

software MBASIC

Le quattro operazioni (1ª parte)

Da tre puntate ci stiamo occupando della gestione da parte dell'MBASIC delle espressioni e, dopo aver analizzato delle subroutine chiamate più volte dalle varie funzioni, da questo numero effettueremo l'analisi di come il nostro interprete esegue le quattro operazioni «fondamentali»: non è evidentemente un lavoro inutile in quanto la conoscenza dei vari «entry point» delle routine ci sarà molto utile per affrontare le funzioni più complesse quali quelle logaritmiche e quelle trigonometriche.

Una prima considerazione che si può fare è che l'MBASIC non si ferma certo se sommando due numeri interi o meglio moltiplicandoli si ottiene un valore non più esprimibile come intero e perciò su due byte: automaticamente in questo ed in altri casi passa alla notazione reale (in singola precisione) e poi se necessario in doppia precisione. Il problema è evidentemente quello di assegnare un ri-

sultato di un certo «tipo» ad una variabile di tipo opportuno: un valore espresso in doppia precisione e perciò con parecchie cifre significative perderà evidentemente le proprie qualità se forzato di una variabile di tipo intero.

Vediamo ora, riagganciandoci a quanto detto nel n. 49, come si arriva alla gestione delle quattro operazioni.

Il seguito della routine 19FEH

Ricordiamo che questa routine è quella a partire dalla quale vengono calcolate le espressioni: l'avevamo analizzata fino ad una «RET C» ed ora riprenderemo dall'indirizzo 1A11H, come si può vedere dal listato n. 1.

Ci stiamo dunque addentrando nell'analisi del comportamento dell'interprete quando si incontra un byte relativo ad una operazione aritmetica o logica: dall'oramai famosa puntata de «I trucchi del CP/M» del n. 38 di MC, citata più volte nei numeri scorsi,

possiamo vedere qual è la codifica sotto forma di byte delle operazioni menzionate, codifica che prevede appunto valori maggiori di 0EFH.

In particolare i valori 0EFH, 0F0H e 0F1H corrispondono agli operatori «maggiore», «uguale» e «minore» rispettivamente, che vengono elaborati a partire dall'indirizzo 1A7FH.

Nel caso non si tratti di tali tre valori, allora viene sottratta la costante 0F2H e controllato che il valore così ottenuto sia compreso tra 00H e 0BH: nel caso particolare del valore nullo, corrispondente alla somma («+»), allora si controlla se il «tipo» della variabile vale 3 (stringa), nel qual caso si salta alla gestione della concatenazione di stringhe.

Nel caso in cui il valore è minore di 0CH invece siamo in presenza di uno dei nostri operatori, riportati nella tabella 1: il programma successivamente estrae da un'apposita tabella posta in memoria un valore rappresentante la «priorità», la «precedenza» dell'operatore, valore che abbiamo posto nella

1A11H	CP 0F2H		JP Z,0CE1H	i errore
	JP C,1A7FH	i < = >	OR A	
	SUB 0F2H		LD C,(HL)	
	LD E,A		INC HL	
	JP NZ,1A25H		LD B,(HL)	
	LD A,(0A67H)		PUSH BC	
	CP 3		JP M,1A70H	i intero
	...	i concatenazione stringhe	INC HL	
1A25H	CP 0CH		LD C,(HL)	
	RET NC		INC HL	
	LD HL,047FH		LD B,(HL)	
	LD D,0		PUSH BC	
	ADD HL,DE		JP P,1A70H	i singola precisione
	LD A,B		LD HL,0C00H	i doppia precisione
	LD D,(HL)		LD C,(HL)	
	CP D		INC HL	
	RET NC	i operatore con precedenza inferiore	LD B,(HL)	
	PUSH BC		INC HL	
	LD BC,1A06H		PUSH BC	
	PUSH BC		LD C,(HL)	
	LD A,0		INC HL	
	CP 7FH		LD B,(HL)	
	JP Z,1A9DH		PUSH BC	
	CP 51H		ADD A,3	
	JP C,1AABH	i AND,OR,XOR,EGV,IMP	LD C,E	
	AND 0FEH		LD B,A	
	CP 7AH		PUSH BC	
	JP Z,1AABH	i MOD,	LD BC,1AD4H	
1A49H	LD HL,0CD4H		PUSH BC	
	LD A,(0A67H)		LD HL,(0A9DH)	
	SUB 3		JP 19F6H	
			1A70H	
			LD C,E	
			LD B,A	
			PUSH BC	
			LD BC,1AD4H	
			PUSH BC	
			LD HL,(0A9DH)	
			JP 19F6H	

