

```

;***** Tastaturreoutine *****
;* Version
;* Programmierer: M. Herbote --- K EAW ZFT (WFT)
;* V 2 . 1
;* Stand : Ø6.Maerz 1985
;***** symbolische Adressen und ihre Bedeutung *****
;* symbol.adr. ! Bedeutung
;-----!
;* TA.TA1      ! Abfrage nach gedrueckter Taste
;* TA.TA2      ! Einlesen des Tastaturcodes
;* TA.PIOPR    ! programmieren des Tastatur-PIO
;* TA.EØIN     ! Anfangswerte fuer EMERK , UMERK und KGMER
;* TA.LOE      ! loeschen aller Tastenroutineadressen
;* TA.LOES     ! wie TA.LOE mit Belegung Standardroutinen
;* TA.ABFR     ! Tastaturreabfrage und Ausfuehrung Tastenfunktion
;* TA.TFE      ! Werteschleife bis keine Taste mehr gedrueckt
;-----!
;* Tabellen mit den Funktionsadressen der einzelnen Tasten
;* TS.TAB1     ! Funktionstasten PØ1 bis P48
;* TS.TAB2     ! INS MOD , DEL , INS LINE , DEL LINE ,
;*             ! SCH LINKS , C HOCH , SCH RECHTS ,
;*             ! C LINKS , HOME , C RECHTS ,
;*             ! CNZA      , C RUNTER
;* TS.TAB3     ! ØØ , ØØØ , NEG QUIT , SYS , END , ABS
;* TS.TAB4     ! POS QUIT , TABL , TABL , CI , M
;* TS.EINA     ! Auswerten der alphanumerischen Tasten
;* TS.TABU     ! Auswerten der Taste ET2 == HT
;* TS.NEWL     ! Auswerten der Taste ET1 == RS
;* TS.TAB5     ! Ebene Ø , Ebene 1 , Ebene 2 , Ebenen 3
;-----!
;* TS.PTAST    ! I/O-Adresse der Tastatur
;* TS.TK       ! Zwischenspeicher fuer den Tastaturcode
;* TS.PTK      ! Zwischenspeicher fuer die P-Tastennummer
;* TS.EMERK    ! Merker fuer die aktive P-Tastenebene
;* TS.SPMER    ! Tastaturgruppenmerker
;*             ! Bit ! Hexacode ! Funktion
;-----!
;*             ! Ø   ! ØØ-1F      ! Steuer- / Sonder-zeichen
;*             ! 1   ! 2Ø,2C      !
;*             ! 2   ! 2B,2D,2E    ! +-.
;*             ! 3   ! 3Ø-39      ! Ø123456789
;*             ! 4   !           ! !"%"&'()*/:;^=^?ÅÜÜþ_ääü`ß
;*             ! 5   ! 4Ø-5A,23,24 ! A...Z$# Grossbuchstaben
;*             ! 6   ! -61-7A      ! a...z Kleinbuchstaben
;*             ! 7   ! 8Ø-FF      ! Funktionstasten
;* TS.UMERK    ! =Ø INS MOD aus =1 INS MOD ein
;* TS.KGMER    ! =Ø Programmier- =1 Schreibmaschinen- Tastatur
;* TS.ZSMER    ! =Ø Schnellschreiben nicht erlaubt
;* TS.ZEIT1    ! Zeitkonstante 1 fuer die Warteschleife
;* TS.ZEIT2    ! Zeitkonstante 2 fuer die Warteschleife
;* TS.MIO      ! =Ø I/O -      =1 memory - Bereitstellung
;* TS.MBF      ! =Ø kein      =1 Abspeichern als Bedienfolge
;* TS.BFOL1    ! Pointer Bedienfolgespeicher bei Aufbau
;* TS.BFOL2    ! Pointer Bedienfolgespeicher bei Abarbeitung
;* TS.STACK    ! Zeiger zum aktuellen Keller
;*****
```

```
*****
*          Monitorroutine
* Version
* Programmierer: M. Herbote --- K EAW ZFT (WFT)
* V 2 . 1
* Stand      : Ø5.April 1984
*****
```

1. Aufteilung des Monitorbildes

	1	a	i	k	b	m
*	1	!A	!	SSA LSA	!	!
*	c		AF	SSAF LSAF	!	!
*	j	SZA	!SZAFA	S	!	!
*	l	!LZA	!LZAF	L	!	!
*	d		!		!	!
*	n	!	!		!	!

