

```

###      ### ###      #####      #####      #####
##### ##### ### ###      ### ###      ### ###      ###
### ### ### ### ###      ###      ### ###      ###
###      ### ### ###      #####      ### design
###      ### ### ###      ### ###      ###      ###
###      ### ### ###      ### ###      ###      ###
###      ### ### ###      ### ###      ###      ###
###      ### ###      #####      ###      ###      ###

```

BOLLETTINO DEL CLUB UTENTI MICRO DESIGN

MARZO - APRILE 1985

In questo numero:

- Console CSA-001 per microprocessore Z80.
- Precisazioni sul PASCAL.
- Il foglio elettronico MICROCALC.
- Modifiche ai progetti CFD-001 e CVP-001 per funzionamento a 4Mhz utilizzando le eprom di tipo 2732.
- Taratura kit CFD-011 e CFD-012 con oscilloscopio.

***** CONSOLLE STOP ON ADDRESS *****

Da tempo i nostri soci ci richiedevano uno strumento utile per il collaudo dell' hardware e del software, versatile, di semplice uso e soprattutto dal costo accessibile a tutti.

Essendo quest'ultimo un vincolo assai stringente abbiamo dovuto scartare il progetto di un emulatore vero e proprio a causa dell'elevato numero di componenti (costosi) necessari per ottenere un prodotto dalle caratteristiche tecniche appena soddisfacenti.

Vi proponiamo, invece, un progetto che, pur avendo prestazioni piu' ridotte di un emulatore, vi consente ugualmente di indagare sul funzionamento, in generale, di tutto il vostro sistema, di scoprire malfunzionamenti e vedere come "gira" il software.

E' un prodotto che non viene inserito sul bus Micro Design e cio' lo rende uno strumento "universale", adatto per qualunque sistema che utilizza il microprocessore Z80.

Quest'ultima caratteristica sara' particolarmente gradita a tutti i nostri soci che progettano hardware e software (talvolta non per il nostro micro) e che necessitano di uno strumento economico per il collaudo dei loro prototipi e per la verifica dei loro programmi.

Inoltre, la console STOP ON ADDRESS e' sicuramente utile a tutti coloro che dispongono di un altro sistema con cpu Z80 ma del quale non hanno la necessaria documentazione per conoscerne a fondo alcune caratteristiche (tipo alcuni indirizzi di memoria a cui fanno riferimento certe particolari routine, in quale parte del sistema operativo si esegue quell' OUTPUT, ecc.).

E' noto, infatti, che non tutti i sistemi sono "aperti" come il nostro e che se si vuole mettere le mani in sistemi "chiusi"

si diventa matti a cercare le necessarie informazioni presso il rivenditore, dal costruttore o dall'amico "esperto".

Con la nostra console non sarete piu' costretti a supplicare liste, chiedere l'indirizzo di memoria dove risiede quel tale byte, farvi descrivere la mappa di memoria e di I/O di quel personal, ecc.

Vi e' poi un altro aspetto che ci ha convinto della necessita' di questa console: la sua natura "didattica".

Chi utilizza il nostro micro e', di solito, una persona che sa "mettere le mani" nel suo sistema e sa trarsi d'impaccio se un programma non gli funziona o una scheda gli si e' guastata; ma spesso, non essendo un professionista, non conosce a fondo lo Z80 e tutti i suoi segnali di ingresso/uscita oppure la differenza fra istruzione e ciclo macchina oppure, piu' in generale, l'esatta interazione tra l'hardware e il software. E ancora, abbiamo molti nuovi soci che si sono avvicinati al nostro sistema proprio per capire a fondo il mondo dei microprocessori: la console STOP ON ADDRESS e' lo strumento che, unito ad uno dei tanti testi disponibili in commercio, potra' chiarir loro qualunque dubbio.

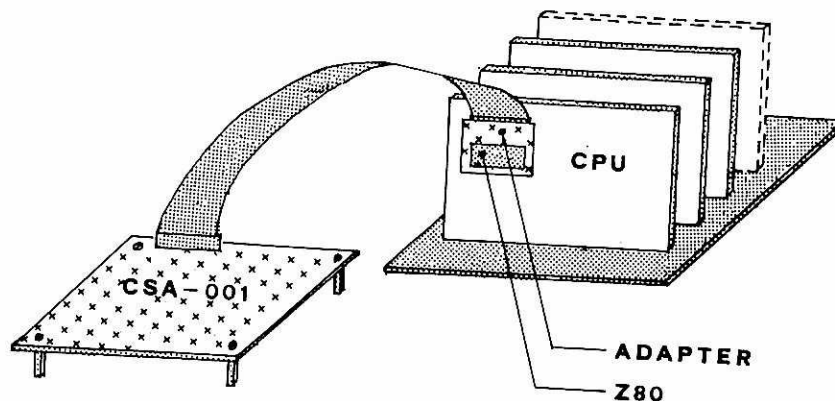
Cosa importante, per le indagini hardware l'unica attrezzatura di corredo (utile ma non obbligatoria) e' una semplice sonda logica (come quella presentata sulle pagine del nostro bollettino) od eventualmente un tester: non occorre possedere un oscilloscopio.

Ma vediamo ora di cosa esattamente si tratta e come funziona.

La console STOP ON ADDRESS consiste di una scheda di dimensioni mm 25 x 16 da cui parte un cavo piatto a 50 vie al cui termine si trova una basetta Adapter sulla quale sono montati uno zoccolo e un connettore entrambi a 40 pin.

Occorre quindi estrarre il microprocessore Z80 dalla scheda cpu del vostro sistema e inserire al suo posto il connettore della basetta Adapter. Il microprocessore deve essere invece inserito nello zoccolo montato sulla basetta Adapter.

La figura che segue vi chiarira' meglio il concetto.