Listato 1 - È il seguito della routine 19FEH mostrata nel numero 49 di MC.

colonna centrale della tabella 1. Da questa vediamo che tanto più alto è tale valore, tanto prima viene eseguita una certa operazione rispetto ad un'altra successiva: ecco che dunque in un'espressione complessa vengono dapprima effettuati gli eventuali elevamenti a potenza, poi la moltiplicazione e la divisione (con pari priorità), poi l'operatore «/» (divisione intera), poi l'istruzione MOD (il resto di una divisione intera), poi la somma e la sottrazione (con pari priorità), proseguendo con gli operatori logici, il primo dei quali è l'AND, seguito poi dall'OR, dallo XOR, dall'operatore EQV (EQUIValence) assai poco usato e da IMP (IMPLICazione logica), stretto parente del precedente in quanto a notorietà.

In particolare il valore relativo all'operazione «precedente»: l'operatore avente il valore più «basso» verrà così abbandonato o meglio lasciato «pendente», in attesa, fino a che un operatore successivo a priorità minore gli permetta l'esecuzione.

Al limite il termine dell'espressione abiliterà l'esecuzione dell'ultimo operatore rimasto.

Come già avevamo detto preceden-

Indirizzo	valore	operazione
047FH	79H	+
0480H	79H	-
0481H	7CH	*
0482H	7CH	/
0483H	7FH	~
0484H	50H	AND
0485H	46H	OR
0486H	3CH	XOR
0487H	32H	EQV
0488H	28H	IMP
0499H	7AH	MOD
049AH	7BH	\

Tabella 1 - Valori delle priorità associate ad ogni operatore: più alto è il valore e prima verrà eseguita l'operazione.

temente nel caso in cui una certa operazione viene lasciata in sospeso, viene salvato nello stack lo «stato» dell'espressione, consistente nel valore dell'accumulatore (il FAC) in quell'istante, il tipo dell'operatore in attesa, la sua priorità nonché un indirizzo di ritorno: ogni volta che si incontra un'operazione nuova si effettua dunque subito il confronto tra le priorità e nel caso si «resuscita» l'operazione lasciata in sospeso a scapito della nuova operazione che ora andrà ad aspettare momenti più propizi.

Supponendo dunque che l'operazione debba essere eseguita, allora ci saranno alcuni confronti per attivare una certa routine a seconda dell'operazione da compiere: in particolare se si tratta di un elevamento a potenza,

	04A9H	049FH	0495H
	interi	reali	d.prec.
+	2B22H	2579H	2C17H
-	2B16H	2576H	2C10H
*	2B42H	26F4H	2055H
/	1B75H	2754H	2E35H

Tabella 2 - «Entry point» delle routine che implementano le quattro operazioni, nel caso di operandi interi, reali ed in doppia precisione.

allora si salterà ad 1A9DH (per ora non ce ne occuperemo), mentre se si tratta di un'istruzione logica o relativa alle divisioni tra interi allora si salterà ad 1AABH (anche di questo parleremo solo in seguito).

Ecco dunque che arrivati all'indirizzo 1A49H sono rimaste le quattro operazioni: in questo caso il valore del FAC deve essere salvato nello stack, per essere poi utilizzato nell'operazione quando nell'analisi dell'espressione arriveremo al secondo operando. Ora a seconda del tipo del primo operando (che potrebbe essere benissimo il risultato di operazioni precedenti e perciò posto nel FAC), verranno salvati nello stack un certo numero di coppie di byte: due per il tipo «intero», quattro per il tipo «reale» ed otto per il tipo «doppia precisione».

Fatto ciò siamo arrivati all'indirizzo 1A70H da dove viene salvata nello stack anche la coppia BC contenente in B il tipo dell'operando ed in C il codice di priorità dell'operazione.

