

```

0001 0000 ; Lader.asm v3.0
0002 0000
0003 0000 ; Intel-Hex Datei ueber Serielle-Schnittstellenkarte lesen
0004 0000 ; Zeilenweise interpretieren (Zieladresse wird dem Hex-File entnommen)
0005 0000
0006 0000 ; Stand:
0007 0000 ; 21.04.2016 Hex-Vorpann testen, nach 16Byte Fehlermeldung
0008 0000 ; 19.04.2016 Umstellung auf Seriellesinterface der CF-Karte
0009 0000
0010 0000 ; 14.03.2015 Mit "Bootlader" aus EPROM 4000h ins RAM ab 2900h kopieren
0011 0000 ; 12.03.2015 Leseroutine auf Zeilenbuffer umgestellt
0012 0000 ; 17.02.2015 UART-Init ueberarbeitet und direkt am Programmanfang gesetzt
0013 0000 ; 01.02.2015 auf 04000h assembliert - alter Platz vom Basicinterpreter
0014 0000 ; 30.01.2015 Hex-Interpreter integriert
0015 0000 ; keine Ueberpruefung der Buffergrenzen
0016 0000 ; 27.01.2015 zeitfenster eingebaut
0017 0000 ; (Empfang wartet max 3sek auf Daten)
0018 0000 ; (Senden wartet max 12ms auf Empfaenger)
0019 0000 ; Buffergrenze auf 16kByte gesetzt
0020 0000 ; Meldetexte
0021 0000
0022 0000 ;-----
0023 0000 ;
0024 0000 ; PIO Ansteuerung
0025 0000
0026 0000 PIO_A equ 0c0h ; CAh Daten Kanal A
0027 0000 PIO_B equ 0d0h ; DAh Daten Kanal B
0028 0000 PIO_C equ 0e0h ; EAh Daten Kanal C
0029 0000 PIO_Ctl equ 0f0h ; FAh Steuerwort
0030 0000 ;
0031 0000 PIO_BA equ 0c1h ; Betriebsart
0032 0000 ; Kanal A bidirektional, Handshake
0033 0000 ; Kanal B als Ausgabe, Kanal C Bit
0034 0000 ;
0035 0000 ; AVR Befehle
0036 0000
0037 0000 AVRres equ 00h ; Spur 0 Sektor 0 ansteuern
0038 0000 SetSpur equ 10h ; Spurnummer setzen
0039 0000 SetSekt equ 20h ; Sektornummer setzen
0040 0000 Rdsekt equ 30h ; Daten vom aktuellen Sektor lesen
0041 0000 Wrsekt equ 40h ; Daten in den aktuellen Sektor sch
0042 0000
0043 0000 v24stat equ 5Fh ; Status der V24 Schnittstelle
0044 0000 ; Bit 3 CTS Signal
0045 0000 ; Bit 4 RTS Signal
0046 0000 ; Bit 5 Senderegister leer
0047 0000 ; Bit 7 Empfangsregister voll
0048 0000 v24open equ 50h ; V24 vorbereiten (RTS setzen)
0049 0000 v24close equ 51h ; noch nicht implementiert
0050 0000 v24btx equ 52h ; ein byte senden ohn
0051 0000 v24brx equ 53h ; ein byte empfangen
0052 0000 v24sttx equ 54h ; String senden Ende 00h mit
0053 0000 v24strx equ 55h ; Zeile empfangen bis CR
0054 0000 ;
0055 0000 AVRstat equ 0F0h ; Status ermittelt
0056 0000 ;
0057 0000 ;
0058 0000 ;-----
0059 0000 ;
0060 0000 ;
0061 0000 buffer EQU 2e10h ; Zeilenbuffer
0062 0000 SRCBEG EQU 2FBFh ; Startadresse
0063 0000 SRCEND EQU 2FC1h ; Endadresse
0064 0000 DSTBEG EQU 2FC3h ; Anfang Zieladresse/Parameter
0065 0000 DSTEND EQU 2FC5h ;
0066 0000 RAMBEG equ 37B0h ;
0067 0000 ;
0068 0000 DIGI1 equ 2E00h ;
0069 0000 ;
0070 0000 ; Monitorprogramme
0071 0000 ;
0072 0000 vwarm equ 1003h ; Videomonitor
0073 0000 co equ 0049h ; (C) Monitorausgabe
0074 0000 ci equ 0043h ; (A) Tastencode, Rueksprung erst nach Betaetigung,
0075 0000 csts equ 0052h ; Status Tastatur, gedruekt Z=0, sonst Z=1
0076 0000 dely1 equ 000bh ; warte 1ms
0077 0000 copy equ 0088h ; SRCBEG=Quelle start, SRCEND=Quelle End, DSTBEG=Ziel
0078 0000 onsec equ 001bh ; warte 1 Sekunde
0079 0000 string equ 1021h ; Textausgabe bis 00h
0080 0000 adrot equ 103Ch ; (DE) vierstellig ausgeben (HL)=Position
0081 0000 error EQU 1027h ; Error Videomonitor
0082 0000 ;
0083 0000 ;=====
0084 0000 ;
0085 0000 org 2900h ; Sprung vom Ladeprogramm