2. Steuervektor Monitorroutine

ZPOL:	DB	Ø	; Zeilennr. Leseposition
SPOL:	DB	Ø	; Spaltennr. Leseposition
ZPOS:	DB	Ø	; Zeilennr. Schreibposition
SPOS:	DB	Ø	; Spaltennr. Schreibposition
ZMIN:	DB	Ø	; Merker Anfangszeile
SMIN:	DB	Ø	; Merker Anfangsspalte
ZMAX:	DB	Ø	; Merker Endzeile
SMAX:	DB	Ø	; Merker Endspalte
ZANZ:	DB	Ø	; Merker Zeilenanzahl BWS
SPANZ:	DB	-Ø	; Merker Spaltenanzahl BWS
BWSA:	DA	Ø	; Anf.adr. Bildwiederholspeicher
DIREL:	DB	Ø	; Kommando forts. Lesepos.
DIRES:	DB	Ø	; Kommando forts. Schreibpos.
CMD:	DB	Ø	; Routinenkommando
TK:	DB	Ø	; zu schreibendes bzw. ; zu lesendes Zeichen
RADR:	DA	Ø	; Benutzeradresse Rollfunktion
;			
MCZ:	DB	Ø	; Zeichenmerker Cursorposition
MCUR:	DB	Ø	; Merker zum Cursor ; MCUR = Ø Cursor aus
;			; MCUR = 1 Cursor ein und gesetzt ; MCUR = 2 Cursor ein und Zeichen
CZEIT:	DB	Ø	; Zeitvariable Cursorblitzen

 3.Kommandos und ihre Bedeutung

- * = Ø ZANZ := 24, SPANZ := 8Ø
- * = 1 ZMIN, SMIN := 1, ZMAX, SMAX := ZANZ, SPANZ
- * = 2 ZMIN, SMIN := 1, ZMAX := ZANZ - 2, SMAX := SPANZ
- * = 3 ZMIN, ZMAX := SYSZ, SMIN := 3, SMAX := 16
- * = 4 gesamtes Bild loeschen
- * = 5 Bildfenster loeschen
- * = 6 Cursor setzen
- * = 7 Cursor loeschen

Schreibposition	Leseposition
= 1Ø ZPOS, SPOS := HL	= 3Ø ZPOL, SPOL := HL
= 11 HL := ZPOS, SPOS	= 31 HL := ZPOL, SPOL
= 12 HL := SZA	= 32 HL := LZA
= 13 HL := SZAF	= 33 HL := LZAF
= 14 HL := SSA	= 34 HL := LSA
= 15 HL := SSAF	= 35 HL := LSAF
= 2Ø Zeichen schreiben	= 4Ø Zeichen lesen
= 21 akt. Position + 1	= 41 akt. Position + 1
= 22 HOME-Fkt. Schreibpos.	= 42 HOME-Fkt. Lesepos.
= 23 n. Zeile Anf. "	= 43 n. Zeile Anf. "
= 24 Vektor schreiben	= 44 Vektor lesen
= 5Ø HL := AF	
= 51 BC := Laenge Bildspeicher	
= 52 HL := Ende Bildspeicher	
= 53 Bildfensterinhalt eine Zeile nach oben	
= 54 Bildfensterinhalt eine Zeile nach unten	
= 55 Bildfensterinhalt eine Spalte nach links	
= 56 Bildfensterinhalt eine Spalte nach unten	