Subito dopo viene salvato un indirizzo di ritorno (nel nostro caso 1AD4H) dopodiché si salta a 19F6H e perciò ancora all'interno della routine di gestione dell'espressione.

Quando successivamente eseguendo questa routine si avrà una RET, allora si salterà all'indirizzo 1AD4H, dove avverrà la vera e propria esecuzione

288AH	FAC reale --> stack
2A05H	intero --> reale
2865H	cambia segno reale
2A7EH	INT(reale)
28A5H	FAC reale --> BC,DE
28A8H	(HL) --> BC,DE
2897H	(HL) --> FAC reale
2836H	(C06,C07) >0 => A = 0
	<0 => A = 1
	<0 => A = 0FFH

Tabella 3 - «Entry point» di alcune routine di uso generale che si incontreranno più volte nell'analisi dei comandi dell'MBASIC.

dell'operazione (somma, sottrazione, moltiplicazione o divisione).

Per non appesantire con un lungo listato e relativo commento il già complicato discorso, riportiamo nelle tabelle 2 e 3 i risultati sotto forma di entry point delle routine relative alle quattro operazioni, nonché di alcune routine onnipresenti ed importanti.

In particolare a partire dall'ultimo indirizzo citato sia ha un controllo tra i tipi dei due operandi, che potranno essere uguali oppure differenti: nel caso che siano uguali allora si utilizzerà direttamente la tabellina 2, dalla quale si ricava l'indirizzo della routine relativa all'operazione da compiere in funzione del tipo degli operandi. Sono appunto questi «entry point» ed in special modo quelli relativi ad operazioni reali che ritroveremo nell'«esplosione» delle funzioni logaritmo-trigonometriche.

Invece se il tipo degli operandi è diverso allora si effettuerà la conversione a favore del tipo «più potente», per avere sempre la migliore precisione possibile: tra operando in doppia precisione e reale o intero si avrà la conversione del reale o dell'intero in doppia precisione, effettuando la chiamata alla routine che esegue l'istruzione CDBL (Convert Double). Invece tra operando in singola precisione ed intero si avrà la conversione di quest'ultimo a reale per mezzo della routine che implementa la funzione CSNG (Convert Single precision).

Nei tre casi in cui le coppie di operandi sono dello stesso tipo, come detto si effettuerà un'operazione più semplice, che consiste nel ricavare da tre tabelle l'indirizzo a cui saltare: come è riportato nella tabella 2, la routine relativa ad operazioni tra interi (che inizia all'indirizzo 1B09H) si riferisce alla tabella posta a partire da 04A9H, mentre la routine relativa a quantità reali (posta a partire da 1B58H) fa riferimento alla tabellina posta all'indirizzo 049FH. Infine la routine che inizia all'indirizzo 1B1CH, relativa alle quattro operazioni applicate ad operandi espressi in doppia precisione, fa riferimento alla tabella di indirizzi posta a 0495H.

Considerazioni e Routine varie

Nella tabella 3 abbiamo riportato l'entry point di altrettante routine che si incontrano spesso analizzando l'implementazione degli operatori e delle funzioni: ne parliamo brevemente adesso, salvo ritornare per i dettagli se

ci sarà bisogno, in quanto già «a colpo d'occhio» potremo capire il significato di certe chiamate a subroutine oppure salti condizionati ad indirizzi che altrimenti non direbbero granché.

Prima ancora della descrizione delle routine, vediamo come viene effettuata una generica operazione tra due operandi (operazione che perciò si dice «binaria», ma non perché coinvolga numeri binari...) ed in particolare vediamo «dove» sono posti gli operandi.

Per quanto riguarda le operazioni tra interi, il primo operando (quello, per intenderci, che già era stato calcolato o «caricato») si troverà nel FAC (ricordiamo che la sigla sta per «Floating ACcumulator») ed in particolare agli indirizzi 0C04H e 0C05H, rispettivamente per la parte bassa e la parte alta del valore espresso in logica complemento a 2; invece il secondo operando (quello incontrato più recentemente) «galleggerà» nello stack, pronto ad essere posto nella coppia di registri DE.

