

**BILDUNGSCOMPUTER  
robotron A5105**

**RBASIC-Kurzbeschreibung**

**VEB ROBOTRON-MESSELEKTRONIK >OTTO SCHÖN< DRESDEN  
Lingnerallee 3, Postfach 211, Dresden, DDR-8012**

**RBASIC-Kurzbeschreibung  
BILDUNGSCOMPUTER  
robotron A5105**

---

**I N H A L T**

<b>1. Konstanten, Variable, Felder</b>	2
- Zahlendarstellung	
- Einfache Variable	
- Felder	
<b>2. Operatoren</b>	3
- Arithmetische Operationen	
- Vergleichsoperationen	
- Logische Operationen	
<b>3. RBASIC-Kommandos</b>	4
- Grundkommandos	
- Speichern und Laden von Programmen	
- Kommandos zur Dateiverwaltung	
<b>4. RBASIC-Grundanweisungen</b>	7
<b>5. RBASIC-Funktionen</b>	11
- Numerische Standardfunktionen	
- Zeichenkettenfunktionen	
- Sonstige Funktionen	
<b>6. Grafische Anweisungen und Funktionen</b>	14
- Textausgabe im Grafikmodus	
<b>7. Speichern und Laden von Daten</b>	15
- Sequentielles Laden	
- Dateien mit direktem Zugriff	
<b>8. Hardwareorientierte Anweisungen und Funktionen</b>	18
<b>9. SCREENS und Farbcodierungen</b>	19

---

**VEB ROBOTRON-MESSELEKTRONIK >OTTO SCHÖN< DRESDEN  
Lingnerallee 3, Postfach 211, Dresden, DDR-8012**

## 1. Konstanten, Variable, Felder

### Zahlendarstellung

Darstellung	Präfix	Beispiel
dezimal	--	1.23, 567, -.89E35
hexadezimal	&H	&HFFF : 4095 dezimal
oktal	&O	&O377 : 255 dezimal
binär	&B	&B11111111 : 255 dezimal

Bereiche entsprechend den jeweiligen numerischen Variablentypen, hexadezimal, oktal und binär, immer ganzzahlig.

### Einfache Variable

Typ	Name	Bereich
reel, einfach (4 Bytes, 6 Stellen)	XY!	$\pm 9.99999E-64$ bis $\pm 9.99999E+62$ und 0
reel, doppelt (8 Bytes, 14 Stellen)	XY (oder XY#)	$\pm 9.9999999999999E-64$ $\pm 9.9999999999999E+62$
integer (2 Bytes)	XY%	-32768 bis 32767
Zeichenkette	XY\$	0 bis 255 Zeichen

### Felder

Dimension	numerisches Feld	Zeichenkettenfeld
1	XY(5)	XY\$(5)
2	XY%(7,8)	XY\$(4,9)
:	:	:

Felder mit bis zu 11 Elementen (Index 0 bis 10) können ohne Dimensionierung verwendet werden. Größere Felder müssen vorher DIMensioniert werden:

DIM A(N,M) setzt die Maximalwerte für die Indizes fest, reserviert Speicher für  $(N+1) * (M+1)$  Elemente, erstes Element : A(0,0)

## 2. Operatoren

Die Operatoren sind in der Folge ihrer Priorität (von der höchsten zur niedrigsten) aufgeführt.

### Arithmetische Operationen

Operator	Operation	Eingabeformat
^	Potenz	X ^ Y
-	Negation (Minus)	-X
*	Multiplikation	X * Y
/	Division	X / Y
\	ganzzahlige Division	X \ Y
MOD	Rest bei Division mit Integerzahlen	X MOD Y
+	Addition	X + Y
-	Subtraktion	X - Y

### Vergleichsoperationen

Operator	Bezeichnung	Beispiel
=	gleich	X = Y , X\$ = Y\$
<	kleiner	X < Y , X\$ < Y\$
>	größer	X > Y , X\$ > Y\$
<>, ><	ungleich	X <> Y , X\$ >< Y\$
<=, =>	kleiner oder gleich	X <= Y , X\$ <= Y\$
>=, =<	größer oder gleich	X >= Y , X\$ => Y\$

