Befehlsliste

MDOS, BASIC

olivetti

Druck Nr. 1729/1/M40 ST/10.82

BASIC-Anweisungen

Das Handbuch dient der Information, sein Inhalt ist ohne ausdrückliche schriftliche Vereinbarung nicht Vertragsgegenstand. Technische Änderungen behalten wir uns vor. Die angegebenen Daten sind lediglich Nominalwerte.

EINLEITUNG

Die vorliegende Befehlsliste beschreibt die Systembefehle, Dienstprogramme, BASIC-Anweisungen und Standardfunktionen unter dem Betriebssystem

MDOS

Der Inhalt gilt ab Releasestufe 1.Ø aufwärts. Neben der Beschreibung und dem Format der einzelnen Sprachelemente werden im Anhang die Fehlercodes erläutert und einige für die Programmierung nützliche Hinweise in Form von Tabellen dargestellt.

Für Anregungen, die der Verbesserung oder Korrektur der vorliegenden Befehlsliste dienen, sind wir Ihnen dankbar.

INHALT

I SYSTEMBEFEHLE

- Befehle im Command - Mode
- Befehle im Calculator- und Debugging - Mode
- Befehle innerhalb von Prozeduren

II DIENSTPROGRAMME

III BASIC - ANWEISUNGEN

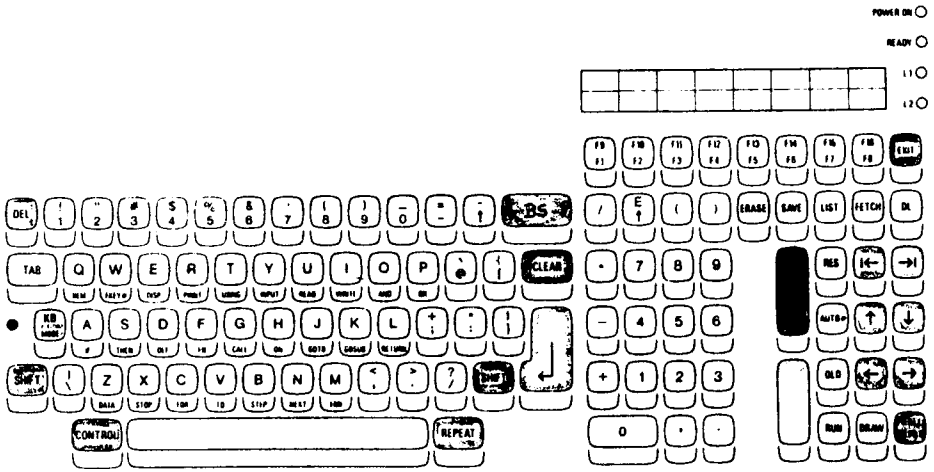
- Allgemeine Anweisungen
- Standardfunktionen

IV SPEZIELLE BASIC - ANWEISUNGEN

- Anweisungen zur Steuerung von Peripherie
- Anweisungen für graphische Darstellungen (PLO, GDI)
- Anweisungen für Matrizenrechnen

V ANHANG

- Meldungen des Betriebssystems
- Liste der Fehlercodes
- ISO - Code- Tabelle
- Tabelle der Options
- Speicherbedarf von Daten
- Tabelle für Bildschirmsteuerung



TASTATUR–ANORDNUNG

FUNKTIONSBESCHREIBUNG DER STEUERTASTEN

DEL DEL	Erzeugt das ISO-DEL-Zeichen
KB MODE KB MODE	Erlaubt es, die Tastatur wie eine Schreibmaschine zu benutzen: Kleinbuchstaben durch bloßes Drücken der entsprechenden Taste, Großbuchstaben durch gleichzeitiges Drücken der SHIFT-Taste. Die BASIC-Ganzworte können im KB-Mode nicht verwendet werden
SHIFT SHIFT	Obere Bedeutung, Großbuchstaben bzw. BASIC-Ganzworte bei doppelt belegten Tasten
CONTROL CONTROL	Erzeugt zusammen mit einer entsprechenden Taste ein Kontrollzeichen
BS BS	Löscht ein Zeichen links vom Cursor
CLEAR CLEAR	Löscht eine Fehlermeldung Zusammen mit SHIFT wird auch die Eingabezeile gelöscht und die automatische Zeilennummerierung abgebrochen
	Beendet eine Eingabe. Wird auch als CR (Carriage Return)-Taste oder als EOL (End of Line)-Taste bezeichnet
REPEAT REPEAT	Wiederholfunktion für andere Tasten
	Weist das Ergebnis einer Calculator-Mode-Berechnung einem Calculator--Mode-Register zu

	Hat die gleiche Funktion wie die CR-Taste (Eingabeabschluß)
EXIT	Unterbricht das laufende Programm, das System wechselt in den Debugging-Mode. Jedes weitere Drücken bewirkt die Ausführung genau einer Programmzeile (Step-by-Step) SHIFT EXIT Programmfortsetzung (Continue) CONTROL EXIT Abbruch eines Programms (Break)
	Cursor an den Zeilenanfang
	Cursor an das Zeilenende
	Die folgende Zeile erscheint als Eingabezeile
	Die vorhergehende Zeile erscheint als Eingabezeile
	Cursor um eine Position nach links
	Cursor um eine Position nach rechts
PR ALL NO PRT	Der Ausdruck auf dem Systemdrucker wird unterdrückt Aufhebung durch nochmaliges Drücken Zusammen mit SHIFT: Alle Meldungen in der Systemzeile und alle eingegebenen Zeichen werden auf dem Systemdrucker gedruckt. Aufhebung durch nochmaliges Drücken

ZEICHENERKLÄRUNG

{ }	Einer der angeführten Parameter muß verwendet werden
[ ]	Die angegebenen Parameter können verwendet werden oder fehlen
S	Unterstrichene Parameter werden vom System eingefügt, falls keiner der Parameter angegeben wird
.....	Der in der letzten Klammer angeführte Parameter kann beliebig oft wiederholt werden
NAME	In diesem Schrifttyp angeführte Wörter und Buchstaben sowie alle Sonderzeichen wie: (), ; * + - müssen in dieser Form eingegeben werden
Name	Alle in dieser Schrift geschriebenen Wörter sind in den Formaten Platzhalter und werden in der Anwendung der Befehle entsprechend ersetzt

ERKLÄRUNG DER VERWENDETEN BEGRIFFE

ERKLÄRUNG DER VERWENDETEN BEGRIFFE

AKTUELLE

- LÄNGE EINES STRINGS — Länge eines Strings (einer Stringvariablen, eines Stringausdruckes) nach dem zuletzt ausgeführten Statement (i.a. ist die aktuelle Länge kleiner als die deklarierte Länge).
- DIMENSION EINER MATRIX — Dimension einer Matrix nach dem zuletzt ausgeführten MAT... Statement des Types MATREAD (m,n) etc.
- FILEGRÖSSE — Der Platz in einem Datenfile, der von Daten belegt ist

ARGUMENT — einzusetzende Größe in einer Funktion

BIBLIOTHEK — Logische Zusammenfassung von Files unter einem Namen durch Platzzuweisung auf einer Disk. Das Inhaltsverzeichnis einer Bibliothek (Katalog) kann in maximal drei Teilbibliotheken unterteilt werden

DELIMITER — Positive ganze Zahl zwischen 0 und 255. Das zugehörige ISO-Zeichen wird als Trennzeichen verwendet (siehe BUILD, ASSIGN...)

DISK — Der Begriff "Disk" steht allgemein für externe Datenträger (z.B. Floppy-Disk, Festplatte)

FELD — Einfach- oder doppelt indizierte numerische oder alphanumerische Variable
- num. Feld: Feld, dessen Elemente numerische Variable sind (Vektor, Matrix)
- alphanum. Feld: Feld, dessen Elemente Stringvariable sind

FILEDESIGNATOR — Positive ganze Zahl, die in einem Programm einem File als Identifikationsnummer zugeordnet wird.

FILENAME — Name eines Files mit max. 6 Zeichen. Beginnt mit einem Großbuchstaben und enthält kein Sonderzeichen. Gehört das File zur Package- bzw. Common-Teilbibliothek, ist das Zeichen "*" bzw. "+" Bestandteil des Namens.

IMAGEFELD — "Druckmaske", folgende Zeichen sind möglich:

#	Zeichen für Ziffer (Für das Vorzeichen ist ein zusätzliches Ziffernzeichen zu reservieren)	
.	Dezimalpunkt	
$	Dollarzeichen	
↑↑↑↑	Exponentenfeld (immer 4 Zeichen)	
'LLL..L	linksbündige	alphanumerische Ausgabe
'RRR..R	rechtsbündige	
'CCC..C	zentrierte	

Alle anderen Zeichen, die im IMAGEFELD verwendet werden, werden unverändert übernommen.

INDIZIERTE VARIABLE — Variable der Form A (I), B (I,J), C$ (I), D$ (I,J), I,J: numerische Ausdrücke.

KONSTANTE

- NUM. KONSTANTE — reelle Zahl
- STRINGKONSTANTE — in Anführungszeichen eingeschlossene Folge von ISO-Zeichen

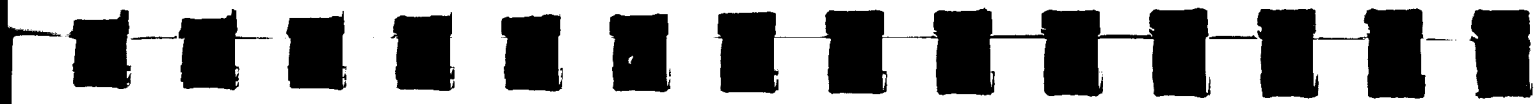
LAUFVARIABLE — numerische, nicht indizierte Variable

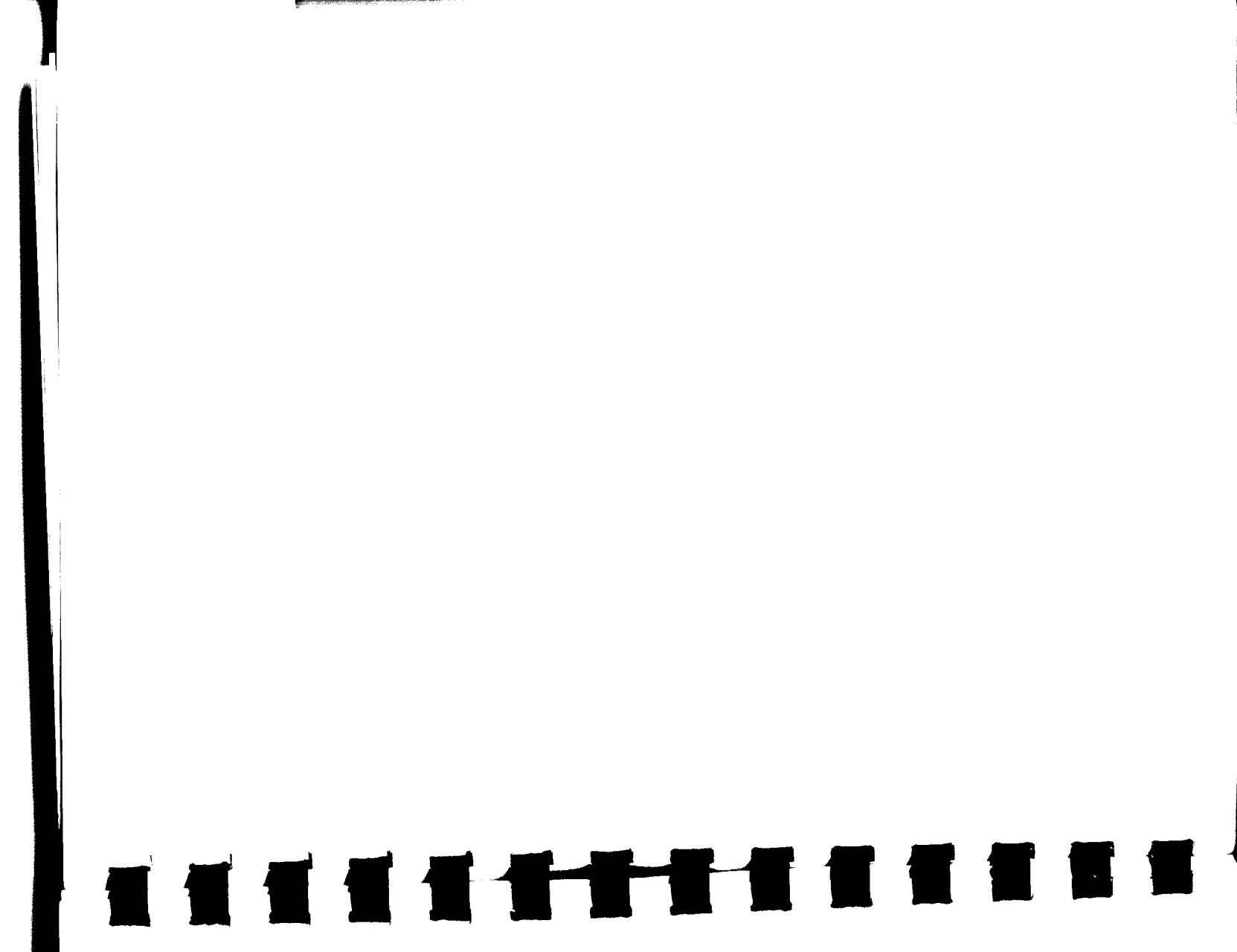
LIB.-NAME	Name einer Bibliothek mit max. 6 alphanumerischen Zeichen. Beginnt mit einem Großbuchstaben und enthält keine Kleinbuchstaben und Sonderzeichen (LIB.- ≙ LIBRARY)
LIB.-REF.	Identifikation einer Bibliothek durch Angabe von Bibliotheksname und den Namen der Einheit, die die Bibliothek enthält (LIB.-REF. ≙ LIBRARY REFERENCE) Lib.-Ref. kann folgende Formate annehmen: (Lib.-Name, Unit-Name) Lib.-Name (, Unit-Name)
MATRIX	Zweifach indizierte Variable
NUMERISCHER AUSDRUCK	Ein aus numerischen Konstanten oder Variablen und den arithmetischen Operationen sowie Klammern, Standardfunktionen oder vom Benutzer definierten Funktionen gebildeter Ausdruck, der als Ergebnis einen numerischen Wert liefert. Erfordert die Ausführung eines Statements einen ganzzahligen Wert (z.B. TAB (X), A(U)...), wird immer gerundet.
PASSWORD	Beliebige Folge von max. 6 Zeichen außer Blank und Komma zum Schutze von Bibliotheken
STRING	Folge von Zeichen
STRINGAUSDRUCK	Ausdruck, der aus Stringkonstanten, Stringvariablen und Standard- oder definierten Funktionen, die als Ergebnis einen String liefern, bestehen kann. Die einzige erlaubte Operation ist die String-addition (+)

STRINGVARIABLE	Variable, in der eine Folge von Zeichen gespeichert werden kann. Falls nicht anders definiert, beträgt die Länge 16 Zeichen (max. 1023 Zeichen)
TEILBIBLIOTHEK	Teil eines Inhaltsverzeichnisses einer Bibliothek mit folgenden Bezeichnungen: Package-, Common- und User-Teilbibliothek. Die Teilbibliotheken unterscheiden sich durch unterschiedliche Schutzmöglichkeiten
UNIT-NAME	Logischer Name einer Diskstation (Einheit)
VEKTOR	Einfach indizierte Variable
VOLUME LABEL	Kennzeichnung einer Disk durch max. 6 alphanumerische Zeichen. Beginnt mit einem Großbuchstaben und enthält keine Kleinbuchstaben und Sonderzeichen

Bemerkungen:

Beim Eingeben von Systembefehlen kann es aufgrund des Befehlsformates durch das Weglassen einzelner Parameter vorkommen, daß als letztes Zeichen ein Komma steht. Dieses Komma (eventuell auch mehrere) muß dann entfallen.