Viceversa il primo operando dal FAC verrà spostato nella coppia HL e con tali coppie di registri verranno effettuate le operazioni richieste: per l'addizione, la sottrazione e la moltiplicazione vale la regola che il calcolo viene portato avanti fino a che è possibile e cioè fino a che non si ha un trabocco del risultato al di fuori dei 16 bit di HL (e questo per la moltiplicazione è una limitazione); in tal caso, l'operazione viene dirottata nel campo dei reali (in singola precisione), non prima di avere convertito entrambi gli operandi in quantità in singola precisione.

Per la divisione invece, dato che esistono i due operatori «/» e MOD prettamente dedicati a quantità intere, i progettisti hanno scelto di convertire subito entrambi gli operandi in singola precisione per poi andare ad effettuare la divisione tra due valori reali: questo ancora una volta per rispettare il principio di fornire al programmatore sempre la migliore precisione possibile.

Per quanto riguarda invece coppie di operandi espresse in singola e doppia precisione il discorso si fa un tantino più complicato in quanto entrano pesantemente in gioco, specie per l'addizione e la sottrazione, gli «ordini di grandezza» dei due operandi: infatti il programma che implementa l'addizione tra due quantità in singola precisione (ed a maggior ragione in doppia precisione) deve prevedere anche il caso di somma di due quantità, una mol-

```

288AH  EX DE,HL
        LD HL,(0C04H)
        EX (SP),HL
        PUSH HL
        LD HL,(0C06H)
        EX (SP),HL
        PUSH HL
        EX DE,HL
        RET
    
```

Listato 2 - Routine 288AH che sposta il contenuto del FAC (Floating ACcumulator), in singola precisione, nello stack.

to grande e l'altra molto piccola, diciamo ad esempio $5 \cdot 10^{10}$ da sommare a $2 \cdot 10^{-5}$, tenendo evidente in conto il numero di cifre significative spettanti all'uno o all'altro grado di precisione richiesto.

Passando ora alle routine della già citata tabella 3, vediamo innanzitutto la prima (288AH) che permette di trasferire il contenuto del FAC relativo a quantità reali nello stack, senza alterare il contenuto del FAC stesso: è molto utile ad esempio quando vogliamo calcolare il quadrato di una certa quantità (come vedremo ad esempio nello sviluppo della funzione trigonometrica SIN).

In questo caso infatti, supponendo che il primo operando sia nel FAC, lo si può ricopiare nello stack con la chiamata alla subroutine in esame e poi si può effettuare la chiamata alla moltiplicazione tra reali (26F4H) per ottenere appunto il quadrato.

Dimenticavamo di dire che la moltiplicazione in questione (ma ne riparleremo in seguito) opera tra il contenuto del FAC ed il contenuto questa volta di BC e DE (in BC la parte più significativa).

Ecco che perciò subito prima di effettuare la chiamata a 26F4H bisognerà trasferire il secondo operando dallo stack alle due coppie di registri BC e DE, non certo con un'altra chiamata a subroutine, ma con due semplicissime POP.

La seconda routine (2A05H) invece svolge principalmente la funzione di conversione di una quantità intera in una in singola precisione, con un meccanismo che ricalca la struttura di un numero reale (struttura di cui abbiamo parlato in una scorsa puntata).

La terza routine (2865H) non è una routine a sé stante, ma fa parte della routine che implementa la funzione ABS, nella parte riguardante i numeri reali: presa così a sé, effettua semplicemente il cambio del segno della quantità reale posta nel FAC: anche questa routine verrà chiamata più volente.

La successiva routine (2A7EH) nasce anch'essa come parte di una subroutine più generale: si tratta della routine che implementa il comando INT, anche in questo caso relativo a quantità espressa in singola precisione.

Le due successive routine (28A5H e 28A8H) sono in realtà attaccate l'una all'altra: la seconda permette di caricare le coppie di registri BC e DE con i contenuti delle quattro locazioni puntate da HL. Sarà molto utile quando dovremo estrarre (nel caso del seno, ad esempio) dalla memoria i quattro byte rappresentanti un coefficiente di una certa serie.

La 28A5H invece setta il valore di HL a 0C04H, proprio l'indirizzo iniziale del FAC nel caso di quantità reali, per poi eseguire (è proprio all'indirizzo successivo) la routine appena vista di caricamento di BC e DE: in definitiva serve per copiare il contenuto reale del FAC nella coppia BC, DE ed è in un certo senso alternativa alla prima routine vista.

