

UP -Symbol	Adresse		Bemerkung
ADRES	2 FBB		
ADRHI	2 FBC		
ADRSET	0040		Adresse laden
ASCBUF	2 FDA		ASCII-Uhrzeit- Buffer
BRKPT 1	2 FD3		Break-Adr. # 1
BRKPT 2	2 FD6		Break-Adr. # 2
CASBDR	0016		CAS- Baudrate
CASIN	008E		CAS- Eingabe
CASOT	008B		CAS- Ausgabe
CASMOD	005F		
CCKEY	0073		CMD - Input
CI	0043		Consol In (ASCII)
CIFLAG	2 FCC		Flag für CI-Routine
CNVBYT	0067		SSG- Umsetzung
CO	0049		Console Out (Vid.; ESCAPE!)
COPY	0088		RAM kopieren
CSINIT	0055		CAS-IF init.
CSTS	0052		Console Status
DATA	2 FBA		Daten
DATSET	0076		Daten-Mode einstellen
DELY 1	000B		1 ms
DIG 1	2 FB3	1 H	Digit 1
:	:		:
DIG 7	2 FB9	1 H	Digit 7
DLY 100	0013		100 ms (mit Anzeige)
DSP ACC	0097		ACC darstellen <u>10550-Anzeige</u>
DSPBUF	2FB2		LED-Zeile
DSPERR	009D		ERR-Anzeige
DSPH&L	009A		H&L darstellen
DSTBEG	2FC3		Par. 3
DSTENS	2FC5		Par. 4
EXCHAN	0091		COPY mit Adreßumrechnung
EZUDSP	0094		EZU in SSG - Anzeige
FCT AB	00A0		FCT - Sprungverteiler
FCTO	00A0	3 H	FCTO: CAS - Ausgabe
:	:		:
FCTE	00CA	3, H	FCTE: Single Step

UP - Symbol	Adresse		Bemerkung
FCTBUF	2 FBE		FCT-Nr.
FILL	0085		Konst. ins RAM
FQUARZ	0006		Quarz Frequenz in MHz
HBUF	2FCF		
HBUF 2	2FD1		
HHKEY	0070		HEX - Input
HIBUF	2 FCD		I/O - Hilfsroutine
HIBUF 2	2 FDO		CTE - Sprungziel
INTRAP	2 F8C		
INT 3	2 F8F	3 H	
INT 75	2 FAH	3 H	
IOFLG	2 FC8		I/O - Parameter
KALT	0000		Kaltstart
KEYBYT	2 FCB		KEY true ./.. invers
LO	004F		Lister Out (Drucker)
LOINIT	005B		PRT - IF init.
MAXBUF	2 FCA		max. Anz.
MAXRAM	005B		Memory Top
MOOFLG	2 FBD		Mode - Flag
NXTBUF	2 FC9		FCT - Eingaben
ONESEC	001B		1s (Mit Anzeige)
OPCOD 1	2FD5		OpCode - Buf. # 1
OPCOD 2	2FD8		OpCode - Buf. # 2
PO	004C		Puncher Out (Cass)
PRESS	006D		Drücken abwarten
PRTBDR	000E		PRT - Baudrate
PRTMOD	005E		
READUR	007C		EZU lesen
REGA	2FA7	1 A	Register A
⋮	⋮	⋮	⋮
REGS	2FAF	1 H	Stack Pointer (HI)
REGP	2FB0	1 H	Stack Pointer (LO)
REGNR	2FC7		Reg.-Nr.
RELES	006A		loslassen abwarten
RESTOR	0082		Reg.laden; Start
RI	0046		Raeder in (Cass.)
RST1	0008		Braeckpoint setzen
RST2	0010		Braeckpoint löschen
RST3	0018		
RST75	003C		

UP - Symbol	Adresse	Bemerkung
SAVE	007F	Reg. retten
SRCBEG	2FBF	Par. 1
SRCEND	2FC1	Par. 2
TIME	0061	Uhrzeit / Datum - String
TIMPRL	0007	KEY - Prellzeit in ms
TRAP	0024	für FDC
UHRBUF	2FDA	BCD-Uhrzeit-Buffer
V24BDR	001E	V.24 Baudrate
V24INI	0064	V.24 init. (Adresse !)
V24110D	0060	
WARM	0003	Warmstart
WRITUR	0079	EZU stellen

```

0001 0000 *****
0002 0000 *
0003 0000 * PROMO Basis-Symboltabelle 24.10.85 *
0004 0000 *
0005 0000 *****
0006 0000 *
0007 0000 * Gilt für alle übrigen PROMO-Programmteile: HEXMO,
0008 0000 * STP/RAM, EZU/DSP sowie Peripherie-Verwaltung.
0009 0000 * Zum Assemblieren eines dieser Programmteile wie
0010 0000 * folgt vorgehen:
0011 0000 *
0012 0000 * 1. Programmteil nach 9000 ff. ins RAM laden, mit
0013 0000 * Editor bearbeiten und auf Diskette abspeichern.
0014 0000 * 2. Basis-Symboltabelle nach 9000 ff. ins RAM laden
0015 0000 * und Assembler-Durchlauf 1 und 2 starten.
0016 0000 * 3. Quell-Listing von 1. nach 9000 ff. ins RAM
0017 0000 * laden und Assembler-Durchläufe 0 und 2
0018 0000 * starten (an vorhandene Symboltabelle anhängen).
0019 0000 * 4. Bei Bedarf Assembler-Durchlauf 3 starten und/
0020 0000 * oder Objektcode aus F000 ff. ins EPROM laden.
0021 0000 *
0022 0000 LEDPOR EQU 0H *LED Port
0023 0000 PRMLED EQU 0AH *PRM-LEDs
0024 0000 PRMCTL EQU 0BH *PRM Control
0025 0000 EZUDAT EQU 18H *EZU-Daten
0026 0000 EZUCTL EQU 19H *CTL EZU
0027 0000 STPPF EQU 1AH *STP-Flipflop
0028 0000 PRMOT EQU 38H *Thermo-Motor
0029 0000 DATREG EQU 4BH *FDC-Datenregister
0030 0000 FDCTRL EQU 40H *FDC/AdresseBlatch
0031 0000 OUTPOR EQU 80H *Digit 7 (+10h)
0032 0000 SERMOD EQU 0A9H *Mode-Wort SER-IF
0033 0000 SERCTL EQU 0E9H *SIO-CTL
0034 0000 SERDAT EQU 0F9H *SIO-Daten
0035 0000 TIMCTL EQU 0D9H *Timer-CTL
0036 0000 TIMDAT EQU 89H *Timer-Daten
0037 0000 CENSTB EQU 0BAH *Centronics-CTL
0038 0000 CENDAT EQU 0CAH *Centronics-Daten
0039 0000 BELOUT EQU 0B9H *Schnarre
0040 0000 PARPOA EQU 0A8H *PIO-Port A
0041 0000 PARPOC EQU 0C8H *PIO-Port C
0042 0000 PARCTL EQU 0D8H *PIO-CTL
0043 0000 HEXLO EQU 0ECH *HEX 0...7
0044 0000 HEXHI EQU 0DCH *HEX 8...F
0045 0000 COMAND EQU 0BCH *CMD-Keys
0046 0000 *
0047 0000 NXTNR EQU 0H *NXT-Spalte
0048 0000 FCTNR EQU 2H *FCT-Spalte
0049 0000 ADRNR EQU 4H *ADR-Spalte
0050 0000 RUNR EQU 6H *RUN-Spalte
0051 0000 DIODNR EQU 7H *Ge-Diode
0052 0000 *
0053 0000 RST50N EQU 1EH *8085
0054 0000 RST60N EQU 1DH
0055 0000 RST70N EQU 1BH
0056 0000 RSTOFF EQU 17H
0057 0000 *
0058 0000 ROMEXT EQU 1000H *CPU-ROM (upper)
0059 0000 RAMBEG EQU 2800H *CPU-RAM
0060 0000 RAMEND EQU 2E00H *Ende Anwender-Bereich
0061 0000 STACK EQU 2F00H *entspr. MOVID
0062 0000 USRSTK EQU 2F70H *Anwender-Stack
0063 0000 *

```

0064	0000			*				
0065	0000			ORG	OH			*CPU-Platz #0
0066	0000			*				
0067	0000			*				
0068	0000	C3	D0	00	KALT:	JMP	0D0H	*Kaltstart
0069	0003	C3	F7	00	WARM:	JMP	0F7H	*Warmstart
0070	0006	06			FQUARZ:	DB	6H	*Quarzfrequenz in MHz
0071	0007	04			TIMPRL:	DB	4H	*Key-Prellzeit in ms
0072	0008	C3	78	05	RST1:	JMP	578H	*Breakpoint setzen
0073	000B	C3	AF	04	DELY1:	JMP	4AFH	*1 ms
0074	000E	00	48		PRTBDR:	DW	4800H	*PRT-Baudrate
0075	0010	C3	1B	06	RST2:	JMP	61BH	*Breakpoint löschen
0076	0013	C3	8B	04	DLY100:	JMP	48BH	*100 ms (mit Anzeige)
0077	0016	00	12		CASBDR:	DW	1200H	*CAS-Baudrate
0078	0018	C3	8F	2F	RST3:	JMP	INT3	
0079	001B	C3	7F	04	ONESEC:	JMP	47FH	*1 s (mit Anzeige)
0080	001E	00	48		V24BDR:	DW	4800H	*V.24-Baudrate
0081	0020				*			
0082	0020	C3	92	2F	RST4:	JMP	INT4	
0083	0023	00				DB	0H	
0084	0024	E9			TRAP:	PCHL		*für FDC
0085	0025	C3	8C	2F		JMP	INTRAP	
0086	0028	C3	95	2F	RST5:	JMP	INT5	
0087	002B	00				DB	0H	
0088	002C	C3	98	2F	RST55:	JMP	INT55	
0089	002F	00				DB	0H	
0090	0030	C3	9B	2F	RST6:	JMP	INT6	
0091	0033	00				DB	0H	
0092	0034	C3	9E	2F	RST65:	JMP	INT65	
0093	0037	00				DB	0H	
0094	0038	C3	A1	2F	RST7:	JMP	INT7	
0095	003B	00				DB	0H	
0096	003C	C3	A4	2F	RST75:	JMP	INT75	
0097	003F	55				DB	55H	*HEX-Monitor Ver. 5.5
0098	0040				*			
0099	0040	C3	70	03	ADRSET:	JMP	370H	*Adresse laden
0100	0043	C3	09	10	CI:	JMP	1009H	*Console In (ASCII)
0101	0046	C3	CF	0C	RI:	JMP	0CCFH	*Reader In (Cass.)
0102	0049	C3	B0	0E	CO:	JMP	0EBOH	*Console Out (Vid.;ESCAPE!)
0103	004C	C3	E8	0C	PO:	JMP	0CE8H	*Puncher Out (Casa s.)
0104	004F	C3	00	0D	LO:	JMP	0D00H	*Lister Out (Drucker)
0105	0052	C3	0C	10	CSTS:	JMP	100CH	*Console Status
0106	0055	C3	00	0B	CSINIT:	JMP	0B00H	*CAS-IF init.
0107	0058	C3	0E	0B	LOINIT:	JMP	0B0EH	*PRT-IF init.
0108	005B	C3	D3	07	MAXRAM:	JMP	7D3H	*Memory Top
0109	005E	14			PRTMOD:	DB	14H	*8 Bits, je 1 Start u.Stop
0110	005F	14			CASMOD:	DB	14H	
0111	0060	14			V24MOD:	DB	14H	
0112	0061	C3	27	09	TIME:	JMP	927H	*Uhrzeit/Datum-String
0113	0064	32	0B		V24INI:	DW	0B32H	*V24-Init. (Adresse!)
0114	0066	E9				PCHL		*NSC800-TRAP
0115	0067	C3	96	02	CNVBYT:	JMP	296H	*SSG-Umsetzung
0116	006A	C3	4E	04	RELES:	JMP	44EH	*Loslassen abwarten
0117	006D	C3	69	04	PRESS:	JMP	469H	*Drücken abwarten
0118	0070	C3	83	01	HHKEY:	JMP	183H	*HEX-Input
0119	0073	C3	2F	03	CKKEY:	JMP	32FH	*CMD-Input
0120	0076	C3	60	03	DATSET:	JMP	360H	*Daten-Mode einstellen
0121	0079	C3	12	08	WRITUR:	JMP	812H	*EZU stellen
0122	007C	C3	73	08	READUR:	JMP	873H	*EZU lesen
0123	007F	C3	4F	01	SAVE:	JMP	14FH	*Regs retten
0124	0082	C3	21	04	RESTOR:	JMP	421H	*Regs laden, Start
0125	0085	C3	BF	07	FILL:	JMP	7BFH	*Konst. ins RAM
0126	0088	C3	56	07	COPY:	JMP	756H	*RAM kopieren
0127	008B	C3	BC	0B	CASOT:	JMP	0BBCH	*CAS-Ausgabe

```

0128 008E C3 12 0C CASIN: JMP 0C12H *CAS-Eingabe
0129 0091 C3 9C 06 EXCHAN:JMP 69CH *COPY mit Adreßumrechnung
0130 0094 C3 F8 08 EZUDSP:JMP 8F8H *EZU in SSG-Anzeige
0131 0097 C3 9E 09 DSPACC:JMP 99EH *(ACC) darstellen
0132 009A C3 B8 09 DSPH&L:JMP 9B8H *(H&L) darstellen
0133 009D C3 CE 09 DSPERR:JMP 9CEH *Err-Anzeige
0134 00A0 *
0135 00A0 FCTAB: DS 0H * FCT-Sprungverteiler
0136 00A0 C3 B6 0B JMP 0BB6H * FCT0: CAS-Ausgabe
0137 00A3 C3 05 0C JMP 0C05H * FCT1: CAS-Eingabe
0138 00A6 C3 F5 0D JMP 0DF5H * FCT2: CPU-SOD-Ausgabe
0139 00A9 C3 0D 0E JMP 0E0DH * FCT3: CPU-SID-Eingabe
0140 00AC C3 81 0E JMP 0E81H * FCT4: HEX-Dump (Thermodr.)
0141 00AF C3 94 0E JMP 0E94H * FCT5: ASCII-Dump (Thermodr.)
0142 00B2 C3 8D 06 JMP 68DH * FCT6: Exchange
0143 00B5 C3 03 07 JMP 703H * FCT7: Insert
0144 00B8 C3 00 08 JMP 800H * FCT8: Writur
0145 00BB C3 6D 08 JMP 86DH * FCT9: Readur
0146 00BE C3 B0 07 JMP 7B0H * FCTA: Fill
0147 00C1 C3 08 05 JMP 508H * FCTB: Breakpoint
0148 00C4 C3 47 07 JMP 747H * FCTC: Copy
0149 00C7 C3 29 07 JMP 729H * FCTD: Delete
0150 00CA C3 3F 05 JMP 53FH * FCTE: Single Step
0151 00CD C3 00 0A JMP 0A00H * MOVID-Sprungziel: LEDASC
0152 00D0 *
0153 00D0 *
0154 00D0 * MAIN: HEXMO, STP/RAM, EZU/DSP, Peripherie
0155 00D0 *
0156 00D0 *
0157 00D0 ORG 2F8CH *PROMO-Buffer
0158 00D0 *
0159 00D0 * Interrupt-Sprungverteiler
0160 00D0 *
0161 2F8C INTRAP:DS 3H
0162 2F8F INT3: DS 3H
0163 2F92 INT4: DS 3H
0164 2F95 INT5: DS 3H
0165 2F98 INT55: DS 3H
0166 2F9B INT6: DS 3H
0167 2F9E INT65: DS 3H
0168 2FA1 INT7: DS 3H
0169 2FA4 INT75: DS 3H
0170 2FA4 *
0171 2FA4 * Hilfszellen für den Monitor
0172 2FA4 *
0173 2FA7 REGA: DS 1H * REG A
0174 2FA8 REGB: DS 1H * REG B
0175 2FA9 REGC: DS 1H * REG C
0176 2FAA REGD: DS 1H * REG D
0177 2FAB REGE: DS 1H * REG E
0178 2FAC REGF: DS 1H * FLAGS
0179 2FAD REGH: DS 1H * REG H
0180 2FAE REGL: DS 1H * REG L
0181 2FAF REGS: DS 1H * Stack-Pointer (HI)
0182 2FB0 REGP: DS 1H * Stack-Pointer (LO)
0183 2FB1 DS 1H * frei
0184 2FB1 *
0185 2FB2 DSPBUF:DS 1H * LED-Zeile
0186 2FB3 DIG1: DS 1H * Digit 1
0187 2FB4 DIG2: DS 1H * Digit 2
0188 2FB5 DIG3: DS 1H * Digit 3
0189 2FB6 DIG4: DS 1H * Digit 4
0190 2FB7 DIG5: DS 1H * Digit 5
0191 2FB8 DIG6: DS 1H * Digit 6

```

0192	2FB9	DIG7:	DS	1H	* Digit 7
0193	2FB9	*			
0194	2FBA	DATA:	DS	1H	* Daten
0195	2FBB	ADRES:	DS	1H	* Adresse
0196	2FBC	ADRHI:	DS	1H	
0197	2FBD	MODFLG:	DS	1H	* Mode-FLAG
0198	2FBE	FCTBUF:	DS	1H	* FCT-Nr.
0199	2FBF	SRCBEG:	DS	2H	* Par.1
0200	2FC1	SRCEND:	DS	2H	* Par.2
0201	2FC3	DSTBEG:	DS	2H	* Par.3
0202	2FC5	DSTEND:	DS	2H	* Par.4
0203	2FC7	REGNR:	DS	1H	* REG-Nr.
0204	2FC8	IOFLG:	DS	1H	* I/O-Param.
0205	2FC9	NXTBUF:	DS	1H	* FCT-Eingaben
0206	2FCA	MAXBUF:	DS	1H	* max. Anz.
0207	2FCB	KEYBYT:	DS	1H	* KEY true./invers
0208	2FCC	CIFLAG:	DS	1H	* FLAG f. CI-Routine
0209	2FCD	HIBUF:	DS	2H	* I/O-Hilfsroutine
0210	2FCF	HBUF:	DS	1H	
0211	2FCF	*			
0212	2FD0	HIBUF2:	DS	1H	* CTE-Sprungziel
0213	2FD1	HBUF2:	DS	2H	
0214	2FD3	BRKPT1:	DS	2H	* BRK-Adresse #1
0215	2FD5	OPCOD1:	DS	1H	* OpCode-Buf. #1
0216	2FD6	BRKPT2:	DS	2H	* BRK-Adresse #2
0217	2FD8	OPCOD2:	DS	2H	* OpCode-Buf. #2
0218	2FD8	*			
0219	2FDA	UHRBUF	EQU	2FDCH	* BCD-Uhrzeit-Buffer
0220	2FDA	*			
0221	2FDA	ASCBUF	EQU	2FEAH	* ASCII-Uhrzeit-Buf.
0222	2FDA	*			
0223	2FDA	END			

```

0001 0000 *****
0002 0000 *
0003 0000 * HEXMO MOPPEL-HEX-Monitor V 5.5 24.10.85 *
0004 0000 *
0005 0000 *****
0006 0000 *
0007 0000 *
0008 0000 * Monitor für HEX-Tastatur und Siebensegment-
0009 0000 * Anzeige; unterscheidet drei Ebenen, die von
0010 0000 * den Mode-FLAGS (MODFLG) abhängig sind:
0011 0000 *
0012 0000 * 1. MEM-Mode (RAM-Zellen inspiz. u. modifiz.)
0013 0000 * (Mode-Anzeige im Digit 3 ="1")
0014 0000 *
0015 0000 * 2. REG-Mode (Register inspizieren u. modif.)
0016 0000 * (Mode-Anzeige im Digit 3 ="2")
0017 0000 *
0018 0000 * 3. FCT-Mode (Funktionen vorber. u. aufrufen)
0019 0000 * (Mode-Anzeige im Digit 3 ="3")
0020 0000 *
0021 0000 * In der LED-Zeile erscheint im Normalbetrieb
0022 0000 * der Inhalt des Datenfeldes binär; im Ein-
0023 0000 * zelschritt-Betrieb wird hier der Inhalt
0024 0000 * des CPU-FLAG-Registers angezeigt.
0025 0000 *
0026 0000 * Aufbau der Mode-FLAGS (MODFLG):
0027 0000 *
0028 0000 * Bit 7 6 5 4 3 2 1 0
0029 0000 * : x x x : : : :
0030 0000 * : : : : : ..Daten-Mode
0031 0000 * : : : : : ..Adreß-Mode
0032 0000 * : : : : : ..Register-Mode
0033 0000 * : : : : : ..Funktions-Eingabe
0034 0000 * : : : : : ..Einzelschritt-Betrieb
0035 0000 *
0036 0000 *
0037 0000 ORG ODOH
0038 00D0 OFS ODOH
0039 00D0 *
0040 00D0 31 00 2F RESET: LXI SP,STACK *Kaltstart
0041 00D3 CD 4F 01 CALL SAVE- *Regs retten
0042 00D6 CD 09 01 CALL INIT
0043 00D9 21 CB 2F LXI H,KEYBYT *KEYs true oder invers?
0044 00DC 36 FF MVI M,OFFH
0045 00DE DB BC IN COMAND *Ge-Diode abfragen
0046 00E0 FE 7F CPI 7FH *HEX (Ge-Diode) true
0047 00E2 CA FD 00 JZ MO
0048 00E5 34 INR M *KEYBYT=0
0049 00E6 FE 80 CPI 80H *HEX (Ge-Diode) invers
0050 00E8 CA FD 00 JZ MO
0051 00EB FE 00 CPI 0H *ASCII invers
0052 00ED CA 00 10 JZ ROMEXT
0053 00F0 05 DCR M *KEYBYT=FF
0054 00F1 C3 00 10 JMP ROMEXT
0055 00F4 CD B4 03 CALL UPDATE
0056 00F7 AF WARM-: XRA A
0057 00F8 D3 40 OUT FDCTRL *Hilfsfsbank 0
0058 00FA 31 00 2F LXI SP,STACK
0059 00FD CD 75 01 MO: CALL HEXKEY *HEX-Tastatur abfragen
0060 0100 CD C6 02 CALL SSGDSP *SSG-Anzeige aktivieren
0061 0103 CD EB 02 CALL CMDKEY *Befehlstasten abfragen
0062 0106 C3 FD 00 JMP MO *Endlosschleife
0063 0109 *