## Logische Operationen

Operation	Bezeichnung	Eingabeformat
NOT	Negation	NOT X
AND	logisch UND	X AND Y
OR	logisch ODER	X OR Y
XOR	exclusives ODER	X XOR Y
EQV	Äquivalenz	X EQV Y
IMP	Implikation	X IMP Y

X	NOT X	X	Y	X AND Y	X OR Y	X XOR Y	X EQV Y	X IMP Y
0	-1	0	0	0	0	0	-1	-1
-1	0	0	-1	0	-1	-1	0	-1
		-1	0	0	-1	-1	0	0
		-1	-1	-1	-1	0	-1	-1

## 3. RBASIC Kommandos

### Grundkommandos

AUTO [anfzn][,schrittweite]	erzeugt automatisch Zeilennummern mit angegebener Schrittweite
CLEAR [zeichenbereich[,endadr]]	stellt alle Variablen auf Null, stellt Größe des Zeichenbereiches ein sowie die größte verfügbare Speicheradresse; schließt alle geöffneten Dateien
CONT	setzt Programmausführung fort
DELETE [zn][-zn]	löscht angegebene Programmzeilen
DELETE.	löscht letzte angegebene Programmzeile
LIST [anfzn][-][endzn]	zeigt Programmliste auf Bildschirm an

LIST. [-][endzn]	zeigt Programmliste ab letzter bearbeiteter Programmzeile an
LLIST [anfzn][-][endzn]	druckt Programmliste über Drucker aus
NEW	löscht RBASIC-Programm und alle Variablen im Speicher
RENUM [znneu],[znalt],[schrittweite]	numeriert Programmzeilen neu
RUN [zn]	startet das Programm ab Zeilennummer
SCREEN,,,,[tastenclick],[baudrate],[druckertyp],[datenformat]	dient zum Ein- bzw. Ausschalten des Tastenclicks, zum Einstellen der Übertragungsgeschwindigkeit am Kassetteninterface, zum Einstellen des Druckertyps und des V.24-Datenformats für den Druckerausgang (siehe auch Abschnitte 6 und 7)
TROFF	schaltet Programmablaufverfolgung aus
TRON	schaltet Programmablaufverfolgung ein
<u>Speichern und Laden von Programmen</u>	
BLOAD "[gerät:]dateiname"[,R][,versatz]	lädt Speicherbereich (z.B. Maschinencodeprogramm) in den Hauptspeicher bzw. lädt und führt Programm aus
BSAVE "[gerät:]dateiname", anfadr,endadr[,startadr]	speichert angegebenen Speicherbereich auf angegebenes Gerät

CLOAD ["dateiname"]	lädt eine Datei von Kassette
CLOAD? ["dateiname"]	vergleicht ein auf Kassette gespeichertes Programm mit dem Speicher
CSAVE "dateiname"[,baudrate]	speichert Programm auf Kassette
LOAD "[gerät:]dateiname"[,R]	lädt RBASIC-Programm in den Speicher
MERGE "[gerät:]dateiname"	lädt ein RBASIC-Programm (ASCII) und mischt es mit dem im Speicher
RUN "[gerät:]dateiname"	lädt RBASIC-Programm in den Speicher und startet es
SAVE b"[gerät:]dateiname"	speichert Programm ab (... , mit Zusatz A im ASCII-Code]

#### Kommandos zur Dateiverwaltung

CALL FORMAT	initialisiert eine Diskette
COPY "[gerät1:]dateiname" TO "[gerät2:]dateiname"	kopiert eine Datei
FILES ["[gerät:]dateiname"]	zeigt Disketteninhaltsverzeichnis an
KILL "[gerät:]dateiname"	löscht Datei auf Diskette
LFILES ["[gerät:]dateiname"]	druckt Disketteninhaltsverzeichnis
NAME "[gerät:]altname" AS "neuname"	ermöglicht das Umbenennen einer Diskettendatei
RESET	aktiviert eine neue, eben gesteckte Diskette; z.B. nach Diskettenwechsel

