

**ice**  **felix**®

COMPUTER S.A.

**CP/M**

**MANUAL DE UTILIZARE**

## CUPRINS

Generalități	5
Comenzi	6
BAUD	6
DIP	6
DDT	7
DIR	9
DUMP	9
ED	9
ERA	11
FORMAT	11
LOAD	11
PIP	12
REN	14
SACE	14
STAT	15
SUBMIT	16
SYSGEN	16
TYPE	17
USER	17
XSUB	17
Caractere de control CP/M	18
Tipuri de fișiere CP/M	18
Mesaje de eroare CP/M	19

P PRINT  
L USR } => PRINT USR  
13446 13446

## GENERALITĂȚI

### Lansarea unei comenzi CP/M

Pentru lansarea unei comenzi CP/M trebuie introdusă o linie completă de comandă ca răspuns la "prompt"-ul sistemului.

O linie de comandă CP/M se compune dintr-un cuvânt cheie de comandă, o parte opțională și <CR>.

- Cuvântul cheie de comandă identifică o comandă ce urmează a fi executată.
- Partea opțională constă în numele fișierului și opțiuni sau parametri.
- Pentru a încheia comanda se apasă tasta "CR".

### Specificații de fișier CP/M

CP/M identifică fiecare fișier prin specificații proprii.  
O specificație de fișier cuprinde 3 părți:

- specificația drive-ului;
- numele fișierului;
- tipul fișierului.

Termenul "filespec" este o abreviere pentru specificația pe fișier și indică orice combinație validă între specificația de drive, numele fișierului și tipul fișierului.

Acest manual utilizează următoarele simboluri pentru a desemna părțile unei "filespec".

**d:** - reprezintă specificația opțională de drive care trebuie să fie formată dintr-un singur caracter alfabetic de la A la D, urmat de două puncte;

**filename** - reprezintă indicația necesară a numelui fișierului care poate avea 1 pînă la 8 caractere alfabetic sau numerice;

**.typ** - reprezintă indicația opțională a tipului de fișier (extensia), care poate fi alcătuită din 0 pînă la 3 caractere alfabetic sau numerice separate de numele fișierului printr-un punct.

Combi-națiile valide ale elementelor unei "filespec" sînt următoarele:

filename  
d:filename  
filename.typ  
d:filename.typ

Anumite comenzi CP/M pot selecta și prelucra mai multe fișiere cînd în "filename" sau "filetype" apar caractere speciale.

Cele două caractere speciale sînt "?", care înlocuiește orice literă din aceeași poziție și "\*", care înlocuiește orice caracter din acea poziție și orice alte caractere rămase în "filename" sau "filetype".

Manualul indică ce comenzi acceptă caracterele "?" sau "\*" în cadrul "filename" sau "filetype".

### Convenții în cadrul manualului

Manualul prezintă comenzile CP/M în ordine alfabetică folosind următoarele sim-

boluri speciale pentru definirea sintaxei comenzii:

{ }	indică un termen opțional în cadrul DDT și ED
	separă alternativele
n	indică un număr
<CR>	indică apăsarea tastei "CR"
^ sau CTRL	indică apăsarea tastei "CTRL"
o	indică o opțiune sau o listă de opțiuni
R/W	înseamnă "Read-Write"
R/O	înseamnă "Read-Only"
SYS	înseamnă atributul "System"
DIR	înseamnă atributul "Directory"
...	elementul precedent poate fi repetat de câte ori se dorește
*	înlocuiește tot sau o parte a numelui și/sau a tipului fișierului
?	înlocuiește un singur caracter în aceeași poziție

### BAUD

Programul BAUD va fixa temporar, pentru "printer", rata de BAUD specificată în linia de comandă.

BAUD va accepta numai ratele 110, 150, 300, 600, 1200, 4800 și 9600, cu excepția cazului în care rata este precedată de un caracter "@", caz în care este acceptată orice valoare între 110 și 9600.

Alte rate decât valorile specificate nu pot fi exacte din cauza rotunjirii de către circuitul de divizare care generează impulsurile de ceas pentru "UART"-uri.

Noua rată de BAUD pentru "printer" se va păstra pînă la rularea din nou a programului BAUD sau pînă la execuția unui IPL ROM (start rece). Startul cald nu va afecta rata de BAUD a "printer"-ului.

### DIP

Comanda DIP se utilizează pentru schimbul de fișiere, realizînd funcțiile de bază ale copierii fișierelor între diferite discuri. Această comandă este necesară în configurațiile "single drive" unde comanda PIP din CP/M-ul standard nu poate fi utilizată.

Programul începe cu introducerea comenzii DIP într-una din formele:

(1) DIP

(2) DIP "command line"

În ambele cazuri DIP este încărcat în TPA și executat.

În primul caz va apare un caracter "\*" prin care se cere introducerea de la consolă a liniilor de comandă ce vor fi citite de DIP. Aceasta se face pînă la introducerea unei linii de comandă goale (este suficient un <CR>). Fiecare linie de comandă realizează o funcție de copiere descrisă în liniile următoare.

A doua formă a comenzii DIP este echivalentă cu prima, dar în acest caz este executată automat unica linie de comandă, execuția terminîndu-se imediat, fără a mai fi cerute de la consolă și alte linii de comandă.

Formatul fiecărei linii de comandă este:

### DESTINATION = SOURCE

unde "destinația" este fișierul sau drive-ul care va primi informația și "sursa" este fișierul de copiat. Ambele fișiere trebuie să fie rezidente pe disc.

Fișierele "sursă" și "destinație" trebuie să figureze pe drive-uri logic diferite. Deși sistemul este "single disk drive", sistemul de operare trebuie să vadă fișierele "sursă" și "destinație" pe drive-uri logic diferite, dacă discurile "sursă" și "destinație" nu sînt aceleași. "Hardware"-ul sistemului va selecta "single drive"-ul în cadrul sistemului și ca drive A., și ca drive B.:

Specificațiile fișierelor "sursă" și "destinație" pot fi ambigue (se pot utiliza caracterele "\*" și "?").

Exemple:

A:X=B:Y <CR>

X=B:Y <CR>

B:X.ASM=A:Y.ASM <CR>

B:=A:Y.ASM <CR>

B:=\*.\* <CR>

Ultimul format este utilizat pentru copierea tuturor fișierelor de pe drive-ul curent (trebuie să fie A:), pe alt disc.

Ca și PIP, comanda DIP acceptă opțiuni scrise între paranteze drepte, după specificația fișierului "sursă".

Există doar 2 opțiuni:

Q - Opțiunea de interogare. Programul va interoga utilizatorul în legătură cu fiecare din fișierele cuprinse într-o specificație ambiguă, dacă este sau nu necesară copierea lui. Dacă operatorul răspunde cu "Y" sau "y", fișierul este copiat, altfel nu.