La penultima routine (2897H) permette di spostare il contenuto di quattro celle di memoria consecutive, la prima delle quali puntata da HL, nel FAC: per ottenere ciò dapprima effettua la chiamata alla 28A8H e poi scarica il contenuto di BC e DE rispettivamente in 0C06H, 0C07H ed in 0C04H, 0C05H.

L'ultima routine invece è una nostra vecchia piccola conoscenza, in quanto l'abbiamo già vista nella scorsa puntata all'interno della subroutine che implementa la funzione SGN: in particolare, come riportato nella tabella, serve a settare l'accumulatore (A) dello Z80 con un valore pari a 0, 1 o 0FFH a seconda se nei due byte più significativi del FAC reale (0C06H e 0C07H) è contenuto un valore relativo a una quantità reale rispettivamente nulla, maggiore oppure minore di zero.

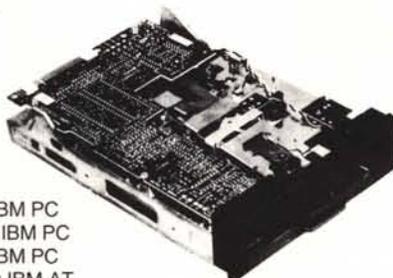
Concludiamo dunque questa puntata segnalando che nel listato numero 2 abbiamo riportato il disassemblato della routine 288AH, molto semplice, ma istruttiva in quanto contiene al suo interno due istruzioni EX (SP), HL, con le quali si può caricare nello stack una coppia di word (costituenti il valore contenuto nel FAC), lasciando però sempre «affiorante» un indirizzo di ritorno: in caso contrario e cioè con due semplici PUSH l'indirizzo di ritorno scenderebbe di due livelli ed un eventuale RET porterebbe lo scompiglio. Nella prossima puntata dunque inizieremo ad analizzare, dopo tante premesse, l'implementazione delle quattro operazioni nei tre casi di operandi, interi, reali ed in doppia precisione.

SIG.ri RIVENDITORI

CHINON DISK DRIVE

10 VOLTE
PIÙ SILENZIOSI
DEGLI ALTRI!!!

GARANZIA
1 ANNO



TIPI:
F-502 360K x IBM PC
F-502L 360K x IBM PC
F-561 1 MB x IBM PC
F-506 1,6 MB x IBM AT
F-051 180K x Apple (1S)
F-T36 360K x Apple (2S)

DISPONIBILI ORA I NUOVI MODELLI CON CHIAVETTA
PREZZI: DA LIT. 288.000



300 B.P.S.
SOLO
L. 189.000

101 Data Modem - Modem interfacciabile con RS-232 con tutti i micro e personal computer (Apple, IBM, Atari, Commodore, ecc.). 300 BPS, Auto Dial, Auto Answer, Full Duplex. Led indicatori di C. Data Receive Data, Transmit Data, Power On.

II E COMPATIBILE + PAD NUMERICO

128K RAM
+ 80 Colonne



II E compatibile al 100% dotato di comodo Pad numerico. Viene fornito completo di scheda 80 colonne + 64K L. 760.000

II E COMPAT. CON TASTIERA SEPARATA

128K RAM
+ 80 Colonne



II E compatibile al 100%. Versatile realizzazione con tastiera separata, intelligente. Pad numerico, Tasti funzione definibili, Autorepeat e «Bip» disinsensibile. Possibilità di inserimento di N. Due Drive Slim all'interno del Cabinet. L. 920.000

HARD DISK + CAVI E CONTROLLER

- Teac 10 MB L. 1.620.000
- Shugart 10 MB L. 1.890.000
- Seagate 20 MB L. 1.990.000
- Mitsubishi 20 MB L. 2.460.000

CASSETTE BACKUP 20 MB

Streamer completo di interfaccia e soft di gestione per IBM PC/XT prodotto dalla Teac consente di fare la copia di 20 MB su cassetta in 11' occupa il posto di un drive slim. L. 2.250.000

CHI VI DA UN ASSORTIMENTO COSÌ COMPLETO CON PREZZI SUPER COMPETITIVI ???