```



```

0064 0109      *      INIT      nimmt die Grundeinstellungen vor
0065 0109      *
0066 0109 21 70 2F INIT: LXI H,USRSTK *Anwender-Stack-Pointer
0067 010C 7D      MOV A,L      *sortieren für REGBUF
0068 010D 6C      MOV L,H      *(H&L vertauschen)
0069 010E 67      MOV H,A
0070 010F 22 AF 2F SHLD REGS      *User-Stack definieren
0071 0112 3E 00      MVI A,0H      *Res.
0072 0114 32 00 00 STA OH
0073 0117 3E 00      MVI A,0H      *REG A löschen
0074 0119 32 BD 2F STA MODFLG      *alle Monitor-FLAGS löschen
0075 011C 32 C8 2F STA IOFLG      *Paralldruck/Shift Lock
0076 011F CD 6D 03 CALL ADRFIX      *Adr=RAMBEG
0077 0122 CD 63 03 CALL DAT      *Daten-Mode
0078 0125 3E 00      MVI A,0H
0079 0127 32 C7 2F STA REGNR      *"REG A" voreinstellen
0080 012A D3 1A      OUT STPFF      *Single-Step-Flipflop
0081 012C D3 40      OUT FDCTRL      *Adreßlatch löschen
0082 012E D3 00      OUT LEDPOR      *LEDs dunkel
0083 0130 D3 0B      OUT PRMCTL      *Read Enable
0084 0132 3E 0C      MVI A,0CH      *Thermodrucker-Motor
0085 0134 D3 38      OUT PRTMOT      *ausschalten
0086 0136 3E 20      MVI A,20H      *PRT-Schnittstelle
0087 0138 D3 A9      OUT SERMOD      *(PRTOT auf LOW)
0088 013A 3E 82      MVI A,82H      *Steuerwort für 8255
0089 013C D3 D8      OUT PARCTL      *A=Out; B=In; C=Out
0090 013E 3E F0      MVI A,0F0H      *Prommer-LEDs
0091 0140 D3 0A      OUT PRMLD      *ausschalten
0092 0142 3E F4      MVI A,0F4H      *Ruhepegel
0093 0144 D3 19      OUT EZUCTL      *für EZU
0094 0146 3E 17      MVI A,RSTOFF   *RSTs sperren
0095 0148 30      SIM
0096 0149 00      NOP      *wg. NSC800-JR-Befehl
0097 014A DB 4B      IN DATREG      *Reset FDC DRQ line
0098 014C D3 B9      OUT BELOUT      *Schnarre aktivieren
0099 014E C9      RET
0100 014F      *
0101 014F      *      SAVE      rettet sämtliche Register
0102 014F      *
0103 014F 32 A7 2F SAVE-: STA REGA      *REG A retten
0104 0152 7D      MOV A,L      *H&L vertauschen
0105 0153 6C      MOV L,H
0106 0154 67      MOV H,A
0107 0155 22 AD 2F SHLD REGH      *H&L retten
0108 0158 F5      PUSH PSW      *FLAGS in den Stack
0109 0159 E1      POP H      *und von dort nach REG L
0110 015A 7D      MOV A,L
0111 015B 32 AC 2F STA REGF      *FLAGS retten
0112 015E 21 A8 2F LXI H,REGB      *Pointer auf REG-Buffer
0113 0161 70      MOV M,B      *REG B retten
0114 0162 23      INX H
0115 0163 71      MOV M,C      *REG C retten
0116 0164 23      INX H
0117 0165 72      MOV M,D      *REG D retten
0118 0166 23      INX H
0119 0167 73      MOV M,E      *REG E retten
0120 0168 21 00 00 LXI H,0H
0121 016B 39      DAD SP      *Stack-Pointer nach H&L
0122 016C 23      INX H      *SP um Zwei erhöhen
0123 016D 23      INX H      *(wegen CALL-Einsprung)
0124 016E 7D      MOV A,L      *sortieren
0125 016F 6C      MOV L,H      *(H&L vertauschen)
0126 0170 67      MOV H,A
0127 0171 22 AF 2F SHLD REGS      *SP retten

```

```

0128 0174 C9 RET
0129 0175 *
0130 0175 * HEXKEY fragt die HEX-Tasten ab und schiebt
0131 0175 * Eingaben ins Daten- oder Adreßfeld
0132 0175 * nach
0133 0175 *
0134 0175 CD 83 01 HEXKEY:CALL HHKEY- *HEX-Tastenzeilen einlesen
0135 0178 DO RNC *keine Taste: RET
0136 0179 CD B6 01 CALL SHIFT *Eingabe nachschieben
0137 017C CD 3D 02 CALL CONV T *HEX/SSG-Umsetzung
0138 017F CD 4E 04 CALL RELES- *Loslassen abwarten
0139 0182 C9 RET
0140 0183 *
0141 0183 * HHKEY fragt beide HEX-Zeilen ab (ein-
0142 0183 * schließlich Entprellen)
0143 0183 * no key: ZERO-FLAG on (Z=1)
0144 0183 * key: CARRY-FLAG on (CY=1) und
0145 0183 * Key-Nr. in REG A und REG C
0146 0183 *
0147 0183 CD 97 01 HHKEY-:CALL HKEY *erste Abfrage
0148 0186 DO RNC *keine HEX-Taste gedrückt
0149 0187 47 MOV B,A *Input in REG B retten
0150 0188 CD 9F 04 CALL DELYP *Prellzeit abwarten
0151 018B CD 97 01 CALL HKEY *Abfrage wiederholen
0152 018E B8 INPCMP:CMP B *beide Abfragen vergleichen
0153 018F CA 95 01 JZ SETCY- *gleich: CY on
0154 0192 37 STC *ungleich: CY off
0155 0193 3F CMC
0156 0194 C9 RET
0157 0195 37 SETCY: STC
0158 0196 C9 RET
0159 0197 *
0160 0197 * HKEY wie HHKEY, aber ohne Entprellen
0161 0197 *
0162 0197 *
0163 0197 DB EC HKEY: IN HEXLO *Tasten 0...7 abfragen
0164 0199 CD A5 01 CALL KEYIN *Input-Sub
0165 019C D8 RC *Taste aktiv: RET
0166 019D DB DC IN HEXHI *Tasten 8...F abfragen
0167 019F 0E 08 MVI C,8H *Anfangswert für 2.Zeile
0168 01A1 CD A7 01 CALL KEYIN2 *Input-Sub
0169 01A4 C9 RET
0170 01A5 *
0171 01A5 * KEYIN ermittelt (bei aktivierter Taste)
0172 01A5 * die Bit-Nr. innerhalb der Zeile
0173 01A5 * no key: ZERO-FLAG on (Z=1)
0174 01A5 * key: CARRY-FLAG on (CY=1) und
0175 01A5 * KEY-Nr. in REG A und REG C
0176 01A5 *
0177 01A5 0E 00 KEYIN: MVI C,0H *Anfangswert für 1.Zeile
0178 01A7 21 CB 2F KEYIN2:LXI H,KEYBYT *KEYs true./invers?
0179 01AA AE XRA M *invert.und/oder FLAGS setzen
0180 01AB C8 RZ *kein Input: RET
0181 01AC *
0182 01AC 0F KEYLOP:RRC *HIGH-Bit ins CY-Bit
0183 01AD DA B4 01 JC KEYEND *Schieben beendet
0184 01B0 0C INR C *Bit-Zähler erhöhen
0185 01B1 C3 AC 01 JMP KEYLOP *weiter schieben
0186 01B4 79 KEYEND:MOV A,C *Ergebnis in REG A übergeben
0187 01B5 C9 RET
0188 01B6 *
0189 01B6 * SHIFT schiebt HEX-Eingaben ins Daten-
0190 01B6 * oder Adreßfeld nach (abh. von MODFL
0191 01B6 *

```

0192	01B6	21 BA 2F	SHIFT: LXI	H,DATA	*Pointer auf Datenfeld
0193	01B9	3A BD 2F	LDA	MODFLG	*Mode-FLAGS holen
0194	01BC	0F	RRC		*FLAG ins Carry-Bit
0195	01BD	DA D7 01	JC	SFTDAT	*Daten-Mode: JMP
0196	01C0	0F	RRC		
0197	01C1	DA C5 01	JC	SFTADR	*Adreß-Mode
0198	01C4	C9	RET		
0199	01C5	0F	SFTADR:RRC		*Reg-Flag testen
0200	01C6	DA EB 01	JC	SFTREG	*Register-Mode: JMP
0201	01C9	23	INX	H	*Pointer auf Adreßfeld
0202	01CA	CD F4 01	CALL	NIBADD	*Nibble nachschieben
0203	01CD	7E	MOV	A,M	*ADRLO erneut holen
0204	01CE	71	MOV	M,C	*aufdatiertes Byte nach ADRLO
0205	01CF	E6 F0	ANI	0FOH	*4 LSB löschen
0206	01D1	0F	RRC		
0207	01D2	0F	RRC		
0208	01D3	0F	RRC		
0209	01D4	0F	RRC		
0210	01D5	4F	MOV	C,A	*für NIBADD ablegen
0211	01D6	23	INX	H	*Pointer auf ADRHI
0212	01D7	CD F4 01	SFTDAT:CALL	NIBADD	*Nibble nachschieben
0213	01DA	71	MOV	M,C	*modif.Byte ins RAM
0214	01DB	3A BD 2F	LDA	MODFLG	
0215	01DE	47	MOV	B,A	*MODFLG retten
0216	01DF	E6 08	ANI	8H	*FCT-FLAG maskieren
0217	01E1	C8	RZ		*kein FCT-Mode: RET
0218	01E2	78	MOV	A,B	*MODFLG holen
0219	01E3	E6 FE	ANI	0FEH	*DAT-FLAG löschen
0220	01E5	F6 02	ORI	2H	*ADR-FLAG setzen
0221	01E7	32 BD 2F	STA	MODFLG	
0222	01EA	C9	RET		
0223	01EB	79	SFTREG:MOV	A,C	
0224	01EC	C6 06	ADI	6H	*verschieben: A = 0
0225	01EE	E6 07	ANI	7H	*MSB unterdrücken
0226	01F0	32 C7 2F	STA	REGNR	*hier: 0...7
0227	01F3	C9	RET		
0228	01F4		*		
0229	01F4		*		
0230	01F4		*	NIBADD	schiebt das eingegebene Nibble in
0231	01F4		*		das von H&L adressierte Byte nach
0232	01F4		*		(Ergebnis in REG C)
0233	01F4		*		
0234	01F4	7E	NIBADD:MOV	A,M	*altes Byte holen
0235	01F5	E6 0F	ANI	0FH	*obere Hälfte löschen
0236	01F7	07	RLC		
0237	01F8	07	RLC		
0238	01F9	07	RLC		
0239	01FA	07	RLC		
0240	01FB	B1	ORA	C	*neues Nibble anhängen
0241	01FC	4F	MOV	C,A	*in REG C retten
0242	01FD	C9	RET		
0243	01FE		*		
0244	01FE		*	READOT	aktualisiert das Datenfeld (ent-
0245	01FE		*		sprechend Adreßfeld bzw. Reg-Nr.)
0246	01FE		*		
0247	01FE	3A BD 2F	READOT:LDA	MODFLG	*Mode-FLAGS holen
0248	0201	47	MOV	B,A	*und retten
0249	0202	E6 08	ANI	8H	*FCT-FLAG maskieren
0250	0204	C0	RNZ		*FCT-Mode: RET
0251	0205	78	MOV	A,B	*MODFLG zurück holen
0252	0206	E6 04	ANI	4H	*REG-FLAG maskieren
0253	0208	C2 13 02	JNZ	READRG	*REG-Mode: JMP
0254	020B	2A BB 2F	LHLD	ADRES	*Adreßfeld auslesen
0255	020E	7E	READAT:MOV	A,M	*adressierte Daten holen

0256	020F	32	BA	2F		STA	DATA	*und ins Datenfeld bringen
0257	0212		C9			RET		
0258	0213	21	A7	2F	READRG:	LXI	H,REGA	*Anfang REGBUF
0259	0216	3A	C7	2F		LDA	REGNR	*Reg-Nr. holen
0260	0219		5F			MOV	E,A	
0261	021A	16	00			MVI	D,OH	*Versatz erzeugen
0262	021C		19			DAD	D	*H&L=Adr. REGBUF
0263	021D	C3	0E	02		JMP	READAT	
0264	0220				*			
0265	0220				*	WRITIN		schreibt die Daten aus dem Daten-
0266	0220				*			feld ins RAM (MEM bzw. REGBUF)
0267	0220				*			
0268	0220	3A	BD	2F	WRITIN:	LDA	MODFLG	*Mode-FLAGS holen
0269	0223	E6	04			ANI	4H	*REG-FLAG maskieren
0270	0225	C2	30	02		JNZ	WRITRG	*REG-Mode: JMP
0271	0228	2A	BB	2F		LHLD	ADRES	*Adreßfeld auslesen
0272	022B	3A	BA	2F	WRITDA:	LDA	DATA	*Daten holen
0273	022E		77			MOV	M,A	*und überschreiben
0274	022F		C9			RET		
0275	0230	21	A7	2F	WRITRG:	LXI	H,REGA	*Anfang REGBUF
0276	0233	3A	C7	2F		LDA	REGNR	*Reg-Nr. holen
0277	0236		5F			MOV	E,A	
0278	0237	16	00			MVI	D,OH	*Versatz erzeugen
0279	0239		19			DAD	D	*H&L=Zieladresse
0280	023A	C3	2B	02		JMP	WRITDA	
0281	023D				*			
0282	023D				*	CONVT		setzt die HEX-Daten aus dem
0283	023D				*			Daten- und Adreßfeld in den SSG-
0284	023D				*			Code um und überschreibt diesen
0285	023D				*			in den Display-Buffer DSPBUF
0286	023D				*			
0287	023D	3A	BA	2F	CONVT:	LDA	DATA	
0288	0240		47			MOV	B,A	*Daten-Byte retten
0289	0241	3A	BD	2F		LDA	MODFLG	
0290	0244	E6	80			ANI	80H	*STP-FLAG maskieren
0291	0246	CA	4D	02		JZ	CNVLED	*kein Single-Step: JMP
0292	0249	3A	AC	2F		LDA	REGF	*FLAG-Register auslesen
0293	024C		47			MOV	B,A	
0294	024D	78			CNVLED:	MOV	A,B	*LED-Byte holen
0295	024E	32	B2	2F		STA	DSPBUF	*LED-Zeile
0296	0251	11	B3	2F		LXI	D,DIG1	*Anzeige-Digit 1
0297	0254	3A	BA	2F		LDA	DATA	*Datenfeld (wieder) nach REG A
0298	0257	CD	96	02		CALL	CNVBY-	*SSG-Umsetzung von (DATA)
0299	025A	11	B6	2F		LXI	D,DIG4	*Anzeige-Digit 4
0300	025D	3A	BD	2F		LDA	MODFLG	*Mode-FLAGS holen
0301	0260	E6	04			ANI	4H	*REG-Mode?
0302	0262	C2	74	02		JNZ	CNVREG	*ja: JMP
0303	0265	3A	BB	2F	CNVADR:	LDA	ADRES	
0304	0268	CD	96	02		CALL	CNVBY-	*SSG-Umsetzung von (ADRLO)
0305	026B	3A	BC	2F		LDA	ADRHI	
0306	026E	CD	96	02		CALL	CNVBY-	*SSG-Umsetzung von (ADRHI)
0307	0271	C3	83	02		JMP	DPSET	
0308	0274	3A	C7	2F	CNVREG:	LDA	REGNR	*REG-Nr.
0309	0277	21	BC	02		LXI	H,REGTAB	*REG-SSG-Codes
0310	027A	CD	A6	02		CALL	CNVSG2	*SSG-Umsetzung von (REGNR)
0311	027D		AF			XRA	A	
0312	027E		12			STAX	D	*DIG5 löschen
0313	027F		13			INX	D	
0314	0280		12			STAX	D	*DIG6 löschen
0315	0281		13			INX	D	
0316	0282		12			STAX	D	*DIG7 löschen
0317	0283				*			
0318	0283				*	DPSET		Dezimalpunkt ins Daten- oder
0319	0283				*			Adreßfeld einblenden

```

0320 0283          *
0321 0283 11 B3 2F DPSET: LXI  D,DIG1  *Pointer auf Digit 1
0322 0286 3A BD 2F      LDA  MODFLG
0323 0289 E6 01          ANI  1H      *DAT-FLAG maskieren
0324 028B C2 91 02      JNZ  DPCTE  *Daten-Mode: JMP
0325 028E 13            INX  D
0326 028F 13            INX  D
0327 0290 13            INX  D      *Pointer auf Digit 4
0328 0291 1A          DPCTE: LDAX  D      *SSG-Code holen
0329 0292 F6 80          ORI  80H    *MSB setzen (=Dez.Punkt)
0330 0294 12          STAX  D      *SSG-Code wieder ablegen
0331 0295 C9            RET
0332 0296          *
0333 0296          *      CNVBYT      setzt (REG A) in den SSG-Code um;
0334 0296          *      überschreibt ihn in 2 RAM-Zellen
0335 0296          *      (ab (D&E)); (D&E)=(D&E)+2
0336 0296          *
0337 0296 47          CNVBY-:MOV  B,A      *Byte retten
0338 0297 CD A3 02      CALL CNVSSG  *SSG-Code holen
0339 029A 78          MOV  A,B      *Byte zurückholen
0340 029B 0F          RRC
0341 029C 0F          RRC
0342 029D 0F          RRC
0343 029E 0F          RRC
0344 029F CD A3 02      CALL CNVSSG  *SSG-Code holen
0345 02A2 C9            RET
0346 02A3          *
0347 02A3          *      CNVSSG      setzt REG A, unteres Nibble in
0348 02A3          *      den SSG-Code um u.schreibt ihn
0349 02A3          *      in die von D&E adr. RAM-Zelle;
0350 02A3          *      erhöht (D&E) um Eins
0351 02A3          *
0352 02A3 21 B2 02 CNVSSG:LXI  H,SSGTAB *Anfang Code-Tabelle
0353 02A6 D5          CNVSSG2:PUSH D      *Buffer-Pointer retten
0354 02A7 16 00          MVI  D,0H    *Versatz bilden
0355 02A9 E6 0F          ANI  0FH    *obere 4 Bits löschen
0356 02AB 5F          MOV  E,A
0357 02AC 19          DAD  D      *(H&L)=Code-Adresse
0358 02AD D1          POP  D      *Buffer-Pointer zurückholen
0359 02AE 7E          MOV  A,M    *SSG-Code holen
0360 02AF 12          STAX  D      *und in DSP-Buffer schreiben
0361 02B0 13          INX  D      *Pointer erhöhen
0362 02B1 C9            RET
0363 02B2          *
0364 02B2          *      SSGTAB      Tabelle der Siebensegment-Codes von
0365 02B2          *      0...9, A...F und H,L,S,P
0366 02B2          *
0367 02B2 3F          SSGTAB:DB  3FH  * 0
0368 02B3 06          DB  06H  * 1
0369 02B4 5B          DB  5BH  * 2
0370 02B5 4F          DB  4FH  * 3
0371 02B6 66          DB  66H  * 4
0372 02B7 6D          DB  6DH  * 5
0373 02B8 7D          DB  7DH  * 6
0374 02B9 07          DB  07H  * 7
0375 02BA 7F          DB  7FH  * 8
0376 02BB 6F          DB  6FH  * 9
0377 02BC 77          REGTAB:DB  77H  * A
0378 02BD 7C          DB  7CH  * b
0379 02BE 39          DB  39H  * C
0380 02BF 5E          DB  5EH  * d
0381 02C0 79          DB  79H  * E
0382 02C1 71          DB  71H  * F
0383 02C2 76          DB  76H  * H

```

```

0384 02C3 38 DB 38H * L
0385 02C4 6D DB 6DH * S
0386 02C5 73 DB 73H * P
0387 02C6 *
0388 02C6 * SSGDSP überschreibt den SSG-Code aus dem
0389 02C6 * Display-Buffer in die Anzeige; hat
0390 02C6 * feste Laufzeit von 10 ms
0391 02C6 *
0392 02C6 11 B9 2F SSGDSP:LXI D,DIG7 *oberste RAM-Zelle im DSPBUF
0393 02C9 13 INX D
0394 02CA 21 CD 2F LXI H,HIBUF *Output-Routine im RAM
0395 02CD 36 D3 MVI M,0D3H *HEX-Code "OUT"
0396 02CF 23 INX H
0397 02D0 23 INX H
0398 02D1 36 C9 MVI M,0C9H *HEX-Code "RET"
0399 02D3 2B DCX H
0400 02D4 0E 80 MVI C,OUTPOR *OUT-Portadresse +1
0401 02D6 1B SSGLOP:DCX D *Adresse im DSPBUF
0402 02D7 79 MOV A,C *alte OUT-Adresse
0403 02D8 D6 10 SUI 10H *Portadresse erniedrigen
0404 02DA B7 ORA A *FLAGS setzen
0405 02DB 4F MOV C,A *OUT-Adresse retten
0406 02DC 77 MOV M,A *OUT-Adresse in OUT-Routine
0407 02DD 1A LDAX D *SSG-Code aus DSPBUF holen
0408 02DE CD CD 2F CALL HIBUF *OUT-Hilfsroutine
0409 02E1 CD AF 04 CALL DELY1- *1-kHz-MUX-Frequenz
0410 02E4 C2 D6 02 JNZ SSGLOP *8 Einzelausgaben abwarten
0411 02E7 CD AC 04 CALL DELY2 *auf 10 ms verlängern
0412 02EA C9 RET
0413 02EB *
0414 02EB * CMDKEY fragt die Befehlstasten ab und
0415 02EB * verzweigt nach Sprungtabelle
0416 02EB *
0417 02EB CD 73 00 CMDKEY:CALL CCKEY *CMD-Keys abfragen
0418 02EE D0 RNC *keine Taste: RET
0419 02EF 21 4B 03 LXI H,JMPTAB *CMD-Sprungverteiler
0420 02F2 47 MOV B,A *Input retten
0421 02F3 3A BD 2F LDA MODFLG
0422 02F6 E6 08 ANI 8H *FCT-FLAG maskieren
0423 02F8 CA 1F 03 JZ BRANCH *CMD-Mode: JMP
0424 02FB 78 MOV A,B *FCT-Mode; Input holen
0425 02FC FE 00 CPI NXTNR *NXT? (nächster Parameter)
0426 02FE CA 1F 03 JZ BRANCH *ja: JMP
0427 0301 FE 04 CPI ADRNR *ADR? (zurück zum Adreß-Mode)
0428 0303 CA 1F 03 JZ BRANCH *ja: JMP
0429 0306 FE 02 CPI FCTNR *FCT? (zurück zum FCT-Feld)
0430 0308 CA 1F 03 JZ BRANCH *ja: JMP
0431 030B FE 06 CPI RUNR *ist es RUN?
0432 030D C0 RNZ *nein: dann RET
0433 030E *
0434 030E 3A BD 2F BRNCH: LDA MODFLG *LEDASC-Sprungziel
0435 0311 E6 F7 ANI 0F7H *FCT-FLAG löschen
0436 0313 32 BD 2F STA MODFLG
0437 0316 21 A0 00 LXI H,FCTAB *FCT-Sprungverteiler
0438 0319 3A BA 2F LDA DATA *"Fx" holen
0439 031C E6 0F ANI 0FH *4 MSB löschen
0440 031E 47 MOV B,A *FCT-Nr. 0...F
0441 031F 78 BRANCH:MOV A,B *Input holen
0442 0320 07 RLC *Input mal 2
0443 0321 80 ADD B *Input mal 3
0444 0322 5F MOV E,A
0445 0323 16 00 MVI D,0H *Versatz in D&E
0446 0325 19 DAD D *(H&L)=Tabellen-Adresse
0447 0326 11 2B 03 LXI D,RETUR *Sub-Rücksprungadresse