SYSTEMBEFEHLE

Befehle im Command-Mode

Name	Funktion	Format	Anmerkung
AUTO	Ermöglicht bei der Programmerstellung die automatische Erzeugung der Zeilennummer entw im Standardformat oder mit Anfangsnummer und Schrittweite	AUT [0#] [Zeilennummer] [,Schrittweite] AUTO# 10,5 AUT ,15	Standardformat: Anfangswert: 10 Schrittweite: 10
CALCULATOR	Einschalten des Calculator Modes	CAL [CULATOR] [n / FL]	n = o, ... 13 Nachkommastellen FL Gleitkommastellen
CATALOG	Listet das Inhaltsverzeichnis der durch "lib-ref" angegebenen Bibliothek	CAT [ALOG] [* / . / : / filename] ,[lib-ref], [P / T / D / O] [,F] CAT :,(,F1),,F	F: Full Information P: Programmfiles T: Textfiles D: Datenfiles O: Objektfiles
COMPILE	Umwandlung eines Textes im Arbeitsspeicher in ein BASIC Programm	COM [PILE]	

Name	Funktion	Format	Anmerkung
CONFIGURE	Erlaubt die Festlegung einer bestimmten Systemkonfiguration	CON [FIGURE] [,MS=n] [,PRO] [,SP= {0 ⋮ 9} F {7 8} {Y N} {E O} {1 2}] CON MS=48,SP=~~0F7YE2~~ CON PRO . 6F7YE2 (J. SCHRÖDER PR 1070)	MS: Größe des Anwenderspeichers $16 \leq n \leq 64$ n=phys. Größe PRO: Autostart der Procedur *SETUP SP: Übertr.param. für den Drucker
CREATE	Anlegen eines externen Datenfiles in einer Bibliothek	CRE [ATE] filename , [lib-ref] ,[S R Z] [,n] CRE HAUS,(BAU,F2) CRE *ABC,(,F2),Z,10000	n: Länge des Files in Bytes. (Defaultwert N = 4096 Bytes S = sequentielles File R = Randomfile Z = Randomfile (einfach genaue Nullen)
DATE	Speicherung des Datums auf System-Disk	DAT [E] Datum	Datum: 6 bel. Zeichen

Name	Funktion	Format	Anmerkung
DCHANGE	Erlaubt den Wechsel einer Disk (ohne den Arbeitsspeicher zu löschen)	DCH [ANGE] unit name DCH F1 DCH F2	Nach dem Wechsel **SHIFT EXIT** drücken
DECOMPILE	Umwandlung eines BASIC-Programmes im Arbeitsspeicher in einen Text	DEC [OMPILE]	
DELETE LINE	Löschen einer oder mehrerer Zeilen im Arbeitsspeicher	DEL [ETE LINE] [Zeilennr.] [, Zeilennr.] DEL 40,200 DEL 80	Falls keine Zeilennummer angegeben wird, wird die laufende Zeile gelöscht
DRAW	HARDCOPY des Bildschirminhalts auf Systemdrucker	DRA [W] [A]	Auch im Debugging-Mode möglich A = Alphaschirm **DRAW** ohne Operand **A** nur bei **OPT GDI** möglich

Name	Funktion	Format	Anmerkung
ENVIRONMENT	Beschreibung der aktuellen Systemkonfiguration	ENV [IRONMENT]	
ERASE	Löschen des Bildschirminhalts	ERA [SE] [G / A] ERA	Auch im Debugging-Mode möglich A = Alphaschirm G = Grafikschirm
EXEC	Aufruf eines im System enthaltenen Dienstprogrammes	EXE [C] Dienstprogramm, [Parameter] ... EXE DCO,F2,F1	Parameter: siehe Beschreibung der Dienstprogramme
FETCH	Eine Zeile des Files im Arbeitsspeicher erscheint in der Systemzeile (für Korrekturen)	FET [CH] [Zeilennummer] FET 80	Falls keine Zeilennummer angegeben wird, erscheint die laufende Zeile in der Systemzeile

Name	Funktion	Format	Anmerkung
LBCLOSE	Schließen von Bibliotheken	LBC [LOSE] {lib-ref / *} LBC (STAT,F1) LBC *	*: Es werden alle offenen Bibliotheken geschlossen
LBOPEN	Öffnen einer Bibliothek	LBO [PEN] lib-ref [,Password] LBO (WERT,F1),AY1508 LBO ERFASS	Ist die Bibliothek mit einem Password versehen, so muß dieses angegeben werden
LBRESTORE	Die momentan offenen Bibliotheken werden geschlossen und an deren Stelle die als Standard (LBSTORE) definierten geöffnet	LBR [ESTORE] LBR	
LBSTORE	Bewirkt das automatische Öffnen der aktuell offenen Bibliotheken bei jedem Laden des Systems und nach LBRESTORE	LBS [TORE] LBS	

Name	Funktion	Format	Anmerkung
LDIMAGE	Ausgabe eines gespeicherten Bildes auf den Bildschirm	LDI [MAGE] [Filename] [,lib-ref.] LDI BILD1,(,F1)	Nur bei Option GDI (auch im Debugging-Mode möglich)
LDKEYS	Den Funktionstasten wird der gespeicherte Standardinhalt zugewiesen (siehe STK)	LDK [EYS] LDK	
LINK	In das Programm im Arbeitsspeicher wird ein als Text abgespeicherter Programmteil oder eine Funktion eingefügt	LIN [K] Filename, [lib-ref.] Zeilennummer [,α] LIN FILE1,(STAT,F2),1000,N LIN FILE3,,250	α: A,...,Z
LIST	Listen eines Programmes oder Textes	LIS [T] [Zeilennummer] [,{Zeilennummer / Zeilennummer ,X}] LIST LIS 50,350,X	X: Bei Textfiles erfolgt der Ausdruck ohne Zeilennummern

Name	Funktion	Format	Anmerkung
LVTOC	Druck eines Verzeichnisses der Bibliotheken einer Einheit oder aller offenen Bibliotheken	LVT [OC] [{Unit Name / *}] LVT F1 LVT*	* Alle offenen Bibliotheken
MERGE	Kombination eines Textes im Arbeitsspeicher mit einem Textfile einer Bibliothek	MER [GE] Filename, [lib-ref.] , [Zeilennummer] , [/] MER TEXT1,,5000,/ MER TEXT2,(.F1)	Wird der Parameter "/" angegeben, so werden ersetzte Zeilen nicht ausgedruckt
MODIFY	Ändert einen Filenamen und/oder die reservierte Länge von Datenfiles	MOD [IFY] Filename, [lib-ref.] ,[Filename$_2$] [,n] MOD FILE4,,FILES MOD DAT1,(STAT,F1),DATA,2048	
NEW	Löscht den Arbeitsspeicher und ermöglicht die Eingabe eines neuen Programmes	NEW NEW	

Name	Funktion	Format	Anmerkung
OLD	Lädt ein in einer Bibliothek gespeichertes Programm- oder Textfile in den Arbeitsspeicher	OLD Filename [,lib-ref.] OLD FRITZ OLD EINGAB,(,F1)	
OPTION	Vorwahlbefehl für die Aktivierung bestimmter Teile des Betriebssystems und Neuladen des Systems	OPT [IONS] [{PLO / GDI}] OPT GDI OPT,	Option: PLO:Plot GDI:Graphic Display Die Options belegen Platz im Anwenderspeicher
PREPARE	Kontrolle eines Programms und Aufbereitung für die Ausführung (Preexecution) sowie Ausgabe des noch freien Platzes im Arbeitsspeicher in worten (falls die 'PRINT ALL'-Taste aktiviert ist).	PRE [PARE] [Filename [,lib-ref.]] PRE *PRINT,(DRUCK,F2) PRE	Wird nach PRE (ohne Parameter) SAVE oder REPLACE ausgeführt, entfällt bei weiteren Aufrufen die Preexecution

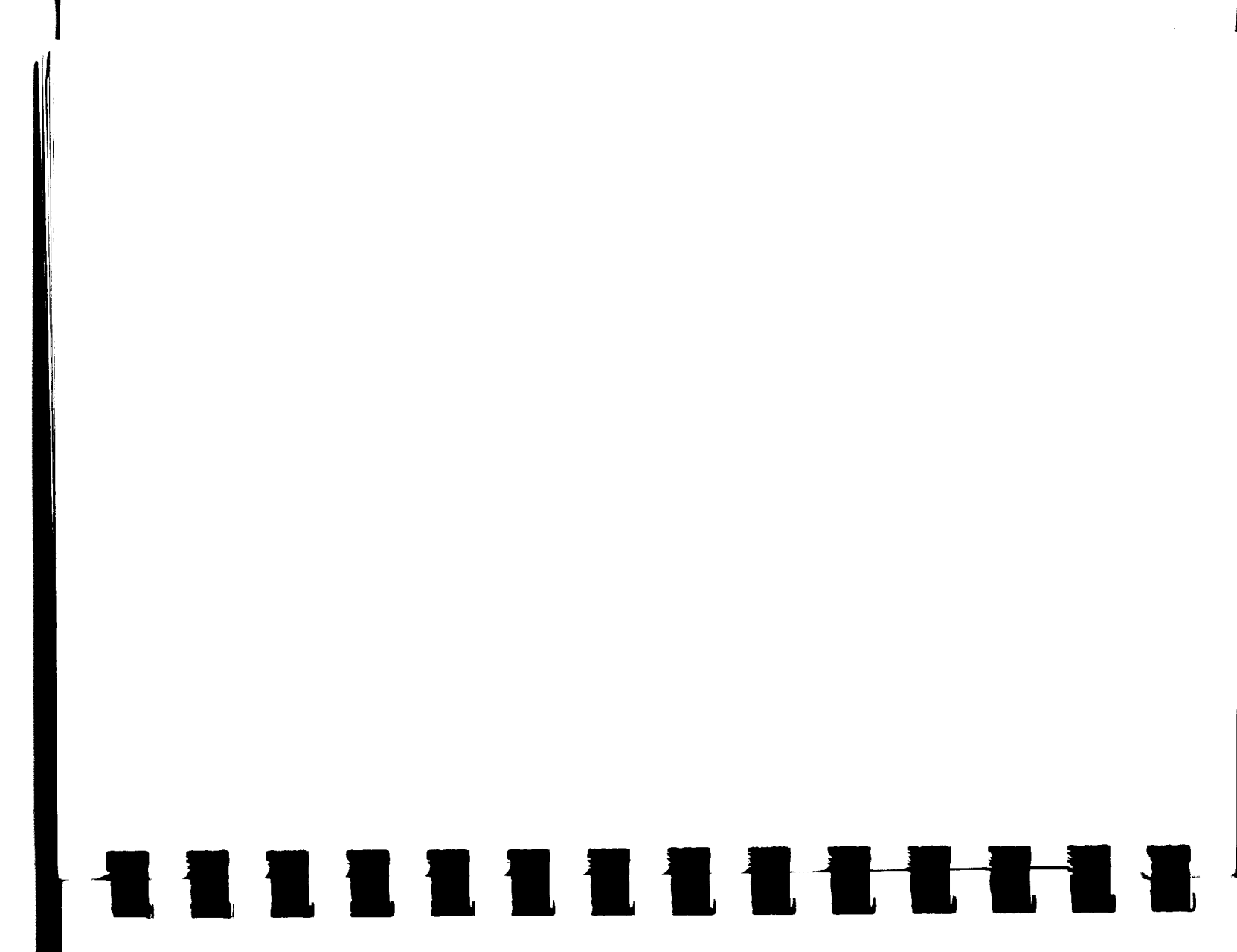
Name	Funktion	Format	Anmerkung
PROCEDURE	Ausführung einer katalogisierten Prozedur	PRO [CEDURE] Filename , [lib-ref.] , [A] , [Zeilennummer 1] [,Zeilennummer 2] Bsp.: PRO TASK1,(NEU,F1) PRO JOB3,,A,,240	A: siehe auch Automatische Input-Übernahme
PURGE	Löschen eines Files in einer Bibliothek	PUR [GE] Filename [,lib-ref.] PUR FILE1,(,F2) PUR +ALT	
REPLACE	Speichert ein von der Disk geladenes Programm- oder Text-File (nach Änderung) wieder an seinen Platz zurück	REP [LACE] [lib-ref.] REP ,LIBR1 REP	

Name	Funktion	Format	Anmerkung
RESEQUENCE	Umnumerierung eines Programmes oder Textes im Arbeitsspeicher	RES [EQUENCE] [Zeilennr.] [,Schrittweite] RES ,20 RES 100,30	Default: 10,10 Schrittweite:positive ganze Zahl
REVERSE	Umkehrung des Bildes auf dem Bildschirm von positiv auf negativ und umgekehrt	REV [ERSE] REV	Auch im Debugging-Mode möglich
RUN	Startet die Ausführung eines Programmes	RUN [Filename [,lib-ref] / Zeilennummer] RUN SERV,(,F2) RUN 380	
SAVE	Abspeicherung eines Programmes oder Textes vom Arbeitsspeicher in eine Bibliothek	SAV [E] Filename, [lib-ref,] [,MSG=n] [,COM=m] SAV ALG1,(MATH,F2),COM=104 SAV PROG,,MSG=0	n = 0 oder 1 m = Anzahl Worte für Common-Bereich

Name	Funktion	Format	Anmerkung
SECURE	Verhindert Listing und Editing von Programmen und das Überschreiben von Datenfiles	SEC [URE] Filename, [lib-ref.] [,n] SEC PROG2,,100	n: Zeilennummer des Programmes, ab der es geschützt werden soll
SHIFT	Erhöhen der Zeilennummern eines Programmes oder Textfiles um einen konstanten Wert ab der definierten Zeile	SHI [FT] Zeilennummer, Konstante SHI 1410,100	
SPACE	In der Systemzeile wird der in einer Bibliothek verfügbare Speicherplatz in Bytes angegeben	SPA [CE] [lib-ref.] SPA (HAUS,F1) SPA (,F2)	
STIMAGE	Speicherung des Bildschirminhaltes des Grafikschirmes in ein sequentielles Datenfile	STI [MAGE] [Filename] [,lib-ref.] [,n] STI BILD1,(,F1)	Nur bei Option GDI (auch im Debugging-Mode möglich) n: Buffergröße in K

Name	Funktion	Format	Anmerkung
STKEYS	Speichert die aktuelle Belegung der Funktionstasten als Standardbelegung	STK [EYS] STK	
TEXT	Löscht den Arbeitsspeicher und ermöglicht das Eingeben eines beliebigen Textes	TEX [T] TEX	Der Text muß Zeilennummern haben
TRANSCODE	Ein Text wird in ein Datenfile umgewandelt oder ein Datenfile in einen Text	TRA [NSCODE] {T / D} ,Filename, [lib-ref.] [,#] TRA T,DAT1,BIBL.* TRA D,FILE1	T,D: Zielfile der Umwandlung Wird # angegeben, werden die Zeilennummern mitgespeichert bzw. dem Datenfile entnommen
TRC	Bei der Ausführung eines Programmes werden die Zeilennummern der jeweils ausgeführten Statements angezeigt	TRC ON / OF	

Name	Funktion	Format	Anmerkung
TRUNCATE	Bei sequentiellen Datenfiles wird die reservierte Länge auf die aktuelle Länge des Files reduziert	TRU [NCATE] Filename [,lib-ref.] TRU DAT2,C,F1)	
VALIDATE	Schließt nach nicht regulärem Programmende offen gebliebene Datenfiles	VAL [IDATE] Filename [,lib-ref.] VAL EINGAB VAL HILFE,(NEU,F2)	Die Verfügbarkeit der zuletzt in das File gespeicherten Daten ist nicht immer gewährleistet

