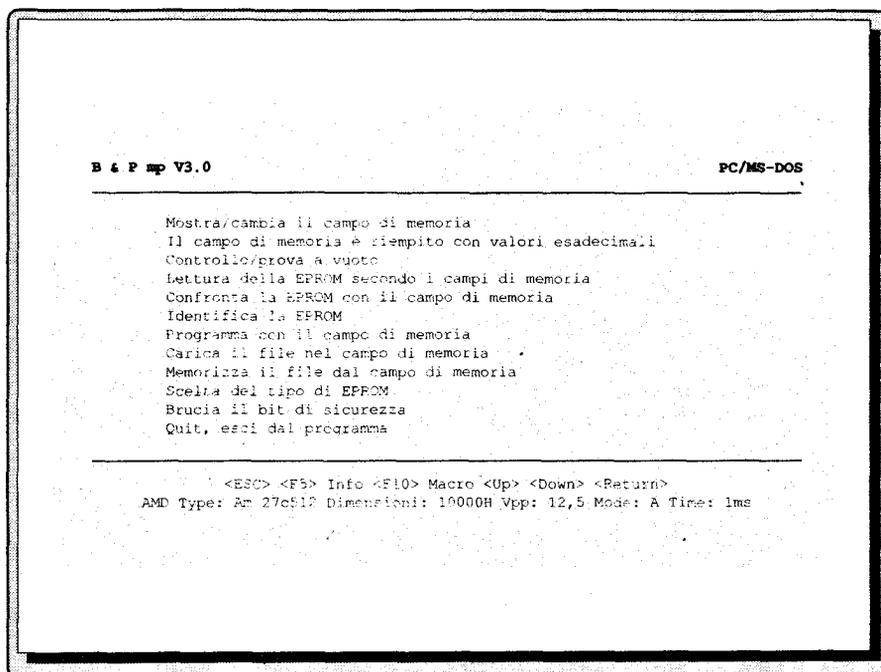


Figura 3. Come si presenta la schermata del menù di apertura.

In alcune condizioni di esercizio, ai piedini 1 ( $V_{pp}$ ) e 23 (A11) si verifica un aumento della corrente assorbita, che non può essere sopportato da un segnale a livello TTL.

Si utilizzano allora T11 e T13 come buffer, che mettono a disposizione 5 V stabilizzati, cioè i 6 V ridotti della caduta di tensione sui diodi zener D9/D12 e della caduta in conduzione dei transistor.

Intorno a T5 è costruito un circuito di protezione contro il cortocircuito.

Tra la base e l'emettitore è inserito il sensore di corrente R38, che viene inserito nella linea proveniente da IC10.

Se la corrente di programmazione aumenta molto al di sopra di 50 mA, T5 va in conduzione (con un ritardo causato da R1/C19) e setta il flip flop costruito con i due invertitori N7/N8. Il segnale d'uscita di N8 esclude le uscite di IC6 (1G, piedino 1) e porta l'uscita di allarme a livello alto.

Contemporaneamente, si accende il LED rosso D1.

## Elenco componenti

### Semiconduttori

D1: LED rosso,  $\phi$  3 mm  
 D2, D3: zener 18 V/0,5 W  
 D4+D6: 1N4002  
 D7: LED verde,  $\phi$  3 mm  
 D8, D10, D11: 1N4148  
 D9, D12: zener 39 V/0,3 W  
 G1: rettificatore B80 C1500  
 T1+T4: BC546  
 T5+T13: BC327  
 IC1+IC4: 4094  
 IC5: 4021  
 IC6: 74LS139  
 IC7: 7407  
 IC8: 7406  
 IC9: 7806  
 IC10: LM317

