

KLEINCOMPUTER

KC85/3

Ergänzungsbeschreibung

veb mikroelektronik
>wilhelm pieck<
mühlhausen
im veb kombinat mikroelektronik

VEB Mikroelektronik "Wilhelm Pieck"
Mühlhausen

im VEB Kombinat Mikroelektronik

E r g ä n z u n g

zur Dokumentation des Kleincomputers KC85/2 für den
KC85/3

Hierzu gehören:

C0111 BASIC-Interpreter	KC85/2
D001 Grundgerät	KC85/2
Systembeschreibung HC-CAOS	Version 2.1 und 2.2

I n h a l t

1.	Einleitung	2
2.	Änderung	2
3.	Erweiterung des Betriebssystems	2
3.1.	Menüerweiterung	2
3.2.	Erweiterung der Tastenfunktionen	4
3.3.	Erweiterung der Systemprogramme	4
4.	Erweiterung des BASIC-Interpreters	8

1. Einleitung

Der KC85/3 ist eine Weiterentwicklung des KC85/2. Deshalb behält die KC85/2-Dokumentation (D001 Grundgerät, BASIC-Interpreter und die Systembeschreibung HC-CAOS) zusammen mit dieser Ergänzung und unter Berücksichtigung der unter Punkt 2. aufgeführten Änderungen ihre Gültigkeit.

Beim KC85/3 ist der BASIC-Interpreter als Firmware bereits im Computer gespeichert. Der Funktionsumfang sowohl des Betriebssystems HC-CAOS 3.1 wie auch des BASIC-Interpreters wurden erweitert.

2. Änderung der Speicherbelegung

Die in der KC85/2-Systembeschreibung dargestellte Speicherplatzbelegung wurde wie folgt geändert:

Funktionstasten - Speicher von B900H bis B99BH

Fenstervektor - Speicher von B99CH bis B9FFH

BASIC - ROM von C000H bis DFFFH

CAOS - ROM von E000H bis FFFFH

Der BASIC-ROM kann mit dem Kommando SWITCH 2 0 und der CAOS-ROM mit dem Kommando JUMP m abgeschaltet werden.

3. Erweiterungen des Betriebssystems

3.2. Menüerweiterung

Nach dem Einschalten des Computers meldet sich das um die Anweisung BASIC, REBASIC, KEY, KEYLIST und DISPLAY erweiterte Betriebssystem. Die Anweisungen BASIC und REBASIC wirken wie für den KC85/2 beschrieben

KEY n

KEY Funktionstastennummer

Das Menüwort KEY dient der Belegung der Funktionstasten F1 bis F6 und der Zweitbelegung F7 bis FC.

Bei Ausführung der KEY-Anweisung im angegebenen Format befindet sich der Computer im Eingabemodus. Nun wird die einzugebende Tastenfolge für die benannte Funktionstaste definiert. In diesem Eingabemodus können neben den üblichen alphanumerischen und Sonderzeichen auch Zeichen für die auf der Tastatur verfügbaren Editier- und Steuerfunktionen durch Druck auf die jeweilige Steuer- und Editiertaste eingegeben werden. Die einzige Korrekturmöglichkeit besteht in diesem Modus durch Drücken der CLR-Taste.

Dabei rückt der Cursor eine Position nach links, das davorliegende Zeichen wird gelöscht und kann neu eingegeben werden. Der Eingabemodus wird mit der STOP-Taste beendet. Mit dem Drücken der betreffenden Funktionstaste (F1 bis F6) wird nun stets die im KEY-Eingabemodus festgelegte Tastenfolge ausgegeben. bzw. die darin als Steuer- oder Editierzeichen enthaltenen Funktionen ausgeführt.

Beispiel:	Eingabe
	KEY 2 KEY Funktionstastennummer ENTER-Taste als Abschluß
	REBASIC Tastenfolge für Tastenbelegung STOP-Taste als Abschluß

KEYLIST

KEYLIST erfordert keinen Parameter. Durch Aufrufen dieses Menüwortes werden alle Funktionstastenbelegungen aufgelistet.

DISPLAY aaaa eeee (ss)

Die Anweisung DISPLAY bewirkt die Ausgabe des Speicherinhaltes von Adresse aaaa bis Adresse eeee (hexadezimale Angabe der Adressen).