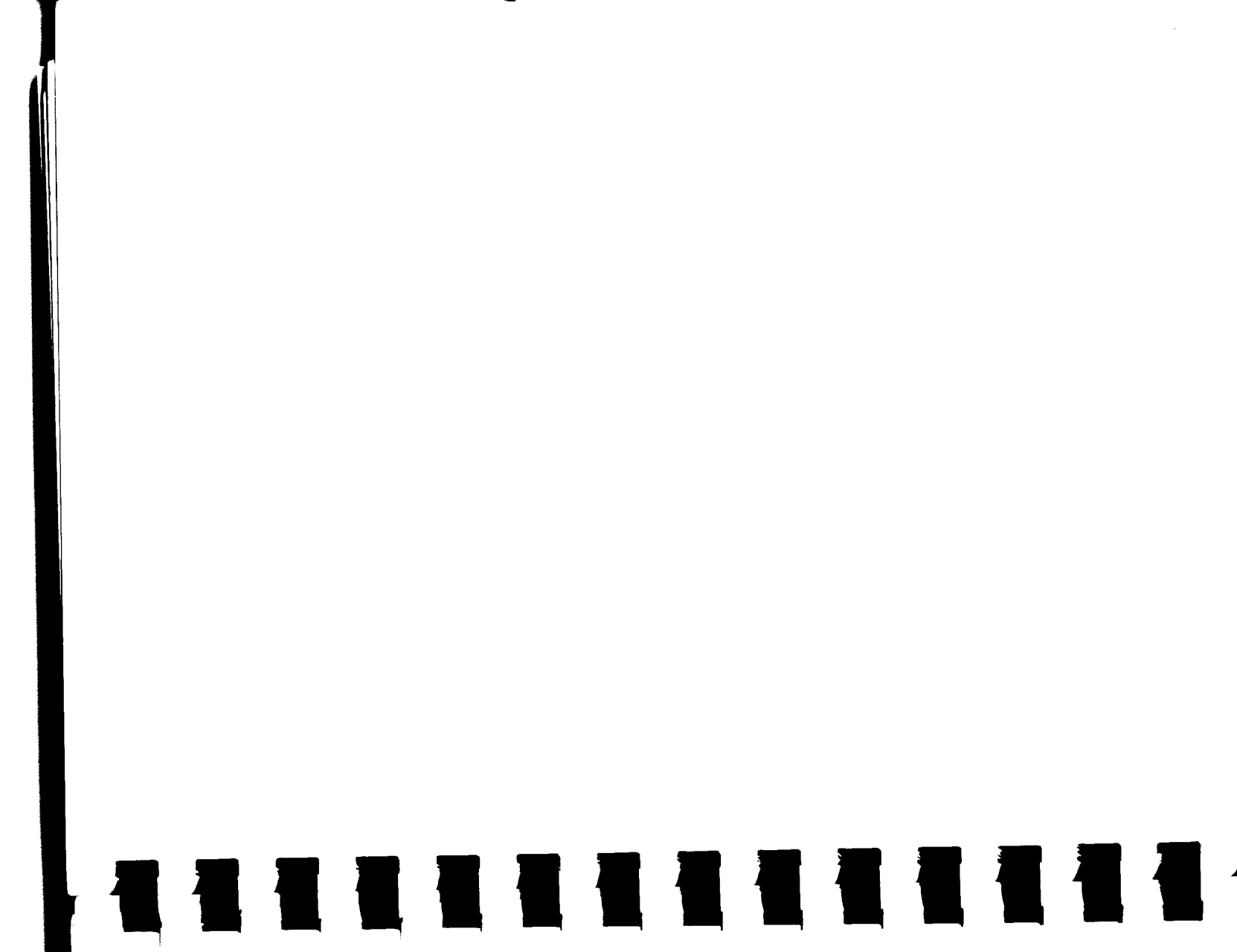


Befehle im Calculator- und Debugging-Mode

Name	Funktion	Format	Anmerkung
FKEY	Belegung einer Funktionstaste mit einer Stringkonstanten	FKEY# n, Stringkonstante [:] FKEY#4,CAT :,C,F1),.F.	1≤n≤16 Der Doppelpunkt übernimmt die Funktion der EOL-Taste Löschen der Tastenbelegung: FKEY#n,
SDEG	Vorwahl der Maßeinheit für Winkeldarstellung in Altgrad	SDE [G]	
SGRAD	Vorwahl der Maßeinheit für Winkeldarstellung in Neugrad	SGR [AD]	
SRAD	Vorwahl des Bogenmaßes zur Winkeldarstellung	SRA [D]	
START	Fortsetzung der Prgramm-ausführung ab der angegebenen Zeile	STA [RT] Zeilennummer	Nur im Debugging-Mode möglich
STOP	Programmstop bei der angegebenen Zeilennummer	STO [P] Zeilennummer	Nur im Debugging-Mode möglich

Befehle innerhalb von Prozeduren

Name	Funktion	Format	Anmerkung
IF	Ermöglicht die bedingte Verzweigung in einer Prozedur	IF CC=n, Zeilennummer IF CC=3.150	$0 \leq n \leq 9$
IN	Festlegung der Inputzeilen für automatischen Input	IN = {Num. Konstante / Stringkonstante} [/ {Num. Konstante / Stringkonstante} ...] IN= 3.45,"EINGABE",2	Für die Ausführung muß im Aufruf der Prozedur der Parameter A angegeben werden
PRO	Wiederaufnahme einer Prozedur nach einem Fehler	PRO	Muß über Tastatur eingegeben werden
TEST	Ermöglicht eine unterschiedliche Fehlerbehandlung in der Ausführung von Prozeduren	TEST {ON / OFF / #nnn}	On: Unterdrückung von Fehlern OFF: Ende der Fehlerunterdrückung # : Meldung des Fehler mit Code n n n



DIENSTPROGRAMME

Name	Funktion	Format	Anmerkung
DCOPY	Kopieren von Disks	EXE [C] DCO [PY] ,Unit-Name1, Unit-Name2 [,[Volume Label] [,V]] EXE DCO,F1,F2 EXE DCO,F2,F1,,V	Unit-Name 1:Original Unit-Name 2:Kopie Volume Label:bewirkt Überprüfung V:bewirkt das Mitkopieren des Volume-Labels
DINIT	Initialisieren einer Disk	EXE [C] DIN [IT] ,Unit-Name [, [Volume Label] [,{P / L}]] EXE DIN,F2 EXE DIN,F1,TEST	P: physisch L: logisch
FLCOPY	Kopieren eines Files von einer (Teil-)Bibliothek in eine andere (oder die gleiche) Bibliothek	EXE [C] FLC [OPY] ,IN= [lib-ref,] ,[Password 1] , Filename1, OUT= [lib-ref,] ,[Password2] [,{Filename2 / * / .}] EXE FLC,IN=(,F2),B88,FILE2,OUT=(DATA,F2),,*	Bei Bibliotheken die mit einem Password versehen sind, ist dieses anzugeben
FLPRINT	Ausgabe des Inhalts eines Datenfiles	EXE [C] FLP [RINT] ,Filename [,lib-ref] EXE FLP,DAT3	

Name	Funktion	Format	Anmerkung
LBCREATE	Anlegen einer Bibliothek auf einer Disk	EXE [C] LBC [REATE] ,lib-ref, [Password] [,SIZE=Z] [,*=n_1] [,+=n_2] [,NP=n_3] EXE LBC,(DATEN,F2),,SIZE=100,*=3 EXE LBC,(TEST,F1),PASS,SIZE=400	Z:Größe der Bibliothek in K (1024 Bytes) n_i (i=1,2,3): Anzahl Sektoren, die für den Katalog angelegt werden
LBEMPTY	Löschen aller Files einer Bibliothek	EXE [C] LBE [MPTY] ,lib-ref. [,Password] EXE LBE,NEU EXE LBE,(ALT,F2),PAS1	
LBPROTECT	Schützen von Teil-Bibliotheken vor Anwendung bestimmter Systembefehle	EXE [C] LBP [ROTECT] , [lib-ref.] , [Password] [*,*] EXE LBP,(DATA,F2),,*	

Name	Funktion	Format	Anmerkung
LBRENAME	Änderung des Namens der Bibliothek und/ oder des Passwords	EXE [C] LBR [ENAME] ,lib-ref. , [altes Password] , { neuer Bibliotheksname [,neues Password] / , neues Password } EXE LBR,(DAT,F2),,DAT2	
LBSCRATCH	Löschen einer Bibliothek	EXE [C] LBS [CRATCH] ,lib-ref. [,Password] EXE LBS,(M40,F1),ST EXE LBS,(HELP,F2)	Befinden sich auf einer Disk mehr als 2 Bibliotheken, kann der freigewordene Platz erst nach RESTRUCT wieder verwendet werden
LIBCOPY	Kopieren einer (Teil-) Bibliothek in eine andere Bibliothek	EXE [C] LIB [COPY] IN=[lib-ref.], [Password1], [*/:], [,Filename] OUT=[lib-ref.], [Password2] EXE LIB,IN=(,F1),,,OUT=(DAT,F2),,PROG EXE LIB,IN=BIBL,KR,:,OUT=(TEST,F1)	Bei Bibliotheken, die mit einem Password versehen sind, ist dieses anzugeben

Name	Funktion	Format	Anmerkung
RESTRUCT	Neuorganisation einer Disk nach dem Löschen einer oder mehrerer Bibliotheken	EXE [C] RES [TRUCT] [,Unit-Name] EXE RES EXE RES,F2	
VOLLABEL	Kennzeichnung einer Disk mit einem Volume Label	EXE [C] VOL [LABEL] ,Unit-Name , Volume Label EXE VOL,F1,ARB	

BASIC - ANWEISUNGEN

Allgemeine Anweisungen

Name	Funktion	Format	Anmerkung
APPEND:	Erlaubt das Anfügen von Daten an ein bestehendes sequentielles Datenfile	APPEND : Filedesignator 50 APPEND:3 20 APPEND:A+B*C	Filedesignator: num. Ausdruck
ASSIGN	Die durch den Delimiter begrenzten Teilstrings werden auf eine Liste von Variablen aufgeteilt	ASSIGN Stringausdruck, {num. Var. / Stringausdr.} [, {num. Var. / Stringvar.}]...;Delimiter 30 ASSIGN A$,A,B,C;42	Delimiter:0 ≤ d ≤ 255 Trennzeichen numerisch; aus ISO-Code-Tabelle
BASSIGN	Wirkung wie ASSIGN jedoch erfolgt die Zuweisung in internem Format	BASSIGN Stringausdruck , {num. Var. / Stringvar.} [, {num. Var. / Stringvar.}]... 50 BASSIGN B1$,E,F,C$	

Name	Funktion	Format	Anmerkung
BBUILD	Die Werte einer Liste von Ausdrücken werden in internem Format einer Stringvariablen zugewiesen (Umkehrung von BASSIGN)	BBUILD Stringvariable, {num.Ausdr. / Stringausdr.}[,{num.Ausdr. / Stringausdr.}]... 100 BBUILD A$,B$,X2,SQR(3)	
BEEP	Ausgabe eines akustischen Signals (0,2 sec.)	BEEP 20 BEEP	
BPAD	Eine Stringvariable (in internem Format) erhält binäre Füllzeichen bis zur deklarierten Länge	BPAD Stringvariable 40 BPAD A$	

Name	Funktion	Format	Anmerkung
BUILD	Die Werte einer Liste von Ausdrücken werden im Standardformat einer Stringvariablen zugewiesen, wobei ein Trennzeichen eingefügt werden kann	BUILD Stringvariable, {num.Ausdruck / Stringausdr.} [, {num.Ausdr. / Stringausdr.} ...] [;Delimiter] 30 BUILD X$,"P6060",Z;32 50 BUILD A$,V,B$;40	Delimiter: numerisches Trennzeichen (aus ISO-Code-Tabelle)
BUILD USING	Wirkung wie BUILD, jedoch erfolgt die Zuweisung der einzelnen Werte in einem im Programm definierten Format	BUILD USING {Zeilennr. / Stringvar.} ,Stringvar. , {num.Ausdr. / Stringausdr.} [[{num.Ausdr. / Stringausdr.} ...]] 80 BUILD USING 20,A$,B$	
CALL	Aufruf einer Assemberroutine. (Nur für P6066)	CALL Stringausdr.1 [, {num.Ausdr. / Stringausdr.} ....] [;DEB] 70 CALL A$,B1,A();DEB 80 CALL "ROUT4",X,Y	Stringausdruck 1: Name des Objektfiles DEB: Aufruf des Assembler-Debug

Name	Funktion	Format	Anmerkung
CHAIN	Beendet die Ausführung des laufenden Programmes und startet die Ausführung eines anderen Programmes	CHAIN { Stringausdruck } 40 CHAIN "PROG2" 20 CHAIN X2$	
COMMON	Definition der Variablen für die der Speicherplatz im Common-Bereich reserviert wird	COMMON { num. Feld () einf. String-Var. String-Feld () } [, { num.Feld () einf. String-Var. String-Feld () } ...] 75 COMMON A(),B$,C$()	Siehe Parameter COM = m im Systembefehl SAVE

Name	Funktion	Format	Anmerkung
CONVERT	1) Die ISO-Codes der Zeichen eines Stringausdruckes werden der Reihe nach den Elementen eines numerischen Vektors zugewiesen	CONVERT Stringausdr. TO num. Vektor LENGTH num. Var. 30 CONVERT T$ TO A LENGTH X 40 CONVERT "FRITZ" TO F LENGTH H	Die num. Variable erhält die aktuelle Länge des Stringausdruckes die Klammern bei der Bezeichnung des Vektors sind wegzulassen
	2) Die Werte eines num. Vektors werden als ISO-Zeichen interpretiert und der Reihe nach einer Stringvariablen zugewiesen	CONVERT num. Vektor TO Stringvar. LENGTH num. Ausdr. 80 CONVERT B TO Z$ LENGTH Z1 90 CONVERT X TO A1$ LENGTH A*C	Die aktuelle Länge der Stringvaraiblen ist gleich dem Wert des num. Ausdrucks
DATA	Baut ein internes Datenfile auf	DATA { num. Konstante / Stringkonstante } [, { num. Konstante / Stringkonstante }] ... 30 DATA "14.MAI",7,308.2,"MEIN"	

Name	Funktion	Format	Anmerkung
DCL	Festlegung der max. Länge für Stringvariable und einfacher Genauigkeit für num. Variablen	DCL { S({num. Var. / num. Feld ()}[,{num. Var. / num. Feld ()}]..)[,S({num.Var. / num. Feld ()}).]... / n({Stringvar. / Stringfeld ()}[,{Stringvar. / Stringfeld ()}]..)[,n({Stringvar. / Stringfeld ()}).]... / SINGLE }	1 ≤ n ≤ 1023
		40 DCL S(A,B,A()),50A$,30(B$,C$) 50 DCL SINGLE	
DEF	Definiert eine einzeilige Funktion	DEF { FNα (Parameter) = num. Ausdruck) / FNα$ (Parameter) = Stringausdruck) }	α:A...Z, Parameter liste: Folge von einfachen num. oder alphanum. Variablen die durch Komma getrennt sind. Max. 15 Parameter
		50 DEF FNC(A,B,C)=SQR(A)*B-C 70 DEF FNA$(A$,X$)="PREIS"+A$+X$	

Name	Funktion	Format	Anmerkung
DEF/FNEND	Definiert eine mehrzeilige Funktion die mit DEFFN beginnt, und mit FNEND abgeschlossen wird.	DEF {FNα / FNα$} [({num. Var. / Stringvar.} [, {num. Var. / Stringvar.}])] [{num. Var. / Stringvar.} [, {num. Var. / Stringvar.}] ...] ⋮ [LET] {FN* = num. Ausdr. / FN* $ = Stringausdruck} [[LET] {FN* = num. Ausdr. / FN* $ = Stringausdr.}] ⋮ FNEND 20 DEF FNB(X,Y) 30 IF X>Y THEN 60 40 LET FN*=Y 50 GOTO 70 60 LET FN*=X 70 FNEND	Anzahl lokaler Variabler und Parameter ≤ 15
DELAY	Bewirkt eine Unterbrechung der Programmausführung für eine bestimmte Zeit	DELAY n 30 DELAY 100	n: Zehntelsekunden positive ganze Zahl mit max. 4 Stellen