V - Opțiunea de verificare. Verifică dacă datele au fost corect scrise pe discul "destinație" prin citirea lor după operația de scriere.

Exemplu:

B:=\*.\*[QV] <CR>

Exemplul realizează copierea tuturor fișierelor de pe discul din drive-ul A: pe discul din drive-ul logic B, întrebînd operatorul și verificînd datele.

Programul va cere utilizatorului mai întîi încărcarea discului "sursă", va afișa apoi numele fiecărui fișier încărcat, încărcarea fișierelor făcîndu-se pînă la umplerea memoriei sau a listei fișierelor, apoi va cere utilizatorului să încarce discul "destinație", după care va scrie fișierele pe discul "destinație". La sfîrșitul operației va apare mesajul: DIP COMPLETE

### DDT

Sintaxa: DDT filespec

DDT este utilizat în depanarea programelor 8080. Dacă se specifică DDT fără "filespec", DDT intră în zona de memorie liberă și așteaptă o comandă. Caracterul de comandă poate fi urmat de unul sau mai multe argumente: valori hexazecimale, specificații de fișier, sau altă informație, depinzînd de comandă. Argumentele sînt separate unul de

celălalt prin virgule sau spații.

Spațiile sînt permise între caracterul de comandă și primul argument. Părțile opționale ale comenzii sînt incluse între acolade.

Exemple:

A » DDT  
A » DDT PROGRAM.COM

Comenzi în cadrul DDT:

As (Assemble)  
introduce date în limbaj de asamblare

D{s,{f}} (Display)  
afișează memoria în HEX și ASCII

Fs,f,bc (Fill)  
umple un bloc de memorie cu valoarea "bc"

G{s}{,b1{b2}} (Go)  
începe execuția

GO - se iese din DDT

Hc1,c2 (Hex)  
suma și diferența hexazecimală

Ifilespec (Input)  
setează linia de comandă de intrare pentru comenzi R sau W

L{s,{f}} (List)  
listează memoria în forma mnemonică

Ms,f,d (Move)  
mută un bloc de memorie

R{o} (Read)  
citește un fișier de pe disc

Ss (Set)  
configurarea octet de memorie

T{n} (Trace)  
trasează execuția programului

U{n} (Untrace)  
execuția programului fără trasare

X{r} (Examine)  
examinează și modifică registrele CPU

Simbol	Semnificație
bc	byte constant
b1	primul punct de întrerupere
b2	al doilea punct de întrerupere
c1	primul byte sau cuvînt constant
c2	al doilea byte sau cuvînt constant
d	destinația datei
f	adresa finală
n	numărul de instrucțiuni de executat
o	offset la adresa de încărcare
r	registrul CPU 8080
s	adresa de început

### DIR

Sintaxa: **DIR d:filename.typ**

DIR afișează numele fișierelor de pe discul specificat, dar nu afișează fișierele setate cu SYS.

Exemple:

```
DIR
DIR B:
DIR B:MYFILE.TEX
DIR A*.ASM
DIR PROG???.PRN
DIR PROGRAM.*
```

### DUMP

Sintaxa: **DUMP filespec**

DUMP afișează la consolă, în format hexazecimal, conținutul fișierului de pe disc specificat de "filespec". Conținutul fișierului este afișat pe linii de 16 octeți, adresa de început a fiecărei linii fiind dată în hexazecimal în stînga liniei. Afișările prea lungi pot fi întrerupte apăsînd tasta "rubout" în timpul afișării.

### ED

Sintaxa: **ED d:filename.typ**

ED este editorul CP/M. Cu ajutorul subcomenzilor ED se pot crea sau modifica fișierele sursă.

Exemplu:

```
ED TEST.DAT
```

Caractere de control:

În ED tasta "CTRL" apare ca o săgeată (^)  
 ^L <CR> <LF> între şirurile de date  
 ^X şterge linia  
 ^Z terminator/separator de date, ieşire din modul "insert"  
 rubout şterge un caracter  
 break anulează comanda

Comanda	Acţiunea
nA	anexează "n" linii din fişierul original la bufferul de memorie
OA	anexează fişierul pînă la umplerea pe jumătate a bufferului
#A	anexează fişierul pînă la umplerea bufferului (sau pînă la sfîrşitul fişierului)
B,-B	mută cursorul la începutul (B) sau sfîrşitul (-B) bufferului
nC,-nC	mută cursorul cu "n" caractere înainte (C) sau înapoi (-C) în buffer
nD,-nD	şterge "n" caractere înainte (-D) sau după (D) cursor
E	salvează noul fişier şi revine în CP/M
Fşir de date { ^Z }	găseşte şirul de caractere
H	salvează noul fişier apoi reeditează, considerînd noul fişier drept fişierul original
I	intră în modul "insert"; pentru ieşire se utilizează ^Z
Işir de date { ^Z }	inserează un şir de date
Jşir de căutat ^Zşir de inserat Xşir la care se termină ştergerea { ^Z }	juxtapunere şiruri de date
nK,-nK	şterge "n" linii de la cursor
nL,-nL,OL	mută cursorul cu "n" linii
nMcomenzi	execută comenzile de "n" ori
n,-n	mută cursorul cu "n" linii şi afişează linia
n:	trece la linia "n"
n:comanda	execută comanda din linia "n"
Nşir de date { ^Z }	extinderea şirului găsit
O	se revine în fişierul original
nP,-nP	mută cursorul cu 23 de linii înainte sau înapoi şi afişează 23 de linii la consolă
Q	abandonează noul fişier; se revine în CP/M
R	citeşte fişierul X\$\$\$\$\$\$\$.LIB în buffer
Rspecificăţie de fişier	citeşte "nume fişier.LIB" în buffer
Sşir de şters ^Zşir de inserat { ^Z }	substitue un şir de date
nT,-nT,OT	tipăreşte "n" linii
U,-U	translatarea în litere mari
V,-V	validează (V), invalidează (-V) afişarea numărului de linii
OV	afişează spaţiul liber rămas în buffer
nW	scrie "n" linii în noul fişier
nX	scrie "n" linii în fişierul X\$\$\$\$\$\$\$.LIB temporar
OX	şterge fişierul X\$\$\$\$\$\$\$.LIB

Comenzi:  
 ERA

Sintaxa: ERA d:filename.typ

ERA şterge un fişier sau un grup de fişiere care nu sînt R/O. ERA acceptă caracterele "\*" şi "?" în specificaţia numelor de fişiere.