Beispiel:

Eingabe: DISPLAY BA00 BA0F 02

Anzeige: Bildschirm

Speicheradresse	Hexadezimale Angabe	ASCII-Darstellung
BA00	41 42 43 44 45 46 47 48	A B C D E F G H
BA08	49 4A 4B 4C 4D 4E 4F 50	I J K L M N O P

Es werden jeweils 8 Bytes in hexadezimaler Darstellung und als ASCII-Zeichen nebeneinander aufgelistet. Die Anzahl der gleichzeitig auf dem Bildschirm angezeigten Zeilen wird durch den Parameter ss festgelegt.

Wird der Parameter ss nicht angegeben, sind 8 Zeilen für die Bildschirmanzeige vereinbart.

Ist die Anzeige der letzten Speicherzelle eeee nach acht oder ss Zeilen noch nicht erreicht, kann die Anzeige durch Drücken einer beliebigen Taste, mit Ausnahme der BRK- und STOP-Taste, fortgesetzt werden.

Die BRK-Taste unterbricht das Auflisten des Speicherbereiches. Durch die STOP-Taste wird das Unterprogramm MODIFY eingeschaltet, wobei der Speicherinhalt nur in der hexadezimalen Darstellung geändert werden kann. Der ASCII-Teil wird dagegen nicht aktualisiert. Für MODIFY gelten alle Zeichenvereinbarungen wie im Bedienhandbuch D001 beschrieben.

3.2. Erweiterung der Tastaturfunktionen

Ist die Zweitbelegung durch die SHIFTLCK-Taste eingestellt, so erscheint der Cursor zweifarbig.

Mit der Tastenkombination

SHIFT und INS

kann ein Tastenklick ein- bzw. ausgeschaltet werden.

Der Tastencode für diese Funktion ist wie folgt festgelegt:

Taste	Erstbelegung/Code	Zweitbelegung/Code
INS	INS 1AH	CLICK 14H

3.3. Erweiterung der Systemunterprogramme

Außer den in der Systembeschreibung aufgeführten Unterprogrammen können die folgenden Unterprogramme im KC85/3 - System genutzt werden:

Name: MBIN UP-Nr. 37H

FKT.: Byteweise Eingabe von Kassette mit Namensvergleich beim 1. Block

PE: RG A - Daten

D - Steuerbyte Bit 3 = 1 Init (1. Block)
Bit 6 = 1 Close (FF. Block)

HL - Name (11 Bytes) nur bei Init

VR: AF, DE, HL Stack: 25

Bem.: Nach dem Blockweisen Einlesen werden die Daten byteweise dem Puffer entnommen.

Name: MBOT UP-Nr. 38H

FKT.: Byteweise Ausgabe auf Kassette

PE: RG A - Daten

D - Steuerbyte Bit 3 = 1 Init (1. Block)
Bit 6 = 1 Close (FF. Block)

HL - Name (11 Bytes) nur bei Init (Adreßzeiger)

VR: AF, DE, HL

Bem.: Mit den auszugebenden Bytes wird der Kassettenpuffer gefüllt und dann blockweise ausgegeben.

Name: KEY UP-Nr. 39H

FKT.: Belegen einer F-Taste (Aufruf der Menükommandoroutine)
PE: A - Nr. der Taste (1 - 0CH), bei unzulässiger Nr. sofortige Rückkehr
PA: -
VR: AF, BC, DE, HL Stack: 17
Bem.: Dieses Programm fordert Tastatureingaben an.

Name: KEYLI UP-Nr. 3AH

FKT.: Anzeige der Belegung der F-Tasten (Aufruf der Menükommandoroutine "KEYLIST")
PE: -
PA: -
VR: AF, BC, HL Stack: 23

Name: DISP UP-Nr. 3BH

FKT.: HEX-/ASCII-Dump (Aufruf der Menükommandoroutine "DISPLAY")
PE: (ARGN) ≤ 2 Anzahl der Zellen = 8
 > 2 Anzahl der Zellen = C
HL - Anfangsadresse
DE - Endadresse
C - Zeilenzahl
PA: -
VR: AF, BC, DE, HL
Bem.: Taste BRK - Abbruch
 Taste STOP - Modify Stack: 23

Name: WININ UP-Nr. 3CH

FKT.: Initialisierung eines neuen Fensters
PE: A - Fensternummer (0 - 9)
 HL - Fensteranfang
 DE - Fenstergröße
PA: CY = 0 - Fehler (Nr., Anfang der Größe)
VR: AF, BC, DE, HL

Name: WINAK **UP-Nr. 3DH**

FKT.: Aufruf eines Fensters über seine Nummer bei Abspeicherung
des aktuellen Fenstervektors