```

0448	0329	D5			PUSH D	*im Stack ablegen
0449	032A	E9			PCHL	*Verzweigung über Tabelle
0450	032B			*		
0451	032B	CD 4E 04	RETUR:	CALL RELES-		*CMD-Loslassen abwarten
0452	032E	C9		RET		*zurück zu MONI
0453	032F			*		
0454	032F			*	CKEY	fragt die CMD-Zeile ab (ein-
0455	032F			*		schließlich Entprellen) und
0456	032F			*		unterdrückt die Dioden-Nr.
0457	032F			*	no key:	ZERO-FLAG on (Z=1)
0458	032F			*	key:	CARRY-FLAG on (CY=1) und
0459	032F			*		Key-Nr. in REG A und REG C
0460	032F			*		
0461	032F	CD 3D 03	CKEY-:	CALL CKEY		*erste Abfrage
0462	0332	D0		RNC		*keine CMD-Taste gedrückt
0463	0333	47		MOV B,A		*Input in REG B retten
0464	0334	CD 9F 04		CALL DELYP		*Prellzeit abwarten
0465	0337	CD 3D 03		CALL CKEY		*Abfrage wiederholen
0466	033A	C3 8E 01		JMP INPCMP		*Abfragen vergl.(in HHKEY)
0467	033D			*		
0468	033D			*	CKEY	wie CCKEY, aber ohne Entprellen
0469	033D			*		
0470	033D	DB BC	CKEY:	IN COMAND		*CMD-Tasten abfragen
0471	033F	CD A5 01		CALL KEYIN		*Input-Sub
0472	0342	D0		RNC		*keine Taste aktiv: RET
0473	0343	FE 07		CPI DIODNR		*nur Ge-Diode drin?
0474	0345	C2 95 01		JNZ SETCY		*nein: CY on
0475	0348	37		STC		*ja: CY off
0476	0349	3F		CMC		
0477	034A	C9		RET		
0478	034B			*		
0479	034B			*		Sprungverteiler für CMD-Verarbeitung
0480	034B			*		
0481	034B	C3 BB 03	JMPTAB:	JMP NXT		
0482	034E	C3 01 04		JMP BST		
0483	0351	C3 91 03		JMP FCT		
0484	0354	C3 82 03		JMP REG		
0485	0357	C3 73 03		JMP ADR		
0486	035A	C3 63 03		JMP DAT		
0487	035D	C3 1A 04		JMP RUN		
0488	0360			*		
0489	0360			*		Mode-Keys DAT,ADR/FCTF,REG,FCT (MODFLGs)
0490	0360			*		
0491	0360	22 BB 2F	DATSE-:	SHLD ADRES		*H&L ins Adreßfeld
0492	0363	3A BD 2F	DAT:	LDA MODFLG		
0493	0366	E6 FD		ANI OFDH		*ADR-FLAG löschen
0494	0368	F6 01		ORI 1H		*DAT-FLAG setzen
0495	036A	C3 B1 03		JMP MODCTE		
0496	036D	21 00 28	ADRFIX:	LXI H,RAMBEG		*RAM-Anfangsadresse
0497	0370	22 BB 2F	ADRSE-:	SHLD ADRES		*ins Adreßfeld laden
0498	0373	3E 08	ADR:	MVI A,8H		*ein Strich
0499	0375	32 B5 2F		STA DIG3		*(ADR-Mode-Kennzeichen)
0500	0378	3A BD 2F		LDA MODFLG		
0501	037B	E6 F2		ANI OF2H		*DAT+REG+FCT-FLAGS löschen
0502	037D	F6 02		ORI 2H		*ADR-FLAG setzen
0503	037F	C3 B1 03		JMP MODCTE		
0504	0382	3E 48	REG:	MVI A,48H		*zwei Striche
0505	0384	32 B5 2F		STA DIG3		*(REG-Mode-Kennzeichen)
0506	0387	3A BD 2F		LDA MODFLG		
0507	038A	E6 FE		ANI OFEH		*DAT-FLAG löschen
0508	038C	F6 06		ORI 6H		*ADR+REG-FLAGS setzen
0509	038E	C3 B1 03		JMP MODCTE		
0510	0391	3E 49	FCT:	MVI A,49H		*drei Striche
0511	0393	32 B5 2F		STA DIG3		*(FCT-Mode-Kennzeichen)

0512	0396	3E	07		MVI	A,7H	*Max. für 4 Parameter
0513	0398	32	CA	2F	STA	MAXBUF	
0514	039B	AF			XRA	A	
0515	039C	32	C9	2F	STA	NXTBUF	*NXTBUF löschen
0516	039F	3E	FF		MVI	A,OFFH	
0517	03A1	32	BA	2F	STA	DATA	*"FF" ins Datenfeld
0518	03A4	21	00	00	LXI	H,0H	
0519	03A7	22	BB	2F	SHLD	ADRES	*"0000" ins Adreßfeld
0520	03AA	3A	BD	2F	LDA	MODFLG	
0521	03AD	E6	F9		ANI	0F9H	*ADR+REG-FLAGS löschen
0522	03AF	F6	09		ORI	9H	*FCT+DAT-FLAGS setzen
0523	03B1	32	BD	2F	MODCTE:STA	MODFLG	
0524	03B4	CD	FE	01	UPDATE:CALL	READOT	*Daten ins Datenfeld
0525	03B7	CD	3D	02	CALL	CONVT	*SSG-Umsetzung
0526	03BA	C9			RET		
0527	03BB				*		
0528	03BB				*	NXT	erhöht Adr/Reg und schreibt
0529	03BB				*		Daten(feld) ins RAM
0530	03BB	3A	BD	2F	NXT: LDA	MODFLG	
0531	03BE	E6	08		ANI	8H	*FCT-FLAG
0532	03C0	C2	DD	03	JNZ	NXTCTE	*FCT-Mode: JMP
0533	03C3	CD	20	02	CALL	WRITIN	*Daten einschreiben
0534	03C6	C2	D0	03	JNZ	NXTREG	*REG-Mode: JMP
0535	03C9	23			INX	H	*Adreßfeld hochzählen
0536	03CA	22	BB	2F	SHLD	ADRES	*und ablegen
0537	03CD	C3	B4	03	JMP	UPDATE	*Anzeige aktualisieren
0538	03D0	3A	C7	2F	NXTREG:LDA	REGNR	
0539	03D3	3C			INR	A	*Reg-Nr. hochzählen
0540	03D4	27			DAA		*Dezimalkorrektur
0541	03D5	E6	0F		NXTSTO:ANI	0FH	*Übertrag unterdrücken
0542	03D7	32	C7	2F	STA	REGNR	*neue Reg-Nr. ablegen
0543	03DA	C3	B4	03	JMP	UPDATE	*Anzeige aktualisieren
0544	03DD	21	CA	2F	NXTCTE:LXI	H,MAXBUF	*Maximalwert f. Eingaben
0545	03E0	3A	C9	2F	LDA	NXTBUF	*Eingabe-Zähler
0546	03E3	BE			CMP	M	*Maximalwert erreicht?
0547	03E4	D0			RNC		*ja: RET
0548	03E5	2A	BB	2F	LHLD	ADRES	*letzter Parameter
0549	03E8	EB			XCHG		*nach D&E retten
0550	03E9	21	BF	2F	LXI	H,SRCBEG	*Anf. Parameter-Buffer
0551	03EC	4F			MOV	C,A	*Versatz erzeugen
0552	03ED	06	00		MVI	B,0H	
0553	03EF	09			DAD	B	*SRCBEG + NXTBUF
0554	03F0	73			MOV	M,E	*Parameter in Buffer
0555	03F1	23			INX	H	
0556	03F2	72			MOV	M,D	
0557	03F3	21	00	00	LXI	H,0H	
0558	03F6	22	BB	2F	SHLD	ADRES	*Adreßfeld wieder löschen
0559	03F9	21	C9	2F	LXI	H,NXTBUF	*Eingabe-Zähler +2
0560	03FC	34			INR	M	
0561	03FD	34			INR	M	
0562	03FE	C3	B4	03	JMP	UPDATE	*Adr=0000 anzeigen
0563	0401				*		
0564	0401				*	BST	schaltet Adr/Reg zurück
0565	0401				*		
0566	0401	CD	FE	01	BST: CALL	READOT	*Pointer ermitteln
0567	0404	C2	0E	04	JNZ	BSTREG	*REG-Mode: JMP
0568	0407	2B			DCX	H	*Adreßfeld erniedrigen
0569	0408	22	BB	2F	SHLD	ADRES	*und ablegen
0570	040B	C3	B4	03	JMP	UPDATE	*Anzeige aufdatieren
0571	040E	3A	C7	2F	BSTREG:LDA	REGNR	*Reg-Nr. holen
0572	0411	3D			DCR	A	*und erniedrigen
0573	0412				*		
0574	0412	F2	D5	03	JP	NXTSTO	*kein neg. Übertrag: JMP
0575	0415	3E	09		MVI	A,9H	*auf Anfangswert setzen

0576	0417	C3 D5 03	JMP	NXTSTO	*neue Reg-Nr. ablegen
0577	041A		*		
0578	041A		*	RUN	Programmstart bei der im Adreßfeld
0579	041A		*		stehenden Adresse; lädt vorher alle
0580	041A		*		Register (einschließlich FLAGS)
0581	041A		*		
0582	041A	AF	RUN:	XRA A	*REG A löschen
0583	041B	32 B2 2F		STA DSPBUF	*LEDs dunkel schalten
0584	041E	CD 4E 04		CALL RELES-	*auf Loslassen warten
0585	0421	3A BD 2F	RESTO-:	LDA MODFLG	
0586	0424	E6 80		ANI 80H	*STP-Mode aktiv?
0587	0426	C4 08 00		CNZ RST1	*ja: Breakpt.setzen
0588	0429	3A AC 2F		LDA REGF	*FLAGS holen
0589	042C	4F		MOV C,A	
0590	042D	C5		PUSH B	*und in den Stack
0591	042E	F1		POP PSW	*von dort ins CPU-FLAG-Reg.
0592	042F	2A AF 2F		LHLD REGS	*Anwender-Stack-Pointer holen
0593	0432	7D		MOV A,L	*und sortieren
0594	0433	6C		MOV L,H	*(H&L vertauschen)
0595	0434	67		MOV H,A	
0596	0435	F9		SPHL	*in CPU-Stack-Pointer
0597	0436	21 A8 2F		LXI H,REGB	*Pointer auf REG-Buffer
0598	0439	46		MOV B,M	*REG B laden
0599	043A	23		INX H	
0600	043B	4E		MOV C,M	*REG C laden
0601	043C	23		INX H	
0602	043D	56		MOV D,M	*REG D laden
0603	043E	23		INX H	
0604	043F	5E		MOV E,M	*REG E laden
0605	0440	2A BB 2F		LHLD ADRES	*Startadresse holen
0606	0443	E5		PUSH H	*und im Stack ablegen
0607	0444	2A AD 2F		LHLD REGH	*Daten für H&L holen
0608	0447	7D		MOV A,L	*und sortieren
0609	0448	6C		MOV L,H	*(H&L vertauschen)
0610	0449	67		MOV H,A	*H&L sind geladen
0611	044A	3A A7 2F		LDA REGA	*REG A laden
0612	044D	C9		RET	*Programmstart
0613	044E		*		
0614	044E		*	RELES	wartet auf Loslassen einer aktiven
0615	044E		*		Taste und aktiviert während dieser
0616	044E		*		Zeit die SSG-Anzeige
0617	044E		*		
0618	044E	CD 61 04	RELES-:	CALL RLS	*Abfrage-Unterprogramm
0619	0451	DA 5B 04		JC RELDSP	*Taste betätigt: JMP
0620	0454	CD 9F 04		CALL DELYP	*Prelzeit abwarten
0621	0457	CD 61 04		CALL RLS	*erneute Abfrage
0622	045A	DO		RNC	*keine Taste: RET
0623	045B	CD C6 02	RELDSP:	CALL SSGDSP	*SSG-Anzeige aktivieren
0624	045E	C3 4E 04		JMP RELES-	*neuer Abfrage-Zyklus
0625	0461		*		
0626	0461		*	RLS	fragt nacheinander HEX- und CMD-
0627	0461		*		Keys ab (kein Entprellen)
0628	0461		*	no key:	ZERO-FLAG on (Z=1)
0629	0461		*	key:	CARRY-FLAG on (CY=1)
0630	0461		*		
0631	0461	CD 97 01	RLS:	CALL HKEY	*HEX-Keys abfragen
0632	0464	D8		RC	*HEX-Taste aktiv: RET
0633	0465	CD 3D 03		CALL CKEY	*CMD-Keys abfragen
0634	0468	C9		RET	
0635	0469		*		
0636	0469		*	PRESS	springt bei Tastendruck zurück
0637	0469		*		zum Warmstart; sonst Return
0638	0469		*		
0639	0469	CD 8B 04	PRESS-:	CALL DLY10-	*Anzeige aktivieren

```

0640 046C CD 83 01 CALL HHKEY-
0641 046F DA 76 04 JC PRELES *HEX-KEY aktiv: JMP
0642 0472 CD 2F 03 CALL CCKEY-
0643 0475 D0 RNC *keine Taste: RET
0644 0476 CD 4E 04 PRELES:CALL RELES- *Loslassen abwarten
0645 0479 CD 6D 03 CALL ADRFIX *Adr=RAM-Beginn
0646 047C C3 03 00 JMP WARM
0647 047F *
0648 047F *
0649 047F *****
0650 047F *
0651 047F * Laufzeit-Unterprogramme
0652 047F *
0653 047F *****
0654 047F *
0655 047F F5 ONESE-:PUSH PSW * 1 s
0656 0480 3E 0A MVI A,0AH * 10 d
0657 0482 CD 8B 04 ONELOP:CALL DLY10-
0658 0485 3D DCR A
0659 0486 C2 82 04 JNZ ONELOP
0660 0489 F1 POP PSW
0661 048A C9 RET
0662 048B *
0663 048B F5 DLY10-:PUSH PSW * 100 ms
0664 048C 3E 0A MVI A,0AH
0665 048E F5 DELYLP:PUSH PSW
0666 048F E5 PUSH H
0667 0490 D5 PUSH D
0668 0491 C5 PUSH B
0669 0492 CD C6 02 CALL SSGDSP
0670 0495 C1 POP B
0671 0496 D1 POP D
0672 0497 E1 POP H
0673 0498 F1 POP PSW
0674 0499 3D DCR A
0675 049A C2 8E 04 JNZ DELYLP
0676 049D F1 POP PSW
0677 049E C9 RET
0678 049F *
0679 049F F5 DELYP: PUSH PSW *KEY-Prellzeit
0680 04A0 3A 07 00 LDA TIMPRL
0681 04A3 CD AF 04 DLOP: CALL DELY1-
0682 04A6 3D DCR A
0683 04A7 C2 A3 04 JNZ DLOP
0684 04AA F1 POP PSW
0685 04AB C9 RET
0686 04AC *
0687 04AC CD AF 04 DELY2: CALL DELY1- * 2 ms
0688 04AF F5 DELY1-:PUSH PSW * 1 ms
0689 04B0 3A 06 00 LDA FQUARZ *4/6/8 MHz!
0690 04B3 87 ADD A
0691 04B4 87 ADD A
0692 04B5 3C INR A *adjust to 99%
0693 04B6 87 ADD A
0694 04B7 87 ADD A
0695 04B8 87 ADD A *Zeitfaktor *32
0696 04B9 3D DELOP: DCR A
0697 04BA C2 B9 04 JNZ DELOP
0698 04BD F1 POP PSW
0699 04BE C9 RET
0700 04BF *
0701 04BF END

```

```

0001 0000 *****
0002 0000 *
0003 0000 * STP/RAM Einzelschritt- und RAM-Funkt. 24.10.85 *
0004 0000 *
0005 0000 *****
0006 0000 *
0007 0000 *      Drei Betriebsarten möglich:
0008 0000 *
0009 0000 *      1 Einzel-Byte-Betrieb
0010 0000 *      (Hardware-Einzel-Wort-Betrieb mit
0011 0000 *      Einzelschritt-Modul; RAM oder ROM)
0012 0000 *      2 Breakpoint-Betrieb
0013 0000 *      (Software-Haltepunkt im RAM setzen)
0014 0000 *      3 Einzelschritt-Betrieb
0015 0000 *      (Software-Single-Step im RAM)
0016 0000 *
0017 0000 *      Bei den Betriebsarten 2 und 3 wird der
0018 0000 *      Inhalt des CPU-FLAG-Registers in der
0019 0000 *      LED-Zeile angezeigt:
0020 0000 *
0021 0000 *      Bit   7   6   5   4   3   2   1   0
0022 0000 *      FLAG  S   Z   x   c   0   P   x   C
0023 0000 *      (Sign, Zero, Parity und Carry;
0024 0000 *      c=Aux.CARRY nicht testbar,
0025 0000 *      x ist undefiniert)
0026 0000 *
0027 0000 ORG      500H
0028 0500 OFS      500H
0029 0500 *
0030 0500 *****
0031 0500 *
0032 0500 *      Einzel-Byte-Betrieb mit Einzelschritt-Modul
0033 0500 *
0034 0500 *****
0035 0500 *
0036 0500 2A BB 2F FCTBYT:LHLD ADRES      *Startadresse holen
0037 0503 3E 40          MVI  A,40H      *BYT-Flipflop setzen
0038 0505 D3 1A          OUT  STPF  *(auf dem STP-Modul)
0039 0507 E9          PCHL      *Byte-Mode starten
0040 0508 *
0041 0508 *****
0042 0508 *
0043 0508 *      Breakpoint-Betrieb im RAM
0044 0508 *
0045 0508 *****
0046 0508 *
0047 0508 21 03 00 FCTBRK:LXI  H,WARM      *Warmstart-Adresse
0048 050B CD 4A 05 BRKMOD:CALL BRKINI    *Breakpoints vorbereiten
0049 050E 2A BB 2F          LHLD ADRES      *Break-Adresse
0050 0511 22 D3 2F          SHLD BRKPT1    *retten
0051 0514 7E          MOV  A,M          *OpCode holen
0052 0515 32 D5 2F          STA  OPCOD1    *und retten
0053 0518 36 D7          MVI  M,0D7H      *"RST2" einsetzen
0054 051A 3A AC 2F BRESTO:LDA  REGF      *RESTOR für FCT B
0055 051D 4F          MOV  C,A          *REG C: CPU-FLAGS
0056 051E C5          PUSH B
0057 051F F1          POP  PSW          *CPU-FLAGS setzen
0058 0520 2A AF 2F          LHLD REGS      *User-Stack-Pointer
0059 0523 7D          MOV  A,L          *sortieren
0060 0524 6C          MOV  L,H          *(H&L vertauschen)
0061 0525 67          MOV  H,A
0062 0526 F9          SPHL          *in CPU-Stack-Pointer
0063 0527 21 A8 2F          LXI  H,REGB      *Pointer auf REG-Buffer