Name	Funktion	Format	Anmerkung
DEPAD	Entfernt die durch n angegebene Füllzeichen einer Stringvariablen (siehe auch PAD)	DEPAD Stringvariable, n 50 DEPAD A$,32	n = 0...255
DIM	Legt die Dimensionen von indizierten Variablen fest	DIM Feldname [(Zeilen [,Spalten])] [,Feldname [(Zeilen [,Spalten])]] 10 DIM H(35),Q(30,20)	
DISP	Ausgabe über Display im Standardformat	DISP [{num.Ausdr. / Stringausdr. / TAB(num.Ausdr.)} [{; / ,} {num.Ausdr. / Stringausdr. / TAB(num.Ausdr.)}] [;/,] ...] 30 DISP A;"WURZEL AUS A=";SQR(A) 50 DISP "QUERSCHNITT",	
DISP "@"	Alphanumerische Ausgabe auf dem Bildschirm (im Modus 23 Zeilen à 80 Zeichen)	DISP Controllstring Ausgabeliste 70 DISP "8"+CHR$(15)+CHR$(42)+CHR$(69)+"Matrix:"	Controllstring: Stringausdruck mit Byte 1: ESC Byte 2: Operationscode Byte 3: Zeile Byte 4: Spalte Ausgabeliste siehe "PRINT (USING)"

Name	Funktion	Format	Anmerkung
DISPUSING	Ausgabe über Systemzeile in definiertem Format	DISPUSING { Zeilennr. / Stringvar. }, { num.Ausdr. / Stringausdr. } [, { num.Ausdr. / Stringausdr. }]...	
		50 DISP USING 80,A,B,C$ 60 DISP USING A$,C$,"WERT",T	
END	Gibt das physische Ende eines Programmes an	END	
		3000 END	
FILE:	Schließt das File, das dem Filesdesignator zugeordnet ist. Der Stringausdruck liefert den Namen des Files, der unter Beibehaltung des Filedesignators geöffnet wird	FILE: Filedesignator , { Stringsausdr. / * }	Filedesignator: num. Ausdruck Der Stringausdruck muß einen Filenamen als Ergebnisstring haben
		30 FILE: 2,"DATEI4" 50 FILE: A*B,X$	

Name	Funktion	Format	Anmerkung
FILES	Öffnet und legt die Maximalzahl der Files fest, die bei der Ausführung des Programmes gleichzeitig geöffnet werden	FILES { Filename / * } [; { Filename / * }] .... 20 FILES DATEN;DATA 60 FILES STAMM;*;UMSATZ	
FKEY#	Belegung einer Funktionstaste mit einer Stringkonstanten	FKEY #n, Stringkonstante [:] 40 FKEY #8,"ANTON": 50 FKEY #1,3:	n = 1... 16 Der Doppelpunkt übernimmt die Funktion der EOL-Taste
FNEND	Beendet die Definition einer mehrzeiligen Funktion (siehe DEFFN)	FNEND siehe DEF / FNEND	
FOR	Beginn einer Schleife, (siehe auch: NEXT)	FOR Laufvar. = num.Ausdr. TO num.Ausdr. [STEP num.Ausdr.] 30 FOR I=1 TO 10 STEP 2 70 FOR J=X1 TO X2 STEP X1*Y	Die Angabe einer Schrittweite 1 kann unterbleiben

Name	Funktion	Format	Anmerkung
GOSUB	Aufruf eines Unterprogrammes (Siehe RETURN)	GOSUB Zeilennummer 50 GOSUB 3000	
GOTO	Unbedingter Sprung zur angegebenen Zeilennummer	GOTO Zeilennummer 60 GOTO 850	
IF...THEN	Sprung zur angegebenen Zeilennummer, wenn die Bedingungen erfüllt sind	IF ({num.Ausdr. / Stringausdr.} o {num.Ausdr. / Stringausdr.}) [{AND / OR} ({num.Ausdr. / Stringausdr.} o {num.Ausdr. / Stringausdr.})] THEN Zeilennr. 50 IF (A=B) OR (C=D) THEN 380 80 IF A$<B$ THEN 210	O steht für eine der Vergleichsoperationen: =, <, <=, >=, >, <>
:(IMAGE)	Definition eines Formates (das dann mit ... USING aufgerufen wird)	: {Text / Image-Feld} [{Text / Image-Feld}] ... 70 #### 'LLLLLL ##.## WERT: ##	

Name	Funktion	Format	Anmerkung
INPUT	Ermöglicht die Eingabe von num. und alphanum. Daten über die Tastatur	INPUT { num.Var. / Stringvar. } [, { num.Var. / Stringvar. }] ... 90 INPUT A$,X,Y	??→Es wurden zu wenig Daten eingegeben; der Rest ist nachzuholen INCORRECT FORMAT-RETYPE LINE → Die Eingabe ist formatgerecht zu wiederholen
INTERRUPT ENABLE	Bewirkt die Ausführung der angegebenen Funktion bei Auftreten von internen Interrupts	INTERRUPT ENABLE (I , Stringausdr.) 50 INTERRUPT ENABLE (I,"A")	I: Interner Interrupt (Error-Erkennung) Stringausdruck: Enthält Namen der Multiline-Funktion
LET	Wertzuweisung an Variable	[LET] { num.Var. [= num.Var.] ... = num.Ausdr. / Stringvar. [= Stringvar.] ... = Stringausdr. }	
MAT...	Kennzeichnet Matrixoperationen (siehe Abschnitt 4: Matrizenrechnen)		

Name	Funktion	Format	Anmerkung
NEXT	Ende einer Schleife (siehe FOR)	NEXT Laufvariable 20 NEXT I	
ON...GOSUB	Aufruf eines Unterprogrammes in Abhängigkeit vom Wert des num. Ausdruckes	ON num.Ausdr. GOSUB Zeilennr. [, Zeilennr.] ... 25 ON F GOSUB 230,80,1020 70 ON A+B GOSUB 100,200,300,400	
ON...GOTO	Sprung zu einer Zeile in Abhängigkeit vom Wert des num. Ausdrucks	ON num.Ausdr. GOTO Zeilennr. [, Zeilennr.] ... 60 ON F GOTO 2000,3000	
PAD	Eine Stringvariable wird bis zur deklarierten Länge mit dem zu n gehörenden ISO-Zeichen aufgefüllt.	PAD Stringvariable , n 80 PAD A$,32	n = 0...255 (ISO-Zeichen)

Name	Funktion	Format	Anmerkung
PRINT	Ausgabe von Daten im Standardformat	PRINT [{ num.Ausdr. / Stringausdr. / TAB(num.Ausdr.) } [{ ; / , } { num.Ausdr. / Stringausdr. / TAB(num.Ausdr.) }] [; / ,] ...] 40 PRINT A$,A,B$,B 50 PRINT "TEXT: ",X$	
PRINT USING	Ausgabe von Daten in definiertem Format	PRINT USING { Zeilennr. / Stringvar. } , { num.Ausdr. / Stringausdr. } [, { num.Ausdr. / Stringausdr. }] 50 PRINT USING 100,B$,"ZEIT",X 10 PRINT USING X$,X,Y,B$,Z$	Bei Masken-Overflow Ausgabe von ** ...
RANDOMIZE	Bei Aufruf von RND wird eine von der Standardfolge verschiedene Folge von Zufallszahlen erzeugt.	RAN [DOMIZE] 30 RAN	

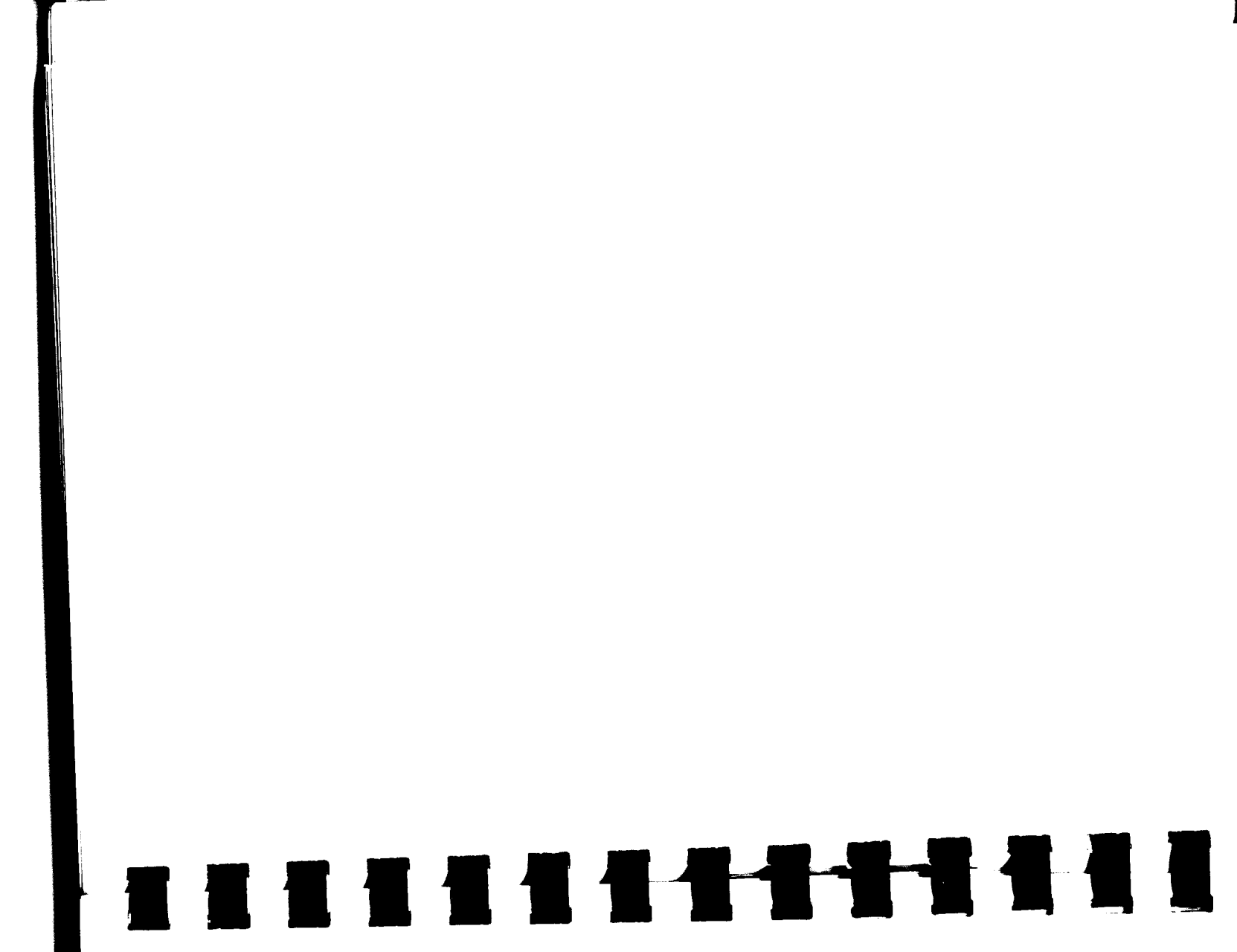
Name	Funktion	Format	Anmerkung
READ	Lesen von Daten aus einem internen File und Zuweisung an Variable (siehe DATA)	READ {num.Var. / Stringvar.} [, {num.Var. / Stringvar.}] ... 10 READ A,B,C$,C,A$(3)	
READ:	Lesen von Daten aus einem externen Daten- oder Textfile	READ : Filedesignator, {num.Var. / Stringvar.} [, {num.Var. / Stringvar.}] ... [EOF Zeilennr.] 90 READ #1,A$ 75 READ #X,B$ EOF 200	EOF: Bei logischem Fileende Sprung zur Zeilennummer
REM	Ermöglicht das Einfügen von Kommentaren in ein Programm	REM beliebiger Text 45 REM EINGABEROUTINE 55 REM----------	

Name	Funktion	Format	Anmerkung
RESTORE	Setzt den Pointer des internen Datenfiles auf das erste Element.	RESTORE Bsp.: 20 RESTORE	
RESTORE:	Versetzt das angegebene sequentielle File in den Lesemodus und setzt den Pointer auf das erste Element	RESTORE : Filedesignator 30 RESTORE :2 50 RESTORE :A+C	Filedesignator: num. Ausdruck:
RETURN	Rücksprung aus einem Unterprogramm (siehe GOSUB, ON...GOSUB)	RETURN 80 RETURN	
RKB	Ermöglicht die Zuweisung einer beliebigen Zeichenfolge über die Tastatur an eine Stringvariable	RKB Stringvariable 20 RKB C$	

Name	Funktion	Format	Anmerkung
SCRATCH:	Setzt in einem sequentiellen File den Pointer auf das erste Element, löscht den Inhalt und versetzt das File in den Schreibmodus	SCRATCH: Filedesignator 30 SCRATCH #3 90 SCRATCH #X*A	Filedesignator: num. Ausdruck
SETW:	Der Pointer eines Random-Files wird auf das angegebene Wort (4-Byte Wort) gesetzt.	SETW #Filedesignator TO Wortnummer 55 SETW #1 TO 50 30 SETW #X-Y TO INT(Z/2)	Filedesignator: num. Ausdruck Wortnummer: num. Ausdruck
STOP	Unterbricht die Ausführung des Programmes	STOP 60 STOP	System geht in Debugging-Mode Fortsetzung mit SHIFT EXIT

Name	Funktion	Format	Anmerkung
TRACE ON	Bewirkt den Ausdruck der Zeilennummern der Statements, die im folgenden ausgeführt werden	TRACE ON 40 TRACE ON	
TRACE OFF	Hebt TRACE ON auf	TRACE OFF 80 TRACE OFF	
WHERE:	Abfrage der Pointer-Position, des Daten typs und gegebenenfalls der Länge von Strings auf dem angegebenen externen File	WHERE: Filedesignator , n_1 [,n_2 [,n_3]] 70 WHERE : 3,P,L,T 15 WHERE : X,F	Filedesignator: num. Ausdruck n_1 n_2 n_3: numerische Variable

Name	Funktion	Format	Anmerkung
WRITE:	Schreibt Daten auf das angegebene externe Datenfile	WRITE#filedesignator , {num.Ausdr. / Stringausdr.} [, {num.Ausdr. / Stringausdr.}] ... [EOF Zeilennr.] 50 WRITE #4,A,Z$,"ENDE" 90 WRITE # B,C$+D$,SIN(X)	Filedesignator: num. Ausdruck EOF: Bei Fileende Sprung zur Zeilen-nummer



Standardfunktionen

Name	Funktion	Format	Anmerkung
ABS	Der Absolutwert eines num. Ausdruckes wird gebildet.	ABS (num. Ausdruck)	
ACS	Arcuscosinus (Funktionswert im Bogenmaß)	ACS (num. Ausdruck)	
ASN	Arcussinus (Funktionswert im Bogenmaß)	ASN (num. Ausdruck)	
ATN	Arcustangens (Funktionswert im Bogenmaß)	ATN (num. Ausdruck)	
COS	Cosinus (Argument im Bogenmaß)	COS (num. Ausdruck)	

Name	Funktion	Format	Anmerkung
COT	Cotangens (Argument im Bogenmaß)	COT (num. Ausdruck)	
DEG	Umwandlung von Bogenmaß in Altgrad	DEG	
DET	Berechnung der Determinante der zuletzt invertierten Matrix	DET	
EXP	Exponentialfunktion (e^x)	EXP (num. Ausdruck)	
HCS	Cosinushyperbolicus	HCS (num. Ausdruck)	
HSN	Sinushyperbolicus	HSN (num. Ausdruck)	
HTN	Tangenshyperbolicus	HTN (num. Ausdruck)	

Name	Funktion	Format	Anmerkung
INT	Größte ganze Zahl, die kleiner oder gleich dem Argument ist	INT (num. Ausdruck)	INT(-1.2) = -2
LGT	Dekadischer Lorgarithmus	LGT (num. Ausdruck)	
LOG	Natürlicher Logarithmus	LOG (num. Ausdruck)	
PI	Erzeugung der Konstanten π = 3.141592... in doppelter Genauigkeit	PI	
RAD	Umwandlung von Altgrad in Bogenmaß	RAD	
RND	Erzeugung einer zwischen 0 und 1 liegenden Zufallszahl (siehe RANDOMIZE)	RND	

Name	Funktion	Format	Anmerkung
SGN	Die Funktion liefert + 1 bei positivem, - 1 bei negativem Argument und ist 0, wenn das Argument 0 ist	ruck)	
SIN	Sinus (Argument im Bogenmaß)	SIN (num. Ausdruck)	
SQR	Quadratwurzel	SQR (num. Ausdruck)	
TAN	Tangens (Argument im Bogenmaß)	TAN (num. Ausdruck)	
LEN	Liefert als Ergebnis die aktuelle Länge des angegebenen String-ausdruckes. (Anzahl der Zeichen)	LEN (Stringausdruck) 55 A=LEN(A$)	

Name	Funktion	Format	Anmerkung
SCN	Liefert die Position, an der das erste Zeichen des Stringausdrucks 2 in dem Stringausdruck 1 ab Position n das m-te mal auftritt.	SCN (Stringausdr.1 , Stringausdr.2 , m , n) 70 C=SCN(A$,B$,A,B)	m,n: numerische Ausdrücke
TAB	Die Funktion ermöglicht in DISP oder PRINT die Ausgabe von Daten ab einer bestimmten Position	TAB (num. Ausdruck) 40 PRINT TAB(15);A$	
IOC	Liefert als Ergebnis Informationen über den Status einer peripheren Einheit (Q oder 1)	IOC (num. Ausdruck) 80 A=IOC(8)	IOC bezieht sich auf das zuletzt ausgeführte TEST oder WAIT
BLN$	Vergleich Bit und Bit zweier Strings, gemäß den Regeln der Bool'schen Algebra	BLN$ (num.Ausdr. , Stringausdr.1 , Stringausdr.2) 50 B$=BLN$(A*B,A$,C$) 80 PRINT BLN$(8,Y$,X$)	num. Ausdr.: Werte zwischen 0 und 15 die die Verknüpfungsregel definieren.