PE: RG A - Fensternummer (0 - 9)

PA: CY = 0 - falsche Nummer

VR: AF, BC, DE, HL Stack: 3

Bem.:

Name: LINE **UP-Nr. 3EH**

FKT.: Zeichnen einer Linie auf dem Bildschirm von X0/Y0 nach
X1/Y1

PE: (ARG 1) - X0 - X-Koordinate-Anfang

 (ARG 2) - Y0 - Y-Koordinate-Anfang

 (ARG 3) - X1 - X-Koordinate-Ende

 (ARG 4) - X1 - Y-Koordinate-Ende

 (FARB) - Bildpunktfarbe (0 ... 1FH)

PA: -

VR: AF, BC, DE, HL Stack: 8

Name: CIRCL **UP-Nr. 3FH**

FKT.: Zeichnen eines Kreises auf dem Bildschirm mit Mittelpunkt
XM/YM und Radius R

PE: (ARG 1) - XM - X-Koordinate-Mittelpunkt

 (ARG 2) - YM - Y-Koordinate-Mittelpunkt

 (ARG 3) - R - Radius

 (FARB) - Bildpunktfarbe (0 ... 1FH)

PA: -

VR: AF, BC, DE, HL Stack: 10

Name: SQR **UP-Nr. 40H**

FKT.: Berechnen der Quadratwurzel

PE: HL - 16 Bit

PA: -A - Ergebnis 8 Bit

VR: AF, HL, DE Stack: 0

Name: MULT **UP-Nr. 41H**

FKT.: Berechnung des Produktes zweier 8-Bit-Zahlen

PE: D, C - Faktoren (8 Bit)

PA: BA - Produkt (16 Bit)

VR: AF, BC, DE Stack: 0

Name: CSTBT **UP-Nr. 42H**

FKT.: Negation des Bits 4 des Steuerbytes (STBT) des Bildschirmprogrammes (Ausführung der Steuerzeichen/Abbildung der Steuerzeichen)

PE: -

PA: -

VR: - Stack: 1

Bem.: Dieses Programm dient der Umschaltung zur Darstellung der Steuerzeichensymbole auf dem Bildschirm

Name: INIEA **UP-Nr. 43H**

FKT.: Initialisierung eines E/A-Kanals über Tabelle

PE: HL - Anfangsadresse der Tabelle

PA: HL - 1. Byte nach der Tabelle

VR: HL Stack: 0

Bem.: Tabellenaufbau

1. Byte = E/A-Adresse

2. Byte = Anzahl der Initialisierungsbytes (N)

3. Byte =)

.)

.) Initialisierungsbytes

.)

N. Byte =)

Name: INIME **UP-Nr. 44H**

FKT.: Initialisierung mehrerer E/A-Kanäle über Tabelle

PE: HL - Anfangsadresse der Tabelle

D - Anzahl der Kanäle

PA: HL - 1. Byte nach der Tabelle

VR: D, HL, F Stack: 1

Bem.: Die E/A-Tabelle besteht aus (D) Tabellen entsprechend UP-Nr. 43H (INIEA)

Name: ZKOUT **UP-Nr. 45H**

FKT.: Ausgabe einer über Register HL adressierten Zeichenkette

PE: HL - Anfang der Zeichenkette

PA: HL - Ende der Zeichenkette = 0 Byte

VR: HL, AF Stack: 22

Bem.: Die auszugebende Zeichenkette besteht aus ASCII-Zeichen und wird mit 00H abgeschlossen (vgl. UP-Nr. 23 OSTR). Das Programm wird vorrangig bei Programmverteiler V bis VI eingesetzt.

Bsp.: LD HL, TXT

LD E, 45

CALL PV5

.

.

.

TXT DEFB 0CH ;CLS

 DEFB 0A ;CUD

 DEFB '=== Testprogramm ==='

 DEFW 0A0D Newline

 DEFB 0

PV V (Adresse 0F015H)

wie PV3 mit Ein-/Ausschalten des IRMs und Um- bzw. Rückschalten des Stackpointers auf den Systemstackbereich.

PV VI (Adresse 0F01EH)

wie PV3 jedoch UP-Nr. über (ARGC).

Beim Setzen der UP-Nr. muß der IRM ebenfalls eingeschaltet werden!