A questo punto il sistema puo' essere acceso.
Sulla console sono presenti:

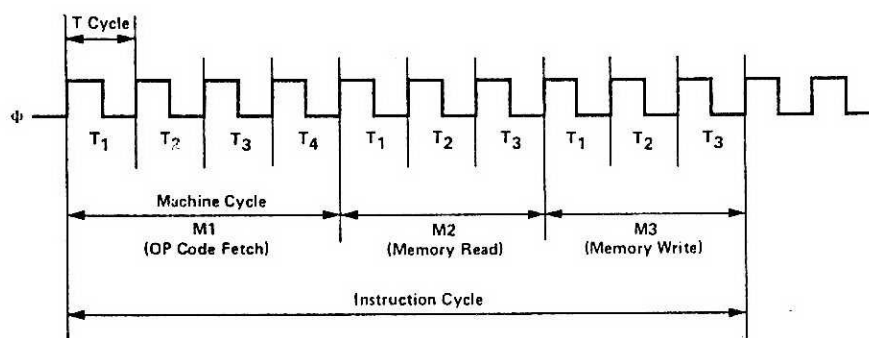
- 1) Interruttori e pulsanti che ne programmano il funzionamento.
- 2) Display e led che forniscono indicazioni visive di indirizzi, dati e segnali del microprocessore Z80.

Apriamo ora una breve parentesi sul funzionamento dello Z80 per poter poi meglio chiarire come funziona e a cosa puo' servire la CSA-001 anche ai soci meno esperti di hardware.

La cpu Z80 esegue le istruzioni attraverso la successione di un ben definito numero di operazioni base, che sono:

- Lettura o scrittura di memoria
- Lettura o scrittura di dispositivi di I/O
- Riconoscimento di interrupt

Tutte le istruzioni sono semplicemente una serie di queste operazioni base. Ciascuna di queste operazioni necessita da 3 a 6 periodi di clock, ma puo' diventare piu' lunga mediante utilizzo della linea WAIT, che permette appunto di inserire un qualsivoglia numero di periodi di clock aggiuntivi. Eccovi l'esempio di una generica timing relativa ad una istruzione dello Z80:

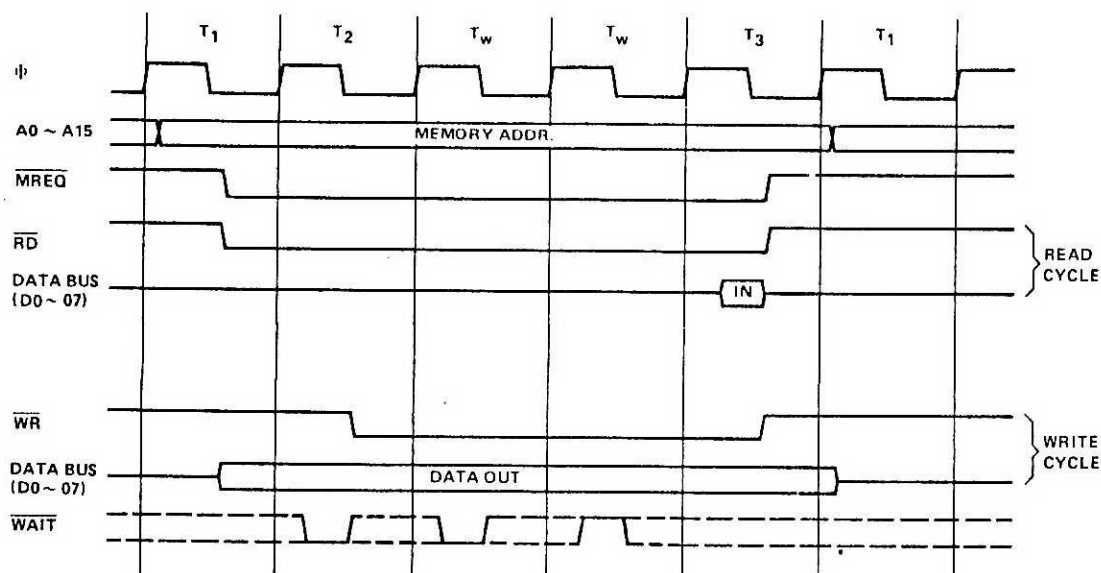


BASIC CPU TIMING EXAMPLE

L'esempio riporta il caso di una istruzione costituita da 3 cicli macchina.

Il primo ciclo macchina di ogni istruzione (M1) e' chiamato FETCH e serve per leggere il codice operativo dell'istruzione da eseguire. I cicli che seguono sono sempre movimenti di dati tra la cpu e la memoria o i dispositivi di I/O.

Vi riportiamo ora la timing dettagliata dei cicli di memoria (sia il caso di lettura che quello di scrittura) con l'aggiunta di 2 stati di wait (Tw).



MEMORY READ OR WRITE CYCLES WITH WAIT STATES

Si noti l'utilizzo della linea Wait: a partire dallo stato T2 se tale linea si trova bassa il microprocessore introduce degli stati Tw.

Finche' la linea wait non si disattiva il microprocessore continua a introdurre stati di wait ed il ciclo macchina interessato non termina.

E' importante notare che nello stato Tw le linee indirizzi, le linee dati e tutti segnali di controllo sono nello stato logico 0 o 1 ma stabili.

Chiariti alcuni principi base di funzionamento della cpu torniamo a parlare della CSA-001.

Il principio di base del funzionamento della console e' quello di sospendere il funzionamento del microprocessore mediante utilizzo del segnale di WAIT oppure di HOLD. Una volta fermato lo Z80, e' possibile, come anzi detto, osservare sui display l'indirizzo e il dato che fanno riferimento al ciclo macchina in questione e sui led il valore di tutti i segnali di stato e di controllo della cpu Z80.

I led indicano, se accesi, che la linea corrispondente e' attiva. Vi sono pure indicazioni relative alla presenza del clock, al reset, alle attivazioni delle linee di interrupt, ecc..

Gli interruttori a disposizione consentono di impostare uno dei seguenti modi di funzionamento:

- 1) RUN
- 2) STEP
- 3) FIND

Il primo modo, RUN, e' quello di normale funzionamento del vostro sistema: in tale stato e' come se la console STOP ON ADDRESS non fosse nemmeno stata inserita e potete tranquillamente lavorare e far eseguire tutti i vostri programmi.