Exemple:

```
ERA PROGRAM.BAS
ERA B:LETTER.DAT
ERA A:LETTER.*
ERA *.BAK
ERA B:*.*
```

FORMAT

Înainte utilizării unui disc nou este necesară operaţia "formare", chiar înaintea comenzii SYSGEN. Această operaţie afectează tot discul distrugînd orice informaţie scrisă anterior şi înregistrează informaţia necesară controller-ului de floppy disk.

Pentru a "forma" un disc, se va proceda conform instrucţiunilor următoare (caracterele introduse de operator sînt subliniate):

```
FORMAT <CR>
DESTINATION DRIVE NAME (OR RETURN TO REBOOT) A
DESTINATION DISK ON A, THEN TYPE RETURN
se introduce discul nou <CR>
ARE YOU AWARE THAT THIS FUNCTION DESTROYS
ALL THE FILES STORED ON THIS DISK?Y
HAVE YOU SAVED ALL YOUR IMPORTANT FILES?Y
ARE YOU SURE?Y
se "formează" discul
FUNCTION COMPLETE
VERIFY?Y
FUNCTION COMPLETE
DESTINATION DRIVE NAME (OR RETURN TO REBOOT) <CR>
A >
```

Opţional, comanda verifică discul după "formare". Dacă apare o eroare în timpul verificării, apare mesajul "DISK ERROR" şi verificarea este abandonată. În acest caz este necesară o a doua încercare de "formare" a discului. Se pot încărca oricîte discuri cu o singură comandă FORMAT, urmînd instrucţiunile date de calculator.

**LOAD**

Sintaxa: LOAD filespec

Comanda LOAD citeşte fişierul specificat (care trebuie să conţină codul maşină în format hexazecimal) şi produce un fişier imagine în memorie care să poată fi executat ulterior.

Numele fișierului specificat trebuie să fie de forma x.HEX și de aceea este suficient să se specifice doar numele "x" în comandă.

Comanda LOAD creează un fișier numit x.COM care poate fi încărcat efectiv în memorie și executat la cererea utilizatorului. Acesta va specifica numele "x" al fișierului după apariția "prompt"-ului sistemului.

În general CCP-ul (procesorul de comandă a consoli) citește numele "x" și caută o funcție fixă a acestui fișier. Dacă nu o găsește, caută în "directory"-ul discului sistem un fișier cu numele X.COM. Dacă acesta este găsit, codul mașină este încărcat în TPA și programul este executat. Astfel utilizatorul nu are nevoie să încarce fișierul HEX, decât o singură dată, el putând fi apoi executat de orice număr de ori specificându-se doar numele său primar. În acest mod utilizatorul poate crea comenzi CP/M noi.

Operația se poate face pe un disc alternativ, dacă numele fișierului este precedat de un nume de drive.

Astfel, "LOAD B:BETA" transferă în TPA programul LOAD al discului curent încărcat și operează pe drive-ul B.

De notat că fișierul BETA.HEX trebuie să conțină înregistrări valide în cod mașină hexazecimal de format INTEL (de exemplu: cele produse de ASM) care încep la adresa 100H, începutul TPA.

În plus, adresele înregistrării hexazecimale trebuie să fie în ordine crescătoare; golurile din regiunile de memorie sînt umplute cu zerouri, de comanda LOAD, în același timp cu citirea înregistrării hexazecimale.

Deci LOAD nu trebuie utilizat decât pentru crearea fișierelor ".COM" standard pentru CP/M care operează în TPA.

Programele care ocupă alte zone de memorie decât TPA pot fi încărcate cu DDT.

## PIP

Sintaxa:

PIP destination = source

PIP d: | filespec = filespec [opțiuni]

PIP filespec = d: [opțiuni]

PIP d: filename.type = filespec1, filespec2, ...

PIP filespec | device: = filespec [opțiuni] | device: [opțiuni]

PIP < CR >

\*

PIP copiază fișiere, combină fișiere și transferă fișiere între dispozitivele periferice.

Prima specificație de fișier este destinația, a doua este sursa. Specificațiile fișierelor sursă cu opțiuni pot fi repetate, separate de virgule, în vederea combinării a două sau mai multor fișiere într-un fișier concatenat. Sursă sau destinație poate fi orice dispozitiv logic CP/M. PIP, dat fără nici o specificație, afisează un "\*" și așteaptă comenzi executînd câte o linie de fiecare dată, pînă la prima linie liberă.

Exemple:

COPIERE de pe un disc pe altul:

PIP B: = A:DRAFT.TEX

PIP B:DRAFT.TEX = A:

COPIEREA unui fișier și redenumirea lui:

PIP B:NEWDRAFT.TEX = A:OLDDRAFT.TEX

PIP NEWDRAFT.TEX = OLDDRAFT.TEX

COPIEREA mai multor fișiere:

PIP < CR >

\* B: = \*.COM[RW]

< CR >

PIP B: = \*.TEX[AV]

PIP B: = C:DRAFT.\*

PIP B: = \*.\*

PIP B: = C:.\*

COMBINAREA mai multor fișiere:

PIP B:NEW.DAT = FILE1.DAT,FILE2.DAT

COPIERE, redenumire și preluare de la utilizatorul 1:

PIP NEWDRAFT.TEX = OLDDRAFT.TEX[G1]

COPIERE pc/de pe dispozitive logice:

PIP B:FUNFILE.SUE = CON:

PIP LST: = CON:

PIP LST: = B:DRAFT.TEX[T8]

PIP PRN: = B:DRAFT.TEX

Opțiuni PIP:

Opțiune	Descriere
A	arhiva: copiază numai fișierele modificate
B	transfer prin intermediul unui buffer
Dn	șterge orice caracter după coloana "n"
E	ecou la consolă
F	filtrează "form-feeds" din fișierul copiat
Gn	preia fișierul de la utilizatorul "n"
H	test pentru formatul Hex
I	eliminarea înregistrărilor nule din fișierele hexa
L	translatarea literelor mari în litere mici
N	numără liniile de ieșire, incrementînd cu 1, cu începere de la 1. Numărul liniei este urmat de ":", iar zerourile nesemnificative sînt ignorate
N2	la fel ca N, dar zerourile nesemnificative nu sînt ignorate și după numărul liniei se introduce un TAB
O	transferă fișierul obiect, ^Z este ignorat
Pn	fixează lungimea paginii la "n" (în lipsa opțiunii, "n" = 60)
Qs ^Z	oprește copierea de pe fișierul sursă, la întîlnirea șirului "s"
R	citește fișierele care au fost puse pe SYS și ignoră atributul R/O
Ss ^Z	începe copierea de pe fișierul sursă la întîlnirea șirului "s"
Tn	setează TAB-urile la valoarea "n"
U	translatează literele mici în litere mari
V	verifică dacă datele au fost scrise corect
W	scrie peste fișierele care au fost puse pe Read-Only
Z	bitul de paritate este forțat pe zero

### Dispozitive logice:

Următoarele 4 dispozitive logice sînt definite în IOBYTE utilizînd STAT și trebuie definite și în BIOS.