Name: MENU **UP-Nr. 46H**

FKT.: Ausschrift des aktuellen Menüs und Übergang in die Kommandoeingabe

PE: (IX+9) - Prologbyte

Bem.: Das Programm dient zur Anzeige des aktuellen Menüs bei möglicher Änderung des Prologbytes. Es erfolgt kein Löschen des Bildschirms und keine Generierung der Titelzeile des Systems. Prologbyte des Systems ist 7FH, mögliche andere Anwenderprologbytes können DDH, FDH usw. sein.

4. Erweiterungen des BASIC-Interpreters

Anweisung: RANDOMIZE

Format: RANDOMIZE

Bemerkung: Mit Hilfe der RANDOMIZE-Anweisung wird der Zufallsgenerator mit einem zufälligen Anfangswert gestartet. Wiederholungen von Zufallszahlenfolgen sind damit weitgehend ausgeschlossen.

Anweisung: LINE

Format: LINE XA, YA, XE, YE (F)

Bemerkung: Die Anweisung zeichnet eine Linie auf dem Bildschirm, die im Punkt, mit den Koordinaten XA, YA beginnt und im Punkt, mit den Koordinaten XE, YE endet. Die Farbe der Geraden wird durch den Vordergrundfarbparameter F ($0 \leq F \leq 31$) festgelegt. Entfällt der Parameter F, so erscheint die Linie in der zuletzt gewählten Grafikfarbe.

Beispiel: LINE 10, 100, 300, 100, 26
OK

Anweisung: CIRCLE

Format: CIRCLE XM, YM, R (,F)

Bemerkung: Diese Anweisung zeichnet auf den Bildschirm einen Kreis, unter Angabe der Mittelpunktkoordinaten XM, YM und des Radius R. Die Farbe der Kreislinie wird durch den Vordergrundfarbparameter F ($0 \leq F \leq 31$) festgelegt. Entfällt der Parameter F, so erscheint die Kreislinie in der zuletzt gewählten Grafikfarbe.

Beispiel: 10 CLS
20 FOR R = 1 TO 100 STEP 16
30 CIRCLE 159, 127, R, 16 * RND (1)
40 NEXT

Funktion: CSRLIN

Format: CSRLIN (n)

Bemerkung: CSRLIN (n) liefert als Funktionswert die Nummer der Zeile, in der sich der Cursor befindet.
n = 0 - Zeilennummer bezüglich des maximalen Fensters
n > 0 - Zeilennummer bezüglich des aktuellen Fensters
n < 0 - FC-ERROR

Beispiel: 10 WINDOW 0, 31, 0, 39:CLS
15 PRINTCHR\$ (17)
20 RANDOMIZE
30 FOR I = 1 TO 100
40 INK (16 * RND (1))
50 PRINTAT (32 * RND (1), 38 * RND (1):CSRLIN (0)
60 NEXT
70 INK 7:PRINTCHR\$ (18)

Anweisung: SWITCH

Format: SWITCH mm, kk

Bemerkung: Die BASIC-Anweisung SWITCH ermöglicht das Schalten von Modulen auch innerhalb eines Programms. In folgenden Angaben unterscheidet sich die SWITCH BASIC-Anweisung von der SWITCH Systemanweisung:

	<u>Systemanweisung</u>	<u>BASIC-Anweisung</u>
SWITCH	<ul style="list-style-type: none"> - (%) Promptzeichen des Systems - Eingabe des Parameters m - Status lesen u. schalten - Durch Leerzeichen getrennte Eingaben - Modulsteckplatzadresse Angabe hexadezimal 	<ul style="list-style-type: none"> - (>) Promptzeichen von BASIC - Eingabepflicht der Parameter m und k - Nur schalten - Parameter durch Komma getrennt - Modulsteckplatzadresse Angabe dezimal

Beispiele: Schalten des V24-Moduls

```
SWITCH C 1
0C EE 01
```

```
SWITCH 12,1
```

Drucken des Namens Emil

```
10 SWITCH 12,1
```

```
Zuweisen V24-Modul
```

```
20 PRINT # 2 "EMIL"
```

```
Druckanweisung
```

```
30 SWITCH 12,0
```

```
Abschalten V24-Modul
```

Anweisung: OPEN

Format: OPEN r # n "NAME"

Bemerkung: OPEN eröffnet eine Kanaloperation mit Namensübergabe auf dem Kanal n für dateifähige Geräte (z.B. Floppy) oder Geräte mit Namensverwaltung. Der Parameter r legt fest, ob es sich um eine Aus- oder Eingabe handelt:
r = 1 Eingabe
r = 0 Ausgabe

Anweisung: CLOSE

Format: CLOSE r # n

Bemerkung: CLOSE schließt den Kanal n (nach Dateneingabe r = 1 oder nach Datenausgabe r = 0). CLOSE führt der BASIC-Interpreter automatisch bei Programmende oder -abbruch (Taste oder Error) aus.