Il modo STEP e' quello classico del funzionamento passo-passo. Il microprocessore viene sospeso in un ciclo macchina e da quel momento sarete in grado di farlo avanzare "passo-passo" agendo su un apposito pulsante. Potrete osservare sui led il tipo di ciclo macchina e quali segnali dello Z80 sono, in quell'istante, attivi.

Il terzo modo, FIND, e' quello normalmente piu' utile per la verifica dei programmi: e' ovvio, infatti, che non e' possibile, osservare lo svolgimento di un programma (di migliaia di passi!) col metodo STEP. E' stato inserito pertanto un predispositore a 4 cifre esadecimali che vi consente di impostare un BREAKPOINT ad un indirizzo da voi opportunamente scelto.

Una volta raggiunto tale indirizzo (programmabile come indirizzo di memoria o di input/output) la consolle STOP ON ADDRESS sospende il funzionamento del micro e da quel momento, passando al funzionamento STEP potrete vedere come il programma prosegue.

Alternando il funzionamento STEP al funzionamento FIND e' possibile seguire tutto lo svolgersi di un programma.

Abbiamo detto che i modi di "fermare" il microprocessore sono due:

- 1) utilizzando la linea WAIT
- 2) " la linea HOLD

Un apposito deviatore consente di scegliere l'uno o l'altro. La scelta dipende dal tipo di memoria che il vostro sistema possiede e se state collaudando dell'hardware oppure del software.

Da un punto di vista esterno (cioe' le visualizzazioni dei display e dei led) i due modi sono identici.

La fondamentale differenza e' pero' la seguente:

- nel modo "WAIT" il microprocessore e' fermato a meta' ciclo, i suoi segnali sono attivi (indirizzi, dati e controlli), tutte le decodifiche del sistema sono attive e "ferme", cioe' osservabili, ad esempio, con l'aiuto di una sonda logica (verificando che vi siano solo quelle corrette).
In questo caso, pero', se il sistema sotto controllo utilizza delle memorie dinamiche, il loro rinfresco viene interrotto e non appena il sistema viene fatto ripartire sicuramente succedera' di tutto tranne quello desiderato.
- Nel modo "HOLD", al contrario, il microprocessore termina il ciclo (tutti i suoi segnali vengono pero' memorizzati sulla consolle e visualizzati) e viene poi messo in stato di hold: non appena il micro Z80 ha rilasciato il bus, una apposita logica, presente sulla consolle, provvede alla simulazione di un "REFRESH CYCLE" simile allo Z80, adempiendo, cosi', al necessario compito di rinfresco delle ram dinamiche.

Entrambi i due modi di funzionamento, WAIT e HOLD, sono necessari. IL primo risulta particolarmente utile per il controllo dell'hardware, in quanto tutti i segnali del ciclo interessato sono attivi, il secondo, invece, risulta obbligatorio nel debug del software in un sistema dotato di ram dinamiche.

Vogliamo evidenziare che i 2 modi sono stati previsti per **rendere generale l'utilizzo della consolle STOP ON ADDRESS a tutti i sistemi dotati di microprocessore Z80.**

Infatti la maggior parte dei sistemi che utilizzano tale microprocessore dispone di memorie di tipo dinamico, a causa del loro favorevole rapporto costo/bit.

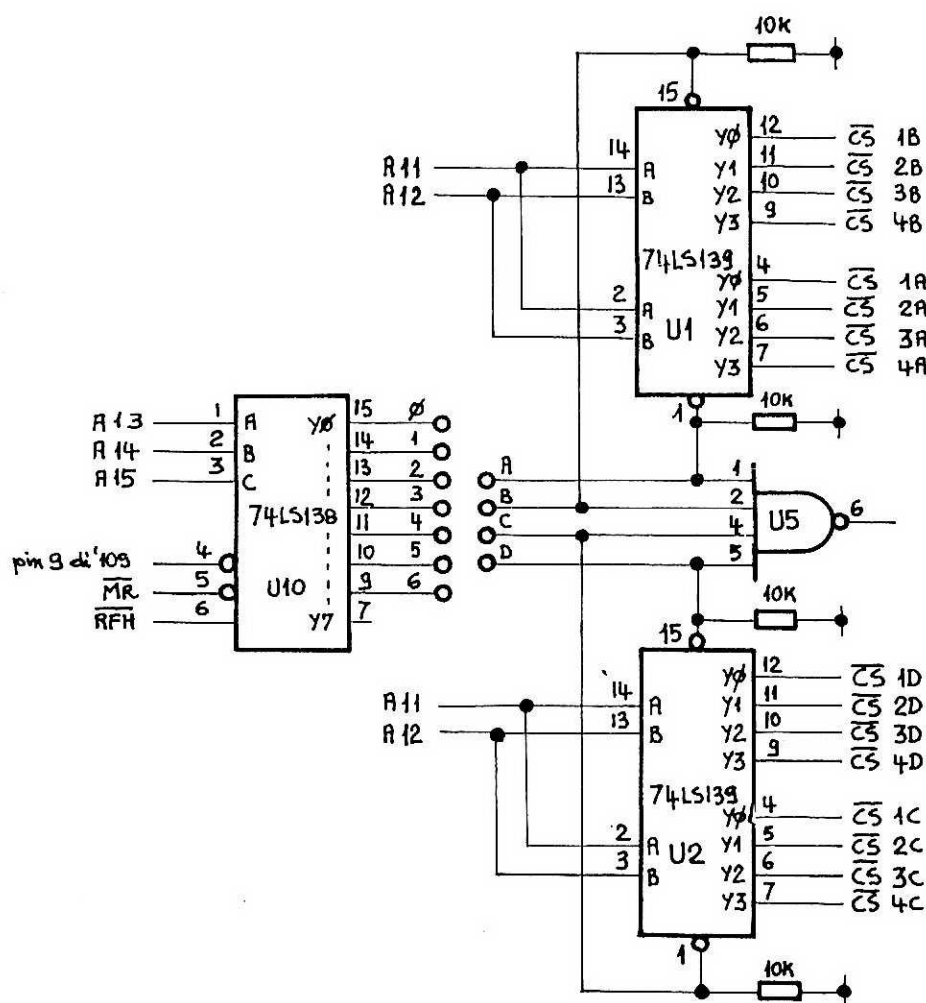
Ricordiamo, inoltre, che per facilitare soprattutto le indagini hardware la nuova CPU-001 e' stata progettata tenendo

conto dell'impiego della nuova consolle e pertanto coloro che già la usano potranno utilizzare sempre il modo WAIT (pur avendo ben 64 Kbytes di ram dinamica).