#### 4. RBASIC-Grundanweisungen

BEEP	erzeugt Piepton
CHR\$(n) = zeichenkette n=0,...,255	ermöglicht die Änderung des Zeichensatzes
CLS	löscht den (aktiven) Bildschirm
COLOR [vordergrundfarbe],[hintergrundfarbe],[randfarbe]	bestimmt die Farbe im Textmodus
DATA konstante[,konstante]...	erstellt Datentabelle, die von einer READ-Anweisung benutzt wird
DEF FN funktion[(parameter[,parameter]...)]	=ausdruck definiert eine Anwenderfunktion
DEFDBL buchstabe1[-buchstabe2]	definiert doppelt-genaue Variable (8 Bytes)
DEFINT buchstabe1[-buchstabe2]	definiert ganzzahlige Variable (2 Bytes)
DEFSNG buchstabe1[-buchstabe2]	definiert einfach-genaue Variable (4 Bytes)
DEFSTR buchstabe1[-buchstabe2]	definiert Zeichenkettenvariable
DIM var(maxindex[,maxindex]...)[,var( ),...]	legt Namen, Datentyp, Größe und Dimension von Feldern fest
END	beendet ein Programm, schließt Datei
ERASE feldname[,feldname]...	löscht Felder
ERROR fehlernummer	setzt einen bestimmten Fehler
FOR var=anwert TO endwert [STEP schrittweite]	
NEXT var	wiederholte Programmausführung NEXT schließt Schleife ab
GOSUB zn	ruft ein BASIC-Unterprogramm auf

GOTO zn	verzweigt zu der angegebenen Zeile	
IF ausdruck THEN anweisung [:anweisung]...[ELSE anweisung [:anweisung]]	erzeugt Programmverzweigung in Abhängigkeit vom Wert des Ausdrucks	
INPUT ["hinweis";]var[,var]...	liest Daten von der Tastatur während der Programmabarbeitung	
INTERVALL	OFF	verhindert   Unterbrechung
	ON	ermöglicht   durch
	STOP	hält   Zeitablauf
KEY fkt_tastenummer,zeichenkette	definiert Zeichenkette für Funktionstaste	
KEY LIST	zeigt Funktionstastenbelegung an	
KEY OFF	löscht Funktionstastenbelegung	
KEY ON	zeigt Funktionstastenbelegung an	
KEY(n)	OFF	verhindert   Unterbrechung
	ON	ermöglicht   durch
	STOP	hält   Funktionstaste n
[LET]var = ausdruck	ordnet Wert des Ausdrucks der Variablen var zu	
LINE INPUT ["hinweis";]var	liest Zeichenkette über Tastatur auf Zeichenkettenvariable	
LOCATE zeile,spalte[,kursor]	positioniert den Cursor im Textmodus, kursor=0 schaltet Cursor aus	
LPRINT ausgabeliste	gibt Daten auf den Drucker aus	
LPRINT USING format;ausgabeliste	gibt Daten auf den Drucker im bestimmten aus	
MID\$(zeichenkette1,m[,n] = zeichenkette2	ersetzt einen Teil der Zeichenkette 1 durch die Zeichenkette 2; beginnend mit der Stelle m, n Zeichen ersetzend	