```

```

0064 052A 46          MOV  B,M          *REG B laden
0065 052B 23          INX  H
0066 052C 4E          MOV  C,M          *REG C laden
0067 052D 23          INX  H
0068 052E 56          MOV  D,M          *REG D laden
0069 052F 23          INX  H
0070 0530 5E          MOV  E,M          *REG E laden
0071 0531 2A BF 2F        LHLD SRCBEG      *Startadresse holen
0072 0534 E5          PUSH H           *und im Stack ablegen
0073 0535 2A AD 2F        LHLD REGH        *Daten für H&L holen
0074 0538 7D          MOV  A,L          *und sortieren
0075 0539 6C          MOV  L,H          *(H&L vertauschen)
0076 053A 67          MOV  H,A
0077 053B 3A A7 2F        LDA  REGA        *REG A laden
0078 053E C9          RET              *Start bis zum Breakpoint
0079 053F          *
0080 053F          *****
0081 053F          *
0082 053F          *      Einzelschritt-Betrieb im RAM
0083 053F          *
0084 053F          *****
0085 053F          *
0086 053F 21 00 00 FCTSTP:LXI H,0H      *Warmstart-Adresse
0087 0542 11 03 00          LXI  D,3H        *errechnen
0088 0545 19          DAD  D           *(./.LXI H,3h)
0089 0546 CD 4A 05 STPMOD:CALL BRKINI *Breakpoints vorbereiten
0090 0549 C9          RET
0091 054A          *
0092 054A          *      BRKINI      Breakpoints vorbereiten (STP-FLAG
0093 054A          *                  setzen, REG F löschen, CTE-Adresse
0094 054A          *                  im Hilfsbuffer HIBUF ablegen)
0095 054A          *
0096 054A 22 D1 2F BRKINI:SHLD HBUF2      *CTE-Adresse (Warmstart)
0097 054D 3E C3          MVI  A,0C3H     *"JMP"
0098 054F 32 D0 2F        STA  HIBUF2     *CTE-Adresse für HIBUF
0099 0552 3A BD 2F        LDA  MODFLG     *Mode-FLAG holen
0100 0555 E6 F7          ANI  0F7H       *FCT-FLAG löschen
0101 0557 F6 80          ORI  80H        *STP-Bit setzen
0102 0559 32 BD 2F        STA  MODFLG
0103 055C AF          XRA  A          *REG F löschen
0104 055D 32 AC 2F        STA  REGF
0105 0560 21 00 00        LXI  H,0H       *Breakpoint-Adr.löschen
0106 0563 22 D3 2F        SHLD BRKPT1
0107 0566 22 D6 2F        SHLD BRKPT2
0108 0569 21 B9 2F        LXI  H,DIG7
0109 056C 36 00          MVI  M,0H
0110 056E 2B          DCX  H
0111 056F 36 50          MVI  M,50H     * "r"
0112 0571 2B          DCX  H
0113 0572 36 1C          MVI  M,1CH     * "u"
0114 0574 2B          DCX  H
0115 0575 36 54          MVI  M,54H     * "n"
0116 0577 C9          RET
0117 0578          *
0118 0578          *      REST1      ersetzt den nächstfolgenden OpCode
0119 0578          *                  durch einen Breakpoint (RST2=D7h)
0120 0578          *
0121 0578 2A BB 2F REST1: LHLD ADRES      *aktuelle Adresse holen
0122 057B CD CE 05          CALL REST11     *Adr.des nächsten OpCodes
0123 057E 7E          MOV  A,M
0124 057F 32 D5 2F        STA  OPCOD1     *OpCode retten
0125 0582 36 D7          MVI  M,0D7H    *"RST2" einsetzen
0126 0584          *
0127 0584          *      REST12     setzt bei den Befehlen PCHL, RST,

```

0128	0584		*		RET, JMP bzw. CALL einen zweiten
0129	0584		*		Breakpoint an der möglichen Ziel-
0130	0584		*		adresse ein (ebenfalls RST2=D7h)
0131	0584		*		
0132	0584	1A		REST12:LDAX D	*OpCode erneut holen
0133	0585	FE E9		CPI 0E9H	*ist es PCHL?
0134	0587	C2 93 05		JNZ RSTCT2	*nein: JMP
0135	058A	2A AD 2F		LHLD REGH	*Zieladresse in H&L
0136	058D	7D		MOV A,L	*H&L vertauschen
0137	058E	6C		MOV L,H	
0138	058F	67		MOV H,A	
0139	0590	C3 C4 05		JMP RSTEND	
0140	0593	CD 09 06		RSTCT2:CALL RSTBEF	*ist es RSTn?
0141	0596	CA A7 05		JZ RSTCT3	*nein: JMP
0142	0599	1A		LDAX D	
0143	059A	E6 38		ANI 038H	*Ziel-Adreßbits maskieren
0144	059C	3C		INR A	*Adresse hinter "C3"
0145	059D	6F		MOV L,A	
0146	059E	26 00		MVI H,0H	*H&L: 01...39h
0147	05A0	5E		MOV E,M	*Zieladresse (lower)
0148	05A1	23		INX H	
0149	05A2	56		MOV D,M	*Zieladresse (upper)
0150	05A3	EB		XCHG	*H&L: RST-Sprungziel
0151	05A4	C3 C4 05		JMP RSTEND	
0152	05A7	CD 01 06		RSTCT3:CALL RETBEF	*ist es RET?
0153	05AA	CA BA 05		JZ RSTCT4	*nein: JMP
0154	05AD	2A AF 2F		LHLD REGS	*User-Stack-Pointer
0155	05B0	7D		MOV A,L	*sortieren
0156	05B1	6C		MOV L,H	*(H&L vertauschen)
0157	05B2	67		MOV H,A	
0158	05B3	5E		MOV E,M	*Stack auslesen
0159	05B4	23		INX H	
0160	05B5	56		MOV D,M	*D&E: RET-Adresse
0161	05B6	EB		XCHG	
0162	05B7	C3 C4 05		JMP RSTEND	
0163	05BA	CD F9 05		RSTCT4:CALL JMPBEF	*ist es JMP/CALL?
0164	05BD	C8		RZ	*nein: RET
0165	05BE	13		INX D	*Zieladresse holen
0166	05BF	1A		LDAX D	
0167	05C0	6F		MOV L,A	*untere Adreßhälfte
0168	05C1	13		INX D	
0169	05C2	1A		LDAX D	
0170	05C3	67		MOV H,A	
0171	05C4	22 D6 2F		RSTEND:SHLD BRKPT2	*Break-Adr.#2 retten
0172	05C7	7E		MOV A,M	*OpCode holen
0173	05C8	32 D8 2F		STA OPCOD2	*und retten
0174	05CB	36 D7		MVI M,0D7H	*"RST2" einsetzen
0175	05CD	C9		RET	
0176	05CE		*		
0177	05CE		*	REST11	ermittelt die Adresse des nächsten
0178	05CE		*		OpCodes (H&L beim Einsprung: Adresse
0179	05CE		*		des aktuellen OpCodes)
0180	05CE		*		
0181	05CE	E5		REST11:PUSH H	
0182	05CF	EB		XCHG	*D&E: Adr.aktueller OpCode
0183	05D0	06 03		MVI B,3H	*Schrittweite Dreiwort-Befehl
0184	05D2	CD F1 05		CALL DRWBEF	*ist es Dreiwort-Befehl?
0185	05D5	C2 E0 05		JNZ RSTCTE	*ja: JMP
0186	05D8	05		DCR B	*Schrittweite Zweiwort-Befehl
0187	05D9	CD E9 05		CALL ZWWBEF	*ist es Zweiwort-Befehl?
0188	05DC	C2 E0 05		JNZ RSTCTE	*ja: JMP
0189	05DF	05		DCR B	*Schrittweite Einwort-Befehl
0190	05E0	48		RSTCTE:MOV C,B	
0191	05E1	06 00		MVI B,0H	*B&C: 1, 2, oder 3

```

0192 05E3 E1 POP H
0193 05E4 09 DAD B *H&L: nächster OpCode
0194 05E5 22 D3 2F SHLD BRKPT1
0195 05E8 C9 RET
0196 05E9 *
0197 05E9 * untersucht OpCode-Typ (Adresse steht
0198 05E9 * beim Einsprung in D&E); ist er in der
0199 05E9 * betreffenden Tabelle enthalten, ist
0200 05E9 * das ZERO-FLAG beim Rücksprung off
0201 05E9 * (Z=0); andernfalls ist Z=1.
0202 05E9 *
0203 05E9 21 50 06 ZWWBEF:LXI H,OP2TAB *Zweiwort-Befehl?
0204 05EC 0E 12 MVI C,12H *Tabellenlänge 18d
0205 05EE C3 0E 06 JMP OPLOP
0206 05F1 21 62 06 DRWBEF:LXI H,OP3TAB *Dreiwort-Befehl?
0207 05F4 0E 1B 14 MVI C,1BH *Tabellenlänge 27d
0208 05F6 C3 0E 06 JMP OPLOP
0209 05F9 21 6A 06 JMPBEF:LXI H,JMPTAB *Jump/Call-Befehl?
0210 05FC 0E 12 MVI C,12H *Tabellenlänge 18d
0211 05FE C3 0E 06 JMP OPLOP
0212 0601 21 7C 06 RETBEF:LXI H,RETAB *Return-Befehl?
0213 0604 0E 09 MVI C,9H *Tabellenlänge 9d
0214 0606 C3 0E 06 JMP OPLOP
0215 0609 21 85 06 RSTBEF:LXI H,RSTAB *Restart-Befehl?
0216 060C 0E 08 MVI C,8H *Tabellenlänge 8d
0217 060E 1A OPLOP: LDAX D *OpCode holen
0218 060F BE CMP M *(REG A)=Tab.wert?
0219 0610 CA 18 06 JZ OPEND *ja: JMP
0220 0613 23 INX H
0221 0614 0D DCR C
0222 0615 C2 0E 06 JNZ OPLOP
0223 0618 AF OPEND: XRA A *REG A löschen
0224 0619 B9 CMP C *Tab. durchlaufen?
0225 061A C9 RET *ja: Z=1
0226 061B *
0227 061B * REST2 ersetzt Breakpoints durch die ur-
0228 061B * sprünglichen OpCodes
0229 061B *
0230 061B 22 CD 2F REST2: SHLD HIBUF *H&L vorübergehend retten
0231 061E 32 CF 2F STA HBUF *REG A ebenfalls
0232 0621 E1 POP H *gleicht RST-Interrupt aus
0233 0622 2B DCX H *wegen Einwort-Interrupt
0234 0623 22 BB 2F SHLD ADRES *aktualisierte Adresse
0235 0626 2A D3 2F LHLD BRKPT1 *Break-Adresse #1 holen
0236 0629 3A D5 2F LDA OPCOD1 *und geretteten OpCode #1
0237 062C 77 MOV M,A *wieder einsetzen
0238 062D 2A D6 2F LHLD BRKPT2 *Break-Adresse #2 holen
0239 0630 3A D8 2F LDA OPCOD2 *und geretteten OpCode #2
0240 0633 77 MOV M,A *wieder einsetzen
0241 0634 3A CF 2F LDA HBUF *REG A laden
0242 0637 2A CD 2F LHLD HIBUF *H&L ebenfalls
0243 063A CD 7F 00 CALL SAVE *sämtliche REGs in RAM-Buffer
0244 063D 21 00 00 LXI H,0H *Break-Adressen löschen
0245 0640 22 D3 2F SHLD BRKPT1
0246 0643 22 D6 2F SHLD BRKPT2
0247 0646 E5 PUSH H *Eintragungen im User-Stack
0248 0647 E5 PUSH H *löschen
0249 0648 E1 POP H
0250 0649 E1 POP H
0251 064A 31 00 2F LXI SP,STACK *Monitor-Stack!
0252 064D C3 D0 2F JMP HIBUF2 *zum Warmstart
0253 0650 *
0254 0650 *****
0255 0650 *

```

Address	Hex	Op	Op Code	Op Name	Notes	
0256	0650	*		Befehlstabellen		
0257	0650	*				
0258	0650	*****				
0259	0650	*				
0260	0650	06	OP2TAB:DB 06H	*MVI B	Zweiwort-Befehle	
0261	0651	0E	DB 0EH	*MVI C		
0262	0652	16	DB 16H	*MVI D		
0263	0653	1E	DB 1EH	*MVI E		
0264	0654	26	DB 26H	*MVI H		
0265	0655	2E	DB 2EH	*MVI L		
0266	0656	36	DB 36H	*MVI m		
0267	0657	3E	DB 3EH	*MVI A		
0268	0658	*				
0269	0658	C6	DB 0C6H	*ADI		
0270	0659	CE	DB 0CEH	*ACI		
0271	065A	D6	DB 0D6H	*SUI		
0272	065B	DE	DB 0DEH	*SBI		
0273	065C	E6	DB 0E6H	*ANI		
0274	065D	EE	DB 0EEH	*XRI		
0275	065E	F6	DB 0F6H	*ORI		
0276	065F	FE	DB 0FEH	*CPI		
0277	0660	*				
0278	0660	D3	DB 0D3H	*OUT		
0279	0661	DB	DB 0DBH	*IN		
0280	0662	*				
0281	0662	01	OP3TAB:DB 01H	*LXI B	Dreiwort-Befehle	
0282	0663	11	DB 11H	*LXI D		
0283	0664	21	DB 21H	*LXI H		
0284	0665	22	DB 22H	*SHLD		
0285	0666	2A	DB 2AH	*LHLD		
0286	0667	31	DB 31H	*LXI SP		
0287	0668	32	DB 32H	*STA		
0288	0669	3A	DB 3AH	*LDA		
0289	066A	*				
0290	066A	C2	JMPTAB:DB 0C2H	*JNZ	Sprungbefehle	
0291	066B	C3	DB 0C3H	*JMP uncond.		
0292	066C	CA	DB 0CAH	*JZ		
0293	066D	D2	DB 0D2H	*JNC		
0294	066E	DA	DB 0DAH	*JC		
0295	066F	E2	DB 0E2H	*JPO		
0296	0670	EA	DB 0EAH	*JPE		
0297	0671	F2	DB 0F2H	*JP		
0298	0672	FA	DB 0FAH	*JM		
0299	0673	*				
0300	0673	C4	CALTAB:DB 0C4H	*CNZ	Aufruf v.Unt.progr.	
0301	0674	CC	DB 0CCH	*CZ		
0302	0675	CD	DB 0CDH	*CALL uncond.		
0303	0676	D4	DB 0D4H	*CNC		
0304	0677	DC	DB 0DCH	*CC		
0305	0678	E4	DB 0E4H	*CPO		
0306	0679	EC	DB 0ECH	*CPE		
0307	067A	F4	DB 0F4H	*CP		
0308	067B	FC	DB 0FCH	*CM		
0309	067C	*				
0310	067C	C0	RETAB: DB 0C0H	*RNZ	Rückspr.v.Unt.progr.	
0311	067D	C8	DB 0C8H	*RZ		
0312	067E	C9	DB 0C9H	*RET uncond.		
0313	067F	D0	DB 0D0H	*RNC		
0314	0680	D8	DB 0D8H	*RC		
0315	0681	E0	DB 0E0H	*RPO		
0316	0682	E8	DB 0E8H	*RPE		
0317	0683	F0	DB 0F0H	*RP		
0318	0684	F8	DB 0F8H	*RM		
0319	0685	*				

0320	0685	C7	RSTAB:	DB	0C7H	*RST0
0321	0686	CF		DB	0CFH	*RST1
0322	0687	D7		DB	0D7H	*RST2
0323	0688	DF		DB	0DFH	*RST3
0324	0689	E7		DB	0E7H	*RST4
0325	068A	EF		DB	0EFH	*RST5
0326	068B	F7		DB	0F7H	*RST6
0327	068C	FF		DB	0FFH	*RST7
0328	068D		*			
0329	068D		*			
0330	068D		*****			
0331	068D		*			
0332	068D		*	RAM-Verwaltung		
0333	068D		*			
0334	068D		*****			
0335	068D		*			
0336	068D		*	EXCHAN	kopiert Speicherbereiche um und	
0337	068D		*		paßt dabei diejenigen Adressen	
0338	068D		*		an, die im kopierten Block	
0339	068D		*		liegen.	
0340	068D		*			
0341	068D	2A BB 2F	FCTXCH:	LHLD	ADRES	
0342	0690	22 C3 2F		SHLD	DSTBEG	*Zieladresse
0343	0693	CD 9C 06		CALL	EXCHA-	
0344	0696	2A C3 2F		LHLD	DSTBEG	*Zieladresse holen
0345	0699	C3 76 00		JMP	DATSET	*und Daten-Mode einstellen
0346	069C		*			
0347	069C	CD 88 00	EXCHA-:	CALL	COPY	*umkopieren
0348	069F	2A BF 2F		LHLD	SRCBEG	*Adreßdifferenz bilden
0349	06A2	EB		XCHG		
0350	06A3	2A C3 2F		LHLD	DSTBEG	
0351	06A6	22 D3 2F		SHLD	BRKPT1	*Anfang Zielbereich
0352	06A9	7D		MOV	A,L	
0353	06AA	93		SUB	E	
0354	06AB	6F		MOV	L,A	*Differenz (lower)
0355	06AC	7C		MOV	A,H	
0356	06AD	9A		SBB	D	
0357	06AE	67		MOV	H,A	*Differenz (upper)
0358	06AF	22 C5 2F		SHLD	DSTEND	
0359	06B2	EB		XCHG		
0360	06B3	2A C1 2F		LHLD	SRCEND	
0361	06B6	19		DAD	D	*Ende im Zielbereich
0362	06B7	22 D6 2F		SHLD	BRKPT2	*für Endabfrage ablegen
0363	06BA	2A D3 2F	EXLOP:	LHLD	BRKPT1	*aktueller OpCode
0364	06BD	22 CD 2F		SHLD	HIBUF	*Adresse retten
0365	06C0	CD CE 05		CALL	REST11	*nächst.OpCode n.BRKPT1
0366	06C3	EB		XCHG		*D&E: Adr.nächster OpCode
0367	06C4	2A D6 2F		LHLD	BRKPT2	*H&L: Endadresse Zielbereich
0368	06C7	CD DC 06		CALL	COMP16	*Ende?
0369	06CA	DO		RNC		*ja: dann Rücksprung
0370	06CB	2A CD 2F		LHLD	HIBUF	*Adresse zurückholen
0371	06CE	EB		XCHG		
0372	06CF	CD F1 05		CALL	DRWBEF	*ist es Dreiwort-Befehl?
0373	06D2	C4 E2 06		CNZ	MODIFY	*ja: CALL
0374	06D5	C3 BA 06		JMP	EXLOP	
0375	06D8		*			
0376	06D8		*	CMPEND	vergleicht (H&L) mit SRCEND; ist	
0377	06D8		*		(H&L) größer (=Bereichsende),	
0378	06D8		*		ist beim Rücksprung CY=1	
0379	06D8		*			
0380	06D8	EB	CMPEND:	XCHG		*D&E: RAM-Adresse
0381	06D9	2A C1 2F		LHLD	SRCEND	
0382	06DC	7A	COMP16:	MOV	A,D	*16-Bit-Vergleich
0383	06DD	BC		CMP	H	*(H&L) mit (D&E)

0384	06DE	D8		RC		*Adr. größer SRCEND
0385	06DF	7B		MOV	A,E	
0386	06E0	BD		CMP	L	*Adr. gleich SRCEND: Z=1
0387	06E1	C9		RET		*Adr. größer SRCEND: CY=1
0388	06E2		*			
0389	06E2		*	MODIFY		modifiziert die in einem Dreiwort-
0390	06E2		*			Befehl enthaltene Absolutadresse,
0391	06E2		*			wenn diese im Bereich SRCBEG...
0392	06E2		*			SRCEND liegt
0393	06E2		*			
0394	06E2	13		MODIFY:INX	D	*D&E: Adr. des akt. OpCodes
0395	06E3	1A		LDAX	D	*Absolutadresse (lower)
0396	06E4	6F		MOV	L,A	
0397	06E5	13		INX	D	
0398	06E6	1A		LDAX	D	*Absolutadresse (upper)
0399	06E7	67		MOV	H,A	*H&L: Absolutadresse
0400	06E8	CD D8 06		CALL	COMPEND	*Vergleich mit SRCEND
0401	06EB	D0		RNC		*größer als SRCEND: RET
0402	06EC	2A BF 2F		LHLD	SRCBEG	*D&E: Absolutadresse
0403	06EF	CD DC 06		CALL	COMP16	*Vergleich mit SRCBEG
0404	06F2	CA F6 06		JZ	MOD	*Gleichh.: auch modifiz.!
0405	06F5	D8		RC		*kleiner als SRCBEG: RET
0406	06F6	2A C5 2F MOD:		LHLD	DSTEND	*Adreßdifferenz
0407	06F9	19		DAD	D	*zur Absol.adr. addieren
0408	06FA	EB		XCHG		*D&E: modifiz. Adresse
0409	06FB	2A CD 2F		LHLD	HIBUF	*aktueller OpCode
0410	06FE	23		INX	H	*zweites Byte im Befehl
0411	06FF	73		MOV	M,E	*modifiz.Adr. (lower)
0412	0700	23		INX	H	*drittes Byte im Befehl
0413	0701	72		MOV	M,D	*modifiz.Adr. (upper)
0414	0702	C9		RET		
0415	0703		*			
0416	0703		*			
0417	0703		*	INSERT		Fügt an der aktuellen RAM-Adresse
0418	0703		*			NOPs (=00) ein (RAM-Ende beachten!)
0419	0703		*			hierbei keine Adreßanpassung!
0420	0703		*			
0421	0703	3A BB 2F	FCTINS:	LDA	ADRES	*Anzahl holen
0422	0706	F5		PUSH	PSW	
0423	0707	5F		MOV	E,A	
0424	0708	16 00		MVI	D,0H	
0425	070A	2A BF 2F		LHLD	SRCBEG	
0426	070D	E5		PUSH	H	
0427	070E	19		DAD	D	*Zieladresse
0428	070F	22 C3 2F		SHLD	DSTBEG	
0429	0712	21 00 2E		LXI	H, RAMEND	
0430	0715	22 C1 2F		SHLD	SRCEND	
0431	0718	CD 56 07		CALL	COPY-	*umkopieren
0432	071B	E1		POP	H	
0433	071C	F1		POP	PSW	
0434	071D	E5		PUSH	H	
0435	071E	36 00	INSLOP:	MVI	M,0H	*NOP einfügen
0436	0720	23		INX	H	
0437	0721	3D		DCR	A	
0438	0722	C2 1E 07		JNZ	INSLOP	
0439	0725	E1		POP	H	
0440	0726	C3 76 00		JMP	DATSET	*Daten-Mode
0441	0729		*			
0442	0729		*			
0443	0729		*	DELETE		Löscht RAM-Zellen und rückt
0444	0729		*			RAM-Inhalt nach (bis RAM-Ende)
0445	0729		*			hierbei keine Adreßanpassung!
0446	0729		*			
0447	0729	3A BB 2F	FCTDEL:	LDA	ADRES	*Anzahl holen

```

0448 072C 5F          MOV  E,A
0449 072D 16 00      MVI  D,0H
0450 072F 2A BF 2F   LHLD SRCBEG
0451 0732 22 C3 2F   SHLD DSTBEG
0452 0735 E5             PUSH H
0453 0736 19             DAD  D          *Startadresse
0454 0737 22 BF 2F   SHLD SRCBEG
0455 073A 21 00 2E   LXI  H,RAMEND
0456 073D 22 C1 2F   SHLD SRCEND
0457 0740 CD 56 07   CALL COPY-     *umkopieren
0458 0743 E1             POP  H
0459 0744 C3 76 00      JMP  DATSET     *Daten-Mode
0460 0747
0461 0747
0462 0747 *****
0463 0747
0464 0747 *          RAM-Bereiche verschieben
0465 0747
0466 0747 *****
0467 0747
0468 0747 2A BB 2F FCTCPY: LHLD ADRES   *dritter Parameter
0469 074A 22 C3 2F          SHLD DSTBEG
0470 074D CD 56 07          CALL COPY-     *kopieren ausführen
0471 0750 2A C3 2F          LHLD DSTBEG   *Zieladresse holen
0472 0753 C3 76 00          JMP  DATSET   *und Daten-Mode einstellen
0473 0756
0474 0756 *          COPY          kopiert Speicherbereiche
0475 0756
0476 0756 *          SRCBEG:      Anfang Ursprungsbereich
0477 0756 *          SRCEND:      Ende Ursprungsbereich
0478 0756 *          DSTBEG:      Anfang Zielbereich
0479 0756
0480 0756 2A C1 2F COPY-: LHLD SRCEND
0481 0759 EB             XCHG          *D&E: Ende Quellbereich
0482 075A 2A C3 2F          LHLD DSTBEG   *H&L: Anf. Zielberich
0483 075D CD A9 07          CALL MAXCOM   *Vergleich D&E mit H&L
0484 0760 44             MOV  B,H
0485 0761 4D             MOV  C,L      *B&C: Anf. Zielbereich
0486 0762 2A BF 2F          LHLD SRCBEG   *H&L: Anf. Quellbereich
0487 0765 D2 98 07          JNC  COPYUP
0488 0768 50             MOV  D,B
0489 0769 59             MOV  E,C      *D&E: Anf. Zielbereich
0490 076A CD A9 07          CALL MAXCOM   *Vergleich D&E mit H&L
0491 076D E5             PUSH H
0492 076E 2A C1 2F          LHLD SRCEND
0493 0771 EB             XCHG          *D&E: Ende Quellbereich
0494 0772 E1             POP  H        *H&L: Anf. Quellbereich
0495 0773 D2 98 07          JNC  COPYUP
0496 0776
0497 0776 D5             PUSH D
0498 0777 E5             PUSH H
0499 0778 7D             MOV  A,L      *DSTEND berechnen
0500 0779 2F             CMA
0501 077A 6F             MOV  L,A
0502 077B 7C             MOV  A,H
0503 077C 2F             CMA
0504 077D 67             MOV  H,A
0505 077E EB             XCHG          *D&E: Kompl.Anf.Quellber.
0506 077F 2A C1 2F          LHLD SRCEND
0507 0782 19             DAD  D
0508 0783 EB             XCHG          *D&E: Diff.SRCEND - SRCBEG
0509 0784 2A C3 2F          LHLD DSTBEG
0510 0787 19             DAD  D
0511 0788 23             INX  H        *H&L: Ende Zielbereich