Vi illustriamo ora alcuni esempi pratici di utilizzo della nostra consolle CSA-001.

1) Ricerca di un guasto sulla memoria MRE-001.

Nello schema elettrico che segue abbiamo riportato in modo chiaro ed evidente la sezione decodifica della nostra espansione di memoria MRE-001 e a tale schema faremo riferimento nella descrizione seguente.



LOGICA DI DECODIFICA DELLA SCHEDA MRE-001

Il primo passo da compiere è l'individuazione di un semplice programmino adatto a "sollecitare" la scheda sotto osservazione che potrà essere scritto ed eseguito col programma monitor.

Pertanto si predispone l'indirizzo di partenza della memoria in oggetto sopra i primi 8 K del sistema (ad es. 8000H), affinché almeno il programma Monitor possa funzionare (si ipotizza che la scheda MRE-001 si sia guastata).

Un esempio di programma, inserito in memoria a partire da 100H, può essere il seguente:

```

100      3A      (LDA 8000H)
101      00
102      80
103      C3      (JMP 100H)
104      00
105      01

```

Si mette in esecuzione tale programma col comando G100.

Si posiziona la CSA-001 in modo STEP e si osservano i seguenti cicli macchina:

Indirizzo	Dato	
0100	3A	Fetch
0101	00	Memory read (1° op.)
0102	80	Memory read (2° op.)
8000	xx	Memory read del dato presente nella locazione di indirizzo 8000H
0103	C3	Fetch (salto a 100H)
0104	00	Memory read
0105	01	Memory read
0100	3A	Fetch
..		

Il procedimento da seguire in queste analisi è il seguente:

- 1) Si individua il segnale fondamentale e si verifica che sia attivo.
- 2) In caso negativo si prosegue a ritroso tutta la catena di porte logiche che devono determinare la sua attivazione fino all'isolamento e individuazione di un livello scorretto o di una pista interrotta o in corto con altre.

Nel nostro caso è conveniente proseguire fino al ciclo macchina di memory read all'indirizzo 8000H e, con una sonda logica, si osserva il valore presente sul pin 20 dell'integrato di memoria che deve rispondere all'indirizzo 8000H (dipende da come avete inserito i ponticelli P1 ma noi ipotizziamo, come esempio, che si tratti del chip di memoria 1A).

Tale pin è il "Chip Select" della memoria ram e deve trovarsi nello stato logico 0.

Possono aversi 2 casi:

- a) la selezione non è attiva (pin 20 al livello logico 1). In questo caso il guasto è nella logica di decodifica e pertanto è necessario percorrere la strada della selezione. Si osservano, in successione, i seguenti segnali e si verifica che abbiano un valore corretto:

segnale:	valore logico OK
pin 4 di U1 (uscita del 74LS139)	0
" 2 di U1 (indirizzo A11)	0
" 3 di U1 (" A12)	0
" 1 di U1 (enable del 74LS139)	0
" 1 di U5 (" " ")	0
" 12 di U10 (uscita del 74LS138)	0

" 1 di U10 (indirizzo A13)	0
" 2 di U10 (" A14)	0
" 3 di U10 (" A15)	1
" 4 di U10 (enable di U10)	0
" 9 di U3 (uscita 74LS109)	0
" 5 di U10 (" di U10, MREQ)	0
" 6 di U10 (" di U10, RFH)	1

Si noti che l'ordine con cui si osservano i segnali e' abbastanza rigido in quanto occorre scegliere la strada piu' veloce che porta alla verifica dell'equazione di una rete logica e quindi del buon funzionamento degli integrati che la realizzano.

b) La selezione del chip di memoria 1A e' attiva. In questo caso si osserva sulla CSA-001 il dato presente all'indirizzo 8000H, dato che avremo provveduto a caricare precedentemente col comando S del monitor. Se il dato e' corretto significa che la memoria all'indirizzo 8000H funziona e pertanto e' necessario indagare su altre locazioni, facilmente riscontrabili col comando T del monitor o col programma MEMTEST. Se, invece il dato non e' corretto occorre considerare due strade:

- b1) verifica dei pin di indirizzo del chip di memoria " del segnale OE, pin 20 di U1A (deve essere 0).
- b2) controllo del buffer dei dati U9 (controllare il pin 19, cioe' l'abilitazione, ed il pin 1, la direzione (0 se trattasi di lettura, 1 se trattasi di scrittura). Potrebbe pure essere necessario, per la sua verifica, analizzare una operazione di scrittura ed in tal caso sara' sufficiente sostituire il contenuto della locazione 100H con 32H (STA al posto di LDA): fermando la CSA all'indirizzo 8000 si vedra' sul display dati cio' che avremo caricato precedentemente nell'accumulatore.

Non appena si incontra un errore nella logica basta procedere, generalmente, col seguente criterio:

- Si controllano le saldature (eventualmente si rifanno per maggior sicurezza).
- Si controllano i possibili corto circuiti.
- Si sostituisce l'integrato che "riceve" un segnale corretto ma che non genera l'uscita desiderata.

2) Il sistema all'accensione non parte.

Questo tipo di malfunzionamento, che non consente nemmeno l'utilizzo del monitor per la verifica delle operazioni fondamentali, e' assai fastidioso ma la consolle CSA-001 fornisce indicazioni immediate e precise.