### Resistori

R1+R13: 10 k $\Omega$   
 R14+R23: 3,3 k $\Omega$   
 R24, R26: 47 k $\Omega$   
 R27+R30: 240  $\Omega$   
 R31, R32: 330  $\Omega$   
 R33+R35: 2,2 k $\Omega$   
 R36, R37: 3,9 k $\Omega$   
 R38: 8,2  $\Omega$   
 R39: 18 k $\Omega$   
 R50+R52: 1 k $\Omega$   
 RN1, RN2: rete di resistori  
 SIL 4 x 47 k $\Omega$

### Condensatori

C1: 1000  $\mu$ F/16 V, elettrolitico radiale

C2: 470  $\mu$ F/16 V, elettrolitico radiale  
 C3, C4: 220  $\mu$ F/16 V, elettrolitici radiali  
 C5, C6: 4,7  $\mu$ F/16 V, elettrolitici radiali  
 C7+C10: 100 nF, ceramici

### Varie

S1: deviatore unipolare  
 S2: interruttore bipolare di rete  
 F1: fusibile da 63 mA, ritardato, con supporti per circuito stampato  
 Tr1: trasformatore piatto, 2 x 9 V/7 VA  
 1: zoccolo Textool a 28 piedini, tipo 228-3345-00-0605  
 1: spina Centronics

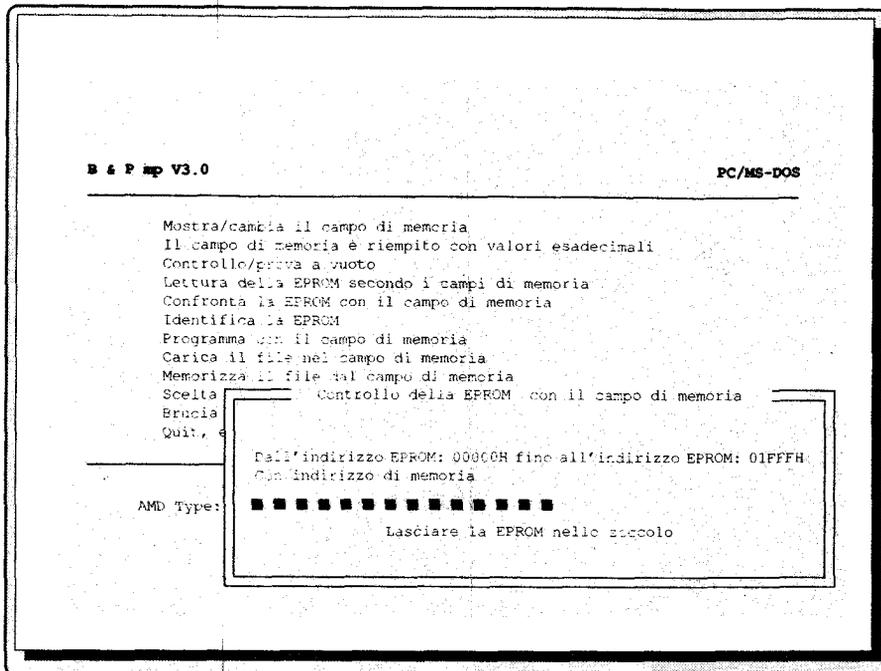


Figura 4. La situazione viene visualizzata costantemente in tutti i processi di scrittura e lettura.

Il flip flop può essere resettato solo dal software oppure, come al solito, spegnendo e riaccendendo il programmatore.

Si chiarisce così anche il significato dei rimanenti bit dei quattro byte di controllo: con essi vengono pilotati i diversi transistor. Un'eccezione è formata dai 5 bit Q2-Q6 del byte 4: Q2-Q5 vanno agli ingressi del decodificatore binario a 2 bit IC6, che decodifica in tutto 7 possibili configurazioni di bit.

Q6 va direttamente all'ingresso di abilitazione dell'uscita di IC4. Il 4094 necessita di un livello logico 1 al piedino 15 perché la configurazione di bit memorizzata possa apparire alle uscite. Con un livello basso, le linee di uscita rimangono ad alta impedenza; questa condizione viene stabilita mediante Q6, quando la EPROM deve essere letta.