Name	Funktion	Format	Anmerkung
CHR$	Es wird das ISO-Zeichen erzeugt, das dem numerischen Wert des Arguments entspricht.	CHR$ (num.Ausdr.) 20 C$=CHR$(40) 60 DISP CHR$(A+B*C)	m,n: numerische Ausdruck
EXT$	Einem Stringausdruck wird ein Teilstring von Position m bis Position n (incl.) entnommen.	EXT$ (Stringausdr. , m , n) 30 E$=EXT$("MEIER",2,5) 70 V$=EXT$(V$,A+B,A*B)	
REP$	Im Stringausdruck 1 wird der Stringausdruck 2 durch den Stringausdruck 3 ersetzt.	REP$ (Stringausdr.1 , Stringausdr.2 , Stringausdr.3 , m , n) 50 A$=REP$("AABBAABB","BB","DD",1,4) 90 C$=REP$(A$+B$,E$,F$,V+W,Z)	m: Anzahl der Ersetzungen n: Position, bei der die Suche nach Stringausdruck 2 beginnt m= -1: jedesmal Ersetzung

Steuerung von Peripherie

Name	Funktion	Format	Anmerkung
BUFFER	Für einen I/O-Kanal wird im Arbeitsspeicher ein Puffer für den Datenaustausch mit einer peripheren Einheit reserviert.	BUFFER # per.-Einh. , Puffergröße 40 BUFFER #2,128	"per-Einh." ganze Zahl zwischen 0 ung 255 zur Definition des Kanals. "Puffergröße" ganze Zahl zwischen 1 und 65535 (byte)
CMD	Sendet Steuer- und Prüfbefehle an die Peripherie	CMD # per.-Einh. , Befehls-Code [, Befehls-Code] ... [AND GO] 20 CMD #12,3 40 CMD #T,4 AND GO	"per.-Einh."num. Ausdruck als Adresse der Peripherie. "Befehls-Code" num. Ausdruck "AND GO" letzter Steuerbefehl wird überlappt mit anderen Operationen in der CPU ausgeführt.

Name	Funktion	Format	Anmerkung
INTERRUPT ENABLE	Bewirkt die Ausführung der angegebenen Funktion bei Auftreten von externen Interrupts	INTERRUPT ENABLE (E , Stringausdr. , Intmsk [, Priorität]) 30 INTERRUPT ENABLE (E,A$,B$,2)	E:Externer Interrupt Stringausdruck: Name der Funktion Intmsk: Stringausdruck zur Definition der Kanäle, für die Interrupt behandelt wird. Priorität: num. Ausdruck für die Kanaladresse mit der höchsten Priorität.
RECEIVE	Empfangen von Daten von einer peripheren Einheit	RECEIVE # per.-Einh. , Stringausdruck [AND GO] 55 RECEIVE #4,B$ AND GO 10 RECEIVE #E,A$	"Per.-Einh.":num. Ausdruck als Adresse der Peripherie. "AND GO": Die Anweisung wird überlappend ausgeführt.

Name	Funktion	Format	Anmerkung
SEND	Senden von Daten an eine periphere Einheit.	SEND # per.-Einh. , Stringausdruck [AND GO] 30 SEND #5,A$ AND GO 30 SEND #40,C$+"****"	"Per.-Einh.":num. Ausdruck als Adresse der Peripherie. "AND GO": Die Anweisung wird überlappend ausgeführt
TEST	Status der peripheren Einheit wird in das Arbeitsregister übertragen. Laufende I/O-Operationen mit dieser Einheit werden nicht unterbrochen.	TEST # per.-Einh. 50 TEST #32	"per.-Einh.": num. Ausdruck als Adresse der Peripherie.
WAIT	Ende der I/O-Operationen wird abgewartet und Status der Peripherie in das Arbeitsregister übertragen	WAIT # per.-Einh. 50 WAIT #E	"per.-Einh.": num. Ausdruck als Adresse der Peripherie

Name	Funktion	Format	Anmerkung
FUNKTION FÜR	PERIPHERIE		
IOC	Liefert als Ergebnis Informationen über den Status einer peripheren Einheit (0 oder 1)	IOC (num. Ausdruck) 50 IF IOC(6)=P THEN 300	IOC bezieht sich auf das zuletzt ausgeführte TEST oder WAIT

Graphische Darstellungen (PLO ,GDI)

Name	Funktion	Format	Anmerkung
CPLOT	Ausgabe der Strings bezogen auf CSIZE.	CPLOT { Stringausdr. [; [Stringausdr.]] ... [,%] } ,% 20 CPLOT "Y=LOG(X)"+T$ 50 CPLOT A$;B$,%	,%: Bewirkt Zeilenschaltung.
CSIZE	Definition der Zeichengröße und der Schreibrichtung. Gilt für nachfolgende CPLOT- oder CTAB-Anweisungen	CSIZE Breite , Höhe , Winkel 40 CSIZE .3,.3,0 90 CSIZE 1/6,1/6,PI/2	Parameter sind num. Ausdrücke. Breite, Höhe: ≥ 0.1 Zoll Winkel: im Bogenmaß
CTAB	Tabulation um n-Zeichen und m-Zeilen gemäß CSIZE	CTAB n , m 20 CTAB 3,8	Parameter: n und m sind numerische Ausdrücke

Name	Funktion	Format	Anmerkung
DISP"ɵ"	Alphanumerische Ausgabe auf dem Graphikschirm (im Modus 39 Zeilen à 80 Zeichen)	DISP Controllstring Ausgabeliste 70 DISP "ɵW"+"4A"+A$ 40 DISP "ɵY"+CHR$(35)+CHR$(42)+"HELP"	Controllstring: Stringausdruck mit Byte 1: ESC Byte 2: Operations-Code (A,B, S, W,X,Y,Z) Byte 3: Zeile Byte 4: Spalte Ausgabeliste: siehe Statement "PRINT" bzw. "PRINT USING"
DISP USING	Alphanumerische Ausgabe auf dem Grafikschirm in definiertem Format	DISP USING { Zeilennr. / Stringvar } , [Controllstring] [Ausgabeliste] 90 DISP USING 10,"ɵZ"+A$+B$	"Verschiebung" num. Ausdruck mit: 0.1≤Verschiebung≤8 bur bei OPT PLO möglich

Name	Funktion	Format	Anmerkung
DOT	Zeichnen eines Punktes der durch Koordinaten definiert ist.	DOT X , Y 70 DOT A,B/2 40 DOT 4,6	X,Y:numerische Ausdrücke, die die Abszisse und die Ordinate des Punktes angeben.
DRAW	Ausgabeanweisung für das Bild über den System-Printer oder Hardcopy	DRAW [Verschiebung] 30 DRAW 50 DRAW 3	"Verschiebung" num. Ausdruck mit: 0.1 ≤ Verschiebung ≤ 8
ERASE	Löschen des Grafikschiminhaltes	ERASE [ON / OFF] 50 ERASE	
EXTERNAL PLOTTER	Vorwahl für einen externen peripheren Plotter.	EXTERNAL PLOTTER 25 EXTERNAL PLOTTER	

Name	Funktion	Format	Anmerkung
FRAME	Festlegung der Größe der Zeichenfläche	FRAME Breite , Höhe 40 FRAME 6,4.36	"Breite", "Höhe": numerische Ausdrücke mit: 0.03 Breite 8 0.03 Höhe 936 Für Grafikschirm haben die Parameter keine Wirkung
IDOT	Zeichnen eines Punktes durch Angaben von Relativkoordinaten dx und dy	IDOT dx , dy 10 IDOT 7,8 20 IDOT 4,X	dx,dy: numerische Ausdrücke
INIMAGE	Vorbereitung zum Generieren eines neuen Bildes.	INIMAGE [OFF / Filename] [, n] 50 INIMAGE BILD1,4 65 INIMAGE 90 INIMAGE OFF	"Filename" Name des sequentiellen Datenfiles."n" = Größe des Puffers 2 ≤ n ≤ 64 K bytes

Name	Funktion	Format	Anmerkung
IPLOT	Ziehen einer Verbinddungslinie zu dem Punkt der durch Relativkoordinaten dx und dy bestimmt wird	IPLOT dx , dy 30 IPLOT -2,5.5	"dx", "dy": num. Ausdrücke
LDIMAGE	Laden eines gespeicherten Bildes	LDIMAGE [Filename] 10 LDIMAGE BILD3 90 LDIMAGE	"Filename" als Name des Plotfiles
MOVE	Positionieren auf einen durch Koordinaten definierten Punkt	MOVE X, Y 65 MOVE X,Y2 30 MOVE A-INT(X/2),3	"X", "Y": numerische Ausdrücke die Abszisse und Ordinate eines Punktes bezeichnen
OFFSET	Verschiebt den Koordinatenursprung im Koordinatensystem.	OFFSET X, Y 20 OFFSET -2,4 60 OFFSET Z1,Z2/2	"X", "Y": num. Ausdrücke

Name	Funktion	Format	Anmerkung
PLOT	Ziehen einer Geraden zu dem durch Koordinaten definierten Punkt	PLOT X , Y 75 PLOT SIN(X),Z1	"X", "Y": num. Ausdrücke
POINTER	Darstellen des graphischen Pointers auf dem Bildschirm	POINTER num.Var.1 , num.Var.2 40 POINTER A,B	Übernahme der Werte durch SHIFT EXIT
REVERSE	Umkehrung des Bildes in negative Darstellung	REVERSE 50 REVERSE	
SCALE	Festlegung des gültigen Wertebereiches innerhalb von FRAME und damit Definition des Maßstabes.	SCALE X-min , X-max , Y-min , Y-max 25 SCALE -1,1,2,10 40 SCALE A,B,C,D	"X-min", "X-max", "Y-min", "Y"-max", num. Ausdrücke mit den Bedingungen X-min < X-max und Y-min < Y-max

Name	Funktion	Format	Anmerkung
STIMAGE	Abspeichern des Bildschirminhaltes auf ein Datenfile	STIMAGE [Filename] [,Puffergröße] 80 STIMAGE 40 STIMAGE PLOT1,5	Nur bei Option GDI (Auch im Debugging-Mode möglich) 2 ↵ n ↵ 64
XAXIS	Zeichnen einer Achse in X-Richtung	XAXIS Y [,tic] [,Start-Punkt , End-Punkt] 20 XAXIS 0,1,-1,10	"Y", "tic","Start-Punkt" und "End-Punkt" als numerische Ausdrücke und "tic" ≠ 0
YAXIS	Zeichnen einer Achse in Y-Richtung.	YAXIS X [,tic] [,Start-Punkt , End-Punkt] 60 YAXIS 0,1,-5,3	"X", "tic", "Start-Punkt" und "End-Punkt" als numerische Ausdrücke und "tic" ≠ 0

MATRIZENRECHNEN

Name	Funktion	Format	Anmerkung
MAT... =	Elementweise Zuweisung der Werte einer Matrix an eine andere (zeilenweise)	MAT Matrix = Matrix 50 MAT A=B	
MAT... + MAT... -	Matrizenaddition oder Matrizensubtraktion	MAT Matrix = Matrix { + / - } Matrix 70 MAT A=B+C 80 MAT B=C-A	
MAT... *	1) Multiplikation einer Matrix mit einem Skalar (einfache Variable, Konstante, arithm. Ausdruck) 2) Multiplikation zweier Matrizen	MAT Matrix = { (num.Ausdr.) / Matrix } * Matrix 40 MAT A=(5)*A 80 MAT C=A*B	

Name	Funktion	Format	Anmerkung
MAT...CON	Setzt alle Elemente der Matrix gleich 1	MAT Matrix = CON [(num.Ausdr. , num.Ausdr.)] 10 MAT A=CON 50 MAT B=CON(12,20)	
MAT...IDN	Weist der angegebenen Matrix die Einheits-matrix zu. (Elemente der Haupt-diagonale = 1, alle anderen = 0)	MAT Matrix = IDN [(num.Ausdr. , num.Ausdr.)] 30 MAT A=IDN(5,5)	
MAT INPUT	Eingabe der Elemente eines Feldes über Tastatur. Die Zuwei-sung der Werte er-folgt elementweise (durch Komma getrennt)	MAT INPUT Feldname [(num.Ausdr. , num.Ausdr.)] 40 MAT INPUT A(8,9) 20 MAT INPUT A$	

Name	Funktion	Format	Anmerkung
MAT...INV	Bildet die inverse Matrix einer quadratischen Matrix	MAT Matrix = INV [(Matrix)] 50 MAT A=INV(B)	
MATREAD	Elementweise Zuweisung von Daten aus einem internen File an ein oder mehrere Felder	MAT READ Feldname [(n.A , n.A)] [,Feldname [(n.A , n.A)]]... 20 MAT READ A,B$ 85 MAT READ A(9,7),B$(9,15)	
MATREAD:	Elementweise Zuweisung von Daten aus einem Datenfile an die angegebenen Felder	MAT READ: Filedesignator,Feld [(n.A,n.A)] [,Feld [(n.A,n.A)]]... [EOF Zeilennr.] 50 MAT READ #1,A,B EOF 100	n.A= numerischer Ausdruck

Name	Funktion	Format	Anmerkung
MATPRINT	Ausdruck einer oder mehrerer Matrizen im Standardformat (untereinander)	MAT PRINT Feldname [{,;} Feldname] ... [{,;}] 10 MAT PRINT A,B,C$	Der Ausdruck der einzelnen Matrizen erfolgt zeilenweise, wobei sich die Trennzeichen auf die Elemente der Matrix beziehen. Die einzelnen Matrizen sind durch eine Leerzeile getrennt
MATPRINT-USING	Wie MATPRINT, die Ausgabe erfolgt jedoch in einem definierten Format	MAT PRINT USING {Zeilennr. / Stringvar.} , Feldname [, Feldname] ... 80 MAT PRINT USING 20,A,B,C$ 60 MAT PRINT USING A$,A,B$	
MAT...TRN	Berechnet die Transponierte einer Matrix (Zeilen und Spaltenvertauschung)	MAT Matrix = TRN (Matrix) 90 MAT A=TRN(B)	

Name	Funktion	Format	Anmerkung
MAT WRITE:	Schreibt die Feldelemente auf das angegebene Datenfile	MAT WRITE: Filedesignator , Feld [, Feld] ... [EOF Zeilennr.] 30 MAT WRITE 5,A,B$ EOF 500	
MAT...ZER	Alle Elemente der Matrix werden 0 gesetzt	MAT Matrix = ZER [(num.Ausdr, , num.Ausdr.)] 45 MAT A=ZER(SQR(A),B*C)	
FUNKTION:			
DET	Berechnung der Determinante der zuletzt invertierten Matrix	DET 10 PRINT A+DET 50 A=DET	

ANHANG

Meldungen des Betriebssystems

Bei der Ausführung von Programmen werden vom Betriebssystem Meldungen über Bildschirm ausgegeben, die einerseits dazu dienen, den Anwender über den jeweiligen Betriebszustand der Maschine zu informieren und andererseits sowohl auf Programm- als auch auf Bedienungsfehler aufmerksam zu machen.