E' sufficiente, infatti, mettere la CSA in funzionamento STEP, resettare e quindi osservare come evolve il programma a partire da 0000 agendo sul pulsante ad azionamento momentaneo.

Eventuali malfunzionamenti nella lettura delle eeprom del monitor residenti sulla CVP-001 o CFD-001 si individuano molto presto e quindi si trattera' di spostare la nostra indagine sulla scheda che si individua essere guasta (video o floppy controller).

Inoltre accade, molto spesso, che il sistema non si muove neppure agendo sul pulsante o mettendo la CSA in funzionamento RUN: i display visualizzano sempre l'indirizzo 0000,

stabile. Basta allora osservare i led che indicano l'attività dei segnali WAIT e RESET e stabilire così se il microprocessore è fermo in wait (non per causa della CSA-001, ovviamente) o se è tenuto in reset. Infatti capita molto spesso, inserendo le schede sul bus di un poco sfalsate, di portare il -12V sull'ingresso di qualche integrato TTL: tale ingresso spesso si guasta in modo da caricare il segnale ad esso applicato, portandolo al livello logico 0 e forzando così una situazione di reset o di wait perenne.

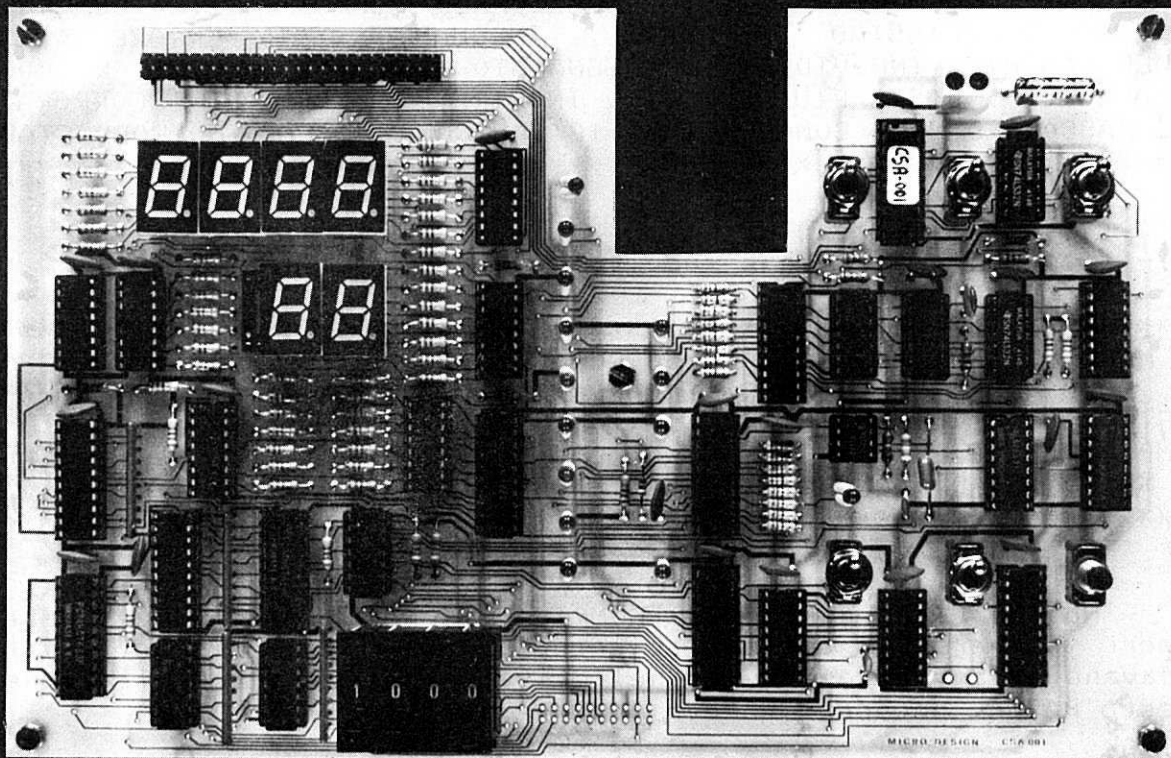
3) Ricerca dello SKEW FACTOR.

Questa operazione, da veri addetti ai lavori, è molto semplice con la nostra CSA-001.

È sufficiente conoscere l'indirizzo di I/O (nel sistema Micro Design e' D2) in cui viene scritto il numero di settore quando si accede ai dati su disco, impostarlo sul predispositore esadecimale della CSA-001, mettere quest'ultima in modo FIND e far partire, sotto CP/M, la lettura di un file tramite un qualunque comando.

La CSA si fermerà all'indirizzo specificato di I/O mostrando, sul display relativo al dato, il valore che rappresenta il numero di settore da leggere. Agendo sul pulsante STEP la consolle riparte e si riferma mostrandoci il successivo valore di settore da leggere: la differenza tra questo ed il precedente rappresenta lo SKEW FACTOR con cui il CP/M installato sul vostro sistema alloca e recupera i files.

La Consolle Stop on Address CSA-001 è in vendita, comprensiva di manuale di installazione ed uso a L. 349.000 + IVA.



PRECISAZIONI SUL PASCAL.

Abbiamo presentato, già da qualche tempo, il linguaggio PASCAL e continuiamo a ricevere richieste di chiarimenti che dimostrano come non sia ancora del tutto chiaro cosa sia, cosa serva e quali siano i vantaggi del PASCAL. Si nota, in modo abbastanza evidente, un certo timore reverenziale nei confronti del nuovo linguaggio, forse dovuto a voci inesatte e al fatto di non aver mai approfondito l'argomento.

Desideriamo quindi ritornare sull'argomento cercando di rispondere alle domande che ci siamo sentiti rivolgere più spesso e illustrandovi poi il programma MC che viene fornito gratuitamente, completo di sorgenti, insieme al PASCAL.