```

```

0512 0789 44          MOV  B,H
0513 078A 4D          MOV  C,L
0514 078B D1          POP  D
0515 078C E1          POP  H
0516 078D          *
0517 078D 7E          COPYDW:MOV A,M      *abwärts kopieren
0518 078E 02          STAX B
0519 078F CD A3 07    CALL COMP
0520 0792 C8          RZ          *fertig: RET
0521 0793 2B          DCX  H
0522 0794 0B          DCX  B
0523 0795 C3 8D 07    JMP  COPYDW
0524 0798          *
0525 0798 7E          COPYUP:MOV A,M     *aufwärts kopieren
0526 0799 02          STAX B
0527 079A CD A3 07    CALL COMP
0528 079D C8          RZ          *fertig: RET
0529 079E 23          INX  H
0530 079F 03          INX  B
0531 07A0 C3 98 07    JMP  COPYUP
0532 07A3          *
0533 07A3 7D          COMP:  MOV  A,L
0534 07A4 BB          CMP   E
0535 07A5 C0          RNZ          *ungleich: RET
0536 07A6 7C          MOV  A,H
0537 07A7 BA          CMP   D      *gleich: Z-FLAG on
0538 07A8 C9          RET
0539 07A9          *
0540 07A9 7C          MAXCOM:MOV A,H
0541 07AA BA          CMP   D
0542 07AB D8          RC          *D&E größer H&L: RET
0543 07AC C0          RNZ
0544 07AD 7D          MOV  A,L
0545 07AE BB          CMP   E
0546 07AF C9          RET          *D&E größer H&L: CY on
0547 07B0          *
0548 07B0          *
0549 07B0          *****
0550 07B0          *
0551 07B0          *      RAM mit Konstante auffüllen
0552 07B0          *
0553 07B0          *****
0554 07B0          *
0555 07B0 3A BB 2F FCTFIL:LDA ADRES  *Konstante holen
0556 07B3 32 C3 2F          STA  DSTBEG
0557 07B6 CD BF 07    CALL FILL-  *Funktion ausführen
0558 07B9 2A BF 2F          LHLD SRCBEG *Startadresse holen
0559 07BC C3 76 00          JMP  DATSET  *und Daten-Mode einstellen
0560 07BF          *
0561 07BF          *      FILL      füllt RAM mit Konstante auf
0562 07BF          *
0563 07BF          *      SRCBEG: Anfangsadresse
0564 07BF          *      SRCEND: Endadresse
0565 07BF          *      DSTBEG: Konstante
0566 07BF          *
0567 07BF 2A C1 2F FILL-: LHLD SRCEND
0568 07C2 EB          XCHG      *D&E: Endadresse
0569 07C3 2A BF 2F          LHLD SRCBEG *H&L: Anfangsadresse
0570 07C6 3A C3 2F          LDA  DSTBEG
0571 07C9 47          MOV  B,A   *REG B: Konstante
0572 07CA 70          FILOP: MOV  M,B
0573 07CB CD A3 07    CALL COMP  *Ende?
0574 07CE 23          INX  H
0575 07CF C8          RZ          *ja: RET

```

```

0576 07D0 C3 CA 07 JMP FILOP
0577 07D3 *
0578 07D3 * MAXRAM ermittelt die höchste RAM-Adresse
0579 07D3 * (ab 8000h) und übergibt sie in
0580 07D3 * den Registern A und B (A=upper)
0581 07D3 *
0582 07D3 21 00 80 MAXRA-:LXI H,8000H *Startadresse
0583 07D6 11 FD 00 LXI D,0FDH *Schrittweite
0584 07D9 06 49 MVI B,49H *Konstante
0585 07DB 19 MAXLOP:DAD D
0586 07DC 4E MOV C,M *RAM-Inhalt retten
0587 07DD 70 MOV M,B *Konstante ins RAM
0588 07DE 7E MOV A,M *Prüflesen
0589 07DF 71 MOV M,C *alten Inhalt zurück
0590 07E0 B8 CMP B *Einschreiben erfolgreich?
0591 07E1 CA DB 07 JZ MAXLOP *ja: weiter
0592 07E4 7C MOV A,H
0593 07E5 3D DCR A *Max-Adr. (upper)
0594 07E6 06 FF MVI B,0FFH *Max-Adr. (lower)
0595 07E8 C9 RET
0596 07E9 *
0597 07E9 *
0598 07E9 END

```

```

0001 0000 *****
0002 0000 *
0003 0000 * EZU/DSP Echtzeit-Uhr- und SSG-Anzeige 24.10.85 *
0004 0000 *
0005 0000 *****
0006 0000 *
0007 0000 *
0008 0000 ORG      800H
0009 0800 OFS      800H
0010 0800 *
0011 0800 *
0012 0800 *****
0013 0800 *
0014 0800 *           Stellen der Echtzeit-Uhr, danach auslesen
0015 0800 *
0016 0800 *****
0017 0800 *
0018 0800 2A BB 2F FCTWUR:LHLD ADRES      *letzte Eing.(Mon/Jahr)
0019 0803 EB          XCHG              *nach D&E
0020 0804 2A C1 2F LHLD SRCEND        *vorl. Eing.(Wotag,Kaltag)
0021 0807 4D          MOV C,L          *nach B&C
0022 0808 44          MOV B,H
0023 0809 2A BF 2F LHLD SRCBEG        *erste Eing.(Std,Min)
0024 080C CD 12 08 CALL WRITU-      *Zeit einschreiben
0025 080F C3 6D 08 JMP FCTRUR      *und anzeigen
0026 0812 *
0027 0812 *           WRITUR      Stellt die 5832-Echtzeit-Uhr
0028 0812 *
0029 0812 *           (H&L):      Stunden, Minuten (Sekunden=00)
0030 0812 *           (B&C):      Wochentag (1=Mo), Kalendertag
0031 0812 *           (D&E):      Monat, Jahres-Einer und -Zehner
0032 0812 *
0033 0812 D5          WRITU-: PUSH D      *MON/JAR RETTEN
0034 0813 C5          PUSH B      *WTG/KTG RETTEN
0035 0814 7C          MOV A,H
0036 0815 F6 80      ORI 80H      *24-h-Format
0037 0817 65          MOV H,L
0038 0818 6F          MOV L,A      *H&L=MIN/STD
0039 0819 0E 08      MVI C,8H      *ADRESSE SEKUNDEN-EINER
0040 081B 79          MOV A,C      *HOLD=H; WRITE=L
0041 081C D3 19      OUT EZUCTL
0042 081E CD 0B 00  CALL DELY1      *Wartezeit 1 ms (150 us)
0043 0821 E5          PUSH H      *MIN/STD RETTEN
0044 0822 2E 00      MVI L,0H      *SEK = 00
0045 0824 CD 45 08  CALL LOUT      *SEKUNDEN LÖSCHEN
0046 0827 E1          POP H      *MIN/STD ZURÜCKHOLEN
0047 0828 CD 41 08  CALL HLOUT     *MIN/STD
0048 082B D1          POP D      *WTG/KTG ZURÜCKHOLEN
0049 082C EB          XCHG
0050 082D 7C          MOV A,H
0051 082E E6 07      ANI 7H
0052 0830 3D          DCR A      *MSM-WTG-FORMAT 0...6
0053 0831 CD 5B 08  CALL OUTPT     *WTG
0054 0834 CD 45 08  CALL LOUT      *KTG
0055 0837 D1          POP D      *MON/JAR ZURÜCKHOLEN
0056 0838 EB          XCHG
0057 0839 CD 41 08  CALL HLOUT     *MON/JAR
0058 083C 3E F4      MVI A,OF4H    *HOLD=L; READ=H
0059 083E D3 19      OUT EZUCTL
0060 0840 C9          RET
0061 0841 *
0062 0841 7C          HLOUT: MOV A,H
0063 0842 CD 4A 08  CALL OUTPUT

```

```

0064 0845 7D          LOUT:  MOV      A,L
0065 0846 CD 4A 08      CALL    OUTPUT
0066 0849 C9              RET
0067 084A          *
0068 084A F5          OUTPUT: PUSH    PSW
0069 084B E6 0F      ANI     OFH
0070 084D CD 5B 08      CALL    OUTPT
0071 0850 F1          POP     PSW
0072 0851 0F          RRC
0073 0852 0F          RRC
0074 0853 0F          RRC
0075 0854 0F          RRC
0076 0855 E6 0F      ANI     OFH
0077 0857 CD 5B 08      CALL    OUTPT
0078 085A C9              RET
0079 085B          *
0080 085B D3 18      OUTPT:  OUT     EZUDAT  *DATEN AUSGEBEN
0081 085D 79          MOV     A,C
0082 085E D3 19      OUT     EZUCTL  *ADRESSE AUSGEBEN
0083 0860 F6 02      ORI     2H      *H-PULS AN WRITE
0084 0862 D3 19      OUT     EZUCTL
0085 0864 E6 FD      ANI     OFDH
0086 0866 D3 19      OUT     EZUCTL
0087 0868 3E 10      MVI     A,10H   *ADRESSE ERHÖHEN
0088 086A 81          ADD     C
0089 086B 4F          MOV     C,A
0090 086C C9              RET
0091 086D          *
0092 086D          *****
0093 086D          *
0094 086D          *      Echtzeit-Uhr auslesen und anzeigen
0095 086D          *
0096 086D          *****
0097 086D          *
0098 086D CD F8 08      FCTRUR:CALL EZUDS-  *EZU auslesen und anzeigen
0099 0870 C3 6D 08      JMP    FCTRUR
0100 0873          *
0101 0873          *      READUR      Liest die 5832-Echtzeit-Uhr aus
0102 0873          *              und überschreibt Uhrzeit und Da-
0103 0873          *              tum in den UHRBUF ins RAM
0104 0873          *
0105 0873          *
0106 0873 21 DC 2F      READU-: LXI     H,UHRBUF
0107 0874 E5          PUSH    H
0108 0877 EB          XCHG   *RAM-ANFANG NACH D,E
0109 0878 0E 0C      MVI     C,0CH  *ANFANGSADRESSE 5832
0110 087A 3E 08      MVI     A,8H   *HOLD=H; READ=L
0111 087C D3 19      OUT     EZUCTL
0112 087E CD 0B 00      CALL    DELY1  *Wartezeit 1 ms (150 us)
0113 0881          *
0114 0881 79          INLOP: MOV     A,C   *ADRESSE HOLEN
0115 0882 D3 19      OUT     EZUCTL *UND AUSGEBEN
0116 0884 E5          PUSH    H      *Wartezeit von 6 us
0117 0885 E1          POP     H
0118 0886 DB 18      IN     EZUDAT  *DATEN EINLESEN
0119 0888 12          STAX   D
0120 0889 13          INX   D
0121 088A 3E 10      MVI     A,10H  *ADRESSE ERHÖHEN
0122 088C 81          ADD     C
0123 088D 4F          MOV     C,A
0124 088E FE DC      CPI     ODCH   *ENDABFRAGE
0125 0890 C2 81 08      JNZ    INLOP
0126 0893 E1          POP     H
0127 0894 CD 9C 08      CALL    SORT   *FÜR UHRBUF FORMATIEREN

```

0128	0897	3E F4		MVI	A,OF4H	*HOLD=L; READ=H	
0129	0899	D3 19		OUT	EZUCTL		
0130	089B	C9		RET			
0131	089C						
0132	089C	E5		* SORT:	PUSH	H	*RAM-ANF.ADR. RETTEN
0133	089D	01 06 00			LXI	B,6H	*OFFSET WOCHENTAG
0134	08A0	09			DAD	B	
0135	08A1	46			MOV	B,M	*WOCHENTAG RETTEN
0136	08A2	E1			POP	H	
0137	08A3	E5			PUSH	H	
0138	08A4	EB			XCHG		*QUELL-POINTER: D,E
0139	08A5	E1			POP	H	
0140	08A6	E5			PUSH	H	*ZIEL-POINTER: H,L
0141	08A7	CD F1 08			CALL	BIT4	*SEKUNDEN-EINER
0142	08AA	CD EB 08			CALL	BIT3	*SEKUNDEN-ZEHNER
0143	08AD	CD F1 08			CALL	BIT4	*MINUTEN-EINER
0144	08B0	CD EB 08			CALL	BIT3	*MINUTEN-ZEHNER
0145	08B3	CD F1 08			CALL	BIT4	*STUNDEN-EINER
0146	08B6	1A			LDAX	D	*STUNDEN-ZEHNER
0147	08B7	E6 03			ANI	3H	
0148	08B9	77			MOV	M,A	
0149	08BA	13			INX	D	
0150	08BB	13			INX	D	
0151	08BC	23			INX	H	
0152	08BD	CD F1 08			CALL	BIT4	*KAL'TAG-EINER
0153	08C0	1A			LDAX	D	*KAL'TAG ZEHNER
0154	08C1	E6 03			ANI	3H	
0155	08C3	77			MOV	M,A	
0156	08C4	13			INX	D	
0157	08C5	23			INX	H	
0158	08C6	23			INX	H	
0159	08C7	CD F1 08			CALL	BIT4	*MONATS-EINER
0160	08CA	1A			LDAX	D	*MONATS-ZEHNER
0161	08CB	E6 01			ANI	1H	
0162	08CD	77			MOV	M,A	
0163	08CE	13			INX	D	
0164	08CF	23			INX	H	
0165	08D0	CD F1 08			CALL	BIT4	*JAHRES-EINER
0166	08D3	CD F1 08			CALL	BIT4	*JAHRES-ZEHNER
0167	08D6	1B			DCX		
0168	08D7	0E 04			MVI	C,4H	
0169	08D9	1A		SHIFT:	LDAX	D	
0170	08DA	77			MOV	M,A	
0171	08DB	1B			DCX	D	
0172	08DC	2B			DCX	H	
0173	08DD	0D			DCR	C	
0174	08DE	C2 D9 08			JNZ	SHIFT	
0175	08E1	78			MOV	A,B	*WOCHENTAG ZURÜCKHOLEN
0176	08E2	E6 07			ANI	7H	
0177	08E4	3C			INR	A	*NORMALFORMAT WTG
0178	08E5	36 00			MVI	M,0H	*WTG-DUMMY
0179	08E7	2B			DCX	H	
0180	08E8	77			MOV	M,A	
0181	08E9	E1			POP	H	
0182	08EA	C9			RET		
0183	08EB			* BIT3:			
0184	08EB	1A			LDAX	D	*BYTE HOLEN
0185	08EC	E6 07			ANI	7H	*3 LSB MASKIEREN
0186	08EE	C3 F4 08			JMP	BITCTE	
0187	08F1	1A		BIT4:	LDAX	D	*BYTE HOLEN
0188	08F2	E6 0F			ANI	0FH	*4 LSB MASKIEREN
0189	08F4	77		BITCTE:	MOV	M,A	
0190	08F5	13			INX	D	*POINTER ERHÖHEN
0191	08F6	23			INX	H	

```

0192 08F7 C9 RET
0193 08F8 *
0194 08F8 *
0195 08F8 * EZUDSP Uhrzeit in die Siebensegment-
0196 08F8 * Anzeige überschreiben
0197 08F8 *
0198 08F8 CD 73 08 EZUDS-:CALL READU- *Uhrzeit auslesen
0199 08FB 11 B3 2F LXI D,DIG1
0200 08FE CD 18 09 CALL EZUSUB *Sekunden umsetzen
0201 0901 13 INX D *DIG3 überspringen
0202 0902 CD 18 09 CALL EZUSUB *Minuten umsetzen
0203 0905 CD 18 09 CALL EZUSUB *Stunden umsetzen
0204 0908 3A B8 2F LDA DIG6
0205 090B F6 80 ORI 80H *DP einblenden
0206 090D 32 B8 2F STA DIG6
0207 0910 CD 13 00 CALL DLY100 *SSG-Buffer anzeigen
0208 0913 AF XRA A *REG A löschen
0209 0914 32 B2 2F STA DSPBUF *LEDs dunkel
0210 0917 C9 RET
0211 0918 *
0212 0918 46 EZUSUB:MOV B,M *erstes BCD-Digit
0213 0919 23 INX H
0214 091A 7E MOV A,M *zweites BCD-Digit
0215 091B 23 INX H
0216 091C 07 RLC
0217 091D 07 RLC
0218 091E 07 RLC
0219 091F 07 RLC
0220 0920 B0 ORA B *beide verknüpfen
0221 0921 E5 PUSH H
0222 0922 CD 67 00 CALL CNVBYT *und in SSG-Code umsetzen
0223 0925 E1 POP H
0224 0926 C9 RET
0225 0927 *
0226 0927 *
0227 0927 * TIME Liest die 5832-Echtzeit-Uhr in
0228 0927 * RAM-Hilfsbuffer aus (in den Uhr-
0229 0927 * zeit-Buffer UHRBUF) und setzt
0230 0927 * die Uhrzeit- und Datums-Informa-
0231 0927 * tion in den ASCII-Code um (ab
0232 0927 * 2FEAh, durch "00" abgeschlossen)
0233 0927 * H&L steht anschl.auf Stringanfang
0234 0927 *
0235 0927 CD 73 08 TIME-: CALL READU- *EZU AUSLESEN
0236 092A EB XCHG *D&E: UHRBUF
0237 092B 1B DCX D
0238 092C 21 FF 2F LXI H,2FFFH *ENDE ASCII-BUF
0239 092F 36 00 MVI M,0H *00 = STRING-ENDE
0240 0931 2B DCX H
0241 0932 36 68 MVI M,"h"
0242 0934 CD 78 09 CALL ASCII2 *SEKUNDEN
0243 0937 2B DCX H
0244 0938 36 3A MVI M,3AH *": "
0245 093A CD 78 09 CALL ASCII2 *MINUTEN
0246 093D 2B DCX H
0247 093E 36 3A MVI M,3AH *": "
0248 0940 CD 78 09 CALL ASCII2 *STUNDEN
0249 0943 2B DCX H
0250 0944 36 3B MVI M,";"
0251 0946 11 E7 2F LXI D,MONZ
0252 0949 CD 78 09 CALL ASCII2 *JAHR
0253 094C 2B DCX H
0254 094D 36 2E MVI M,"."
0255 094F 11 E5 2F LXI D,WTGU