Beispiel: siehe Anweisung PRINT #, INPUT #

Anweisungen: PRINT #, INPUT #

Format: PRINT # n Daten
INPUT # n Daten

Bemerkung: PRINT # ermöglicht die Eingabe von Daten auf einen durch n definierten Kanal.
INPUT # ermöglicht die Eingabe der mit PRINT # ausgegebenen Daten von einem durch n wählbaren Peripheriegerät.
Beide Anweisungen sind nur in Verbindung mit OPEN und CLOSE (Übergabe eines Dateinamens) zu verwenden.
Ein Lesen der mit PRINT # ausgegebenen Daten ist nur durch die Anweisung OPEN und CLOSE möglich.
Sollen Dateien jedoch nur ausgegeben werden, (z.B. auf Drucker), können sie auch mit der Anweisung PRINT # ohne OPEN und CLOSE ausgegeben werden.

Beispiel: 10 CLS: DIM A\$ (4)
20 FOR I = 0 TO 4: INPUT A\$ (1): NEXT
30 PRINT "RECORDER AUF AUFNAHME STELLEN"
40 OPEN 0 # 1 "BEISPIEL"
50 FOR I = 0 TO 4: PRINT # 1 A\$ (I): NEXT
60 CLOSE 0 # 1

10 CLS: DIM A\$ (4)
20 PRINT "RECORDER AUF AUFNAHME STELLEN"
30 OPEN I # 1 "BEISPIEL"
40 FOR I = 0 TO 4: INPUT # 1 A\$ (I): NEXT
50 CLOSE I # 1

Anweisung: LIST #, LOAD #

Format: LIST # n "Programmname"
LOAD # n "Programmname"

Bemerkung: LIST # ermöglicht die Ausgabe eines Programmlistings (ASCII-Zeichen für ASCII-Zeichen) auf ein durch n wählbares Peripheriegerät. Mit LOAD # können die mit LIST # ausgegebenen Programme wieder eingelesen werden. Durch n wird wiederum das Peripheriegerät festgelegt.

Beispiele: LIST # 2 "KALKU" (Ausgabe des Programmlistings KALKU auf Drucker)
LOAD # 1 "TEST" (Einlesen des mit LIST # 1 TEST" auf Magnetband ausgegebenen Programmlistings)

Peripheriecodierung n

für die Anweisungen
OPEN, CLOSE, PRINT #, INPUT #, LIST # und LOAD #

n Peripherie

0 Bildschirm und Tastatur (Angabe # 0 sowie OPEN
und CLOSE können
entfallen)

1 Kassettenrecorder

2) frei für weitere Anwendungen (Drucker, Plot-
ter, Floppy, Lochstreifeneinheit usw.)

3)

Einbinden der Treiberrountinen der Systembeschrei-
bung entnehmen.

Funktion: PTEST

Format: PTEST (X)

Bemerkung: Die Funktion testet, ob ein Bildpunkt gesetzt ist,
d.h., ob die eingestellte Vordergrundfarbe auf dem
Bildschirm erscheint. Die X-Koordinate des zu
testenden Punktes wird als Parameter in Klammern
hinter die Funktion geschrieben.
Als Y-Koordinate benutzt der Computer die zuletzt
genannte Y-Koordinate einer Grafikanweisung. Ist
der Bildpunkt gesetzt, liefert die Funktion als
Ergebnis den Wert 1, andernfalls den Wert 0.

Beispiel: 40 PRESET 0,56
50 A = PTEST (70)
60 PRINT A

Funktion: VGET\$

Format: String = VGET\$

Bemerkung: Die Funktion VGET\$ liefert den Inhalt der Cursor-
position als String. (1 Zeichen)

Beispiel: 10 CLS
20 PRINT "TEXT"
30 LOCATE 0,1:A\$ = VGET\$
40 PRINT AT (10,10); A\$
50 LOCATE 1,0

Funktion: INSTR

Format: INSTR (String, String)

Bemerkung: Die Funktion INSTR (A\$, B\$) ermittelt die Posi-
tion, ab welcher die Zeichenkette A\$ in der Zei-
chenkette B\$ enthalten ist.

Beispiel: 10 A\$ = "PFERDE":B\$ = "BLUMENTOPFERDE"
20 PRINT INSTR (A\$, B\$)