Dabei können drei Arten von Systemmeldungen unterschieden werden:

1. Bedienungshinweise

2. Informationsmeldungen

3. Fehlermeldungen für die Ausführung von

   - BASIC-Statements
   - Systembefehlen
   - Dienstprogrammen

In der Folge wird eine kurze Beschreibung jeder Meldungsart gegeben.

Die Bedienungshinweise und Informationsmeldungen sind selbsterklärend, für die Fehlermeldungen folgt eine komplette Liste der möglichen Codes in dieser Beschreibung, außerdem wird eine allgemeine Anleitung zum Beheben dieser Fehler gegeben.

Diese Meldungen machen den Anwender darauf aufmerksam, daß Eingaben nicht korrekt erfolgt sind.

Werden zum Beispiel als Antwort auf eine Inputanforderung einer **INPUT**-Anweisung zu viele Daten eingegeben, so kommt die Meldung **TOO MUCH INPUT - EXCESS IGNORED.** Es werden in der Folge nur so viele Daten an Variable zugewiesen, als einzelne Eingaben angefordert werden.

Es kann jedoch passieren, daß durch falsche Eingaben das System Eingabezeilen nicht in der vom Anwender beabsichtigten Form interpretiert (z.B. Komma statt Dezimalpunkt bei der Eingabe von Dezimalzahlen).

Sollte das Ursache der Meldung sein, so kann im allgemeinen über die im Programm vorhandenen Korrekturroutinen eine Korrektur der fehlerhaften Eingabe erfolgen.

Ebenfalls bei der Ausführung einer **INPUT**-Anweisung kann die Meldung **INCORRECT FORMAT - RETYPE LINE** auftreten. Es wurde dann anstatt einer vom System erwarteten Zahl ein String eingegeben und das System wartet auf die Eingabe einer korrekten Zeile.

A N M E R K U N G

Wird statt eines Strings eine Zahl eingegeben, so kann das vom System nicht als Fehler behandelt werden, da ja Strings auch aus Folgen von Ziffern bestehen können.

Informationsmeldungen

Durch Informationsmeldungen wird der Anwender über den Betriebszustand des Systems informiert.

Die Meldung **READY** bedeutet, daß das System bereit ist, Systembefehle auszuführen, d.h. sich im **COMMAND-MODE** befindet.

Die Meldung **PROGRAM name READY TO RUN** zeigt bei aktivierter Taste **PRINT ALL** an, daß die Preexecution in Folge eines **PREPARE**-Befehles erfolgreich beendet wurde.

Die Ausführung des Programms kann durch Drücken der Tasten **SHIFT + EXIT** gestartet werden.

Soll das Programm nicht ausgeführt werden, so sind die Tasten **CTRL + EXIT** zu drücken.

Die Meldung **PROGRAM name RUNNING** zeigt an, daß das Programm **name** ausgeführt wird. Diese Meldung kann durch Meldungen des Anwenderprogramms ersetzt sein.

Bei der Ausführung von Befehlen und Dienstprogrammen wird der Anwender über die erforderlichen Bedienungshinweise informiert.

Die genaue Bedeutung dieser Meldung kann dem Anwenderhandbuch, Beschreibung des gewählten Befehles oder Dienstprogrammes, entnommen werden.

Soll im Zuge der Ausführung eines Dienstprogrammes oder der Ausführung von **DCH** eine Disk getauscht werden, so treten Meldungen der Form **LOAD DISK ON...** bzw. **RESTORE DISK** auf.

Es kann dann ohne weitere Eingabe der Tausch durchgeführt werden. Anschließend ist mit **SHIFT + EXIT** fortzusetzen.
Als Warnmeldungen können auftreten:
ACTION ON UNIT... LISTED LIBRARIES WILL BE DELETED
Diese Meldungen zeigen an, daß ein bestehender Inhalt auf einer Disk zerstört wird. Soll die Operation durchgeführt werden, so ist mit **SHIFT + EXIT** fortzufahren. Werden die Tasten **CTRL + EXIT** gedrückt, so kann ohne Zerstörung eines bestehenden Inhalts die Operation abgebrochen werden.

A N M E R K U N G

Wurde **SHIFT + EXIT** nicht gedrückt, so werden Eingaben über die Tastatur mit ERROR 190 beantwortet.

Es ist dann zunächst **CLEAR** und anschließend **SHIFT + EXIT** zu drücken. Nach Drücken von **END OF LINE** wird die Eingabe akzeptiert.

Fehlermeldungen

Diese Meldungen bezeichnen Fehler, die in der Anwendung von

- Systembefehlen
- BASIC-Statements
- Dienstprogrammen

auftreten können.

Diese Meldungen, die durch eine entsprechende Nummer den Fehler genau spezifizieren, können in vier Gruppen eingeteilt werden:

1. Warnmeldungen

Treten beim Initialisieren des Systems Konditionen auf, die nicht den gespeicherten Konfigurationen entsprechen, so werden die entsprechenden Fehlermeldungen gedruckt. Diese Meldungen können sich auf nicht vorhandene Disketten, Bibliotheken oder konfigurierte Peripherie beziehen. Die Meldungen erfordern keine unmittelbare Behebung durch den Anwender.

2. Syntaktische Fehler

Sie treten bei der fehlerhaften Eingabe von Systembefehlen oder BASIC-Statements auf.

3. Preexecution - Fehler

Diese Fehlermeldungen weisen vor der Ausführung eines Programms auf Fehler in der Struktur des Programmes hin.

4. Ausführungsfehler

Diese Fehlermeldungen treten auf, wenn im Zuge der Ausführung eines Programmes, eines Systembefehles oder Dienstprogrammes unzulässige oder die Ausführung verhindernde Konditionen auftreten.

Syntaktische Fehler werden vom System unmittelbar bei der Eingabe erkannt. Wird nach der entsprechenden Fehlermeldung die Taste **CLEAR** gedrückt, so kann die Eingabe sofort korrigiert werden, wobei sich der Pointer in der Eingabezeile an der Stelle befindet, an der das System den Fehler erkannte.

Nach der Korrektur der Zeile kann durch Drücken von **END OF LINE** die Eingabe des Befehls abgeschlossen werden.

Fehler der Preexecution werden entweder bei der Ausführung des Systembefehls **PREPARE** oder **RUN** gemeldet und beziehen sich auf die logische Struktur des Programmes, wobei diese Fehlermeldungen gedruckt werden. Das System geht in den **COMMAND-MODE** über, so daß in einfacher Weise alle notwendigen Korrekturen durchgeführt werden können.

Ausführungsfehler können während der Ausführung eines Programmes entweder aufgrund von Programmfehlern oder aufgrund nicht anschließbarer unzulässiger Datenkombinationen auftreten.

Ausführungsfehler können entweder behebbar oder nicht behebbar sein.

Behebbare Fehler können während der Ausführung des Programmes korrigiert werden und erfordern keinen Abbruch der Programmausführung.

Wird ein behebbarer Fehler erkannt, so wird die entsprechende Fehlermeldung angezeigt und das System schaltet in den **DEBUGGING-MODE.**

Die behebbaren Fehler treten meist infolge von unzulässigen Variablenwerten auf, wobei das System in diesen Fällen eine Standardlösung vorsieht. Wird die Standardlösung akzeptiert, so kann durch Drücken von **SHIFT** + **EXIT** die Ausführung des Programmes wieder aufgenommen werden.

Ist die Standardlösung für den speziellen Fall nicht günstig, so können die Werte der Variablen mit den Möglichkeiten des **DEBUGGING-MODES** korrigiert und anschließend das Programm fortgesetzt werden. Dabei ist jedoch zu beachten, daß die Ausführung mit der nächsten Anweisung fortgesetzt wird.

Tritt ein behebbarer Fehler aufgrund einer Variablen auf, der noch kein Wert zugewiesen wurde (**ERROR 1**), so wird der Wert Null bzw. Nullstring für die Ausführung dieses Statements angenommen, die Variable bleibt jedoch weiterhin nicht initialisiert.

Die Fortsetzung der Programmausführung kann nach einem behebbaren Fehler mit allen Möglichkeiten des **DEBUGGING-MODES** wieder aufgenommen werden. Es kann also entweder mit **SHIFT + EXIT** die Ausführung mit der nächsten Zeile fortgesetzt werden, mit **EXIT** das Programm schrittweise abgearbeitet oder mit **START** ab einer beliebigen Zeile ausgeführt werden.

LISTE DER FEHLERCODES

Fehlermeldungen werden aufgrund eines numerischen Fehlercodes identifiziert. Bei Fehlermeldungen der Preexecution und bei den Ausführungsfehlern wird - wenn möglich - zusätzlich die Zeilennummer der Programmzeile, in der der Fehler auftrat, angegeben.
(Beispiel: ERROR 6 IN LINE 155).

Im folgenden Abschnitt werden die möglichen Fehlercodes mit der Beschreibung angegeben, wobei die Liste in die oben beschriebenen Gruppen unterteilt ist.

Fehler mit den Nummern 1 bis 16 beziehen sich auf behebbare Fehler bei der Ausführung eines BASIC-Programmes:
Fehler mit Codes 40 bis 57 können bei der Preexecution eines Programmes auftreten.

Die Fehlercodes von 63 bis 98 bezeichnen nicht behebbare Ausführungsfehler, Codes zwischen 100 und 128 können bei der fehlerhaften Eingabe von BASIC-Zeilen oder der Compilierung eines Textfiles auftreten.
Die Fehlercodes mit den Nummern 131 bis 160 beziehen sich auf die Systeminstallation und Fehler, die beim Zugriff auf eine Diskstation auftreten.

Nicht behebbare Fehler in der Verbindung mit peripheren Einheiten werden durch die Codes 162 bis 172 gemeldet, und Fehler, die bei der Eingabe oder Ausführung von Systembefehlen auftreten, haben Nummern zwischen 173 und 227.

Die Codes 228 bis 235 zeigen Fehler in der Ausführung von Dienstprogrammen an, behebbare und nicht behebbare Fehler der Plot-Option werden in den Codes 236 bis 254 beschrieben.

Bei fehlerhaften Konditionen bei der Behandlung von Interrupts treten Fehler mit den Nummern 280 bis 284 auf.

Abschließend werden noch die Fehlercodes, die einen Systemabbruch anzeigen und die Meldungen für nicht betriebsbereite Einheiten, angeführt.

1. Behebbare Fehler während der Ausführung von BASIC - Programmen

Fehlercode	Bedeutung
1	Eine numerische oder alphanumerische Variable wurde nicht initialisiert. Es wird für das auszuführende Statement der Wert 0 bzw. der Leerstring angenommen, die Variable gilt jedoch weiterhin als nicht initialisiert.
2	Fehlerhaftes Argument in einer Stringfunktion
3	Overflow (z.B. bei Division durch Ø). Als Ergebnis der Operation wird die größte darstellbare Zahl mit dem richtigen Vorzeichen angenommen.
4	Underflow. Als Ergebnis der Operation wird Null angenommen.
6	Entweder ist das Argument einer Funktion außerhalb der zulässigen Grenzen, oder es soll die Wurzel einer negativen Zahl berechnet werden. Es wird die Wurzel des Absolutbetrages berechnet.
7	Bei einer Stringoperation entsteht ein String mit einer Länge von mehr als 1023 Zeichen. Der String wird nach der 1023. Stelle abgeschnitten.

Fehlercode	Bedeutung
8	Der String, der an eine Stringvariable zugewiesen werden soll, ist länger als die deklarierte Länge der Stringvariablen. Der String wird entsprechend der deklarierten Länge der Stringvariablen abgeschnitten.
9	Es soll der Logarithmus einer negativen Zahl berechnet werden. Es wird der Logarithmus des Absolutbetrages berechnet.
10	Es wird versucht, den Logarithmus von Null zu berechnen. Als Ergebnis wird -9.999999999999E+99 gesetzt.
11	Eine negative Zahl hat einen nicht ganzzahligen Exponenten. Es wird mit dem Absolutbetrag der Basis gerechnet.
12	Der Wert Null hat einen negativen Exponenten. Als Ergebnis wird die größte darstellbare Zahl (+9.999999999999E+99) gesetzt.
13	Die zu invertierende Matrix ist singulär (d.h. die Determinante ist gleich Null). die Ergebnismatrix enthält nicht definierte Werte, eine nachfolgende Verwendung der Funktion DET liefert jedoch ein konkretes Ergebnis.

Fehlercodes	Bedeutung
14	Die adressierte Peripherie ist nicht angeschlossen oder eingeschaltet, oder es tritt während des Betriebes eine abnormale Kondition auf. Die Übertragung wird nicht ausgeführt.
15	Übertragungsfehler während eines Datenaustausches mit einer Peripherie. Die Datenübertragung wird nicht ausgeführt.
16	Beim letzten überlappten Input/Output für die gleiche Peripherie ist ein Übertragungsfehler aufgetreten.

2. Fehler während der Preexecution eines Programmes

Fehlercodes	Bedeutung
40	Unerlaubter Sprung: a) Sprung zu einer nicht existierenden Zeilennummer b) Sprung aus einer mehrzeiligen Funktion oder von außen in den Vereinbarungsteil einer mehrzeiligen Funktion c) Sprung von außen in eine FOR/NEXT-Schleife
41	NEXT ohne zugehörendes FOR oder Überschneidung von FOR/NEXT-Schleifen.
42	Innerhalb der Definition einer mehrzeiligen Funktion wird eine andere mehrzeilige Funktion definiert.
43	Eine aufzurufende Funktion wurde nicht definiert. (Fehler kann möglicherweise nur durch **DEC** und **COM** behoben werden.)
44	Es sind mehr als 15 FOR/NEXT-Schleifen geschachtelt.

Fehlercodes	Bedeutung
45	a) FN*, FN*$, FNEND treten außerhalb der Definition einer mehrzeiligen Funktion auf b) In einer Definition einer mehrzeiligen Stringfunktion wird die Pseudovariable FN* oder in der Definition einer mehrzeiligen numerischen Funktion wird die Pseudovariable FN*$ verwendet.
46	Zwei- oder mehrere ineinander geschachtelte FOR/NEXT-Schleifen haben dieselbe Laufvariable.
47	FOR ohne ein nachfolgendes NEXT mit der gleichen Laufbariablen. Dieser Fehler kann weitere Fehlermeldungen während der Preexecution zur Folge haben.
48	In der Definition einer mehrzeiligen Funktion fehlt die **FNEND**-Anweisung.
49	Eine einfach indizierte und eine doppelt indizierte Variable haben denselben Namen.
50	**END** hat nicht die höchste Zeilennummer im Programm.

Fehlercodes	Bedeutung
51	Das Programm enthält keine **END**-Anweisung.
52	Das auszuführende Programm enthält nicht compilierte Zeilen. Der Fehler tritt auf, falls die bei der Compilierung eines Textfiles gemeldeten Fehler nicht korrigiert wurden.
53	In der Definition einer mehrzeiligen Funktion erfolgt keine Wertzuweisung an die Pseudovariable FN* bzw. FN*$
54	Für eine **PRINT USING**-Anweisdung (**DISP USING**, ...) fehlt die zugehörende Formatvereinbarung.
55	Innerhalb der Definition einer mehrzeiligen Funktion befindet sich eine **STOP**-Anweisung.
56	Der für den **COMMON**-Bereich reservierte Platz ist für die Speicherzuweisung der als **COMMON** deklarierten Variablen zu klein.
57	Das File im Arbeitsspeicher hat keinen Namen.