```


0256	0952	CD 78 09	CALL	ASCII2	*MONAT
0257	0955	2B	DCX	H	
0258	0956	36 2E	MVI	M, "."	
0259	0958	11 E1 2F	LXI	D, STDZ	
0260	095B	CD 78 09	CALL	ASCII2	*KALENDERTAG
0261	095E	2B	DCX	H	
0262	095F	36 2C	MVI	M, 2CH	*", "
0263	0961	2B	DCX	H	
0264	0962	E5	PUSH	H	
0265	0963	21 8E 09	LXI	H, WTGTAB	
0266	0966	3A E4 2F	LDA	WTGL	*WTG-Nr.
0267	0969	E6 07	ANI	7H	
0268	096B	07	RLC		
0269	096C	5F	MOV	E, A	
0270	096D	16 00	MVI	D, 0H	
0271	096F	19	DAD	D	
0272	0970	5E	MOV	E, M	*WTG-SYMBOL (LOWER)
0273	0971	23	INX	H	
0274	0972	56	MOV	D, M	*WTG-SYMBOL (UPPER)
0275	0973	E1	POP	H	
0276	0974	73	MOV	M, E	
0277	0975	2B	DCX	H	
0278	0976	72	MOV	M, D	
0279	0977	C9	RET		
0280	0978		*		
0281	0978		*		
0282	0978		*		
0283	0978	CD 7B 09	ASCII2: CALL	ASCII1	
0284	097B	2B	ASCII1: DCX	H	
0285	097C	13	INX	D	
0286	097D	1A	LDAX	D	
0287	097E	C6 30	ADI	30H	*ASCII-Umsetzung
0288	0980	FE 3A	CPI	3AH	*"9" + 1
0289	0982	D2 8A 09	JNC	STROKE	*größer 9: JMP
0290	0985	FE 30	CPI	"0"	
0291	0987	D2 8C 09	JNC	ASCTE	*größer/gleich 0: JMP
0292	098A	3E 2D	STROKE: MVI	A, "-"	
0293	098C	77	ASCTE: MOV	M, A	*Nibble in ASCBUF
0294	098D	C9	RET		
0295	098E		*		
0296	098E	00	WTGTAB: NOP		
0297	098F	00	NOP		
0298	0990	6F	DB	"o"	
0299	0991	4D	DB	"M"	
0300	0992	69	DB	"i"	
0301	0993	44	DB	"D"	
0302	0994	69	DB	"i"	
0303	0995	4D	DB	"M"	
0304	0996	6F	DB	"o"	
0305	0997	44	DB	"D"	
0306	0998	72	DB	"r"	
0307	0999	46	DB	"F"	
0308	099A	61	DB	"a"	
0309	099B	53	DB	"S"	
0310	099C	6F	DB	"o"	
0311	099D	53	DB	"S"	
0312	099E		*		
0313	099E		*		
0314	099E		*		
0315	099E		*	DSPACC	Akku-Inhalt im Siebensegment-Code und binär in LED-Zeile darstellen; Warmstart bei Tastendruck
0316	099E		*		
0317	099E		*		
0318	099E	F5	DSPAC-: PUSH PSW		
0319	099F	11 B3 2F	LXI	D, DIG1	

```

0320 09A2 CD 67 00 CALL CNVBYT *SSG-Umsetzung
0321 09A5 F1 POP PSW
0322 09A6 32 B2 2F STA DSPBUF *(ACC) in LED-Zeile
0323 09A9 21 00 00 LXI H,0H
0324 09AC 22 B6 2F SHLD DIG4 *obere Digits dunkel
0325 09AF 22 B8 2F SHLD DIG6
0326 09B2 CD 6D 00 DSPLOP:CALL PRESS *auf Taste warten
0327 09B5 C3 B2 09 JMP DSPLOP
0328 09B8 *
0329 09B8 *
0330 09B8 * DSPH&L Inhalt von H&L im Siebensegment-
0331 09B8 * Code darstellen; Warmstart bei
0332 09B8 * Tastendruck
0333 09B8 *
0334 09B8 E5 DSPHL-:PUSH H *H&L retten
0335 09B9 11 B6 2F LXI D,DIG4 *rechte Adreßanzeige
0336 09BC 7D MOV A,L *(REG L) SSG umsetzen
0337 09BD CD 67 00 CALL CNVBYT
0338 09C0 E1 POP H *H&L wiederholen
0339 09C1 7C MOV A,H *(REG H) SSG umsetzen
0340 09C2 CD 67 00 CALL CNVBYT
0341 09C5 21 00 00 DSPCTE:LXI H,0H *untere Digits dunkel
0342 09C8 22 B3 2F SHLD DIG1
0343 09CB C3 B2 09 JMP DSPLOP
0344 09CE *
0345 09CE *
0346 09CE * DSPERR Fehleranzeige "ErrX"
0347 09CE * bei Tastendruck zurück zum
0348 09CE * Warmstart
0349 09CE *
0350 09CE 11 B6 2F DSPER-:LXI D,DIG4 *rechte Adreßanzeige
0351 09D1 CD 67 00 CALL CNVBYT *Fehler-Nummer
0352 09D4 21 B9 2F LXI H,DIG7
0353 09D7 36 79 MVI M,79H * "E"
0354 09D9 2B DCX H
0355 09DA 36 50 MVI M,50H * "r"
0356 09DC 2B DCX H
0357 09DD 36 50 MVI M,50H * "r"
0358 09DF C3 C5 09 JMP DSPCTE
0359 09E2 *
0360 09E2 *
0361 09E2 * BCD-Uhrzeit-Buffer
0362 09E2 *
0363 09E2 ORG UHRBUF
0364 09E2 *
0365 2FDC SEKE: DS 1H
0366 2FDD SEKZ: DS 1H
0367 2FDE MINE: DS 1H
0368 2FDF MINZ: DS 1H
0369 2FE0 STDE: DS 1H
0370 2FE1 STDZ: DS 1H
0371 2FE2 KLTE: DS 1H
0372 2FE3 KLTZ: DS 1H
0373 2FE4 WTGL: DS 1H
0374 2FE5 WTGU: DS 1H
0375 2FE6 MONE: DS 1H
0376 2FE7 MONZ: DS 1H
0377 2FE8 JARE: DS 1H
0378 2FE9 JARZ: DS 1H
0379 2FE9 *
0380 2FE9 * ASCII-Uhrzeit-Buffer
0381 2FE9 *
0382 2FEA ASCBUF:DS 15H
0383 2FEA *
0384 2FFF END

```

```

0001 0000 *****
0002 0000 *
0003 0000 * LEDASC Monitor f.ASCII-Tastatur u.LED-Display *
0004 0000 *
0005 0000 *****
0006 0000 *
0007 0000 * HEX-Monitor mit ASCII-Eingabe und Siebensegment-
0008 0000 * Anzeige; die Eingabe kann wahlweise über die
0009 0000 * an die ASCII-Tastatur angeflanschte HEX-Tasta-
0010 0000 * tur erfolgen oder über folgende ASCII-Tasten:
0011 0000 *
0012 0000 * DAT: Pfeil rechts (neben TAB)
0013 0000 * ADR: Pfeil links (neben TAB)
0014 0000 * REG: "r" (oder "R")
0015 0000 * NXT: "Return"
0016 0000 * BST: Pfeil hoch (neben CTRL)
0017 0000 * RUN: "g" (oder "G")
0018 0000 * FCT: Funktionstaste
0019 0000 *
0020 0000 *
0021 0000 HXTEST EQU 1015H
0022 0000 *
0023 0000 *
0024 0000 ORG 0A00H
0025 0A00 OFS 0A00H
0026 0A00 *
0027 0A00 *
0028 0A00 CD 52 00 LEDASC:CALL CSTS *auf Loslassen warten
0029 0A03 C2 00 0A JNZ LEDASC
0030 0A06 CD 09 01 CALL INIT *Grundeinstellungen
0031 0A09 CD B4 03 CALL UPDATE
0032 0A0C CD 18 0A LEDMO: CALL HEXKY2 *HEX-Tasten abfragen
0033 0A0F CD C6 02 CALL SSGDSP *SSG-Anzeige aktivieren
0034 0A12 CD 60 0A CALL CMDKY2 *Befehlstasten abfragen
0035 0A15 C3 0C 0A JMP LEDMO *Schleife
0036 0A18 *
0037 0A18 CD 52 00 HEXKY2:CALL CSTS *Taste aktiv?
0038 0A1B C8 RZ *nein: RET
0039 0A1C CD 43 00 CALL CI
0040 0A1F 47 MOV B,A
0041 0A20 06 0A MVI B,0AH
0042 0A22 FE 82 CPI 82H *Cursor rechts?
0043 0A24 CA 4E 0A JZ HEXCTE
0044 0A27 06 0B MVI B,0BH
0045 0A29 FE 8B CPI 8BH *Cursor down?
0046 0A2B CA 4E 0A JZ HEXCTE
0047 0A2E 06 0C MVI B,0CH
0048 0A30 FE 80 CPI 80H *Cursor left?
0049 0A32 CA 4E 0A JZ HEXCTE
0050 0A35 06 0D MVI B,0DH
0051 0A37 FE 2F CPI 2FH *Slash?
0052 0A39 CA 4E 0A JZ HEXCTE
0053 0A3C 06 0E MVI B,0EH
0054 0A3E FE 89 CPI 89H *Cursor up?
0055 0A40 CA 4E 0A JZ HEXCTE
0056 0A43 06 0F MVI B,0FH
0057 0A45 FE 8F CPI 8FH *Cursor home?
0058 0A47 CA 4E 0A JZ HEXCTE
0059 0A4A CD 15 10 CALL HXTEST
0060 0A4D C8 RZ
0061 0A4E 48 HEXCTE:MOV C,B
0062 0A4F 79 MOV A,C
0063 0A50 CD B6 01 CALL SHIFT *Eingabe nachschieben

```

0064	0A53	CD 3D 02	CALL CONV	*HEX/SSG-Umsetzung
0065	0A56	CD C6 02	HXL0P: CALL SSGDSP	
0066	0A59	CD 52 00	CALL CSTS	*auf Loslassen warten
0067	0A5C	C2 56 0A	JNZ HXL0P	
0068	0A5F	C9	RET	
0069	0A60		*	
0070	0A60	21 ED 0A	CMDKY2:LXI H,CML0P	*CTE-Sprungziel
0071	0A63	E5	PUSH H	
0072	0A64	DB 0C	IN OCH	*FCT-Spalte abfragen
0073	0A66	21 CC 2F	LXI H,CIFLAG	
0074	0A69	AE	XRA M	
0075	0A6A	FE 20	CPI 20H	
0076	0A6C	CA 91 03	JZ FCT2	
0077	0A6F		*	
0078	0A6F	CD 52 00	CALL CSTS	*Taste aktiv?
0079	0A72	E1	POP H	
0080	0A73	C8	RZ	*nein: RET
0081	0A74	E5	PUSH H	
0082	0A75	CD 43 00	CALL CI	*ja: dann ASCII-Code holen
0083	0A78		*	
0084	0A78	47	MOV B,A	
0085	0A79	3A BD 2F	LDA MODFLG	*FCT-Mode aktiv?
0086	0A7C	E6 08	ANI 8H	
0087	0A7E	78	MOV A,B	
0088	0A7F	CA A5 0A	JZ CMDCTE	*nein: JMP
0089	0A82		*	
0090	0A82	FE 0D	CPI 0DH	*Ret?
0091	0A84	CA 0E 03	JZ BRNCH2	
0092	0A87	FE 20	CPI 20H	*Blank?
0093	0A89	CA 0E 03	JZ BRNCH2	
0094	0A8C		*	
0095	0A8C	FE 3C	CPI 3CH	*Pfeil links?
0096	0A8E	CA 0E 03	JZ BRNCH2	
0097	0A91	FE 2D	CPI 2DH	*Minus?
0098	0A93	CA 0E 03	JZ BRNCH2	
0099	0A96		*	
0100	0A96	FE 72	CPI "r"	*
0101	0A98	CA 0E 03	JZ BRNCH2	
0102	0A9B	FE 52	CPI "R"	*
0103	0A9D	CA 0E 03	JZ BRNCH2	
0104	0AA0	FE 2B	CPI "+"	*
0105	0AA2	CA 0E 03	JZ BRNCH2	
0106	0AA5		*	
0107	0AA5	FE 3E	CMDCTE:CPI 3EH	*Pfeil rechts?
0108	0AA7	CA 63 03	JZ DAT2	
0109	0AAA	FE 2A	CPI 2AH	*Asterisk?
0110	0AAC	CA 63 03	JZ DAT2	
0111	0AAF		*	
0112	0AAF	FE 3C	CPI 3CH	*Pfeil links?
0113	0AB1	CA 73 03	JZ ADR2	
0114	0AB4	FE 2D	CPI 2DH	*Minus?
0115	0AB6	CA 73 03	JZ ADR2	
0116	0AB9		*	
0117	0AB9	FE 72	CPI "r"	*
0118	0ABB	CA 82 03	JZ REG2	
0119	0ABE	FE 52	CPI "R"	*
0120	0AC0	CA 82 03	JZ REG2	
0121	0AC3	FE 2B	CPI "+"	
0122	0AC5	CA 82 03	JZ REG2	
0123	0AC8		*	
0124	0AC8	FE 0D	CPI 0DH	*Ret?
0125	0ACA	CA BB 03	JZ NXT2	
0126	0ACD	FE 20	CPI 20H	*Blank?
0127	0ACF	CA BB 03	JZ NXT2	

0128	OAD2		*			
0129	OAD2	FE 5E		CPI	5EH	*Pfeil hoch?
0130	OAD4	CA 01 04		JZ	BST2	
0131	OAD7	FE 2E		CPI	2EH	*Dez. Punkt?
0132	OAD9	CA 01 04		JZ	BST2	
0133	OADC		*			
0134	OADC	FE 67		CPI	"g"	*
0135	OADE	CA 1A 04		JZ	RUN2	
0136	OAE1	FE 47		CPI	"G"	*
0137	OAE3	CA 1A 04		JZ	RUN2	
0138	OAE6	FE 3F		CPI	"?"	*
0139	OAE8	CA 1A 04		JZ	RUN2	
0140	OAEB		*			
0141	OAEB	E1		POP	H	
0142	OAEC	C9		RET		
0143	OAED	CD C6 02	CMLOP:	CALL	SSGDSP	
0144	OAF0	CD 52 00		CALL	CSTS	
0145	OAF3	C2 ED 0A		JNZ	CMLOP	
0146	OAF6	C9		RET		
0147	OAF7		*			
0148	OAF7	DAT2		EQU	DAT	
0149	OAF7	ADR2		EQU	ADR	
0150	OAF7	REG2		EQU	REG	
0151	OAF7	NXT2		EQU	NXT	
0152	OAF7	BST2		EQU	BST	
0153	OAF7	RUN2		EQU	RUN	
0154	OAF7	FCT2		EQU	FCT	
0155	OAF7	BRNCH2		EQU	BRNCH	
0156	OAF7		*			
0157	OAF7		END			

```

0001 0000 *****
0002 0000 *
0003 0000 * Ansteuerung von Peripheriegeräten      24.10.85 *
0004 0000 *
0005 0000 *****
0006 0000 *
0007 0000 *      Unterprogramme für Peripheriegeräte:
0008 0000 *
0009 0000 *      - CSINIT initialisiert CAS-IF
0010 0000 *      - LOINIT initialisiert PRT-IF
0011 0000 *      - V24INI initialisiert V.24-Schnittstelle
0012 0000 *      - PO Cassetten-Ausgabe
0013 0000 *      - RI Cassetten-Eingabe
0014 0000 *      - LO Drucker-Ausgabe
0015 0000 *      - V24OT Ausgabe über V.24-Schnittstelle
0016 0000 *      - V24IN Eingabe über V.24-Schnittstelle
0017 0000 *      - CPUOT SOD-Ausgabe (CPU)
0018 0000 *      - CPUIN SID-Eingabe (CPU)
0019 0000 *
0020 0000 *
0021 0000 ORG      0B00H
0022 0B00 OFS      0B00H
0023 0B00 *
0024 0B00 ADROT   EQU   103CH      *(H&L) anzeigen
0025 0B00 UPDAT   EQU   2E56H      *Cursor-Update-Adresse
0026 0B00 *
0027 0B00 *****
0028 0B00 *
0029 0B00 *      Schnittstellen initialisieren
0030 0B00 *
0031 0B00 *****
0032 0B00 *
0033 0B00 D5      CSINI-:PUSH D
0034 0B01 E5              PUSH H
0035 0B02 3A 5F 00      LDA  CASMOD      *Mode-Wort für SER-IF
0036 0B05 47              MOV   B,A
0037 0B06 3E C2         MVI   A,0C2H      *Mode-Register-Byte
0038 0B08 2A 16 00      LHLD CASBDR      *CAS-Baudrate
0039 0B0B C3 46 0B      JMP  INITCT
0040 0B0E D5      LOINI-:PUSH D
0041 0B0F E5              PUSH H
0042 0B10 3E 8A         MVI   A,8AH      *8255 initialisieren
0043 0B12 D3 D8         OUT  PARCTL      *PA, PClo = OUT
0044 0B14 3E 01         MVI   A,1H       *PB, PChi = IN
0045 0B16 D3 C8         OUT  PARPOC      *STROBE auf HIGH
0046 0B18 21 03 2E      LXI  H,XTXBUF    *Vorbereitung Thermodrucker
0047 0B1B 22 00 2E      SHLD TXTPOI
0048 0B1E AF           XRA  A
0049 0B1F 32 02 2E      STA  TXTNUM      *Zeichenzähler löschen
0050 0B22 3E 01         MVI   A,1H
0051 0B24 D3 BA         OUT  CENSTB      *INPRM für Centronics
0052 0B26 3A 5E 00      LDA  PRTMOD      *Mode-Wort für SER-IF
0053 0B29 47              MOV   B,A
0054 0B2A 3E A0         MVI   A,0A0H     *Mode-Register-Byte
0055 0B2C 2A 0E 00      LHLD PRTBDR      *PRT-Baudrate
0056 0B2F C3 46 0B      JMP  INITCT
0057 0B32 C3 3B 0B V24IN-:JMP V24CTE
0058 0B35 C3 19 0D V24OT: JMP L07
0059 0B38 C3 CF 0C V24IN: JMP RI-
0060 0B3B *
0061 0B3B D5      V24CTE:PUSH D
0062 0B3C E5              PUSH H
0063 0B3D 3A 60 00      LDA  V24MOD      *Mode-Wort für V.24

```

```

0064 0B40 47          MOV  B,A
0065 0B41 3E 58      MVI  A,58H      *Mode-Register-Byte
0066 0B43 2A 1E 00    LHLD V24BDR     *V.24-Baudrate
0067 0B46 D3 A9      INITCT:OUT SERMOD *Mode-Wort ausgeben
0068 0B48 3E 03      MVI  A,3H      *RESET für 6850
0069 0B4A D3 E9      OUT  SERCTL
0070 0B4C 78          MOV  A,B        *Mode-Wort 6850
0071 0B4D D3 E9      OUT  SERCTL
0072 0B4F 3E 36      MVI  A,36H     *Timer-CTL: Mode 3 binär
0073 0B51 D3 D9      OUT  TIMCTL
0074 0B53 CD 5F 0B    CALL BAUDRT     *Teilerfaktor ermitteln
0075 0B56 7D          MOV  A,L
0076 0B57 D3 89      OUT  TIMDAT     *Teilerfaktor (lower)
0077 0B59 7C          MOV  A,H
0078 0B5A D3 89      OUT  TIMDAT     *Teilerfaktor (upper)
0079 0B5C E1          POP  H
0080 0B5D D1          POP  D
0081 0B5E C9          RET
0082 0B5F          *
0083 0B5F          *      BAUDRT      ermittelt aus der Baudrate und
0084 0B5F          *                      der CPU-Quarzfrequenz FQUARZ den
0085 0B5F          *                      Timer-Teilerfaktor; zulässige
0086 0B5F          *                      Werte: 9600,4800,2400,1200,600,
0087 0B5F          *                      300 und 110 Bd; Quarz: ca.4 MHz (1)
0088 0B5F          *                      ca. 6 MHz (2) oder ca. 8 MHz (3)
0089 0B5F          *
0090 0B5F 7C          BAUDRT:MOV  A,H      *Baudrate (upper) holen
0091 0B60 07          RLC
0092 0B61 3E 00      MVI  A,0H
0093 0B63 8C          ADC  H
0094 0B64 E6 07      ANI  7H        *Versatz
0095 0B66 07          RLC
0096 0B67 5F          MOV  E,A
0097 0B68 16 00      MVI  D,0H
0098 0B6A 21 A6 0B    LXI  H,TAB3     *hohe Frequenz
0099 0B6D 3A 06 00    LDA  FQUARZ     *Quarzfrequenz
0100 0B70 FE 07      CPI  7H
0101 0B72 D2 80 0B    JNC  BDEND
0102 0B75 21 96 0B    LXI  H,TAB2     *mittlere Frequenz
0103 0B78 FE 05      CPI  5H
0104 0B7A D2 80 0B    JNC  BDEND
0105 0B7D 21 86 0B    LXI  H,TAB1     *niedrige Frequenz
0106 0B80 19          BDEND: DAD  D
0107 0B81 5E          MOV  E,M
0108 0B82 23          INX  H
0109 0B83 56          MOV  D,M
0110 0B84 EB          XCHG          *H&L: Teilerfaktor
0111 0B85 C9          RET
0112 0B86          *
0113 0B86          TAB1: DS  0H      * 4.194 MHz
0114 0B86 B5 01      DW  01B5H      * 4800 Bd
0115 0B88 78 4A      DW  4A78H      * 110 Bd
0116 0B8A D4 06      DW  06D4H      * 1200 Bd
0117 0B8C 4E 1B      DW  1B4EH      * 300 Bd
0118 0B8E 6A 03      DW  036AH      * 2400 Bd
0119 0B90 B5 01      DW  01B5H      * ./
0120 0B92 A7 0D      DW  0DA7H      * 600 Bd
0121 0B94 DA 00      DW  00DAH      * 9600 Bd
0122 0B96          *
0123 0B96          TAB2: DS  0H      * 6.144 MHz
0124 0B96 80 02      DW  0280H      * 4800 Bd
0125 0B98 17 6D      DW  6D17H      * 110 Bd
0126 0B9A C4 09      DW  09C4H      * 1200 Bd
0127 0B9C 00 28      DW  2800H      * 300 Bd