 4.Bedeutung von DIRES und DIREL

- * Bit 7 = Ø Schreib-/Lese-position nicht weiter setzen
- * = 1 Schreib-/Lese-position weiter setzen
- * 6 = Ø Schreib-/Lese-position umlaufend auf Zeile
- * = 1 Schreib-/Lese-position umlaufend im Bild
- * 5 = Ø Rollfunktion ausgeschaltet (nur DIRES)
- * = 1 Rollfunktion eingeschaltet (nur DIRES)
- * 4 = Ø Bildfensterinhalt rollen (nur DIRES)
- * = 1 gesamte Zeile rollen (nur DIRES)
- * 3 = Ø mit Umladen BWSA ==³ BWS
- * = 1 ohne Umladen BWSA ==³ BWS

- * 2,1,Ø geben die Richtung an, in welcher die Schreib-/Lese-position weitergesetzt werden soll

Ø1Ø
 (Ø11) (ØØ1)
 1ØØ ØØØ
 (1Ø1) (111)
 11Ø

```
*****
*          BAMOS --- Kassettenmagnetband *
* OS - Version -----
* Programmierer: Hildebrandt --- RVB *
* V 1.2   -----
* Stand    : 01.Januar 1981 *
*****
```

Geraetebedienroutine fuer Kassettenmagnetband
 Anschluss von zwei Laufwerken K5200
 an einer AKB K5020

* E/A-Tabelle Byte 0 Status
 * Byte 1 Fehlerschlüssel
 * Byte 2 Basisadresse der AKB
 * Byte 3 Subadresse
 * Byte 4 Kommando
 * Byte 5 L-Teil Adresse Eintrittspunkt
 * Byte 6 H-Teil Adresse Eintrittspunkt
 * Byte 7 L-Teil Blockadresse
 * Byte 8 H-Teil Blockadresse
 * Byte 9 H-Teil Blocklaenge (Bit 15...Bit 8)
 * Byte 10 L-Teil Blocklaenge (Bit 7...Bit 0)
 * n bei suchen Bandmarke
 * Byte 11 Anzahl Lese-/Schreib wiederholen
 * Byte 12 Anzahl Blocklueckenverlaengerungen

* Status Bit 0 Ø Geraet frei
 * 1 Geraet besetzt
 * Bit 1 Ø
 * 1 Datentraegerende
 * Bit 2 Ø Aufzeichnen erlaubt
 * 1 Aufzeichnen verboten
 * Bit 3 Ø Kassettenseite A
 * 1 Kassettenseite B
 * Bit 4 Ø Geraet bereit
 * 1 Geraet nicht bereit
 * Bit 5 Ø Geraet nicht reserviert
 * 1 Geraet reserviert
 * Bit 6 Ø Aufzeichnungsverf. 38 CM*S**-1
 * 1 Aufzeichnungsverf. 19 CM*S**-1
 * Bit 7 Ø Kommando fehlerfrei ausgefuehrt
 * 1 Fehler (siehe Fehlerschlüssel)

* Kommandos ØØ Aufzeichnen mit RAW (2=Blockl. =256)
 * Ø8 Aufzeichnen ohne RAW
 * Ø2 Wiedergabe (es wird der naechste Block
 * gelesen, Adresse und Laenge werden in
 * der E/A-Tabelle uebermittelt
 * 11 einen Block vorsetzen
 * 15 einen Block rücksetzen
 * 21 umspulen
 * 31 reservieren ein
 * 41 reservieren aus
 * 51 schreiben einer Bandmarke (Steuerblock)*
 * 61 schreiben Schlussluecke
 * 71 suche n-te Bandmarke vorwaerts
 * 75 suche n-te Bandmarke rückwaerts

```
*****
* Fehler-Code      10 falsches Kommando
*                  11 angewaehltes Geraet nicht reserviert
*                  12 Subadresse falsch
*                  13 Pufferl. kleiner 12 oder groesser 256
*                  14 Ende der Aufzeichnungen auf dieser
*                     Kassettenseite (Lesen)
*                  15 angewaehltes Geraet besetzt
*                  16 Ende/Anfang der Aufzeichnungen
*                     (Bandmarke suchen)
*                  17 gelesener Block laenger 260 Byte
*                  18 Bandmarke nicht gefunden
*                  19 Aufzeichnen verboten
*                     (mit Kommando: Aufzeichnen)
*                  21 Fehler nach festgelegter Anzahl
*                     Lesewiederholungen
*                  22 Fehler nach festgelegter Anzahl
*                     Schreibwiederholungen
*                  23 kein Echosignal vom Eingabekanal
*                     bei RAW
*****
```