3. Nicht behebbare Fehler während der Ausführung eines BASIC-Programmes

Fehlercodes	Bedeutung
63	Das System ist so konfiguriert, daß nur Programme mit bereits durchgeführter Preexekution ausgeführt werden können (Betriebssystem RO). Das Programm muß mit einem vollständigen (programmierbaren) System preexekutiert und anschließend auf die betreffende Disk gebracht werden.
65	Während der Programmausführung wird der zur Verfügung stehende Bereich des Anwenderspeichers überschritten (z.B. bei zu tiefer Schachtelung von Unterprogrammen). Das System geht direkt in den Command-Mode über.
66	Der Index einer indizierten Variablen hat einen nicht zulässigen Wert (d.h. kleiner als 1 oder größer als die maximale Dimension dieser Variablen).
67	Die Anweisung verlangt die Vereinbarung von Dimensionen einer Matrix, die größer sind als die maximalen Dimensionen.
68	Beim Aufruf der Programmausführung beginnend mit einer angeführten Zeilennummer (nach **RUN**-Zeilennummer, oder **START** Zeilennummer) tritt ein **NEXT** ohne zugehörendes **FOR** auf.

Fehlercodes	Bedeutung
69	Die Parameter im Aufruf einer Funktion stimmen in ihrem Typ nicht mit den Parametern der Definition dieser Funktion überein.
70	Es soll ein **RETURN**-Statement ohne vorangehendes **GOSUB** ausgeführt werden.
71	Die Summe der Längen der Strings, die den Funktionstasten zugewiesen werden sollen, ist größer als die maximal zulässige Länge (238 Zeichen). Die Belegungen von **FKEY**-Tasten können durch **FKEY n**, gelöscht werden.
72	Die Anzahl der Parameter in einem Funktionsaufruf stimmt nicht überein mit der Anzahl der Parameter in der Definition der Funktion.
73	Die aktuellen Dimensionen einer Matrix erlauben nicht die Ausführung der gewünschten Operationen (z.B. nicht quadratische Matrix bei einer Inversion).
74	Innerhalb einer definierten Funktion wurde die maximal zulässige Anzahl von Aufrufen anderer definierter Funktionen (max. 255) überschritten.

Fehlercodes	Bedeutung
75	Die zur Ausführung des Programmes benötigte Option wurde nicht in den Arbeitsspeicher geladen. Für die Behebung des Fehlers ist der Systembefehl **OPT...** unter Angabe der benötigten Option auszuführen.
76	Es wird versucht, ein File zu öffnen, das bei der vorherigen Ausführung eines Programms nicht physisch geschlossen wurde. Physisch offen gebliebene Files sind im CATALOG mit **OPEN** gekennzeichnet. Ohne genaueste Kenntnis des internen Programmablaufs muß auf der letzten Datensicherungskopie aufgesetzt werden. Alle seit diesem Stand angefallenen Arbeiten sind zu wiederholen!!
77	Es fehlt entweder das **FILES**-Statement oder der Filedesignator ist kleiner als 1 oder größer als die Anzahl der Files in der FILES-Anweisung.
78	Ein File ist nicht entsprechend dem Typ der auszuführenden Operationen geöffnet.
80	Der Wert des Wortdesignators einer **SETW**-Anweisung ist größer als es der Länge des entsprechenden Files entspricht.
81	Innerhalb eines Programmes wird versicht, ein bereits offenes File zu öffnen.

Fehlercodes	Bedeutung
82	Der verfügbare Platz in einem externen File reicht nicht für die Ausführung der verlangten Operation.
83	Das angesprochene File ist nicht geöffnet.
84	Es wurde bei einer Lese- oder Schreiboperation das Fileende ohne **EOF**-Angabe erreicht.
85	Das Argument einer **TAB**-Funktion in einer **PRINT** oder **DISP**-Anweisung ist kleiner als 1.
86	Einer numerischen Variablen soll ein String zugewiesen werden.
87	Die deklarierte Länge der Stringvariablen in einer **BBUILD**-Anweisung reicht nicht für die Ausführung der Anweisung.
88	Zu wenig Daten im internen File bei der Ausführung einer **READ**-Anweisung oder im Stringausdruck bei der Ausführung von **ASSIGN**.
89	die Formatfelder stimmen in ihrem Typ nicht mit den Daten der aufrufenden **PRINT UNSING** oder **DISP USING**-Anweisung überein.
90	Ein Wert kleiner als Ø oder größer als 255 soll in ein **ISO**-Zeichen umgewandelt werden.

Fehlercodes	Bedeutung
91	Der Ausdruck, der die Anzahl der in einer **CONVERT**-Anweisung umzuwandelnden Vectorelemente festlegt, ergibt einen negativen Wert.
92	Der in einer **CHAIN**-Anweisung angegebene Filename hat ein ungültiges Format.
93	Die Daten eines externen Files stimmen in ihrem Typ nicht mit den Variablen der entsprechenden **READ**-Anweisung überein. Derselbe Fehler tritt auch bei der Ausführung einer **BASSIGN**-Anweisung auf, falls der Typ der Daten und der entsprechenden Variablen nicht übereinstimmt.
96	Der Wortdesignator einer **SETW**:-Anweisung hat einen Wert kleiner 1.
97	Eine **SCRATCH**:- oder **APPEND**:-Anweisung bezieht sich auf ein Randomfile.
98	Die Anzahl der angegebenen Parameter in einer **WHERE**:-Anweisung verlangt, daß das sequentielle File für das Lesen geöffnet ist.

4. Fehlermeldungen des BASIC-Compilers

Fehlercodes	Bedeutung
100	Eine Programmzeile besteht nur aus der Zeilennummer.
101	a) Unzulässige Zeilennummer b) Bei Ausführung des Befehls **LINK** würden sich Zeilennummern überschneiden.
102	Unzulässiges Schlüsselwort
103	Unzulässiger Parameter
104	Unzulässiger Ausdruck
105	Der Operator ist für den Typ der Operanden nicht zulässig.
106	In einem Aufruf einer Funktion stimmt die Anzahl oder Typ der Paramter nicht mit den verlangten Parametern überein.
107	Ungültiges Format für einen Filenamen
109	Nicht interpretierbarer Syntaxfehler
110	Der Name der Funktion in der **DEF**-Anweisung kommt bereits in einer DEF-Anweisung mit einer anderen Zeilennummer vor.

Fehlercodes	Bedeutung
111	Es wurden bereits zuviele verschiedene Zeilennummern als Sprungziele definiert. Es sind maximal 255 Sprungziele zulässig. Jeder Funktionsaufruf gilt als eine Verzweigung. Nicht mehr verwendete Sprungziele können nur mit **DEC** und **COM** gelöscht werden.
112	Die Anzahl der verwendeten Variablen übersteigt die maximal zulässige Anzahl (max. 123 numerische bzw. 255 Stringvariable).
113	Unzulässiges Zeichen (z.B. bei verschiedener Anzahl von öffnenden und schließenden Klammern in einem numerischen Ausdruck).
114	Rekursiver Aufruf in der Definition einer einzeiligen Funktion.
115	Unzulässiger Bezug auf eine Variable oder eine Funktion.
116	Es wurden mehr als 26 numerische Felder vereinbart.
117	Speicherüberschreitung bei der Eingabe oder beim Laden eines Programmes oder Textes. Ggf. sind unnötige Options zu löschen.
118	Es wurde bereits eine **FILES**-Anweisung mit einer anderen Zeilennummer eingegeben.
119	Es wurde bereits die max. zulässige Anzahl von Funktionen definiert.
120	Die Zeilennummer, auf die Bezug genommen wird, existiert nicht.
128	Unzureichender Platz für die Compilierung der eingegebenen Zeile.

5. Fehlermeldungen im Zusammenhang mit den Diskstationen

Fehlercodes	Bedeutung
131	Ungültiger Operand zur Angabe des Typs einer Diskstation
132	Ungültiger interner Code für die Bezeichnung einer Peripherie.
133	Die angegebene Diskstation hat bereits einen Namen.
134	Der angegebene interne Code paßt nicht zum angegebenen Typ der Diskstation.
135	Syntaktischer Fehler im symbolischen Namen
136	Der symbolische Name wurde bereits für eine andere Station vergeben.
137	Ungültiges Zeichen im symbolischen Namen einer Diskstation.
138	Es wurde keine Floppy-Disk-Einheit konfiguriert.
146	In einer Assembler-Routine erfolgt ein unzulässiger Bezug auf einen externen Modul.
147	Das angegebene Assembler-Objekt-File ist nicht zusammenhängend (Extent 1) gespeichert.
151	Fehler auf der linken Floppy-Disk-Station
152	Fehler auf der rechten Floppy-Disk-Station

Fehlercodes	Bedeutung
154	Eine konfigurierte Einheit enthält keine Disk oder Floppy-Disk oder es wird eine im System nicht konfigurierte Einheit angesprochen.
155	Die Disk in der angegebenen Einheit ist zerstört
156	Die Konfiguration hat keine System-Disk
157	Die Disk mit dem internen Code AØ ist entweder beschädigt oder nicht betriebsbereit
158	Die Disk mit dem internen Code A1 ist entweder beschädigt oder nicht betriebsbereit
159	Die Disk mit dem internen Code A2 ist entweder beschädigt oder nicht betriebsbereit
160	Die Disk mit dem internen Code A3 ist entweder beschädigt oder nicht betriebsbereit

A N M E R K U N G :

Zu den Fehlern 151, 152, 158 und 160:
Tritt der Fehler wiederholt und auch mit anderen Disks auf, so liegt ein Hardwarefehler vor. Bezieht sich der Fehler auf eine Disk, so kann mit den Dienstprogrammen **LIBCOPY** bzw. **FLCOPY** versucht werden, den bestehenden Inhalt zu kopieren.

6. Nicht behebbare Fehler im Zusammenhang mit peripheren Einheiten

Fehlercodes	Bedeutung
162	**SEND** für Input-Einheit oder **RECEIVE** für Output-Einheit (falsche Peripherie-Adresse)
163	Der mit **SEND** zu übertragende String ist größer als der Puffer des entsprechenden Kanals
164	Ein File ist zerstört und sein Inhalt verloren. Das File wird im Katalog mit Parameter F durch **DEAD** gekennzeichnet (siehe CATALOG)
165	Das Statement bezieht sich auf einen peripheren Kanal, der in der Konfiguration nicht vorhanden ist
166	Für den angesprochenen Kanal wurde kein Puffer definiert
167	Es wurde eine negative Peripherieadresse oder eine negative Command-Nummer angegeben
168	Im **OPTION**-Befehl wurde gleichzeitig **PLO** und **GDI** angegeben. **OPT PLO** und **OPT GDI** sind unverträglich.
169	Eine Peripherieadresse oder eine Command-Nummer sind größer als 255
170	Die in einem **RECEIVE**-Statement angegebene Stringvariable ist länger als der Puffer des entsprechenden Kanals
171	Der mit **CONFIGURE** angegebene externe Drucker ist nicht betriebsbereit a) es kann ohne externen Drucker weitergearbeitet werden b) ist der Drucker wieder betriebsbereit, so muß neu konfiguriert oder initialisiert werden

7. Fehler der Eingabe oder Ausführung von Systembefehlen

Fehlercodes	Bedeutung
172	Der für **OPT GDI** notwendige Grafik-Controller ist im System nicht vorhanden.
173	Es ist keine Bibliothek offen. Für das Arbeiten mit Files muß mindestens eine Bibliothek offen sein.
174	Es sind bereits sechst Bibliotheken offen.
175	Im Befehl **CONFIGURE** wird mehr Speicherkapazität als im System vorhanden ist, oder weniger als 16K verlangt.
177	Die angegebene Diskstation existiert nicht oder es wurde ein falscher Name angegeben.
178	Die angegebene Bibliothek ist auf der durch Namen oder implizit festgelegten Einheit nicht offen. (Siehe Systembefehle **LBOPEN** und **LVTOC**.)
179	Auf der angegebenen Station ist keine Bibliothek geöffnet (Siehe Systembefehle **LBOPEN** bzw. **LVTOC**).

Fehlercodes	Bedeutung
180	Die angegebene Bibliothek ist nicht offen. (Siehe Systembefehl **LBOPEN**)
181	Der zur Verfügung stehende Speicher reicht nicht für die Ausführung des Befehls.
182	Beim Befehl **TRANSCODE** wurde bei der Umwandlung eines Datenfiles in ein Textfile der Parameter angegeben, es sind jedoch im Datenfile keine Zeilennummern enthalten.
183	Die angesprochene Teil-Bibliothek ist in der Bibliothek nicht vorhanden, oder es wird die zulässige maximale File-Anzahl für eine Teil-Bibliothek überschritten.
186	In der Teil-Bibliothek existiert bereits ein File mit gleichem Namen.
187	Das File mit dem angegebenen Namen wird nicht gefunden (z.B. falscher Filename, Bibliothek nicht offen, ...)
188	Auf der Disk ist in einer Bibliothek zu wenig Platz für die Ausführung des Befehls BEMERKUNG: Die Ausführung der Befehle **RUN** und **PRE** ohne Angaben benötigen einen sogenannten listfähigen Platz in der ersten offenen Bibliothek.

Fehlercodes	Bedeutung
189	Ein Datenfile soll so verkleinert werden, daß möglicherweise signifikante Informationen gelöscht würden.
190	Der Befehl wird im gegenwärtigen Status des Systems nicht akzeptiert oder ist nicht bekannt (z.B. falscher Mode, fehlerhafte Eingabe, im konfigurierten System nicht enthalten. **SHIFT+EXIT** wurde vergessen.)
191	Es wurde kein Filename angegeben oder das File im Arbeitsspeicher hat keinen Namen.
192	Unzulässiges Zeichen im Befehl
193	Es fehlen Parameter im Befehl
194	Aufruf einer nicht existierenden Zeilennummer
195	Im Debugging-Mode wird der Befehl "**START Zeilennummer**" nach einem Befehl "**RUN Filename**" nicht akzeptiert, da das auszuführende Programm ohne vorhergehende Preexecution auf der Disk abgespeichert ist. Das gilt auch für den Befehl "**PREPARE Filename**".
196	Unzulässiger Parameter oder fehlerhaftes Format bei der Eingabe eines Befehls.
197	Der Befehl bezieht sich auf eine Zeilennummer, die innerhalb der Definition einer mehrzeiligen Funktion liegt.

Fehlercodes	Bedeutung
198	Es wird mehr Speicherplatz verlangt, als in der Bibliothek vorhanden ist.
199	Die verlangte Operation ist für ein geschütztes File (secured) nicht zulässig.
200	Unerlaubte Operationen, da die Teil-Bibliothek geschützt ist.
201	Der Befehl ist nicht ausführbar, da beide Diskstationen Disks enthalten müssen.
202	In der Ausführung eines Befehls müßte ein File in mehr als 4 Teile aufgeteilt werden.
203	In einem **LIST-** oder **DELETE**-Befehl ist die erste Zeilennummer größer als die zweite.
204	Die Ausführung einer Prozedur wird abgebrochen, da bei der Ausführung eines Programmes mehr Eingabezeilen auszuführen sind, als im Command-File IN-Zeilen existieren oder eine IN-Zeile enthält Strings, die numerischen Variablen zugewiesen werden sollen.
205	Unerlaubte Operation, da die Zeile geschützt ist.
206	Das File im Arbeitsspeicher ist kein Programm.

Fehlercodes	Bedeutung
207	Die Operation ist für den angesprochenen Filetyp nicht ausführbar.
208	Die im **OPTION**-Befehl angeführte Option ist im System nicht vorhanden.
209	Es wurde eine Zeilennummer größer als 9999 erzeugt.
210	In einem **LIST**-Befehl wurde für den Ausdruck eines Programmes der Paramter X angegeben.
211	Leerer Arbeitsspeicher
212	Die zu druckende(n) Zeile(n) existiert(existieren) nicht.
213	Bei der Decompilierung einer Zeile tritt ein Line-Overflow auf.
214	Im Unterprogramm, das als mehrzeilige Funktion mit **LINK** eingefügt werden soll, ist die erste Zeile kein **DEF**-Statement.
215	Die angegebene Bibliothek ist bereits offen.
216	Es wird eine Einheit angesprochen, die bei Installation des Systems nicht konfiguriert wurde.
217	Ungültiges Password für eine Bibliothek.
218	Die angegebene Bibliothek wurde nicht gefunden.