```

```

0128 OB9E 00 05 DW 0500H * 2400 Bd
0129 OBA0 80 02 DW 0280H * ./
0130 OBA2 00 14 DW 1400H * 600 Bd
0131 OBA4 40 01 DW 0140H * 9600 Bd
0132 OBA6 *
0133 OBA6 TAB3: DS 0H * 8.000 MHz
0134 OBA6 41 03 DW 0341H * 4800 Bd
0135 OBA8 0C 8E DW 8E0CH * 110 Bd
0136 OBAA 05 0D DW 0D05H * 1200 Bd
0137 OBAC 15 34 DW 3415H * 300 Bd
0138 OBAE 83 06 DW 0683H * 2400 Bd
0139 OBB0 41 03 DW 0341H * ./
0140 OBB2 0B 1A DW 1A0BH * 600 Bd
0141 OBB4 A1 01 DW 01A1H * 9600 Bd
0142 OBB6 *
0143 OBB6 *****
0144 OBB6 *
0145 OBB6 * Cassetten-Ausgabe
0146 OBB6 *
0147 OBB6 *****
0148 OBB6 *
0149 OBB6 * Das Programm erzeugt ca.5 s langen Vorspann,
0150 OBB6 * um das spätere Einlesen definiert starten
0151 OBB6 * zu können
0152 OBB6 *
0153 OBB6 2A BB 2F FCTCOT:LHLD ADRES *letzten Parameter holen
0154 OBB9 22 C1 2F SHLD SRCEND *und ablegen
0155 OBBC *
0156 OBBC 3A BF 2F CASOT-:LDA SRCBEG *Abfrage wg. Abgleich
0157 OBBF FE AD CPI OADH *ad=Adjust
0158 OBC1 CA 90 0C JZ ADJUST *ja: JMP
0159 OBC4 *
0160 OBC4 CD 00 0B CALL CSINI- *CAS-Ausgabe
0161 OBC7 01 58 02 LXI B,258H *600d * 8,3 ms
0162 OBCA AF OUTLOP:XRA A *REG A löschen
0163 OBCB CD C6 0C CALL CASOT1 *Byte ausgeben
0164 OBCE 0B DCX B
0165 OBCF B9 CMP C
0166 OBD0 C2 CA 0B JNZ OUTLOP
0167 OBD3 B8 CMP B
0168 OBD4 C2 CA 0B JNZ OUTLOP
0169 OBD7 *
0170 OBD7 3E D3 MVI A,0D3H *Startzeichen
0171 OBD9 CD C3 0C CALL CASOT2 *3mal ausgeben
0172 OBDC CD C6 0C CALL CASOT1
0173 OBDF *
0174 OBDF AF XRA A
0175 OBE0 CD C6 0C CALL CASOT1 *"00" hinterher
0176 OBE3 *
0177 OBE3 2A C1 2F LHLD SRCEND *Parameter holen
0178 OBE6 23 INX H
0179 OBE7 EB XCHG
0180 OBE8 2A BF 2F LHLD SRCBEG
0181 OBEB 4E OUTPUT:MOV C,M *Zeichen holen
0182 OBEC CD E8 0C CALL PO- *und ausgeben
0183 OBEF 23 INX H
0184 OBFO 7D MOV A,L *Endabfrage
0185 OBF1 BB CMP E
0186 OBF2 C2 EB 0B JNZ OUTPUT *nicht die Endadresse: JMP
0187 OBF5 7C MOV A,H
0188 OBF6 BA CMP D
0189 OBF7 C2 EB 0B JNZ OUTPUT *nicht die Endadresse: JMP
0190 OBFA *
0191 OBFA AF XRA A

```



```

0192 0BFB CD C3 0C CALL CASOT2 *4mal "00" als
0193 0BFE CD C3 0C CALL CASOT2 *Endzeichen
0194 0C01 CD 0E 0B CALL LOINI- *wieder Drucker einstellen
0195 0C04 C9 RET
0196 0C05 *
0197 0C05 *****
0198 0C05 *
0199 0C05 * Cassetten-Eingabe
0200 0C05 *
0201 0C05 *****
0202 0C05 *
0203 0C05 * Programm im 5-s-Vorspann starten
0204 0C05 *
0205 0C05 2A BB 2F FCTCIN:LHLD ADRES *letzten Parameter holen
0206 0C08 22 C1 2F SHLD SRCEND *und ablegen
0207 0C0B CD 12 0C CALL CASIN-
0208 0C0E C0 RNZ *o.k.; RET
0209 0C0F C3 9D 00 JMP DSPERR *Einlesefehler
0210 0C12 *
0211 0C12 CD 00 0B CASIN-:CALL CSINI-
0212 0C15 *
0213 0C15 06 64 MVI B,64H *100d Bytes erfassen
0214 0C17 0E 0A INLOP0:MVI C,0AH *10d Vorlauf-Bytes
0215 0C19 CD CF 0C INLPO: CALL RI-
0216 0C1C FE 00 CPI OH
0217 0C1E CA 28 0C JZ INLO *"00" erkannt
0218 0C21 05 DCR B
0219 0C22 C2 17 0C JNZ INLOP0
0220 0C25 C3 2F 0C JMP INVERT
0221 0C28 0D INLO: DCR C
0222 0C29 C2 19 0C JNZ INLPO
0223 0C2C C3 4C 0C JMP INLOP2 *auf Startmarke "D3" warten
0224 0C2F *
0225 0C2F 3E C3 INVERT:MVI A,0C3H *Signal invertieren (mit LSB)
0226 0C31 D3 A9 OUT SERMOD
0227 0C33 *
0228 0C33 06 64 MVI B,64H *100d Bytes erfassen
0229 0C35 0E 0A INLOP1:MVI C,0AH *10d Vorlauf-Bytes
0230 0C37 CD CF 0C INLP1: CALL RI-
0231 0C3A FE 00 CPI OH
0232 0C3C CA 48 0C JZ INL1 *"00" erkannt
0233 0C3F 05 DCR B
0234 0C40 C2 35 0C JNZ INLOP1
0235 0C43 3E 00 MVI A,0H *Vorspann-Fehler
0236 0C45 C3 8A 0C JMP INEND *Fehler: Z=1
0237 0C48 0D INL1: DCR C
0238 0C49 C2 37 0C JNZ INLP1
0239 0C4C *
0240 0C4C 0E 03 INLOP2:MVI C,3H *3 Start-Bytes
0241 0C4E CD CF 0C INLOP3:CALL RI-
0242 0C51 FE D3 CPI 0D3H
0243 0C53 C2 4C 0C JNZ INLOP2 ? *"D3" nicht erkannt
0244 0C56 0D DCR C
0245 0C57 C2 4E 0C JNZ INLOP3
0246 0C5A C3 62 0C JMP INLOP4 *3*"D3" erkannt
0247 0C5D 3E 01 MVI A,1H *Startzeichen-Fehler
0248 0C5F C3 8A 0C JMP INEND *Fehler: Z=1
0249 0C62 CD CF 0C INLOP4:CALL RI-
0250 0C65 FE 00 CPI OH
0251 0C67 CA 70 0C JZ INPUT *"00" erkannt
0252 0C6A AF XRA A
0253 0C6B 3E 02 MVI A,2H *Sync-Fehler
0254 0C6D C3 8A 0C JMP INEND *Fehler: Z=1
0255 0C70 *

```

```

0256 0C70 2A C1 2F INPUT: LHL D SRCEND *Parameter holen
0257 0C73 23 INX H
0258 0C74 EB XCHG
0259 0C75 2A BF 2F LHL D SRCBEG
0260 0C78 CD CF 0C INP LOP:CALL RI- *Zeichen einlesen
0261 0C7B C8 RZ *RET bei Datenverlust
0262 0C7C 77 MOV M,A *sonst im RAM ablegen
0263 0C7D 23 INX H
0264 0C7E 7D MOV A,L *Endabfrage
0265 0C7F BB CMP E
0266 0C80 C2 78 0C JNZ INP LOP *nicht die Endadresse: JMP
0267 0C83 7C MOV A,H
0268 0C84 BA CMP D
0269 0C85 C2 78 0C JNZ INP LOP *nicht die Endadresse: JMP
0270 0C88 AF XRA A
0271 0C89 3C INR A *o.k.: Z=0
0272 0C8A F5 INEND: PUSH PSW
0273 0C8B CD OE 0B CALL LOINI- *wieder Drucker einstellen
0274 0C8E F1 POP PSW
0275 0C8F C9 RET
0276 0C90 *
0277 0C90 *
0278 0C90 *****
0279 0C90 *
0280 0C90 * Cassetten-Interface abgleichen
0281 0C90 *
0282 0C90 *****
0283 0C90 *
0284 0C90 * Testpunkt TPO an Masse legen, Testpunkt TP1 mit
0285 0C90 * Timer-Eingang GO verbinden und Abgleich-Routine
0286 0C90 * aufrufen: Eingabe "o ad", gefolgt von Return
0287 0C90 * ("o" für "Cassetten-Ausgabe", "ad" für "Adjust").
0288 0C90 *
0289 0C90 * Sollwert bei 4.000 MHz: 3750d
0290 0C90 * 4.194 MHz: 3690d
0291 0C90 * 6.000 MHz: 3125d
0292 0C90 * 6.144 MHz: 3080d
0293 0C90 * 8.000 MHz: 2500d
0294 0C90 *
0295 0C90 3E 31 ADJUST:MVI A,31H *Mode 0, BCD
0296 0C92 D3 D9 OUT TIMCTL
0297 0C94 3E 00 MVI A,0H
0298 0C96 D3 89 OUT TIMDAT
0299 0C98 3E 50 MVI A,50H *Anfangswert 5000d
0300 0C9A D3 89 OUT TIMDAT
0301 0C9C *
0302 0C9C 3E 01 MVI A,1H
0303 0C9E D3 A9 OUT SERMOD *XOR triggern
0304 0CA0 AF XRA A
0305 0CA1 D3 A9 OUT SERMOD
0306 0CA3 CD 0B 00 CALL DELY1 *1 ms abwarten
0307 0CA6 *
0308 0CA6 3E 01 MVI A,1H *Timer lesen
0309 0CA8 D3 D9 OUT TIMCTL
0310 0CAA DB 89 IN TIMDAT *lower Byte
0311 0CAC 6F MOV L,A
0312 0CAD DB 89 IN TIMDAT *upper Byte
0313 0CAF 67 MOV H,A
0314 0CB0 EB XCHG
0315 0CB1 2A 56 2E LHL D UPDAT *Cursor-Update
0316 0CB4 E5 PUSH H
0317 0CB5 EB XCHG
0318 0CB6 CD 3C 10 CALL ADROT *Zählerstand anzeigen
0319 0CB9 E1 POP H

```

```

0320 0CBA 22 56 2E      SHLD UPDAT
0321 0CBD CD 1B 00      CALL ONESEC      *beruhigen
0322 0CC0 C3 90 0C      JMP ADJUST
0323 0CC3                *
0324 0CC3                *
0325 0CC3                *      CASOT1/2      gibt 1 bzw. 2 Bytes über das
0326 0CC3                *      Magnetband-Interface aus (auf
0327 0CC3                *      der Seriellen Interface-Karte)
0328 0CC3                *
0329 0CC3 CD C6 0C      CASOT2:CALL CASOT1
0330 0CC6 C5                CASOT1:PUSH B
0331 0CC7 4F                MOV C,A
0332 0CC8 F5                PUSH PSW
0333 0CC9 CD E8 0C      CALL PO-          *Zeichen ausgeben
0334 0CCC F1                POP PSW
0335 0CCD C1                POP B
0336 0CCE C9                RET
0337 0CCF                *
0338 0CCF                *      RI          liest ein Byte über das Magnet-
0339 0CCF                *      band-Interface ein (ins REG A)
0340 0CCF                *
0341 0CCF C5                RI-:  PUSH B
0342 0CD0 0E 0C                MVI C,0CH      *max. Wartezeit: 12 ms
0343 0CD2 DB E9      RILOP: IN SERCTL  *Rec.Data full?
0344 0CD4 E6 01                ANI 1H         *Statusbit maskieren
0345 0CD6 C2 E4 0C      JNZ RICTE
0346 0CD9 CD 0B 00      CALL DELY1
0347 0CDC 0D                DCR C
0348 0CDD C2 D2 0C      JNZ RILOP
0349 0CE0 3E 03                MVI A,3H      *Fehler: Datenverlust
0350 0CE2 C1                POP B
0351 0CE3 C9                RET            *Fehler: Z=1
0352 0CE4 DB F9      RICTE: IN SERDAT  *Daten auslesen
0353 0CE6 C1                POP B
0354 0CE7 C9                RET
0355 0CE8                *
0356 0CE8                *      PO          gibt ein Byte über das Magnet-
0357 0CE8                *      band-Interface aus (von REG C)
0358 0CE8                *
0359 0CE8 41                PO-:  MOV B,C
0360 0CE9 0E 0C                MVI C,0CH      *max. Wartezeit 12 ms
0361 0CEB DB E9      POLOP: IN SERCTL  *Transm.Data empty?
0362 0CED E6 02                ANI 2H         *Statusbit maskieren
0363 0CEF C2 FC 0C      JNZ POCTE
0364 0CF2 CD 0B 00      CALL DELY1
0365 0CF5 0D                DCR C
0366 0CF6 C2 EB 0C      JNZ POLOP
0367 0CF9 3E 04                MVI A,4H      *Gegenstelle nicht bereit
0368 0CFB C9                RET            *Fehler: Z=1
0369 0CFC 78                POCTE: MOV A,B
0370 0CFD D3 F9      OUT SERDAT      *Daten übergeben
0371 0CFF C9                RET
0372 0D00                *
0373 0D00                *      LO          gibt ein Byte über die MOPPEL-
0374 0D00                *      Schnittstellen aus; die Auswahl
0375 0D00                *      der vier möglichen LO-Geräte er-
0376 0D00                *      folgt über die I/O-FLAGS
0377 0D00                *
0378 0D00                *      LO7: V.24-Schnittstelle (IOFLAG #7)
0379 0D00                *      LO6: Centronics-Schnittstelle (IOFLAG #6)
0380 0D00                *      LO5: 8255-Parallel-Ausgabe (IOFLAG #5)
0381 0D00                *      LO4: Thermodrucker (IOFLAG #4)
0382 0D00                *
0383 0D00 3A C8 2F      LO-:  LDA IOFLG

```

```

0384 0D03 E6 F0 ANI 0F0H *4 MSB maskieren
0385 0D05 07 RLC
0386 0D06 DA 19 0D JC L07
0387 0D09 07 RLC
0388 0D0A DA 46 0D JC L06
0389 0D0D 07 RLC
0390 0D0E DA 69 0D JC L05
0391 0D11 07 RLC
0392 0D12 DA 93 0D JC L04
0393 0D15 AF XRA A *FLAGS setzen
0394 0D16 3E 05 MVI A,5H *I/O-Byte undefiniert
0395 0D18 C9 RET *Fehler: Z=1
0396 0D19 *
0397 0D19 * Serielle Ausgabe über die Drucker-
0398 0D19 * Schnittstelle
0399 0D19 *
0400 0D19 TIMEOUT EQU 2000H *ca. 8000 ms = 8 s
0401 0D19 *
0402 0D19 C5 L07: PUSH B
0403 0D1A 01 00 20 LXI B,TIMOUT
0404 0D1D DB E9 L0LOP: IN SERCTL *Transm.Data empty?
0405 0D1F E6 02 ANI 2H *Statusbit maskieren
0406 0D21 C2 35 0D JNZ LOCTS
0407 0D24 CD 0B 00 CALL DELY1
0408 0D27 0B DCX B
0409 0D28 AF XRA A
0410 0D29 B8 CMP B
0411 0D2A C2 1D 0D JNZ L0LOP
0412 0D2D B9 CMP C
0413 0D2E C2 1D 0D JNZ L0LOP
0414 0D31 C1 POP B
0415 0D32 3E 07 MVI A,7H *Gegenstelle nicht bereit
0416 0D34 C9 RET *Fehler: Z=1
0417 0D35 DB E9 LOCTS: IN SERCTL *Clear to Send?
0418 0D37 E6 08 ANI 8H *Statusbit maskieren
0419 0D39 C2 35 0D JNZ LOCTS
0420 0D3C 01 00 00 L07CTE: LXI B,0H *FLAGS löschen
0421 0D3F C5 PUSH B
0422 0D40 F1 POP PSW
0423 0D41 C1 POP B
0424 0D42 79 MOV A,C
0425 0D43 D3 F9 OUT SERDAT *Zeichen übergeben
0426 0D45 C9 RET
0427 0D46 *
0428 0D46 * Parallel-Ausgabe über Centronics-
0429 0D46 * Schnittstelle
0430 0D46 *
0431 0D46 C5 L06: PUSH B
0432 0D47 01 00 20 LXI B,TIMOUT
0433 0D4A DB CA L06LOP: IN CENDAT *BUSY-Abfrage
0434 0D4C E6 40 ANI 40H
0435 0D4E C2 62 0D JNZ L06CTE
0436 0D51 CD 0B 00 CALL DELY1
0437 0D54 0B DCX B
0438 0D55 AF XRA A *Endabfrage
0439 0D56 B8 CMP B
0440 0D57 C2 4A 0D JNZ L06LOP
0441 0D5A B9 CMP C
0442 0D5B C2 4A 0D JNZ L06LOP
0443 0D5E C1 POP B
0444 0D5F 3E 06 MVI A,6H *Gegenstelle nicht bereit
0445 0D61 C9 RET *Fehler: Z=1
0446 0D62 C1 L06CTE: POP B
0447 0D63 79 MOV A,C

```

```

0448 OD64 D3 CA OUT CENDAT *Daten ausgeben
0449 OD66 D3 BA OUT CENSTB *STROBE-Puls
0450 OD68 C9 RET
0451 OD69 *
0452 OD69 * Parallel-Ausgabe über 8255
0453 OD69 *
0454 OD69 * Daten: Port A
0455 OD69 * BUSY: Port C, Bit 7 (In)
0456 OD69 * STROBE:Port C, Bit 0 (Out)
0457 OD69 *
0458 OD69 C5 L05: PUSH B
0459 OD6A 01 00 20 LXI B,TIMOUT
0460 OD6D DB C8 L05LOP:IN PARPOC *BUSY-Abfrage
0461 OD6F E6 80 ANI 80H
0462 OD71 CA 85 0D JZ L05CTE
0463 OD74 CD 0B 00 CALL DELY1
0464 OD77 0B DCX B
0465 OD78 AF XRA A *Endabfrage
0466 OD79 B8 CMP B
0467 OD7A C2 6D 0D JNZ L05LOP
0468 OD7D B9 CMP C
0469 OD7E C2 6D 0D JNZ L05LOP
0470 OD81 C1 POP B
0471 OD82 3E 08 MVI A,8H *Gegenstelle nicht bereit
0472 OD84 C9 RET *Fehler: Z=1
0473 OD85 C1 L05CTE:POP B
0474 OD86 79 MOV A,C
0475 OD87 D3 A8 OUT PARPOA *Daten ausgeben
0476 OD89 AF XRA A
0477 OD8A D3 C8 OUT PARPOC *STROBE-Puls
0478 OD8C CD 92 0D CALL L05END *Wartezeit
0479 OD8F 3C INR A
0480 OD90 D3 C8 OUT PARPOC
0481 OD92 C9 L05END:RET
0482 OD93 *
0483 OD93 * Ausgabe über den Thermodrucker als
0484 OD93 * LO-Ausgabegerät
0485 OD93 *
0486 OD93 PRTLIN EQU 2014H *ASCII-Zeile drucken
0487 OD93 *
0488 OD93 TXTPOI EQU 2E00H *Buffer-Pointer
0489 OD93 TXTNUM EQU 2E02H *Zeichenzähler
0490 OD93 TXTBUF EQU 2E03H *Text-Buffer
0491 OD93 *
0492 OD93 79 L04: MOV A,C
0493 OD94 FE 08 CPI 08H *BS?
0494 OD96 CA E3 0D JZ DEL
0495 OD99 FE 0C CPI 0CH *FF?
0496 OD9B CA D5 0D JZ FF
0497 OD9E FE 0D CPI 0DH *CR?
0498 ODA0 CA C8 0D JZ CR
0499 ODA3 FE 7F CPI 7FH *DEL?
0500 ODA5 CA E3 0D JZ DEL
0501 ODA8 *
0502 ODA8 FE 20 CPI 20H *Code im zulässigen Bereich?
0503 ODAA DA B2 0D JC L04RET *nein: JMP und übergehen
0504 ODAD FE A0 CPI 0A0H
0505 ODAF DA B6 0D JC L04CTE
0506 ODB2 00 L04RET:NOP *. /. XRA A wg. Fehlermeldung
0507 ODB3 3E 09 MVI A,9H *unzulässiger Code
0508 ODB5 C9 RET *evtl. Z=1 übergeben
0509 ODB6 *
0510 ODB6 2A 00 2E L04CTE:LHLD TXTPOI
0511 ODB9 71 MOV M,C *Zeichen in Text-Buffer

```

```

0512 ODBA 23          INX H          *Buffer-Pointer erhöhen
0513 ODBB 22 00 2E   SHLD TXTPOI
0514 ODBE           *
0515 ODBE 3A 02 2E   LDA TXTNUM
0516 ODC1 3C          INR A          *Zeichenzähler erhöhen
0517 ODC2 32 02 2E   STA TXTNUM
0518 ODC5 FE 14      CPI 14H        *Buffer voll?
0519 ODC7 C0          RNZ          *nein: RET
0520 ODC8           *
0521 ODC8 AF          CR: XRA A          *Zeichenzähler löschen
0522 ODC9 32 02 2E   STA TXTNUM
0523 ODCC 21 03 2E   LXI H,TXTBUF *Pointer auf Textanfang
0524 ODCF 22 00 2E   SHLD TXTPOI
0525 ODD2 CD 14 20   CALL PRTLIN  *Zeile ausdrucken
0526 ODD5           *
0527 ODD5 3E 14      FF: MVI A,14H  *Buffer löschen
0528 ODD7 21 03 2E   LXI H,TXTBUF
0529 ODDA 36 20      FFLOP: MVI M,20H
0530 ODDC 23          INX H
0531 ODDD 3D          DCR A
0532 ODDE C2 DA 0D   JNZ FFLOP
0533 ODE1 3C          INR A          *FLAGS löschen
0534 ODE2 C9          RET
0535 ODE3           *
0536 ODE3 3A 02 2E   DEL: LDA TXTNUM *Zeichen löschen
0537 ODE6 3D          DCR A
0538 ODE7 F8          RM          *Buffer war leer: RET
0539 ODE8 32 02 2E   STA TXTNUM
0540 ODEB 2A 00 2E   LHLD TXTPOI
0541 ODEE 2B          DCX H
0542 ODEF 36 20      MVI M,20H    *Blank überschreiben
0543 ODF1 22 00 2E   SHLD TXTPOI
0544 ODF4 C9          RET
0545 ODF5           *
0546 ODF5           *
0547 ODF5           *
0548 ODF5           *      Serielle Schnittstellen der CPU ansteuern
0549 ODF5           *
0550 ODF5           *
0551 ODF5           *
0552 ODF5           *      Datenübertragung zwischen SOD des einen und
0553 ODF5           *      SID des anderen Computers; dazu folgende
0554 ODF5           *      Verbindungen herstellen:
0555 ODF5           *      SID, Gerät A an SOD, Gerät B
0556 ODF5           *      SOD, Gerät A an SID, Gerät B
0557 ODF5           *      GND, Gerät A an GND, Gerät B (Masse)
0558 ODF5           *
0559 ODF5 2A BB 2F   FCTSOD:LHLD ADRES  *SOD-Ausgabe
0560 ODF8 23          INX H
0561 ODF9 EB          XCHG        *D&E: Endadresse
0562 ODFA 2A BF 2F   LHLD SRCBEG  *H&L: Startadresse
0563 ODFD 4E          OTLOP: MOV C,M    *Zeichen holen
0564 ODFE CD 28 0E   CALL CPUOT  *und ausgeben
0565 OE01 23          INX H
0566 OE02 7D          MOV A,L      *Endabfrage
0567 OE03 BB          CMP E
0568 OE04 C2 FD 0D   JNZ OTLOP  *nicht die Endadresse: JMP
0569 OE07 7C          MOV A,H
0570 OE08 BA          CMP D
0571 OE09 C2 FD 0D   JNZ OTLOP  *nicht die Endadresse: JMP
0572 OE0C C9          RET
0573 OE0D           *
0574 OE0D 2A BB 2F   FCTSID:LHLD ADRES  *SID-Eingabe
0575 OE10 23          INX H