MOTOR [OFF] [ON]	schaltet Motor des Kassettengerätes		[aus] [ein] um
ON ERROR GOTO zn	setzt Programm bei Programmfehler mit angegebener Zeilennummer fort		
ON ausdruck GOSUB zn[,zn]...	verzweigt zu Unterprogramm je nach Wert des Ausdruckes		
ON ausdruck GOTO zn[,zn]...	verzweigt zu Zeilennummer je nach Wert des Ausdruckes		
ON INTERVALL=zeit GOSUB zn	verzweigt zu Unterprogramm, wenn Unterbrechung durch Zeitablauf verursacht wurde		
ON KEY GOSUB [zn][,zn]...	verzweigt zu Unterprogramm, wenn Unterbrechung durch Funktionstaste verursacht wurde		
ON STOP GOSUB zn	verzweigt zu Unterprogramm, wenn Unterbrechung durch die Tasten [CTRL] + [STOP] verursacht wurde		
ON STRIG GOSUB zn	verzweigt zu Unterprogramm, wenn Unterbrechung durch Leertaste oder Aktionstaste eines Steuerhebels verursacht wurde		
PAUSE [n]	n=0,....,65535	unterbricht die Bearbeitung von Programmen, ohne in Kommando-Modus zu gehen, für n*1/50 Sekunden, Abbruch des Programms mit [CTRL] + [STOP], Fortsetzung des Programms mit [STOP]	
PLAY zeichenkette	erzeugt Töne entsprechend der Zeichenkette		
PRINT ausgabeliste	zeigt numerische Daten oder Zeichenketten am Bildschirm an (Textmodus)		

PRINT USING format;ausgabeliste	zeigt Daten in gewähltem Format an
READ var[,var]...	liest aus einer DATA-Anweisung eingegebene Daten
REM [kommentar]	Kommentar, Kurzform: Hochkomma (')
RESTORE [zn]	bestimmt den Anfang für die nächste READ-Anweisung
RESUME	verzweigt aus Fehlerverarbeitungs-routine zurück
RESUME [ 0 ] [ zn ] [NEXT]	
RETURN [zn]	verzweigt aus einem Unterprogramm zurück
SOUND registernummer,ausdruck	erzeugt Toneffekte durch direktes Schreiben von Daten in das SVG-(Speicher Verwaltung Generator)-Register, registernummer=0-13
STOP	unterbricht ein Programm
STOP OFF	macht Unterbrechung durch
STOP ON	[CTRL]+[STOP]
STOP STOP	
STRIG(n) OFF	n=0,...4
STRIG(n) ON	
STRIG(n) STOP	
SWAP var,var	vertauscht die Werte der Variablen
TIME = ausdruck	setzt den internen Zeitgeber
WINDOW [zeile1,zeile2,spalte1,spalte2]	dient dem Einstellen des aktuellen Bildschirmausschnittes auf Textbildschirm

## 5. RBASIC-Funktionen

### Numerische Standardfunktionen

ABS(X)	absoluter Betrag
ATN(X)	Arcustangens von X
CDBL(X)	wandelt X in Zahl mit doppelter Genauigkeit um
CINT(X)	wandelt X in ganze Zahl um
COS(X)	Cosinus von X (X im Bogenmaß)
CSNG(x)	wandelt X in Zahl mit einfacher Genauigkeit um
EXP(X)	Exponentialfunktion, Potenz von e
FIX(X)	ganzer Teil einer Zahl X
INT(X)	ergibt kleinste Ganzzahl, die nicht größer als X ist
LOG(X)	natürlicher Logarithmus von X
RND(X)	Zufallszahl zwischen 0 und 1
SGN(X)	Vorzeichenfunktion
	falls X positiv ---> 1
	falls X 0 ---> 0
	falls X negativ ---> -1
SIN(X)	Sinus von X (X im Bogenmaß)
SQR(X)	Quadratwurzel von X
TAN(X)	Tangens von X (X im Bogenmaß)