Fehlercodes	Bedeutung
219	Die gewünschte Operation ist für die Disk, die das System enthält, nicht zulässig.
220	Das Programm kann nicht in den Arbeitsspeicher geladen werden, da der Platz für den vereinbarten **COMMON**-Bereich fehlt.
221	Der angegebene Name bezieht sich nicht auf eine Floppy-Disk-Einheit.
222	Die angegebenen Diskeinheiten haben einen verschiedenen Typ.
223	Der Volume-Label stimmt nicht mit dem angegebenen überein.
224	Die Operation verlangt mehr Platz auf einer Disk, als im Moment frei ist.
225	Die verlangte Operation ist nur für Single-Drive-Disk-Stationen vorgesehen.
226	Es wurde eine ungültige Disk eingelegt.
227	Es wurde keine Bibliothek angegeben.

8. Fehler beim Aufruf oder der Ausführung von Dienstprogrammen

Fehlercodes	Bedeutung
228	Es wurde entweder versucht, auf einem Datenträger mehr als 35 Bibliotheken einzurichten oder es ist zu wenig Platz auf der Diskstation.
229	Es existiert bereits eine Bibliothek mit dem gleichen Namen auf derselben Diskstation.
230	Die angegebene Disk kann nicht initialisiert werden, da sie entweder beschädigt ist, oder mehr Alternativspuren angelegt werden mußten als möglich ist.
232	In einem Aufruf von **LBCREATE** ist die Anzahl der Sektoren, die für den Katalog reserviert werden sollen, größer als die Anzahl der Sektoren in der Bibliothek.
234	Es fehlt die Angabe des Namens des Dienstprogrammes.
235	Das aufgerufene Dienstprogramm existiert nicht (z.B. falscher Name oder nicht im konfigurierten System enthalten).

9. Fehlermeldungen, die sich auf PLOT-OPERATIONEN beziehen

Fehlercodes	Bedeutung
236	Behabbar: das in **INIMAGE** genannte File existiert nicht, das Bild wird nur im Arbeitsspeicher generiert.
237	Behebbar: a) Das genannte File ist nicht sequentiell oder zu klein. b) Nach der Preexecution bleibt nur eine Anwenderkapazität kleiner als 1280 Bytes. Das Bild wird nur im Arbeitsspeicher generiert.
238	Für den externen Plotter fehlt die Funktion **FNP** (nicht behebbar).
239	In der Anweisung **DRAW** wird das Bild außerhalb des Zeichenbereiches verschoben (nicht behebbar).
240	Nicht behebbar: Zur Ausführung von **DRAW** fehlt der Thermodrucker.
241	Nicht behebbar: Es wurde weder **INIMAGE** noch **LDIMAGE** ausgeführt.
242	Behebbar: **FRAME** folgt nicht unmittelbar auf **INIMAGE**.
243	Behebbar: Als Abstand für die Markierung der Achsen wurde 0 angegeben. Der Operand wird ignoriert.

Fehlercodes	Bedeutung
244	Behebbar: Nach der Preexecution ist der verfügbare Speicherplatz für den Anwender kleiner als 1280 Bytes, die Ausführung kann dadurch langsamer werden.
245	Nicht behebbar: in **FRAME** ist der Wert für die Breite nicht zulässig.
246	Nicht behebbar: in **FRAME** ist der Wert für die Höhe nicht zulässig.
247	Nicht behebbar: In **SCALE** ist X-Min ≥ X-Max oder Y-Min ≥ Y-Max.
248	Nicht behebbar: Der in **INIMAGE** vereinbarte Puffer ist für die in **FRAME** angegebene Größe des Bildes zu klein.
249	Behebbar: Die für das File vereinbarte Größe ist kleiner als der Puffer im Arbeitsspeicher. Das Bild wird nur im Arbeitsspeicher erzeugt.
250	Behebbar: a) Die Größe des Puffers oder des externen Files ist zu klein, so daß keine weiteren Punkte gespeichert werden können. Alle folgenden **PLOT**-Anweisungen mit Ausnahme von **INIMAGE**, **LDIMAGE** und **DRAW** werden ignoriert. b) In einem Programm, für das kein externes File zur Speicherung des Bildes existiert, soll ein **DRAW**-Statement zum zweiten Mal ausgeführt werden. Alle folgenden **PLOT**-Anweisungen mit Ausnahme von **INIMAGE** und **LDIMAGE** werden ignoriert.

Fehlercodes	Bedeutung
251	Nicht behebbar: In der Anweisung **CSIZE** ergibt die Angabe von Breite und Höhe einen Wert kleiner 0.1
252	Nicht behebbar: Das File in der Anweisung **LDIMAGE** enthält kein Bild.
253	Behebbar: Die im Debugging-Mode aufgerufene Funktion enthält ein **DRAW, INIMAGE** oder **LDIMAGE**-Statement.
254	Behebbar: Bei der Ausführung der Zeichnung wurde die maximal mögliche Anzahl von Schreibvorgängen erreicht. Alle nachfolgenden **PLOT**-Anweisungen mit Ausnahme von **DRAW** werden ignoriert.

10. Fehlermeldungen für die Interrupt-Behandlung

Fehlercodes	Bedeutung
280	Der Funktionsname in einem **INTERRUPT ENABLE**-Statement ist länger als 1 Zeichen.
281	Die Länge der Maske für die Kanäle ist ungleich 16.
282	Während der Behandlung eines internen Interrupts tritt ein weiterer Interrupt auf.
283	Ungültiger Wert für die Priorität.
284	Die Funktion, die von **INTERRUPT ENABLE** aufgerufen werden soll, existiert nicht.

11. Fehlermeldungen für einen nicht normalen Systemzustand

Fehlercodes	Bedeutung
m*A	Falls m eine andere Zahl als 12 oder 16 ist, und der Fehler auch bei anderen als den eingelegten Disks auftritt, so liegt ein Hardwarefehler vor.
12*A	Eine Fehlerhafte Kondition im Betriebssystem ist aufgetreten.
16*A	Tritt der Fehler auch nach einem erneuten Initialisieren des Systems auf, so liegt ein Hardwarefehler vor.
A N M E R K U N G :	Fehlermeldungen, die nach dem numerischen Code *A haben, bewirken ein Löschen des Arbeitsspeichers. Nach Drücken der Tasten **SHIFT+EXIT** wird das System neu geladen.
ABN unit name	Die angegebene Einheit arbeitet nicht korrekt, es ist zu überprüfen, ob der Einschub geschlossen und eine Disk eingelegt wurde.
ABN PRT	Der Drucker arbeitet nicht korrekt.

Fehlercodes	Bedeutung
ABN unit-name DCH OMITTED	Bei einer Diskstation wurde ohne vorherige Eingabe des **DCHANGE**-Befehls ein Einschub geöffnet oder eine Disk gewechselt. Dies kann behoben werden durch Drücken von **CLEAR** und anschließend der **SHIFT+EXIT**-Tasten. Ist danach kein Arbeiten mit dem System möglich, muß es neu initialisiert werden.
ABEND	Meldungen der Form **ABEND ...** werden von aufgerufenen Assembler-Routinen verursacht.
A N M E R K U N G :	Statusmeldungen **ABN** können nach Beheben der Fehlerursache mit **CLEAR** gelöscht und mit **SHIFT+EXIT** der Ablauf fortgesetzt werden.

Liste der darstellbaren Zeichen und ihre entsprechenden dezimalen Werte

Zeichen	dezim. Wert	Zeichen	dezim. Wert	Zeichen	dezim. Wert	Zeichen	dezim. Wert
[illegible]	0	!	33	B	66	c	99
┌	1	"	34	C	67	d	100
⊥	2	#	35	D	68	e	101
┘	3	$	36	E	69	f	102
[illegible]	4	%	37	F	70	g	103
[illegible]	5	&	38	G	71	h	104
[illegible]	6	'	39	H	72	i	105
[illegible]	7	(	40	I	73	j	106
↖	8	)	41	J	74	k	107
→	9	*	42	K	75	l	108
≡	10	+	43	L	76	m	109
↓	11	,	44	M	77	n	110
[illegible]	12	-	45	N	78	o	111
←	13	.	46	O	79	p	112
[illegible]	14	/	47	P	80	q	113
[illegible]	15	0	48	Q	81	r	114
[illegible]	16	1	49	R	82	s	115
[illegible]	17	2	50	S	83	t	116
[illegible]	18	3	51	T	84	u	117
[illegible]	19	4	52	U	85	v	118
[illegible]	20	5	53	V	86	w	119
√	21	6	54	W	87	x	120
π	22	7	55	X	88	y	121
┤	23	8	56	Y	89	z	122
Σ	24	9	57	Z	90	{	123
¥	25	:	58	[	91	|	124
[illegible]	26	;	59	\	92	}	125
[illegible]	27	<	60	]	93	~	126
[illegible]	28	=	61	↑	94	[illegible]	127
[illegible]	29	>	62	_	95	[illegible]	128
[illegible]	30	?	63	`	96	[illegible]	129
[illegible]	31	@	64	a	97	.	...
	32	A	65	b	98	[illegible]	255

Darstellung der ISO-Zeichen der Spalten 0 und 1 der ISO-Code-Tabelle

Taste CONTROL	Darst. des Zeichens	ISO-Zeichen	Dez. Wert
@		NUL	0
A		SOH	1
B		STX	2
C		ETX	3
D		EOT	4
E		ENQ	5
F		ACK	6
G		BEL	7
H		BS	8
I		HT	9
J		LF	10

Taste CONTROL	Darst. des Zeichens	ISO-Zeichen	Dez. Wert
K		VT	11
L		FF	12
M		CR	13
N		SO	14
O		SI	15
P		DLE	16
Q		DC1	17
R		DC2	18
S		DC3	19
T		DC4	20
U		NAK	21

Taste CONTROL	Darst. des Zeichens	ISO-Zeichen	Dez. Wert
V		SYN	22
W		ETB	23
X		CAN	24
Y		EM	25
Z		SUB	26
[		ESC	27
\		FS	28
]		GS	29
↑		RS	30
0		US	31

ISO – CODE – Tabelle

Hexadezimal		0	1	2	3	4	5	6	7
	Dual	0000	0001	0010	0011	0100	0101	0110	0111
0	0000	0 NUL	16 DLE	32 SP	48 0	64 @	80 P	96 `	112 p
1	0001	1 SOH	17 DC1	33 !	49 1	65 A	81 Q	97 a	113 q
2	0010	2 STX	18 DC2	34 "	50 2	66 B	82 R	98 b	114 r
3	0011	3 ETX	19 DC3	35 #	51 3	67 C	83 S	99 c	115 s
4	0100	4 EOT	20 DC4	36 $	52 4	68 D	84 T	100 d	116 t
5	0101	5 ENQ	21 NAK	37 %	53 5	69 E	85 U	101 e	117 u
6	0110	6 ACK	22 SYN	38 &	54 6	70 F	86 V	102 f	118 v
7	0111	7 BEL	23 ETB	39 '	55 7	71 G	87 W	103 g	119 w
8	1000	8 BS	24 CAN	40 (	56 8	72 H	88 X	104 h	120 x
9	1001	9 HT	25 EM	41)	57 9	73 I	89 Y	105 i	121 y
A	1010	10 LF	26 SUB	42 *	58 :	74 J	90 Z	106 j	122 z
B	1011	11 VT	27 ESC	43 +	59 ;	75 K	91 [	107 k	123 {
C	1100	12 FF	28 FS	44 ,	60 <	76 L	92 \	108 l	124 |
D	1101	13 CR	29 GS	45 -	61 =	77 M	93]	109 m	125 }
E	1110	14 SO	30 RS	46 .	62 >	78 N	94 ↑	110 n	126 ~
F	1111	15 SI	31 US	47 /	63 ?	79 O	95 _	111 o	127 DEL

OPTIONS (Systembefehl OPT)

OPTIONS		Erforderlich für folgende Features
PLO	(**PLO**T)	Alle **BASIC**-Anweisungen zur Verwendung des Thermodruckers als Plotter.
GDI	(Graphic **DI**splay)	Alle **BASIC**-Anweisungen und Befehle für den Bildschirm mit Ausnahme von: **BASIC**-Anweisungen **DISP, ERASE, REVERSE,** **DEBUGGING**- und **SYSTEM**-Befehle **ERASE, REVERSE, DRAW A**

Datentyp	Platzbelegung			Bemerkungen
	Arbeitsspeicher	Externer Speicher		
	Bytes	Bytes	Wort	
Numerische Konstante	$4 * \frac{N+1}{2}$	8	2	1. N Anzahl Ziffern der numerischen Konstanten 2. Der Wert des Ausdrucks $\frac{N+1}{2}$ wird auf die nächste ganze Zahl aufgerundet.
Numerische Variable	10	--		Numerische Variable, die einfach genau deklariert sind, werden schneller bearbeitet als doppelt genaue.
Werte einer numerischen Variablen	--	4	1	Einfach genau
		8	2	Doppelt genau
Matrix oder Vektor	$\frac{N*4+10}{N*8+10}$	--		Einfach genau N Anzahl Elemente Doppelt genau der Matrix oder des Vektors
String-konstante	N + 2	$4 * INT\left(\frac{N-1}{4}+2\right)$	$INT\left(\frac{n-1}{4}+2\right)$	1. N Anzahl Zeichen des Strings 2. INT bedeutet, daß der ganzzahlige Anteil des Ausdruckes zu bilden ist.
String-variable	N + 9	--		N aktuelle Länge des Strings
Wert einer Stringvariablen	--	siehe Stringkonstante		
String-array	13+ (L+2)*N	--		1. L aktuelle Länge der Elemente 2. N Anzahl Elemente des Arrays

Speicherbedarf von Daten

Textdarstellung auf dem Bildschirm im Alpha-Mode (23 Zeilen à 80 Zeichen)

Ausgabecode

Beispiel / Buchst.A		normal	blinking	highlight	blink+high	reverse	blink+rev.
Alpha - Schirm mit 23 Zeilen	A	0	16	32	48	64	80
	A	1	17	33	49	-	81
	A	2	18	34	50	-	82
	A	3	19	35	51	67	-
	A	4	20	36	52	68	84
	A	5	21	37	53	69	85
	A	6	22	38	54	70	86
	A	7	23	39	55	71	-
	A	8	24	40	56	72	-
	A	9	25	41	57	73	-
	A	10	26	42	58	74	-
	A	11	27	43	59	75	91
	A	12	28	44	60	76	92
	A	13	29	45	61	77	93
	A	14	30	46	62	78	94
	A	15	31	47	63	79	95

auszugebender String

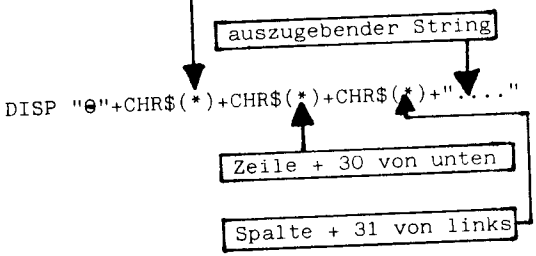
DISP "Θ"+CHR$(*)+CHR$(*)+CHR$(*)+"...."

Zeile + 30 von unten

Spalte + 31 von links

Textdarstellung auf dem Bildschirm im Graphik-Mode (39 Zeilen à 80 Zeichen)

	Dezimal-Code		Iso-Zeichen	Bedeutung
Grafik - Schirm mit 39 Zeilen	65	≙	A	Aufhebung von Code B
	66	≙	B	Nachfolgende PRINT-Anweisung nicht auf Schirm
	83	≙	S	Alle Zeilen um eine Zeile nach oben verschoben
	87	≙	W	Auszugebender String ersetzt alle anderen Zeichen in norm. Schrift
	88	≙	X	Auszugebender String erscheint zusätzlich in negativer Schrift
	89	≙	Y	Auszugebender String erscheint zusätzlich in normaler Schrift
	90	≙	Z	Auszugebender String ersetzt alle anderen Zeichen in neg. Schrift



Das Handbuch dient der Information, sein Inhalt ist ohne ausdrückliche schriftliche Vereinbarung nicht Vertragsgegenstand. Technische Änderungen behalten wir uns vor. Die angegebenen Daten sind lediglich Nominalwerte.

DEUTSCHE OLIVETTI GMBH · FRANKFURT-NIEDERRAD · LYONER STRASSE 34