```

0576	0E11	EB			XCHG		*D&E: Endadresse
0577	0E12	2A	BF	2F	LHLD	SRCBEG	*H&L: Startadresse
0578	0E15	3E	CO		MVI	A,0COH	*SOD auf HIGH
0579	0E17	30			SIM		* (=Ready)
0580	0E18	CD	58	0E	INLOP:	CALL CPUIN	*Zeichen einlesen
0581	0E1B	77			MOV	M,A	*und im RAM ablegen
0582	0E1C	23			INX	H	
0583	0E1D	7D			MOV	A,L	*Endabfrage
0584	0E1E	BB			CMP	E	
0585	0E1F	C2	18	0E	JNZ	INLOP	*nicht die Endadresse: JMP
0586	0E22	7C			MOV	A,H	
0587	0E23	BA			CMP	D	
0588	0E24	C2	18	0E	JNZ	INLOP	*nicht die Endadresse: JMP
0589	0E27	C9			RET		
0590	0E28				*		
0591	0E28				*		
0592	0E28				*	CPUOT	Ausgabe von (REG C) über die CPU-
0593	0E28				*		Schnittstelle SOD
0594	0E28				*		
0595	0E28	20			CPUOT:	RIM	*Ready abfragen
0596	0E29	A7			ANA	A	
0597	0E2A	F2	28	0E	JP	CPUOT	*SID auf LOW: JMP
0598	0E2D	79			MOV	A,C	*Zeichen holen
0599	0E2E	CD	49	0E	CALL	STABI	*Startbit ausgeben
0600	0E31	0E	08		MVI	C,8H	*Bit-Zähler
0601	0E33	0F			OTLP:	RRC	*LSB zuerst ausgeben
0602	0E34	A7			ANA	A	*Pegelabfrage
0603	0E35	FA	3E	0E	JM	HIGH	*Bit ist HIGH: JMP
0604	0E38	CD	49	0E	LOW:	CALL STABI	*LOW-Bit ausgeben
0605	0E3B	C3	41	0E	JMP	OUTCTE	
0606	0E3E	CD	4F	0E	HIGH:	CALL STOBI	*HIGH-Bit ausgeben
0607	0E41	0D			OUTCTE:	DCR C	
0608	0E42	C2	33	0E	JNZ	OTLP	
0609	0E45	CD	4F	0E	CALL	STOBI	*Stopbit ausgeben
0610	0E48	C9			RET		
0611	0E49				*		
0612	0E49	F5			STABI:	PUSH PSW	
0613	0E4A	3E	CO		MVI	A,0COH	*SOD auf HIGH bringen
0614	0E4C	C3	52	0E	JMP	BITCTE	
0615	0E4F	F5			STOBI:	PUSH PSW	
0616	0E50	3E	40		MVI	A,40H	*SOD auf LOW bringen
0617	0E52	30			BITCTE:	SIM	
0618	0E53	CD	7A	0E	CALL	DELY2	*Wartezeit
0619	0E56	F1			POP	PSW	
0620	0E57	C9			RET		
0621	0E58				*		
0622	0E58				*	CPUIN	Eingabe ins REG A über die CPU-
0623	0E58				*		Schnittstelle SID
0624	0E58				*		
0625	0E58	20			CPUIN:	RIM	*SID abfragen
0626	0E59	A7			ANA	A	
0627	0E5A	F2	58	0E	JP	CPUIN	*SID auf LOW: JMP
0628	0E5D	D5			PUSH	D	
0629	0E5E	16	08		MVI	D,8H	*Bit-Zähler
0630	0E60	1E	00		MVI	E,0H	*Schieberegister
0631	0E62	CD	0B	00	CALL	DELY1	*halbe Bit-Dauer warten
0632	0E65	CD	7A	0E	INLP:	CALL DELY2	*Mitte Datenbit
0633	0E68	20			RIM		
0634	0E69	E6	80		ANI	80H	*MSB maskieren
0635	0E6B	B3			ORA	E	*verknüpfen
0636	0E6C	0F			RRC		
0637	0E6D	5F			MOV	E,A	*ins Schieberegister
0638	0E6E	15			DCR	D	
0639	0E6F	C2	65	0E	JNZ	INLP	

```

0640 0E72 07          RLC          *Ergebnis in REG A
0641 0E73 F5          PUSH PSW
0642 0E74 CD 7A 0E    CALL DELY2    *Mitte Stopbit
0643 0E77 F1          POP PSW
0644 0E78 D1          POP D
0645 0E79 C9          RET
0646 0E7A          *
0647 0E7A CD 0B 00 DELY2: CALL DELY1
0648 0E7D CD 0B 00          CALL DELY1
0649 0E80 C9          RET
0650 0E81          *
0651 0E81          *****
0652 0E81          *
0653 0E81          *      Ansteuerung des Thermodruckers
0654 0E81          *
0655 0E81          *****
0656 0E81          *
0657 0E81          PRTHX EQU 2000H
0658 0E81          PRTASC EQU 2003H
0659 0E81          *
0660 0E81 2A BB 2F FCTHEX:LHLD ADRES      *HEX-Dump
0661 0E84 EB          XCHG      *D&E: Endadresse
0662 0E85 2A BF 2F          LHLD SRCBEG *Anfangsadresse
0663 0E88 3E 40          MVI A,40H
0664 0E8A D3 40          OUT FDCTRL *Hilfs-AdreBlatch
0665 0E8C CD 00 20        CALL PRTHX
0666 0E8F AF          XRA A
0667 0E90 D3 40          OUT FDCTRL
0668 0E92 3C          INR A
0669 0E93 C9          RET
0670 0E94          *
0671 0E94 2A BB 2F FCTASC:LHLD ADRES      *ASCII-Ausdruck
0672 0E97 22 C1 2F          SHLD SRCEND *Zeilenzahl
0673 0E9A 2A BF 2F          LHLD SRCBEG *String-Anfang
0674 0E9D 3E 40          MVI A,40H
0675 0E9F D3 40          OUT FDCTRL *Hilfs-AdreBlatch
0676 0EA1 CD 03 20        CALL PRTASC
0677 0EA4 AF          XRA A
0678 0EA5 D3 40          OUT FDCTRL
0679 0EA7 3C          INR A
0680 0EA8 C9          RET
0681 0EA9          *
0682 0EA9          END

```



```

0001 0000 *****
0002 0000 *
0003 0000 * ESCAPE Verarbeitung von ESCAPE-Sequenzen *
0004 0000 *
0005 0000 *****
0006 0000 *
0007 0000 * Anpassung des TEXASS-Textverarbeitungsprogramms
0008 0000 *
0009 0000 * Erweiterung des Mode-FLAGS (MODFLG):
0010 0000 *
0011 0000 *
0012 0000 *      x  2  1  0   x  x  x  x   MODFLG
0013 0000 *      :  :  :
0014 0000 *      :  :  : ..ESCO (wird von "1B" gesetzt)
0015 0000 *      :  :  : .....ESC1 zweiter Parameter
0016 0000 *      :  :  : .....ESC2 dritter Parameter
0017 0000 *
0018 0000 *
0019 0000 VIDCTL EQU  2AH
0020 0000 *
0021 0000 COCTE  EQU  1006H      *MOVID-CO
0022 0000 CLRVID EQU  102DH
   23 0000 CLRLIN EQU  1030H
0024 0000 CRSPOS EQU  1033H
0025 0000 CRSON  EQU  1036H
0026 0000 CRSOFF EQU  1039H
0027 0000 *
0028 0000 UPDATE EQU  2E56H
0029 0000 MODFLG EQU  2FBDH
0030 0000 HIBUF  EQU  2FD3H      *BRKPT #1
0031 0000 HBUF   EQU  2FD4H
0032 0000 HIBUF3 EQU  2FD6H      *BRKPT #2
0033 0000 HBUF3  EQU  2FD7H
0034 0000 *
0035 0000 VIDRAM EQU  3000H
0036 0000 HOME  EQU  3140H
0037 0000 LINZ  EQU  3730H
0038 0000 VIDEND EQU  3780H
0039 0000 *
0040 0000 LIN    EQU  50H        *80d Zeichen/Zeile
0041 0000 LININV EQU  OFFBOH   *-80d +1
   42 0000 *
0043 0000 BCUP   EQU  3140H
0044 0000 DEUP  EQU  3780H
0045 0000 HLUP  EQU  3190H
0046 0000 BCDWN EQU  377FH
0047 0000 DEDWN EQU  313FH
0048 0000 HLDWN EQU  372FH
0049 0000 *
0050 0000 ORG    OEBOH
0051 0EB0 OFS    OEBOH
0052 0EB0 *
0053 0EB0 *
0054 0EB0 F5     ESCAPE:PUSH PSW
0055 0EB1 E5           PUSH H
0056 0EB2 D5           PUSH D
0057 0EB3 C5           PUSH B
0058 0EB4 3E C3       MVI  A,0C3H      *"JMP"
0059 0EB6 32 D3 2F    STA  HIBUF
0060 0EB9 21 06 10    LXI  H,COCTE
0061 0EBC 22 D4 2F    SHLD HBUF
0062 0EBF *
0063 0EBF 3A BD 2F    LDA  MODFLG

```

0064	0EC2	47		MOV	B,A		
0065	0EC3	E6	10	ANI	10H	*ESC-FLAG abfragen	
0066	0EC5	C2	D2	0E	JNZ	ESTEST	*gesetzt: JMP
0067	0EC8	79		MOV	A,C		
0068	0EC9	FE	1B	CPI	1BH	*ESC abfragen	
0069	0ECB	CA	3D	0F	JZ	ESCSET	*ja: ESC-FLAG setzen
0070	0ECE	78		MOV	A,B		
0071	0ECF	C3	33	0F	JMP	ESCEND	
0072	0ED2			*			
0073	0ED2	78		ESTEST:MOV	A,B		
0074	0ED3	E6	40	ANI	40H	*ESC-FLAG 2 abfragen	
0075	0ED5	CA	E1	0E	JZ	ESCTE1	*gelöscht: JMP
0076	0ED8	79		MOV	A,C		
0077	0ED9	32	D7	2F	STA	HBUF3	*CO-Byte ablegen
0078	0EDC	0E	16	MVI	C,16H	*=TSCPOS	
0079	0EDE	C3	FD	0E	JMP	ESPRCS	
0080	0EE1	78		ESCTE1:MOV	A,B		
0081	0EE2	E6	20	ANI	20H	*ESC-FLAG 1 abfragen	
0082	0EE4	CA	F1	0E	JZ	ESCTE2	*gelöscht: JMP
0083	0EE7	79		MOV	A,C		
0084	0EE8	32	D6	2F	STA	HIBUF3	*CO-Byte ablegen
0085	0EEB	78		MOV	A,B		
0086	0EEC	F6	40	ORI	40H	*ESC-FLAG 2 setzen	
0087	0EEE	C3	33	0F	JMP	ESCEND	
0088	0EF1	79		ESCTE2:MOV	A,C		
0089	0EF2	FE	16	CPI	16H	*TSCPOS?	
0090	0EF4	C2	FD	0E	JNZ	ESPRCS	*nein: JMP
0091	0EF7	78		MOV	A,B		
0092	0EF8	F6	20	ORI	20H	*ESC-FLAG 1 setzen	
0093	0EFA	C3	33	0F	JMP	ESCEND	
0094	0EFD			*			
0095	0EFD	79		ESPRCS:MOV	A,C		
0096	0EFE	FE	20	CPI	20H	*kleinstes ASCII-Zeichen	
0097	0F00	D2	20	0F	JNC	ESCL	*ist ASCII: JMP und darstellen
0098	0F03	FE	1A	CPI	1AH	*oberer Grenzwert	
0099	0F05	D2	28	0F	JNC	ESCLR	*überschritten: JMP
0100	0F08	FE	10	CPI	10H	*unterer Grenzwert	
0101	0F0A	DA	28	0F	JC	ESCLR	*unterschritten: JMP
0102	0F0D	21	28	0F	LXI	H,ESCLR	*CTE-Adresse
0103	0F10	E5		PUSH	H	*in den Stack	
0104	0F11	21	43	0F	LXI	H,ESCTAB	
0105	0F14	D6	10	SUI	10H		
0106	0F16	07		RLC			
0107	0F17	5F		MOV	E,A		
0108	0F18	16	00	MVI	D,0H	*D&E: Versatz	
0109	0F1A	19		DAD	D	*H&L: Tabellenpointer	
0110	0F1B	5E		MOV	E,M		
0111	0F1C	23		INX	H		
0112	0F1D	56		MOV	D,M		
0113	0F1E	EB		XCHG		*H&L: Zieladresse	
0114	0F1F	E9		PCHL		*Verzweigung	
0115	0F20	3A	BD	2F	ESCL: LDA	MODFLG	
0116	0F23	E6	8F	ANI	8FH	*ESC-FLAGS löschen	
0117	0F25	C3	33	0F	JMP	ESCEND	
0118	0F28	3A	BD	2F	ESCLR: LDA	MODFLG	*weiter nach RET
0119	0F2B	E6	8F	ANI	8FH	*ESC-FLAGS löschen	
0120	0F2D	21	98	0F	ESCND: LXI	H,RETUR	
0121	0F30	22	D4	2F	SHLD	HBUF	
0122	0F33	32	BD	2F	ESCEND: STA	MODFLG	
0123	0F36	C1		POP	B		
0124	0F37	D1		POP	D		
0125	0F38	E1		POP	H		
0126	0F39	F1		POP	PSW		
0127	0F3A	C3	D3	2F	JMP	HIBUF	

0128	0F3D			*					
0129	0F3D	78		ESCSET:MOV	A,B				
0130	0F3E	F6	10	ORI	10H			*ESC-FLAG 0 setzen	
0131	0F40	C3	2D	OF	JMP	ESCND			
0132	0F43			*					
0133	0F43	2D	10	ESCTAB:DW	TSESCR			*P;Schirm löschen	
0134	0F45	57	0F	DW	TSCHOM			*Q;Cursor home	
0135	0F47	39	10	DW	TSCOFF			*R;Cursor aus	
0136	0F49	36	10	DW	TSCON			*S;Cursor ein	
0137	0F4B	5D	0F	DW	TSR05			*T;Scroll up	
0138	0F4D	7B	0F	DW	TSRU5			*U;Scroll down	
0139	0F4F	99	0F	DW	TSCPOS			*V;Cursor pos.(LL/CC)	
0140	0F51	AE	0F	DW	TSEE0S			*W;Schirm ab Cursor löschen	
0141	0F53	B7	0F	DW	TSELIN			*X;Zeile löschen	
0142	0F55	BD	0F	DW	TSEE0L			*Y;Zeile ab Cursor löschen	
0143	0F57			*					
0144	0F57			TSESCR EQU	CLRVID			*Schirm löschen	
0145	0F57			*					
0146	0F57	21	40	31	TSCHOM:LXI	H,HOME		*Cursor home	
0147	0F5A	C3	33	10	JMP	CRSPOS			
0148	0F5D			*					
0149	0F5D			TSCOFF EQU	CRSOFF			*Cursor aus	
0150	0F5D			*					
0151	0F5D			TSCON EQU	CRSON			*Cursor ein	
0152	0F5D			*					
0153	0F5D	01	40	31	TSR05: LXI	B,BCUP			
0154	0F60	11	80	37	LXI	D,DEUP			
0155	0F63	21	90	31	LXI	H,HLUP			
0156	0F66	DB	2A		UPLOP: IN	VIDCTL		*Blanking abwarten	
0157	0F68	0F			RRC				
0158	0F69	D2	66	0F	JNC	UPLOP			
0159	0F6C	7E			MOV	A,M			
0160	0F6D	02			STAX	B			
0161	0F6E	03			INX	B			
0162	0F6F	23			INX	H			
0163	0F70	7B			MOV	A,E		*Endabfrage	
0164	0F71	BD			CMP	L			
0165	0F72	C2	66	0F	JNZ	UPLOP			
0166	0F75	7A			MOV	A,D			
0167	0F76	BC			CMP	H			
0168	0F77	C2	66	0F	JNZ	UPLOP			
0169	0F7A	C9			RET				
0170	0F7B			*					
0171	0F7B	01	7F	37	TSRU5: LXI	B,BCDWN			
0172	0F7E	11	3F	31	LXI	D,DEDWN			
0173	0F81	21	2F	37	LXI	H,HLDWN			
0174	0F84	DB	2A		DWNLOP: IN	VIDCTL		*Blanking abwarten	
0175	0F86	0F			RRC				
0176	0F87	D2	84	0F	JNC	DWNLOP			
0177	0F8A	7E			MOV	A,M			
0178	0F8B	02			STAX	B			
0179	0F8C	0B			DCX	B			
0180	0F8D	2B			DCX	H			
0181	0F8E	7B			MOV	A,E		*Endabfrage	
0182	0F8F	BD			CMP	L			
0183	0F90	C2	84	0F	JNZ	DWNLOP			
0184	0F93	7A			MOV	A,D			
0185	0F94	BC			CMP	H			
0186	0F95	C2	84	0F	JNZ	DWNLOP			
0187	0F98	C9			RETUR: RET				
0188	0F99			*					
0189	0F99	2A	D6	2F	TSCPOS:LHLD	HIBUF3		*Cursor positionieren	
0190	0F9C	EB			XCHG			*D:Spalte; E:Zeile	
0191	0F9D	21	00	30	LXI	H,VIDRAM			

0192	OFA0	01 50 00		LXI	B,LIN	
0193	OFA3	09	POSLOP:	DAD	B	
0194	OFA4	1D		DCR	E	
0195	OFA5	C2 A3 0F		JNZ	POSLOP	
0196	OFA8	4A		MOV	C,D	
0197	OFA9	09		DAD	B	
0198	OFAA	CD 33 10		CALL	CRSPOS	
0199	OFAD	C9		RET		
0200	OFAE		*			
0201	OFAE	11 80 37	TSEEOS:	LXI	D,VIDEND	*Schirm ab Cursor löschen
0202	OFB1	2A 56 2E		LHLD	UPDATE	
0203	OFB4	C3 C8 0F		JMP	FILOP	
0204	OFB7		*			
0205	OFB7	CD DC 0F	TSELIN:	CALL	CLRSUB	*Zeile löschen
0206	OFBA	C3 30 10		JMP	CLRLIN	
0207	OFBD		*			
0208	OFBD	CD DC 0F	TSEEOL:	CALL	CLRSUB	*Zeile ab Cursor löschen
0209	OFCD	01 50 00		LXI	B,LIN	
0210	OFCD	09		DAD	B	
0211	OFCD	EB		XCHG		*D&E: Zeilenende +1
0212	OFCD	2A 56 2E		LHLD	UPDATE	*H&L: Cursor-Position
0213	OFCD		*			
0214	OFCD	DB 2A	FILOP:	IN	VIDCTL	*Blanking abwarten
0215	OFCA	0F		RRC		
0216	OFCD	D2 C8 0F		JNC	FILOP	
0217	OFCE	36 20		MVI	M,20H	*löschen
0218	OFD0	23		INX	H	
0219	OFD1	7B		MOV	A,E	
0220	OFD2	BD		CMP	L	*Endabfrage
0221	OFD3	C2 C8 0F		JNZ	FILOP	
0222	OFD6	7A		MOV	A,D	
0223	OFD7	BC		CMP	H	
0224	OFD8	C2 C8 0F		JNZ	FILOP	
0225	OFDB	C9		RET		
0226	OFDC		*			
0227	OFDC	01 B0 FF	CLRSUB:	LXI	B,LININV	*B&C: negative Zeilenlänge
0228	OFDF	11 80 37		LXI	D,VIDEND	
0229	OFE2	2A 56 2E		LHLD	UPDATE	
0230	OFE5	EB		XCHG		
0231	OFE6	7C	CLRLOP:	MOV	A,H	*H&L: Zeilenanfang
0232	OFE7	BA		CMP	D	*UPDATE
0233	OFE8	DA FA 0F		JC	CLREND	
0234	OFEB	C2 F6 0F		JNZ	CLRCTE	
0235	OFEE	7D		MOV	A,L	
0236	OFEF	BB		CMP	E	
0237	OFF0	DA FA 0F		JC	CLREND	
0238	OFF3	CA FA 0F		JZ	CLREND	
0239	OFF6	09	CLRCTE:	DAD	B	
0240	OFF7	C3 E6 0F		JMP	CLRLOP	
0241	OFFA	C9	CLREND:	RET		*H&L: Anfang Cursorzeile
0242	OFFB		*			
0243	OFFB		END			