D. - E' difficile scrivere programmi con il PASCAL?

NON E' ASSOLUTAMENTE DIFFICILE PROGRAMMARE IN PASCAL, INFATTI QUESTO LINGUAGGIO VIENE ADOPERATO PER INSEGNARE A PROGRAMMARE AI PRINCIPIANTI. IL PASCAL SI PRESTA IN MODO PARTICOLARE A QUESTO SCOPO PERCHE' ABITUA, CON LE SUE REGOLE, A SCRIVERE PROGRAMMI IN MODO ORDINATO E MODULARE FACILITANDO ENORMEMENTE LA LETTURA E LA COMPrensIONE, ANCHE A DISTANZA DI TEMPO, DI LISTATI PROPRI O SCRITTI DA ALTRI.

D. - Il PASCAL e' un interprete come l' MBASIC o un compilatore come il BASCOM?

IL PASCAL CHE VI PROPONIAMO E' UN VERO COMPILATORE CHE PRODUCE CODICE MACCHINA Z-80 MA HA LA CARATTERISTICA DI POTER COMPILARE PROGRAMMI IN MEMORIA. QUESTA ECCEZIONALE PARTICOLARITA' CONSENTE UN USO VELOCISSIMO DEL PASCAL, IN QUANTO OGNI ERRORE VIENE IMMEDIAMENTE INDIVIDUATO E SEGNALATO PASSANDO IN EDITOR COME ACCADE CON I BASIC PIU' EVOLUTI. INOLTRE L' EDITOR E' INCORPORATO NEL PASCAL ED HA I COMANDI IDENTICI AL CONOSCIUTISSIMO WORD STAR.

D. - Il manuale che viene fornito e' didatticamente valido?

IL MANUALE IN ITALIANO, MOLTO CURATO E PIENO DI ESEMPI, INSEGNA, IN MODO MOLTO EFFICACE, COME SI ADOPERA QUESTA PARTICOLARE INSTALLAZIONE DEL PASCAL, I PRINCIPIANTI CHE DESIDERANO UN LIBRO COMPLETAMENTE DIDATTICO POSSONO LEGGERE "INTRODUZIONE AL PASCAL" DI WILSON E ADDYMAN EDITO DA "IL ROSTRO".

D. - Ho sentito parlare di COMPILATORI PASCAL che, per poter essere adoperati, avevano bisogno di ben tre dischetti che andavano via via sostituiti nel drive durante la compilazione. Questo PASCAL presenta questo inconveniente?

CERTAMENTE NO, TUTTI I FILE NECESSARI: COMPILATORE, EDITOR, LIBRERIA, MESSAGGI DI ERRORE, SI TROVANO SU UN UNICO DISCHETTO CHE PUO' ESSERE UTILIZZATO ANCHE PER CONTENERE IL PROGRAMMA DA

COMPILARE. E' QUINDI ADATTO ANCHE PER CHI POSSIEDE UN SOLO DRIVE DI LIMITATA CAPACITA' (5.25" SINGOLA DENSITA' DA 77 KBYTE).

D. - Puo' utilizzare il PASCAL anche chi ha LA SCHEDA VIDEO LX 529 ?

CERTAMENTE, SE UTILIZZA IL CP/M ORIGINALE MICRO DESIGN, PUO' RICHIEDERE TRANQUILLAMENTE IL COMPILATORE CHE FUNZIONERA' PERFETTAMENTE.

Ed ora alcuni chiarimenti piu' specifici:

D. - Che cosa sono le dichiarazioni?

OGNI PROGRAMMA PASCAL SI COMPONE ESSENZIALMENTE DI DUE PARTI: UNA CHE COMPRENDE LE DICHIARAZIONI E UN' ALTRA CHE COMPRENDE LE ISTRUZIONI. LE DICHIARAZIONI VENGONO USATE PER SPECIFICARE I NOMI E LE PROPRIETA' DEGLI IDENTIFICATORI CHE SI USANO NELLE ISTRUZIONI. IN PRATICA PRIMA DI UTILIZZARE UNA DETERMINATA SEQUENZA DI LETTERE (AD ESEMPIO PER UNA VARIABILE) OCCORRE DICHIARARLA ESPLICITAMENTE. VEDIAMO LA DIFFERENZA TRA DUE PROGRAMMI CHE FANNO LA STESSA FUNZIONE, IL PRIMO SCRITTO IN BASIC E IL SECONDO SCRITTO IN PASCAL.

```
100 FOR X=1 TO 100
110 PRINT X,X*X
120 NEXT X
```

```
PROGRAM QUADRATI;
VAR X:INTEGER;
BEGIN
FOR X:=1 TO 100 DO WRITELN (X,X*X);
END.
```

SI PUO' NOTARE CHE NEL PROGRAMMA IN PASCAL NON E' NECESSARIO NUMERARE LE RIGHE E CHE LE ISTRUZIONI DEL PROGRAMMA SONO RACCHIUSE TRA LE PAROLE "BEGIN" E "END". LE DUE RIGHE PRECEDENTI DICHIARANO CHE IL PROGRAMMA SI CHIAMA "QUADRATI" E CHE LA VARIABILE INDICATA CON "X" E' INTERA. SE CI DIMENTICHIAMO DI DICHIARARE LA VARIABILE X IL COMPILATORE SEGNERA' UN ERRORE, INFATTI E' OBBLIGATORIO SPECIFICARE TUTTI GLI IDENTIFICATORI CHE SI UTILIZZANO. QUESTO, A PRIMA VISTA, PUO' SEMBRARE UNA PERDITA DI TEMPO MA INVECE E' UN NOTEVOLE AIUTO ALLA SCRITTURA DI PROGRAMMI CORRETTI, FACILMENTE LEGGIBILI, ORDINATI E EFFICIENTI. SI PUO' NOTARE COME SIA ALTRETTANTO SEMPLICE SCRIVERE BREVI PROGRAMMI IN PASCAL E IN BASIC. PERO', UTILIZZANDO IL PASCAL, LE SUE REGOLE PRECISE OBBLIGANO AD UNA SCRITTURA "PROFESSIONALE" DEL PROGRAMMA CHE GARANTISCE UNA QUALITA' DECISAMENTE SUPERIORE DEL SOFTWARE.

D - Che cosa significa tipi definibili dall' utente?