Hinweise zur Arbeit mit der Geraetebedienr.

- IM2 einstellen , SP und I-Register laden
- E/A-Tabelle laden (Basisadresse AKB)
- Adresse E/A-Tabelle in IX laden
- programmieren AKB durch Aufruf des UP's
KR.INIT (CALL KR.INIT)
- laden Schluessel fuer Kommando , Subadresse
Adresse Eintrittspunkt
- Aufruf Geraetebedienroutine immer mit
CALL KR.BRKMB
- wird keine Parallelarbeit mit anderen
Programmen gewuenscht , kann wie in
folgendem Beispiel verfahren werden:

```
...
CALL KR.BRKMB ;Aufruf
MARKE:HALT
        JR  MARKE-#
ADREP:..... ;Adresse Eintrittspunkt
...
```

- bei Rueckkehr der Bedienroutine zum Ein-
trittspunkt ist die Adresse des folgenden
Befehls , bei dem der letzte Interrupt
angenommen wurde , noch im SP .
Wird , wie im obigen Beispiel , diese
Adresse nicht mehr benoetigt , so ist der
SP um 2 zu erhoehen.

```
*****
*          Basisroutinen Kassette
* Version -----
* Programmierer: M. Herbote --- K EAW ZFT (WFT)
* V 2 . 1   -----
* Stand      : 10.April 1984
*****
```

* Die unterste Schicht wird von der ROBOTRON - OEM-Routine
* gebildet. Sie enthaelt die Routinen:
* INIT ... Initialisieren der AKB - PIO s und
* BRKMB ... Ausfuehrung eines Kassettenkommandos

* Nachfolgende Tabelle zeigt die implementierten Basis-
* routinen und ihre Kommandos.

* symb. Adr. OEM-Komm. SYS-Komm.

GNRE	31H	KiRE	reservieren ein
GNRA	41H		reservieren aus
GNRW	21H	KiRA	enthaelt GNRE,GNRW,GNRA
GNRP		KiRW	rewind
GNRM		KiR+	einen Block vor
GNLP	71H	KiR-	einen Block zurueck
GNLPN		KiL+	eine Datei vor
GNLM	75H	KiL-	um (D) Dateien vor
GNLMN			eine Datei zurueck
GNCLS	51,61,15,11H	KiCL	um (D) Dateien zurueck
			Abschluss einer Ausg.datei
GNRD	Ø2H		einen Block == ⁵ KS.PUFFS
GNRDR		KiXA,-XE	(K2.TXANF-K2.TXEND) fuellen
EIN			SØ.PFFI fuellen
EINP			einen Block == ⁵ SØ.PFFI
LESV			Eingabe eines Zeichens vw.
LESR			Eingabe eines Zeichens rw.
GNWR	ØØH		einen Block ausgeben
GNWRT		KiY	(K2.TXANF-K2.TXEND) ausgeben
AUS			SØ.PFFA ausgeben
SCHR			Ausgabe eines Zeichens
GNRWR	Ø8H		einen Block zurueckschr.
MRCAT			Zugriff auf MRES-Katalog
MREPO		KiME	MRES-Eingabedatei eroeffnen
MRAPO		KiMA	MRES-Ausgabedatei eroeffnen
MRCLS		KiMC	MRES-Ausgabedatei abschl.