CON: dispozitiv "Consola"  
LST: dispozitiv "List"  
PUN: dispozitiv "Punch"  
RDR: dispozitiv "Reader"

### Dispozitive speciale recunoscute de PIP:

INP: intrare definită de utilizator  
OUT: ieșire definită de utilizator  
PRN: ca LST; TAB-uri de cîte 8 caractere, pagina de 60 de linii, numerotarea liniilor  
EOF: generează CTRL-Z (sfîrșit de fișier)  
NULL: trimite 40 de "null-uri la dispozitivul "Punch"

### Dispozitive fizice:

TTY: Consola, terminal, "reader", "punch", "teletypewriter"  
CRT: Consola, terminal, dispozitiv "CRT"  
PTR: Cititor de bandă sau de cartele  
PTP: Perforator de bandă sau cartele  
LPT: Dispozitiv "List", imprimantă  
UC1: Consolă definită de utilizator  
UR1: "Reader" definit de utilizator  
UR2: "Reader" definit de utilizator  
UP1: "Punch" definit de utilizator  
UP2: "Punch" definit de utilizator  
UL1: Dispozitiv de listare definit de utilizator

### REN

Sintaxa: **REN d:newfile.typ = oldfile.typ**

REN schimbă numele fișierului existent, numit fișierul vechi, cu numele fișierului nou.  
Exemple:

```
REN NEWFILE.DAT = OLDFILE.DAT
REN B:NEWFILE.DAT = OLDFILE.DAT
REN NEWLIST = OLDLIST
```

### SAVE

Sintaxa: **SAVE n d:filename.typ**

SAVE copiază pe disc "n" pagini (1 pagină = 256 octeți) din conținutul zonei temporare de memorie (Transient Program Area - TPA), începînd de la locația 100 Hex. SAVE asignează numele și tipul fișierului specificate în comandă. Caracterele "\*" și "?" nu sînt acceptate. Numărul paginilor de salvat poate fi calculat utilizînd DDT.

### Exemple:

```
SAVE 3 EXTRA.COM
SAVE 40 QUICK
SAVE 10 B:ZOT.COM
```

### STAT

Sintaxa:  
Afișează starea:

```
STAT
STAT d:
STAT filespec
STAT d:DSK
STAT DEV:|VAL:|USR:
```

Modifică starea:

```
STAT d: = R/O
STAT filespec $R/O|R/W|SYS|DIR
STAT filespec $$
```

STAT dă informații despre fișiere, drive-uri și alte dispozitive periferice atașate calculatorului. STAT modifică atributele drive-urilor și fișierelor. STAT, fără altă specificație, informează despre capacitatea de memorie liberă, în Kbytes, pentru drive-ul curent și dacă drive-ul respectiv este Read-Only (R/O) sau este Read-Write (R/W). STAT poate fi utilizat pentru a seta drive-ul pe R/O și CTRL-C pentru repunerea lui pe R/W. "STAT filespec" dă numărul de Kbytes utilizați de un fișier și atributele acestui fișier. "STAT filespec" poate, cu o opțiune, să seteze un fișier sau mai multe pe R/O, R/W, SYS sau DIR. STAT afișează fișierele puse pe SYS între paranteze.

STAT DSK: afișează caracteristicile drive-ului specificat;

STAT USR: arată care dintre numerele de utilizatori conțin fișiere pe o anumită dischetă;

STAT VAL: arată comenzile posibile și dispozitivele externe care pot fi atribuite calculatorului;

STAT DEV: afișează atribuirile dispozitivelor logice curente.

STAT acceptă caracterele "\*" și "?" în cadrul comenzilor.

Exemple:

```
STAT
STAT myfile.txt
STAT C:letter.bak
STAT genledgr.dat $R/O
STAT *.com $R/O
STAT *.bak
STAT B:*.*
```



## Opțiuni:

R/W	Read-Write
R/O	Read-Only
SYS	atribut de sistem
DIR	fără atribut de sistem
S	afișează mărimea fișierului sau fișierelor bazate pe numerele ultimei înregistrări
USR:	afișează numerele de utilizatori care conțin fișiere
DSK:	afișează caracteristicile drive-ului
VAL:	afișează comenzile STAT posibile și dispozitivele de intrare/ieșire
DEV:	afișează atribuiriile dispozitivelor logice curente

## SUBMIT

Sintaxa: **SUBMIT filename parametri**

SUBMIT începe execuția unui fișier de comenzi CP/M, câte o comandă pe fiecare linie. Un fișier SUBMIT trebuie să aibă extensia ".SUB". Orice parametri opționali urmînd după specificația fișierului în linia de comandă sînt substituiți pentru parametrii formali corespunzători (\$1,\$2,...) în fișierul SUBMIT.

Exemple:

```
SUBMIT START
SUBMIT B:START
SUBMIT START C:LETTER
```

În exemplele precedente fișierul START.SUB este un fișier SUBMIT. Liniile de comandă în START.SUB nu apar aici. În exemplul al doilea START.SUB este pe drive-ul B;. În ultimul exemplu parametrii actuali "C:" și "LETTER" urmează după numele fișierului SUBMIT. SUBMIT.COM crează un fișier temporar și substituie parametrii actuali din linia de comandă, pentru parametrii formali (\$1,\$2,...) ce apar în fișierul START.SUB.

"C:" este substituit pentru orice parametri "\$1", care apar în fișierul START.SUB.

## SYSGEN

Sintaxa: **SYSGEN**

SYSGEN copiază sistemul de operare CP/M de pe discul sursă pe un disc destinație. Se pot crea unul sau mai multe discuri sub CP/M cu o singură comandă SYSGEN. SYSGEN, cu o specificație de fișier, nu cere un drive sursă, dar folosește drive-ul specificat drept sursă.

Exemple:

Următorul exemplu creează un disc sub CP/M pe drive-ul B după un disc sub CP/M de pe drive-ul A.

```
A>SYSGEN
SYSGEN VER 2.0
SOURCE DRIVE NAME (OR RETURN TO SKIP)A
```

```
SOURCE ON A, THEN TYPE RETURN <CR>
FUNCTION COMPLETE
DESTINATION DRIVE NAME (OR RETURN TO REBOOT)B
DESTINATION ON B, THEN TYPE RETURN <CR>
FUNCTION COMPLETE
DESTINATION DRIVE NAME (OR RETURN TO REBOOT) <CR>
```

Pentru a crea un nou disc sub CP/M utilizînd un sistem "single drive", se pune un disc nou în A., după apariția indicației DESTINATION DRIVE NAME și se apasă tasta "A".

```
A>SYSGEN CPM32.COM
DESTINATION DRIVE NAME (OR RETURN TO REBOOT)A
DESTINATION ON A, THEN TYPE RETURN <CR>
FUNCTION COMPLETE
```

...