Basta una telefonata ed in 48 ore riceverete quanto ordinato con garanzia 6 mesi od 1 anno e, se non sarete soddisfatti, vi sostituiremo l'articolo con lo stesso modello o con altro materiale a patto che il reso ci pervenga non manomesso, in porto franco, con gli imballi originali entro 18 gg. dalla data di spedizione.

RICHIEDETEVI OGGI STESSO LO «SPECIALE PROMOZIONE»



AT
COMPATIBILE

Versione Base Main Board OK espandibile ad 1 MB, alimentatore 200 W, Cabinet in metallo, tastiera

L. 3.950.000

300 B.P.S.
SOLO
L. 264.000



9201 ACUSTIC DATA MODEM (RS-232)
Modem acustico per qualsiasi tipo di computer. Compatibile con l'interfaccia RS-232. Data rate: 300 bps. Transmit level: - 10 dbm. Receive sensity: - 45 dbm. Communication mode: full duplex. Compatibility: 4 CCITT. Frequency: ± 0,1% (crystal controlled).

MODEM
MODEM
MODEM

PC/XT TURBO

L. 1.680.000

Clock 6,67-4,77 MHz
Main Board Esp. 640K



N. 1 Drive DS/DD 360K controller,
Main Board OK espandibile A 640K,
Alimentatore 130 W, Tastiera K5 S

PC/XT STANDARD (4,77 MHz)

L. 1.420.000

Configurazione come sopra ma con Main Board 128K espandibile a 256K

*** Per le interfacce video vedere listino***

- Monitor Philips Monocr. x IBM L. 227.000
- Monitor Cabel MR Colori x IBM L. 439.000
- Monitor Philips HR Colori x IBM L. 690.000

DATAFLEX PROFESSIONAL 5 1/4

Dischetti professionali 100% Error Free

CON BOX IN PLASTICA OMAGGIO!!!
SCONTI PER QUANTITÀ

SINGOLA F. - DOPPIA D.	DOPPIA F. DOPPIA D.
200 Pezzi 2.230	200 Pezzi 2.870
100 Pezzi 2.350	100 Pezzi 3.150
30 Pezzi 2.550	30 Pezzi 3.400

PREZZI
IVA
ESCLUSA



LA CASA DEL
COMPUTER

IMPORTAZIONE DIRETTA

0587
212.312



VIA MISERICORDIA, 84 - 56025 PONTEDERA (PI)



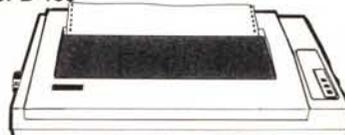
STAMPANTI PER IBM

C.T.I CPB-80



130 C.P.S., 80 Colonne, 2K Buffer Down Load, Set Grafico IBM L. 550.000

C.T.I. CPB-136



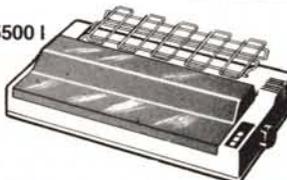
130 C.P.S., 136 Colonne, 2K Buffer Down Load, Set Grafico IBM. L. 860.000

COPAL SC-1200 L



120 C.P.S., 80 Colonne, Near Letter Quality, Set Grafico IBM L. 650.000

COPAL SC-5500 I



180 C.P.S., 132 Colonne, 3K Buffer, Down Load, Near Letter quality, Set Grafico IBM L. 1.150.000

STAR GEMINI 10X
Superofferta!!! Telefonare

PLOTTER E DIGITIZER

- KDC FPL-2000 L. 1.500.000
- Hitachi BIG-3 L. 2.430.000
- Digitizer 11" Tiger 11 L. 1.850.000
- Digitizer 43 KS10-MK2 L. 1.960.000

I plotter sono formato A3, dotati di procedura X-ON/X-OFF ed Ingresso Parallelo/Seriale.

Disponibili a magazzino anche tutti gli accessori: Pennini, pennarelli e relativi adattatori.

RICHIEDETEVI SPECIFICHE

DATO L'INSTABILE MERCATO DEI CAMBI PREGASI TELEFONARE PER CONFERMA PREZZI E DISPONIBILITÀ
— RICHIEDETEVI IL CATALOGO —