IN PASCAL, OLTRE AI SOLITI TIPI: INTERO, REALE, BOOLEANO, ETC. E' POSSIBILE DEFINIRE DEI TIPI DIVERSI CREATI DAL PROGRAMMATORE. UN

ESEMPIO MOLTO USATO PER SPIEGARE MEGLIO IL CONCETTO E' QUELLO DEL PROGRAMMA CHE DEVE UTILIZZARE, ORDINARE, GESTIRE FATTURE IN BASE AL MESE DI EMISSIONE. E' POSSIBILE IN PASCAL, DICHIARARE:

```
TYPE MESE =(GEN,FEB,MAR,APR,MAG,GIU,LUG,AGO,SEP,OTT,NOV,DIC);
```

DEFINENDO COSI' UN NUOVO TIPO DI VARIABILE CHE NON E' NE INTERA NE REALE MA E' "MESE". UNA VARIABILE DICHIARATA DI QUESTO TIPO

```
VAR SCADENZA,MESECORRENTE: MESE;
```

PUO' ASSUMERE UNO DEI DODICI VALORI RACCHIUSI TRA PARENTESI. SU TALI VARIABILI SI POSSONO USARE LE FUNZIONI PRED (predecessore), SUCC (successore), ORD (ordinale), E TUTTI GLI OPERATORI DI RELAZIONE: ("<" ">" "="). E' POSSIBILE QUINDI INDIVIDUARE AD ESEMPIO: SUCC(GEN) CHE E' FEB, PRED(OTT) CHE E' SEP, ORD(MAR) CHE E' 3 (infatti partendo da GEN=1, FEB=2, MAR=3,...). INOLTRE E' POSSIBILE SCRIVERE SCADENZA>OTT CHE E' UNA ESPRESSIONE BOOLEANA VERA SE LA VARIABILE SCADENZA DI TIPO MESE HA O IL VALORE NOV OPPURE DIC. IN PASCAL E' QUINDI FACILE SCRIVERE ISTRUZIONI COME:

```
IF SCADENZA < MESECORRENTE WRITELN ('FATTURA SCADUTA');  
IF SUCC(SCADENZA) = MESECORRENTE WRITELN ('FATTURA IN SCADENZA');
```

LA CUI SEMPLICITA' E IMMEDIATEZZA SONO ORMAI CHIARE.

Il modo migliore per imparare il PASCAL e' quello di studiarne le caratteristiche principali, provare direttamente con un buon compilatore e leggere programmi scritti da persone competenti cercando di capirli a fondo. A questo scopo viene fornito, gratuitamente, insieme al PASCAL, un programma molto interessante che, oltre ad essere utilizzabile come "foglio elettronico", permette di imparare moltissimo dalle liste dei sorgenti. Vi diamo nelle pagine successive una descrizione del funzionamento di tale programma.

INTRODUZIONE AL MICROCALC

Il MicroCalc e' un programma dimostrativo tipo VisiCalc (foglio elettronico) fornito insieme al TURBO-Pascal.

Dato che e' un programma dimostrativo ha alcune limitazioni che pero' possono essere eliminate in quanto viene fornito completo di programma sorgente.

- * Non e' possibile copiare le formule da una cella all'altra.
- * Non e' possibile inserire o eliminare righe e colonne.

Nonostante tali limitazioni il MicroCalc fornisce prestazioni molto interessanti tra cui:

- * Numeri reali a 11 digits floating point
- * Set completo di funzioni matematiche (SIN,COS,LN,EXP etc.).
- * Editor interno per il testo e per le formule.
- * Il testo puo' essere scritto direttamente nelle celle.
- * Le formule sono protette da cancellazioni accidentali.

In aggiunta a tutto questo il MicroCalc fornisce tutte le prestazioni di un normale foglio elettronico:

- * Caricamento di un foglio da disco.
- * Salvataggio di un foglio su disco.
- * Calcolo automatico dei valori dopo ogni modifica (disab.)
- * Memorizzazione del foglio su file.LST.
- * Ripristino del foglio.

Un foglio elettronico equivale funzionalmente a un pezzo di carta su cui e' possibile scrivere testi, numeri e formule che il MicroCalc elabora automaticamente.

La pagina seguente mostra la struttura del foglio elettronico.

	A	B	C	D	
1	22.00				
2	1.00				
3	2.00				
4	3.00				
5	28.00				
"					
"					
A 5	Formula: (A1+A2+A3+A4+A5)				

Nell' esempio le ultime righe indicano che la cella "attiva" e' la A 5 e che contiene una formula: (A1+A2+A3+A4). Questo significa che i numeri contenuti in A1,A2,A3 e A4 devono essere sommati e il risultato deve essere messo in A 5.

La formula puo' essere abbreviata in (A1>A4) con il significato: somma il contenuto delle celle da A1 a A4.

E' possibile muovere il cursore con gli stessi comandi del TURBO editor:

(Su)
Ctrl-E
(Sinistra) Ctrl-S Ctrl-D (Destra)
Ctrl-X
(Giu')

Una cella puo' contenere un numero, una formula oppure testo. Il tipo della cella e le sue coordinate vengono indicate nell' angolo in basso a sinistra dello schermo:

A 5	Formula:	(Significa che la cella "attiva" e' la A 5 e che contiene una formula).
A 1	Text	(La cella A 1 contiene un testo)
A 2	Numeric	(La cella A2 contiene un numero)

Sommario

Le celle sono indicate da A1 fino a G21 per un totale di 147.

Sommario delle funzioni standard e degli operatori

SIN, COS, ARCT, ABS, FACT, EXP, LN, +, -, /, *

Inoltre l'operatore '>' puo' essere usato per indicare un insieme di celle da sommare.

Ingresso dati

Per introdurre dati in una cella occorre spostarsi in quella cella col cursore e battere i dati. Il MicroCalc determina automaticamente se la cella e' numerica o contiene testo basandosi sul primo carattere: numero ==> numerica; lettera ==> testo; (==> formula.