### Zeichenkettenfunktionen

ASC(X\$)		übergibt Zeichencode des ersten Zeichens der Zeichenkette X\$
BIN\$(X)		wandelt X in binäre Zeichenkette um
CHR\$(X)		wandelt Zeichencode in Zeichen um
HEX\$(X)		wandelt Zahl X in hexadezimale Zeichenkette um
INSTR([n,]X\$,Y\$)	n=1,....,255	sucht Zeichenkette Y\$ in X\$, bei Position n beginnend und gibt deren Position an
LEFT\$(X\$,n)		übergibt die ersten n Zeichen der Zeichenkette X\$
LEN(X\$)		übergibt die Länge der Zeichenkette
MID\$(X\$,m[,n])	m,n=1,....,255	übergibt n Zeichen aus X\$, an der Stelle m beginnend
OCT\$(X)		wandelt Zahl X in oktale Zeichenkette um
RIGHT\$(X\$,n)	n=0,....,255	übergibt die letzten n Zeichen der Zeichenkette X\$
SPACE\$(n)	n=0,....,255	übergibt n Leerzeichen
STR\$(X)		wandelt Zahl X in Zeichenkette um
STRING\$(n,m)		liefert n-mal Zeichen mit Code m
STRING\$(n,X\$)		liefert n-mal Anfangszeichen von X\$
VAL(X\$)		wandelt Zeichenkette in Zahl um

### Sonstige Funktionen

CSRLIN	ergibt Zeilennummer der aktuellen Kursorposition
ERL	gibt Fehlerzeile im Programm aus

ERR		ergibt Fehlernummer des letzten Fehlers
FRE(X)		ergibt freien Speicher in Bytes
FRE(X\$)		ergibt freien Platz im Zeichenketten speicherbereich
INKEY\$		liest Zeichen von Tastatur (""" - keine Taste gedrückt)
INPUT\$(n)	n=0,....,255	liest n Zeichen von Tastatur (ohne Anzeige am Bildschirm, ohne <u>ENTER</u> )
LPOS(X)		gibt Position des Zeigers im Druckpuffer an
PLAY(n)	n=0,....,3	überprüft, ob über die Musikkanäle (n) eine Tonausgabe erfolgt oder nicht (ergibt -1, wenn ja; 0, wenn nein)
POS(X)		ergibt Spaltennummer der aktuellen Kursorposition
SPC(n)	n=0,....,255	gibt n Leerzeichen aus (kann nur mit PRINT und LPRINT verwendet werden)
STICK(n)	0 Kursortasten n=0   1 Steuerhebel 1   2 Steuerhebel 2	gibt Richtung des Steuerhebels bzw. der Kursortasten an Richtungen: <pre>       1       8   2       7 - 0 - 3       6   4       5 </pre>
STRIG(n)	0 Leertaste   1,3 Aktionstasten n=0   Steuerhebel 1   2,4 Aktionstasten   Steuerhebel 2	ergibt -1, wenn Leertaste oder Aktionstaste gedrückt; 0, wenn nicht gedrückt
TAB(n)	n=0,....,255	setzt den Tabulator (kann nur mit PRINT und LPRINT verwendet werden)
TIME		gibt Stand des Zeitgebers an

## 6. Grafische Anweisungen und Funktionen

Die Eingabe der Koordinaten in den grafischen Anweisungen kann

- absolut (bezogen auf das absolute Koordinatensystem) durch (X,Y)
  - relativ (bezogen auf die letzte Position des Zeichenkursors) durch STEP(X,Y)
- erfolgen.

CALL BICOP	erzeugt eine Hardcopy des visuellen Bildspeicherbereiches auf einen Drucker
CIRCLE [STEP](x,y),radius,[farbe],[anfswinkel],[endswinkel],[ellipse]	zeichnet einen Kreis, eine Ellipse, einen Kreisbogen um den Mittelpunkt [STEP](x,y)
COLOR [hintergrundfarbe],[palette],[randfarbe]	stellt Farben im Grafikmodus ein
DRAW zeichenkette	zeichnet einen Polygonzug entsprechend zeichenkette
LINE [[STEP](x1,y1)-[STEP](x2,y2),[farbe],[B[F]]]	zeichnet eine Linie oder ein Rechteck
PAINT [STEP](x,y),[zeichenfarbe],[grenzf Farbe]	färbt einen von einer Grenzlinie eingeschlossenen Bereich ein
F=POINT [STEP](x,y)	ergibt Farbcode des Punktes (X,Y)
PRESET [STEP](x,y)[,farbe]	löscht Punkt am Grafikbildschirm
PSET [STEP](x,y)[,farbe]	setzt Punkt am Grafikbildschirm
SCREEN [aktivmodus],[aktivseite],[vismodus],[visseite]	dient zum Einstellen des aktiven Bildschirmmodus und der aktiven und visuellen Bildschirmseiten (siehe Abschnitt 2.4, 4.1 und 5.3)