## TYPE

Sintaxa: **TYPE d:filename.typ**

TYPE afișează pe un ecran conținutul unui fișier ASCII. Pentru a opri afișarea, se apasă orice tastă. TYPE nu acceptă "\*" și "?" în specificațiile de fișier. Un caracter CTRL-P înaintea comenzii TYPE determină ca ieșirea să apară în ecou la imprimantă; cu un alt CTRL-P se poate opri tipărirea.

Exemple:

```
TYPE letter.dat
TYPE a:document.law
TYPE program.bas
TYPE B:reader.asm
```

## USER

Sintaxa: **USER n**

USER și un număr de la 0 la 15 schimbă numărul utilizatorului curent cu numărul specificat de "n". În lipsa comenzii USER, CP/M consideră că există un singur utilizator.

STAT USR: afișează utilizatorii curenți și activi.

Exemple:

```
USER 2
USER 7
```

## XSUB

Sintaxa: **XSUB**

XSUB, utilizat în cadrul unui fișier SUBMIT, permite unor programe să accepte comenzi dintr-un fișier SUBMIT și nu de la consolă. Comanda XSUB trebuie să apară

înaintea unei linii de comandă care necesită o comandă de la consolă. XSUB rămâne activ până la o nouă încălcare a sistemului de operare (cold start).

Exemplu:

```
TYPE SAVER.SUB
XSUB
DDT
I$1.HEX
R
GO
SAVE 1 $2.COM
```

### CARACTERE DE CONTROL CP/M

Tasta	Acțiunea
CTRL-C	abandonarea programului lansat în execuție și reîntoarcerea în CP/M
CTRL-E	impune un carriage return fizic, dar nu trimite comanda la sistem
CTRL-H	la fel ca BACKSPACE
CTRL-J	line-feed; încheie intrarea la consolă
CTRL-M	ca și RETURN
CTRL-P	trănite în ecou la imprimanta ce s-a transmis de la consolă; un al doilea CTRL-P îl anulează pe primul
CTRL-R	reîmpărește linia de comandă curentă, de obicei după apăsarea tastelor RUB sau DEL.
CTRL-S	întrerupe temporar listarea la consolă; un nou CTRL-S reia listarea
CTRL-U	anulează linia; afișează #; cursorul coboară cu o linie și așteaptă o nouă comandă
CTRL-X	șterge toată linia de comandă
CTRL-Z	separator de șiruri de date sau de câmpuri
BACKSPACE	mută cursorul înapoi cu un spațiu; șterge caracterul anterior
DEL	la fel ca RUB
RETURN	carriage return - încheie linia introdusă de la consolă
RUB	șterge caracterul din stînga cursorului; afișează caracterul șters; cursorul se mută la dreapta

### TIPURI DE FIȘIERE CP/M

CP/M identifică fiecare fișier printr-o specificație unică ce cuprinde: specificația de drive, numele fișierului și tipul fișierului. Specificația tipului fișierului este opțională și este formată din 3 caractere separate de numele fișierului printr-un punct. De obicei ea indică un tip special de fișier. Tipurile obișnuite de fișiere și sensurile lor sînt prezentate în continuare.

Tipul de fișier	Indicație
ASM	Fișier sursă în limbaj de asamblare. Asamblorul ASM assemblează sau traduce un fișier ".ASM" în limbaj mașină.
BAK	Fișier auxiliar creat de editorul de text; numele fișierului sursă urmat de acest tip de fișier, indică faptul că fișierul original a fost prelucrat. Fișierul original rămîne pe disc ca fișier auxiliar, deci se poate recurge la el.
BAS	Fișier sursă al programului CBASIC.
COM	Fișier executabil cu 8080.
HEX	Fișier în format hexazecimal.
INT	Fișier în limbaj intermediar pentru programul CBASIC.
IRL	Fișier REL indexat produs de LIB.
LIB	Este utilizat de MAC și RMAC pentru macrobiblioteci. Comanda R din ED citește fișierele de tip "LIB". Comanda X din ED scrie fișiere de tip LIB. Fișier ce poate fi tipărit la consolă sau imprimantă.
OVL	Fișier "overlay"; compilatorul PLI-80 extinde fișiere; se pot crea fișiere "overlay" cu LINK-80.
PLI	Program în limbaj sursă PLI-80.
PR	Fișier relocabil la nivel de pagină.
PRN	Fișier afișabil la consolă sau imprimantă.
REL	Fișier relocabil creat de RMAC și PLI-80 ce poate fi legat de LINK-80.
SUB	Tip de fișier necesar pentru fișierul SUBMIT conținînd una sau mai multe comenzi CP/M. Programul SUBMIT execută comenzi în fișiere de tip ".SUB", furnizînd pentru CP/M un mod de execuție pe loturi de date.
SYM	Fișier tabel de simboluri. MAC, RMAC și LINK-80 dau la ieșire fișiere de tip SYM. SID și ZSID citesc fișierele de tip SYM.
TEX	Fișier sursă pentru TEX-80.
XRF	Fișier creat de XREF conținînd lista referințelor încrucișate.
\$\$\$	Fișier temporar.

### MESAJE DE EROARE CP/M

?  
DDT. Acest mesaj apare cînd DDT nu înțelege instrucțiunea în limbaj de asamblare, cînd fișierul nu poate fi deschis, cînd există o eroare în suma de control într-un fișier "HEX" sau cînd programul încărcat s-a suprapus peste DDT.

#### ABORTED

PIP. A fost întreruptă o operație PIP prin apăsarea unei taste.

#### Mesaje de eroare ASM:

- D Eroare de date: elementele de date nu pot fi plasate în zona de date specificată.
- E Eroare de expresie: expresia nu poate fi evaluată în timpul asamblării.
- L Eroare de etichetare: eticheta nu poate apare în acest context (poate fi etichetă duplicat).
- N Neimplementat: facilitățile neimplementabile, cum ar fi "macro", sînt semnalate.
- O Depășire: expresia este prea complexă pentru a fi evaluată.
- P Eroare de fază: valoarea etichetei se schimbă la cele două treceri prin asamblare.
- R Eroare de registru: valoarea specificată ca registru este incompatibilă cu codul.
- S Eroare de sintaxă: forma improprie a expresiei.

U Eticheta nedefinită: eticheta utilizată nu există.

V Eroare de valoare: forma improprie a unui operand întâlnit într-o expresie.

#### **BAD DELIMITER**

STAT. Verifică linia de comandă din cauza scrierii eronate a datelor.

#### **Bad Load**

Mesaj de eroare CCP sau SAVE.