Quando si muove il cursore tra le celle

^S, ^D, ^E, ^X effettuano i movimenti a sinistra, destra, alto e basso.

Quando si edita una cella

^S, ^D effettuano movimenti a sinistra e destra. ^A, ^F portano all'inizio e alla fine della riga.

DEL, ^G cancellano il carattere a sinistra o a destra del cursore.

ESC rende possibile l'annullamento delle modifiche e l'edit di una cella.

Sommario dei comandi

- / Ripristina lo schermo.
- Q Termina le operazioni uscendo dal MicroCalc.
- L Carica un foglio dal disco.
- S Salva un foglio su disco.
- R Effettua i calcoli.
- P Salva il foglio su file. LST in formato adatto alla stampa.
- F Cambia il formato dei numeri.
- A Attiva o disattiva l'Autocalc (Calcolo automatico).

Nota: per usare la notazione scientifica occorre usare il comando F e introdurre - 1 . Il secondo numero richiesto e' l'ampiezza della cella, nel caso sia superiore a 10, la cella occupa anche parte della successiva.

ESEMPI

Esempi di formule valide:

(A1+(B2-C7)) sottrae la cella C7 dalla B2 e somma alla A1
(A1>A23) somma delle celle: A1,A2,A3,.....,A23
(A1>B5) somma delle celle: A1,....,A5 e B1,....,B5

Le formule possono essere anche molto complicate:

(SIN(A1)*COS(A2)/((1.2*A8)+LN(FACT(A8)+8.9E-3))+(C1>C5))

Per modificare una formula o un testo occorre posizionarsi sulla cella e premere **ESC**, fare le modifiche e battere <RETURN>. Se si commette un errore bisogna premere di nuovo **ESC** in modo da ripristinare il contenuto precedente.

Per provare il MicroCalc potete usare il comando /L e caricare il file: **MCDEMO**.

MODIFICA STAMPATO CFD-001 PER UTILIZZO MEMORIE TIPO 2732

A causa della costante diminuzione sul mercato dei componenti delle memorie tipo 2716 e' necessario eseguire una semplice modifica sullo stampato CFD-001 per potere installare una memoria di tipo 2732 al posto delle attuali due memorie di tipo 2716.

Tale modifica consente, inoltre, grazie alla disponibilita' di 2732 con minore tempo di accesso (maggior velocita'), di potere utilizzare una cpu a 4MHz senza malfunzionamenti (eliminando il problema di selezionare memorie 2716 piu' veloci di quanto dichiarato dai costruttori).

==== MODIFICA ====

- 1) Estrarre le due memorie 2716 dagli zoccoli U20 e U21.
- 2) Alzare i pin 11 e 12 di U22 (74LS139).
- 3) Cortocircuitare i pin 12 e 15 di U22.
- 3A) *TAGLIARE LA CONNESSIONE TRA PIN 21 e 24 di U20.*
- 4) Connettere il pin 14 di U22 al pin 21 di U20.
- 5) Inserire la eprom 2732 nello zoccolo U20.

Con tale modifica la eprom inserita nello zoccolo U20 risponde al campo di indirizzi:

F000 - FFFF

UTILIZZO MEMORIE TIPO 2732 SU CVP-001

E' possibile inserire un chip 2732 al posto della eprom 2716 presente sulla CVP-001 senza alcuna modifica allo stampato. In ogni caso di tale eprom vengono utilizzati solamente 2 Kbytes e pertanto non sono necessari cambiamenti alla logica di selezione.

Occorre notare, pero', che poiche' la linea A11 della 2732 viene a trovarsi ad un "uno" logico, e' necessario ricordare (in caso di nuova programmazione di tale chip) che la meta' utilizzata, di tale eprom 2732, e' quella superiore.

=====

ATTENZIONE!!!!

Prosegue l'offerta di lancio della nuova CPU-001 che, pertanto, continua a essere in vendita a L. 495.000 +IVA.

E' finalmente disponibile il manuale in italiano del programma DBASE II a L. 65.000 + IVA.

=====

TARATURA KIT CFD-011 CON AUSILIO DI OSCILLOSCOPIO

Resettare il sistema MD-001 e introdurre il comando N.

REGOLAZIONE DEL TRIMMER R 8 (PRECOMPENSAZIONE).

- 1) Osservare la durata dell'impulso presente sul pin 31 dell'integrato MD-001.
- 2) Aggiustare il trimmer R 8 fino all'ottenimento della durata desiderata. Tale valore e' specificato sul manuale del driver utilizzato (valore tipico: 150 ns).

REGOLAZIONE DEL TRIMMER R 9 (DATA SEPARATOR 5.25").

- 1) Osservare la durata dell'impulso presente sul pin 29 dell'integrato MD-001.
- 2) Aggiustare il trimmer R 9 fino all'ottenimento di un impulso di durata pari a 1/8 della frequenza.

REGOLAZIONE DEL TRIMMER R10 (DATA SEPARATOR 8").

- 1) Come per R 9.
- 2) Aggiustare R 9 fino all'ottenimento di un impulso di durata pari a 1/8 della frequenza.

REGOLAZIONE DEL COMPENSATORE C11.

- 1) Osservare la frequenza presente sul pin 16 Dell'integrato MD-001.
- 2) Aggiustare il compensatore C11 fino all'ottenimento di una frequenza di 500 KHz.

TARATURA KIT CFD-012 CON AUSILIO DI OSCILLOSCOPIO

Resettare il sistema MD-001 e introdurre il comando N.

REGOLAZIONE DEL TRIMMER R 9 (DATA SEPARATOR 5.25").

- 1) Osservare la durata dell'impulso presente sul pin 29 dell'integrato MD-001.
- 2) Aggiustare il trimmer R 9 fino all'ottenimento di un impulso di durata pari a 1/8 della frequenza.

REGOLAZIONE DEL COMPENSATORE C11.

- 1) Osservare la frequenza presente sul pin 16 Dell'integrato MD-001.
- 2) Aggiustare il compensatore C11 fino all'ottenimento di una frequenza di ~~250~~ 500 KHz.