WINDOW (x1,y1)-[STEP](x2,y2)

definiert Koordinatensystem des Grafikbildschirmes  
(x1,y1): Koordinaten links unten  
(x2,y2): Koordinaten rechts oben

### Textausgabe im Grafikmodus

OPEN "GRP:"AS[#]dateinummer eröffnet die Ausgabedatei

PRINT #dateinummer,[USING format;]ausgabedatei  
schreibt Daten in einem bestimmten Format in eine durch OPEN geöffnete Datei

CLOSE [[#]dateinummer][,...] schließt die Ausgabedatei

## 7. Speichern und Laden von Daten

Sequentielle Dateien Nutzung für Kassette und Diskette)

BLOAD "[gerät:]dateiname"[,{R|S}][,versatz]  
lädt angegebenen Speicherbereich vom angegebenen Gerät (S in Video-RAM, nur für Diskette)

BLOAD "[gerät:]dateiname",Z[,versatz]  
lädt den Zeichensatz vom angegebenen Gerät (nur für Diskette)

BSAVE "[gerät:]dateiname",anfadr,endadr[(startadr|S)]  
speichert angegebenen Speicherbereich auf angegebenes Gerät (S aus Video-RAM, nur für Diskette)

BSAVE "[gerät:]dateiname",zeichennr1,zeichennr2,Z  
speichert den Zeichensatz auf angegebenes Gerät (nur für Diskette)



CLOSE [\*][dateinummer][,dateinummer]...  
schließt Datei, die mit OPEN geöffnet wurde

K=EOF (dateinummer) ergibt nach vollständigem Lesen einer Datei -1, sonst 0

INPUT #dateinummer,var[,var]...  
liest Daten von einer durch OPEN geöffneten Datei und ordnet sie den Variablen der Variablenliste zu

X\$=INPUT\$(n,[#]dateinummer) liest n Zeichen von der angegebenen INPUT-Datei in eine Zeichenkettenvariable

LINE INPUT #dateinummer,var liest Daten von einer durch OPEN geöffneten Datei und ordnet sie der Variablen zu, Datenendezeichen ist RETURN

MAXFILES = ausdruck gibt Anzahl von Dateien an, die gleichzeitig in einem Programm eröffnet werden können

OPEN "[gerät:]dateiname" [FOR modus] AS [#]dateinummer [LEN=satzlänge]  
eröffnet Datei  
modus (nur sequentielle Dateien):  
- INPUT sequent. Eingabedatei  
- OUTPUT sequent. Ausgabedatei  
- APPEND sequent. Ausgabedatei fortsetzen  
satzlänge (nur Direktzugriffsdateien)  
- Wert von 1 bis 256

PRINT #dateinummer,ausgabeliste  
schreibt Daten in eine durch OPEN eröffnete Datei

PRINT #dateinummer,USING format;ausgabeliste  
schreibt Daten im angegebenen Format in eine durch OPEN eröffnete Datei

#### Dateien mit direktem Zugriff (nur mit Diskette)

X#=CVD(8-byte-zeichenkette) konvertiert 8-byte-Zk in 8-byte-Zahl

I%=CVI(2-byte-zeichenkette) konvertiert 2-byte-Zk in 2-byte Zahl

A!=CVS(4-byte-zeichenkette) konvertiert 4-byte-Zk in 4-byte-Zahl

FIELD [#]dateinummer,feldbreite AS zeichenkettenvar[,...]  
vereinbart Pufferspeicher für Direktzugriffsdateien

GET [#]dateinummer[,satznummer]  
liest Satz von Direktzugriffsdatei in den Übergabebereich

N=LOC(dateinummer) zeigt die Nummer des zuletzt gelesenen oder geschriebenen Satzes an