#### **Bdos Err On d:**

Eroare în sistemul principal de operare pe disc (Basic Disk Operating System - BDOS) pe drive-ul indicat. CP/M înlocuiește "d:" cu specificația drive-ului la care a apărut eroarea. Acest mesaj este urmat de una din cele 4 indicații ce vor fi descrise în continuare.

**Notă:** Sistemul principal de intrare/ieșire (Basic Input/Output System - BIOS) transmite codurile la BDOS. Aceste coduri stabilesc ce mesaje afișează BDOS. BIOS depinde de "hardware"-ul și tipul calculatorului. De aceea cauza specificată a acestor mesaje de eroare BDOS poate să varieze.

#### **Bdos Err On d: Bad Sector**

Acest mesaj apare când discul nu a fost încărcat sau este impropriu format, sau când drive-ul nu a fost conectat. Se verifică una dintre aceste situații și se încearcă din nou. Acest mesaj mai poate indica o defecțiune "hardware", un disc uzat sau impropriu și revenirea în CP/M sau tasta "CR" pentru ignorarea erorii.

#### **Bdos Err On d: File R/O**

S-a încercat ștergerea, redenumirea sau setarea atributelor când fișierul era Read-Only. Fișierul trebuia mai întâi pus pe Read-Write (R/W) cu comanda "STAT filespec \$R/W".

#### **Bdos Err On d: R/O**

Drive-ul specificat de "d:" a fost alocat pentru R/O cu o comandă STAT sau discul din drive a fost schimbat fără a fi fost inițializat cu CTRL-C. CP/M termină programul curent la apăsarea unei taste.

#### **Bdos Err On d: Select**

CP/M primește o comandă ce se referă la un disc inexistent. CP/M termină programul curent imediat după apăsarea unei taste. Se apasă tasta "CR" sau CTRL-C pentru revenire.

#### **Break "x" at c**

ED. "x" este unul dintre simbolurile descrise mai jos, iar "c" este comanda aflată în execuție când a apărut eroarea.

# Eroare de căutare. ED nu poate găsi șirul de date specificat într-una din comenzile F, S sau N.

? Comanda nu a fost recunoscută. ED nu recunoaște comanda indicată sau una din comenzile E, H, Q sau O nu este singura pe linie.

O Fișierul specificat într-o comandă R nu poate fi găsit.

> Buffer plin. ED nu mai poate introduce nici un caracter în bufferul de memorie sau șirul specificat într-o comandă F, N sau S este prea lung.

E Comanda întreruptă în timpul execuției. O apăsare a unei taste a întrerupt execuția comenzii.

F Disc sau "directory" plin. Această eroare este urmată de unul din mesajele "disk

full" sau "directory full". Se va recurge la procedurile de revenire, ce urmează după aceste mesaje.

#### **CANNOT CLOSE DESTINATION FILE - {filespec}**

PIP. Un fișier de ieșire nu poate fi închis. Se vor lua măsurile indicate după ce s-a verificat, dacă discul potrivit este introdus în drive și dacă nu este protejat la scriere.

#### **Cannot close, R/O**

#### **CANNOT CLOSE FILES**

CP/M nu poate scrie un fișier. De obicei aceasta se întâmplă când discul este protejat la scriere.

ASM. Un fișier de ieșire nu poate fi închis. Aceasta este o eroare fatală, care oprește execuția ASM. Se va verifica dacă discul este introdus în drive și dacă nu este protejat la scriere.

DDT. Fișierul scris pe disc cu o comandă W nu poate fi închis. Este o eroare fatală care oprește execuția DDT. Se va verifica dacă discul potrivit se află în drive și dacă nu este protejat la scriere.

SUBMIT. Această eroare se poate ivi în timpul prelucrării unui fișier "SUBMIT". Se va verifica dacă discul sistem corect este drive-ul A și dacă discul nu este protejat la scriere. Lucrul cu SUBMIT poate fi reluat după încărcarea sistemului.

#### **CANNOT READ**

PIP. PIP nu poate citi fișierul sursă specificat. "Reader" nu poate fi implementat.

#### **CANNOT WRITE**

PIP. Destinația specificată în comanda PIP este ilegală. Este probabil să fi fost specificat un dispozitiv de intrare drept destinație.

#### **Checksum error**

PIP. S-a întâlnit o eroare de sumă de control într-o înregistrare hexazecimală. Înregistrarea hexazecimală în care s-a produs eroarea trebuie corectată, posibil prin recrearea fișierului hexazecimal.

#### **CHECKSUM ERROR**

LOAD ADDRESS hhhh

ERROR ADDRESS hhhh

BYTES READ:

hhhh:

LOAD. Fișierul conține date incorecte. Se va recrea fișierul hexazecimal de la sursă.

#### **Command Buffer Overflow**

SUBMIT. Bufferul SUBMIT permite înscrierea a cel mult 2048 caractere în fișierul de intrare.

#### **Command too long**

SUBMIT. O comandă din fișierul SUBMIT nu poate depăși 125 de caractere.

#### **CORRECT ERROR, TYPE RETURN OR CTRL-Z**

PIP. A fost întâlnită o eroare de sumă de control într-o înregistrare hexazecimală, în timpul transferului unui fișier hexazecimal. Eroarea trebuie corectată, posibil prin recrearea fișierului hexazecimal.

**DESTINATION IS R/O, DELETE (Y/N)?**

PIP. Fișierul destinație specificat într-o comandă PIP există și este R/O. Dacă se apasă tasta "Y", fișierul destinație este șters înainte de copiere.

**Directory full**

ED. Nu există suficient loc în "directory" pentru ca fișierul să poată fi scris pe discul destinație. Se poate folosi comanda "OXfilespec" pentru a șterge orice fișiere inutile de pe disc, fără a ieși din editor.

SUBMIT. Se vor șterge unele fișiere sau se va folosi un disc nou și se va încerca din nou.

**Disk full**

ED. Nu există destul spațiu pe disc pentru fișierul de ieșire. Această eroare se poate ivi la comenzile W, E, H sau X. Dacă apare în comanda X, se poate repeta comanda scriind înaintea numelui fișierului un drive diferit.

**DISK READ ERROR - {filespec}**

PIP. Fișierul de intrare specificat într-o comandă PIP nu poate fi citit. Acesta este de obicei rezultatul apariției unui fișier de fișier neașteptat. Se va face corecția în fișierul propriu.

**DISK WRITE ERROR - {filespec}**

DDT. O operație de scriere pe disc nu poate fi realizată într-o comandă W, probabil datorită faptului că discul este plin. Se pot șterge unele fișiere inutile sau se poate utiliza un alt disc cu mai mult spațiu.