Per comodità, il significato di tutti i bit è stato raccolto nella Tabella 2.

## Costruzione

Il montaggio dei componenti sulla basetta non è affatto difficile. Montare e saldare come al solito tutti i componenti; soltanto lo zoccolo ZIF dovrà essere saldato sul lato saldature della basetta a doppia faccia con fori metallizzati. La (purtroppo) grande finestra per lo zoccolo verrà praticata sul mobiletto con trapano e lima.

Fissare la basetta sul coperchio del

mobiletto, con il lato saldature e lo zoccolo rivolti verso l'esterno. Gli altri fori sono tutti circolari: due per i LED, uno per il commutatore S1 e due per il passaggio dei cavi di rete e di interfaccia. I collegamenti alla basetta sono eseguiti secondo la consueta disposizione a matrice. Volendo utilizzare un connettore a spina, non ci saranno problemi con i normali connettori ad una sola serie di contatti. Per il collegamento ai 220 V, sarà opportuno utilizzare una morsettiera per circuito stampato. Attenzione comunque a bloccare bene il cavo di rete per prevenire eventuali strappi.

## Il software

Non si limita soltanto al vero e proprio programma di controllo, ma comprende anche alcuni file ausiliari. Tutti i programmi sono stati progettati in Turbo-Pascal, versione 4.0.

Quando il programmatore è stato collegato al computer, tramite il cavo per la stampante, ed è stato acceso, si comincia dapprima con il programma MP.EXE, che comunica il collegamento dell'interfaccia al programmatore (LPT1, LPT2 oppure LPT3), il modo e la frequenza di clock della CPU nel computer nonché la condizione di attività dell'adattatore monitor. Inoltre il programma si configura in maniera autonoma: non è quindi necessario un setup, come avviene di solito.

L'azionamento è possibile in due modi: una voce del menù può essere richiamata battendo la lettera iniziale oppure posizionando il cursore colorato con i tasti-freccia, per poi dare conferma con Return. I diversi punti del menù si spiegano da soli:

- Mostra/modifica il campo di memoria
- Riempi il campo di memoria con valori esadecimali
- Controllo/prova di memoria vuota
- Identifica la EPROM
- Programma con il campo di memoria
- Carica il file del campo di memoria
- Memorizza il file proveniente dal campo di memoria
- Scelta del tipo di EPROM
- Brucia il bit di sicurezza
- Esci

Attenzione soltanto che, prima della programmazione o della lettura di una EPROM, sia stato impostato correttamente il tipo di EPROM. I processi di scrittura o di lettura vengono visualizzati sul monitor e quindi possono essere seguiti in ogni istante.

Nel programma di controllo è contenuto un comodo editor per schermo, con il quale potranno essere modificati presto e facilmente i file EPROM letti o salvati sul dischetto.

Sul dischetto ci sono diversi programmi di conversione dei file: ad esempio, per poter suddividere i file esadecimali in high byte e low-byte (per l'utilizzo delle EPROM organizzate byte-wide dei sistemi a 16 bit).

Il programma Exchange converte i file esistenti nei più diversi formati di dati. MP.EXE necessita sempre del formato binario: all'occorrenza, si dovrà effettuare prima la relativa conversione. La sintassi sarà per esempio: Exchange Infile.hex Outfile.bin.

Le macro permettono di memorizzare in un file le impostazioni da tastiera che verranno eseguite in modo autonomo. Per l'esecuzione delle macro è previsto un editor integrato. Sul dischetto ci sono 3 esempi di macro, 1 file batch e diversi file di prova. Con l'aiuto di questo esempio sarà facile effettuare la programmazione delle macro. Con lo stesso file Batch si potrà far partire il programma di controllo dal piano DOS, chiamando contemporaneamente una macro.

Richieste di software, hardware e consigli dell'Autore si possono indirizzare direttamente a:

Peter Seng  
Ludwig-Dürr-Straße 10  
7320 Göppingen