N=LOF (dateinummer) zeigt Recordanzahl im aktuellen Datei-bereich (Extent) an

LSET zeichenkettenvar=zeichenkette  
trägt Variable linksbündig ein

X\$=MKD\$(8-Byte-zahl) konvertiert 8-Byte-Zahl in 8-byte-Zk

I\$=MKI\$(2-Byte-zahl) konvertiert 2-Byte-Zahl in 2-byte-Zk

A\$=MKS\$(4-Byte-zahl) konvertiert 4-Byte-Zahl in 4-Byte-Zk

PUT [#]dateinummer[,satznummer]  
schreibt Satz aus dem Übergabebereich auf Direktzugriffsdatei

RSET zeichenkettenvar=zeichenkette  
trägt Variable rechtsbündig ein

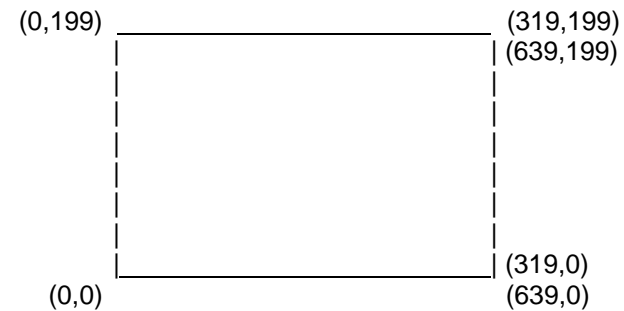
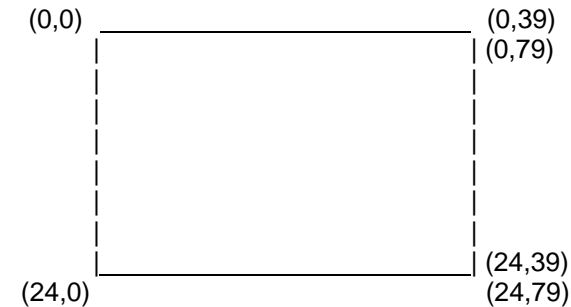
A=VARPTR(#dateinummer) liefert Adresse des File-Control-Blockes der angegebenen Datei

## 8. Hardwareorientierte Anweisungen und Funktionen

DEFUSR [x]=startadresse	legt Startadresse vom x-ten Maschinen-codeprogramm fest
I%=INP(portadresse)	liest Daten eines E/A-Ports
OUT portadresse,ausdruck	schickt ein Byte zum genannten E/A-Port
I%=PEEK(adresse)	liest Inhalt einer Speicheradresse (1 Byte)
POKE adresse,ausdruck	schreibt ein Byte auf die Adresse
Z=USR[x](argument)	liefert Ergebnis des x-ten Maschinencode-programms, das mit einer durch DEFUSR... definierten Adresse beginnt
I%=VARPTR(var)	liest Adresse von Daten, die einer Variablen zugeordnet sind
I%=VDEEK(adresse)	liest Adresse im Bildspeicher (2 Bytes)
VDOKE adresse,ausdruck	schreibt 16-Bit-Wert (0,...,65535) in den Bildspeicher
WAIT portadresse,ausdruck1[,ausdruck2]	wartet auf definierte Signaleingabe

## 9. SCREENs und Farbcodierungen

### SCREENs



Grafik

Farbcode für SCREEN 0,1,8,9 (Text) und SCREEN 5 (Grafik)

0	schwarz	8	dunkelgrau
1	dunkelblau	9	blau
2	dunkelgrün	10	grün
3	grünblau	11	hellblau
4	dunkelrot	12	rot
5	dunkelpurpur	13	purpur
6	ocker	14	gelb
7	grau	15	weiß

Farbpaletten für SCREEN 2 und 3 (Grafik)

Farbe	Palette 0	Palette 1
0	Hintergrundfarbe, (wählbar)	Hintergrundfarbe, (wählbar)
1	dunkelgrün	grünblau
2	dunkelrot	dunkelpurpur
3	ocker	grau