PIP. O operație de scriere pe disc nu poate fi realizată într-o comandă PIP, probabil datorită faptului că discul este plin. Se pot șterge unele fișiere inutile sau se poate utiliza un alt disc cu mai mult spațiu, dând din nou comanda PIP.

SUBMIT. Programul SUBMIT nu poate scrie pe disc fișierul \$\$\$SUB. Se vor șterge unele fișiere sau se va utiliza un disc nou încercându-se din nou comanda SUBMIT.

**ERROR: BAD PARAMETER**

PIP. A fost introdus un parametru ilegal într-o comandă PIP. Se va scrie din nou efectuând corecturile necesare.

**ERROR: CANNOT OPEN SOURCE, LOAD ADDRESS hhhh**

LOAD. Apare dacă LOAD nu poate găsi fișierul specificat sau dacă nu a fost precizat numele de fișier.

**ERROR: CANNOT CLOSE FILE, LOAD ADDRESS hhhh**

LOAD. Este cauzată de o eroare de cod dată de o apelare a unei funcții BDOS. Este posibil și ca discul să fi fost protejat la scriere.

**ERROR: CANNOT CLOSE FILE, LOAD ADDRESS hhhh**

LOAD. Nu poate fi găsit fișierul sursă. Se va verifica conținutul discului.

**ERROR: DISK READ, LOAD ADDRESS hhhh**

LOAD. Apare din cauza unei erori de cod date de o apelare a unei funcții BDOS.

**ERROR: DISK WRITE, LOAD ADDRESS hhhh**

LOAD. Discul destinație este plin.

**ERROR: INVERTED LOAD ADDRESS, LOAD ADDRESS hhhh**

LOAD. Adresa unei înregistrări este prea departe față de adresa înregistrării anterioare. Aceasta este o limitare internă a instrucțiunii LOAD, dar poate fi înlăturată. Se va utiliza DDT pentru citirea fișierului Hex în memorie, apoi se va utiliza comanda SAVE pentru a salva pe disc fișierul din memorie.

**ERROR: NO MOVE DIRECTORY SPACE, LOAD ADDRESS hhhh**

LOAD. "Directory" este plin.

**Error on line "nnn" message**

SUBMIT. Programul SUBMIT își afișează mesajele în formatul arătat mai sus, unde "nnn" reprezintă numărul liniei din fișierul SUBMIT. Eroarea se referă la mesajul ce urmează după numărul liniei.

**FILE ERROR**

ED. Discul sau "directory" sînt pline și ED nu mai poate scrie nimic pe disc. Este o eroare fatală, deci trebuie să existe suficient spațiu pe disc pentru o copie a fișierului înainte de a apela ED.

**FILE EXIST**

S-a încercat crearea sau redenumirea unui fișier utilizînd o specificație care fusese atribuită altui fișier. Se poate șterge fișierul existent sau se va utiliza altă specificație.

REN. Noul nume specificat este numele unui fișier care există deja. Nu se poate schimba numele unui fișier cu numele altui fișier existent. Dacă se dorește înlocuirea unui fișier existent cu o altă versiune a aceluiași fișier, se poate redenumi sau șterge fișierul existent sau se va utiliza PIP.

**File exist, erase it**

ED. Numele fișierului destinație există în momentul plasării fișierului destinație pe alt disc decît cel sursă. El trebuie șters sau trebuie ales alt disc pentru înregistrarea fișierului de ieșire.

**\*\* FILE IS READ-ONLY \*\***

ED. Fișierul specificat într-o comandă ED este "Read-Only". ED poate citi fișierul astfel încît acesta să poată fi examinat, dar nu poate modifica un fișier "Read-Only".

**File Not Found**

CP/M nu poate găsi fișierul specificat. Se va verifica dacă specificația drive-ului este corectă și dacă în drive a fost introdus discul potrivit.

ED. ED nu poate găsi fișierul specificat. Se va verifica dacă s-a dat specificația corectă a drive-ului și dacă în drive se află discul potrivit.

STAT. STAT nu poate găsi fișierul specificat. Mesajul de eroare ar mai putea să apară dacă s-a omis specificarea drive-ului. Se va verifica dacă în drive se află discul potrivit.

**FILE NOT FOUND - {filespec}**

PIP. Un fișier de intrare specificat nu există.

**Filename required**

ED. S-a dat comanda ED fără specificarea numelui fișierului. Se va da din nou comanda ED, urmată de numele fișierului care trebuie editat sau creat.

hhhh?? = dd

DDT. Semnele de întrebare semnifică faptul că DDT nu cunoaște reprezentarea valorii hexazecimale "dd" întâlnite la adresa "hhhh" în limbajul de asamblare 8080. "dd" nu este o instrucțiune-mașină 8080.

#### Insufficient memory

DDT. Nu există memorie suficientă pentru încărcarea fișierului specificat într-o comandă R sau E.

#### Invalid Assignment

STAT. S-a specificat un drive sau un atribut de fișier incorect sau a fost scris greșit numele unui dispozitiv. Acest mesaj de eroare poate fi urmat de o listă a atributelor de fișier valide care pot fi scrise după numele fișierului. Dacă a fost dat un atribut invalid de drive, este afișat mesajul "Use: d: = RO" care arată sintaxa corectă pentru atributele de drive.

#### Invalid control character

SUBMIT. Singurele caractere de control valide în fișierele SUBMIT de tipul SUB sînt ^A pînă la ^Z. De reținut că într-un fișier SUBMIT caracterele de control sînt reprezentate prin tasta "^" și nu prin tasta "CTRL".

#### INVALID DIGIT - {filespec}

PIP. S-a întâlnit o cifră hexazecimală incorectă în timpul citirii unui fișier hexazecimal. Fișierul hexazecimal în care se află cifra incorectă trebuie corectat prin recrearea lui.

#### Invalid Disk Assignment

STAT. Apare dacă specificația de drive este urmată de altceva decît de "= R/O".

#### INVALID DISK SELECT

CP/M a primit o linie de comandă specificînd un drive inexistent sau discul de drive este impropriu format. Programul curent este abandonat prin apăsarea oricărei taste.

#### INVALID DRIVE NAME (Use A, B, C sau D)

SYSGEN. SYSGEN recunoaște doar drive-urile A, B, C sau D ca destinații valide pentru generarea sistemului.

#### Invalid File Indicator

STAT. Eroarea apare dacă nu se specifică R/O, R/W, DIR sau SYS.

#### INVALID FORMAT

PIP. Formatul comenzii PIP este ilegal. A se vedea descrierea comenzii PIP.

#### INVALID HEX DIGIT

LOAD ADDRESS hhhh

ERROR ADDRESS hhhh

BYTES READ:

hhhh

LOAD. Fișierul conține cifre hexazecimale incorecte.

#### INVALID MEMORY SIZE

MOVCPM. Se va specifica o valoare mai mică decît 64KB sau decît capacitatea actuală a calculatorului cu care se lucrează.

#### INVALID SEPARATOR

PIP. A fost plasat un caracter incorect ca separator între două nume de fișiere de intrare.

#### INVALID USER NUMBER

PIP. A fost specificat un număr de utilizator mai mare decît 15. Numerele de utilizator sînt de la 0 la 15.

n?

USER. S-a specificat un număr mai mare ca 15 ca număr de utilizator. De exemplu, dacă s-a transmis USER 18, pe ecran va apare "18?".

#### NO DIRECTORY SPACE

ASM. "Directory" este plin. Se vor șterge unele fișiere pentru a face loc fișierelor PRN și HEX. "Directory" poate, de obicei, să conțină doar 64 nume de fișiere.

#### NO DIRECTORY - {filespec}

PIP. Nu este loc suficient în "directory" pentru fișierul de ieșire. Se vor șterge unele fișiere inutile sau se va lua un disc cu mai mult spațiu în "directory" și se va executa din nou PIP.

#### NO FILE - {filespec}

DIR, ERA, REN, PIP. CP/M nu poate găsi fișierul specificat sau nu există nici un fișier pe disc.

ASM. Fișierul sursă nu poate fi găsit pe drive-ul indicat.

DDT. Fișierul într-o comandă R sau E nu poate fi găsit pe disc.

#### NO INPUT FILE PRESENT ON DISK

DUMP. Fișierul cerut nu există.

#### No memory

Nu există destulă memorie disponibilă pentru încărcarea programului specificat.

#### NO SOURCE FILE ON DISK

SYSGEN. SYSGEN nu găsește CP/M în forma CPMxx.COM sau pe pistele sistem ale discului sursă.

#### NO SOURCE FILE PRESENT

ASM. Asamblorul nu găsește fișierul specificat. A fost tipărită greșit specificația fișierului în linia de comandă sau fișierul nu este de tip ASM.

#### NO SPACE -

SAVE. "Directory" sau discul sînt pline.

#### No SUB file present

SUBMIT. Pentru a opera corect cu SUBMIT trebuie creat un fișier de tipul SUB. Fișierul "SUB" conține de obicei comenzile CP/M. Se va utiliza cîte o singură comandă pe linie.

#### NOT A CHARACTER SOURCE

PIP. Sursa specificată în comanda PIP este ilegală. Probabil s-a specificat un dispozitiv de ieșire drept sursă.

**\*\* NOT DELETED \*\***

PIP. PIP n-a șters fișierul care se poate să fi fost R/O.

**NOT FOUND**

PIP. PIP nu găsește fișierul specificat.

**OUTPUT FILE WRITE ERROR**

ASM. S-a specificat o dischetă protejată la scriere ca destinație pentru fișierele PRN și HEX, sau pe dischetă nu se mai găsește spațiu liber. Se va corecta înaintea asamblării programului.

**Parameter error**

SUBMIT. În cadrul fișierelor SUBMIT de tip SUB parametri valizi sînt de la \$0 la \$9.

**PARAMETER ERROR, TYPE RETURN TO IGNORE**

SYSGEN. Dacă se apasă tasta "RETURN", SYSGEN continuă fără a prelucra parametrul invalid.

**QUIT NOT FOUND**

PIP. Șirul argument al parametrului Q nu a fost găsit în fișierul de intrare.

**Read error**

TYPE. A apărut o eroare în timpul citirii fișierului specificat în comanda TYPE. Se verifică discul și se încearcă din nou. Comanda "STAT filespec" poate depista cauza erorii.

**READER STOPPING**

PIP. Operația "reader" s-a întrerupt.

**Record Too Long**

PIP. PIP nu poate lucra cu o înregistrare mai lungă de 128 bytes.

**Requires CP/M 2.0 or later**

XSUB. XSUB necesită facilitățile versiunii CP/M 2.0 sau ale unei versiuni mai noi.

**Requires CP/M 2.0 or newer for operation**

PIP. Această versiune a comenzii PIP necesită facilitățile versiunii CP/M 2.0 sau ale uneia mai noi.

**START NOT FOUND**

PIP. Șirul argument al parametrului "S" nu este găsit în fișierul sursă.

**SOURCE FILE INCOMPLETE**

SYSGEN. SYSGEN nu poate utiliza actualul fișier CP/M sursă.

**SOURCE FILENAME ERROR**

ASM. Când se dorește asamblarea unui fișier nu se vor utiliza caracterele "\*" și "?" în numele fișierului. Cu o comandă ASM se poate asambla un singur fișier.

**SOURCE FILE READ ERROR**

ASM. Asamblorul nu înțelege informația din fișierul conținând programul în limbaj de asamblare. S-ar putea ca peste fișierul în limbaj de asamblare să fi fost scrise porțiuni

din alt fișier sau ca informația să nu fi fost bine salvată pe dischetă. Pentru localizarea erorii se va da comanda TYPE. Când pe ecran vor apare caracterele străine, se va ști că în acel loc al fișierului a apărut eroarea.

**"SYSTEM" FILE NOT ACCESSIBLE**

S-a încercat deschiderea unui fișier pus pe SYS cu comanda SET.

**\*\* TOO MANY FILES \*\***

STAT. Nu există suficientă memorie pentru ca STAT să clasifice fișierele specificate sau că s-au specificat mai mult de 512 fișiere.

**UNEXPECTED END OF HEX FILE - {filespec}**

PIP. A apărut sfârșitul fișierului înaintea înregistrării de sfârșit de fișier hexazecimal. Se poate face corecția, posibil prin recrearea fișierului Hex.

**Unrecognized Destination**

PIP. Se verifică linia de comandă pentru destinație validă.

**USE: STAT d: = R/O**

STAT. A fost dată o comandă STAT drive incorectă. Singurul atribut de drive valid în STAT este "STAT d: = R/O".

**VERIFY ERROR: - {filespec}**

PIP. În timpul copierii cu opțiunea V, PIP a găsit o neconcordanță între datele scrise și cele aflate în bufferul de memorie. De obicei aceasta indică un defect al discului destinație sau al drive-ului.

**WRONG CP/M VERSION (REQUIRES 2.0)****XSUB ACTIVE**

SUBMIT. A fost invocat XSUB.

**XSUB ALREADY PRESENT**

SUBMIT. XSUB este activ în memorie.

**Your Input?**

Dacă CP/M nu găsește comanda specificată, afișează numele comenzii introduse urmat de un semn de întrebare. Se va verifica dacă a fost introdus numele corect al comenzii și dacă comanda de care este nevoie există ca fișier de tip ".COM" pe discul cu care se lucrează sau pe discul specificat.