```

```

0086 2900
0087 2900 C3 1B 29 lader: jmp lade- ; Intel-Hex empfangen, interpretieren
0088 2903 C3 1A 2A hxi: jmp hxi- ; Intel-Hex Interpreter fuer eine Zeile
0089 2906 C3 E9 29 bytrx: jmp bytrx- ; ein Byte empfangen ueber CF-Karte
0090 2909 C3 F2 29 byttx: jmp byttx- ; ein byte senden
0091 290C C3 FD 29 strrx: jmp strrx- ; Zeile bis CR empfangen
0092 290F C3 98 29 init: jmp init- ; CF-Karte vorbereiten
0093 2912 C3 94 29 inibu: jmp inibu- ; Buffereinstellen
0094 2915
0095 2915 CD 1B 00 Ende: call onesec ;
0096 2918 C3 03 10 jmp vwarm ; zurueck zum Monitor
0097 291B
0098 291B ;=====
0099 291B ; Hauptprogramm
0100 291B
0101 291B ; - laden Intel-Hex-File zeilenweise
0102 291B ; - Intel-Hex Interpreter
0103 291B
0104 291B
0105 291B CD 98 29 lade-: call init- ; CF-Karte vorbereiten
0106 291E ; Bildschirmmenue ausgeben
0107 291E 21 D7 2A lxi h,meld_0 ; Meldung "Ladeprogramm"
0108 2921 CD 21 10 call string
0109 2924 21 06 2B lxi h,meld_1 ; Start-Meldung Datenuebertragung
0110 2927 CD 21 10 call string
0111 292A 3E 50 mvi a,v24open ; RTS setzen damit der Empfang nicht
0112 292C CD DD 29 call piowr ; ueberrannt wird
0113 292F CD 1B 00 call onesec ; Wartezeit bis Datenabgeholt werden
0114 2932 CD 1B 00 call onesec
0115 2935 CD 1B 00 call onesec
0116 2938
0117 2938 CD 94 29 lade_l: call inibu- ;
0118 293B CD FD 29 call strrx- ; Zeile empfangen
0119 293E FE 00 cpi 00h ; Fehler auswerten
0120 2940 C2 8E 29 jnz lade_e4 ; Bufferueberlauf
0121 2943
0122 2943 CD 94 29 call inibu- ; Buffer auf Anfang
0123 2946
0124 2946 CD 1A 2A call hxi- ; eine Zeile interpretieren und im Zielbuffer able
0125 2949 22 C5 2F shld dstend ; Test
0126 294C
0127 294C 0E 2A mvi c,2ah ; Sternchen pro Zeile ausgeben
0128 294E CD 49 00 call co ;
0129 2951
0130 2951 FE 01 cpi 01h ; Fehlercode auswerten
0131 2953 CA 7C 29 jz lade_e1 ; kein ASCII
0132 2956 FE 02 cpi 02h ;
0133 2958 CA 82 29 jz lade_e2 ; keine Startmarke
0134 295B FE 03 cpi 03h ;
0135 295D CA 88 29 jz lade_e3 ; falsche Pruefsumme
0136 2960 FE 04 cpi 04h ;
0137 2962 C2 38 29 jnz lade_l ; weiter bis Dateiende
0138 2965
0139 2965 21 5A 2B lxi h, meld_5 ; alles OK zurueck zum Monitor
0140 2968 CD 21 10 lade_e: call string ; Bildschirmmeldung ausgeben
0141 296B 3E 51 mvi a,v24close ; RTS freigeben
0142 296D CD DD 29 call piowr ;
0143 2970 CD 1B 00 call onesec ;
0144 2973 C3 15 29 jmp Ende
0145 2976
0146 2976 21 8E 2B lade_e0: lxi h,err_1 ;
0147 2979 C3 68 29 jmp lade_e ;
0148 297C
0149 297C 21 8E 2B lade_e1: lxi h,err_1 ;
0150 297F C3 68 29 jmp lade_e ;
0151 2982
0152 2982 21 A6 2B lade_e2: lxi h,err_2 ;
0153 2985 C3 68 29 jmp lade_e ;
0154 2988
0155 2988 21 C4 2B lade_e3: lxi h,err_3 ;
0156 298B C3 68 29 jmp lade_e ;
0157 298E
0158 298E 21 75 2B lade_e4: lxi h,err_0 ;
0159 2991 C3 68 29 jmp lade_e ;
0160 2994
0161 2994 ;=====
0162 2994
0163 2994 ; Unterprogramme
0164 2994
0165 2994 ; Buffer initialisieren
0166 2994 ; (HL) Buffer Startadresse
0167 2994 ; (DE) Buffer Endadresse
0168 2994
0169 2994 21 10 2E inibu-: lxi h,buffer ; ZeilenBuffer max 64Byte
0170 2997 C9 ret

```

```

0171 2998
0172 2998
0173 2998
0174 2998
0175 2998
0176 2998 CD CC 29 init: call pioinit ; PIO und AVR initialisieren
0177 299B 3E 00 mvi a,avrres ; Test ob CF-Erweiterung vorhanden ist;
0178 299D CD DD 29 call piowr ;
0179 29A0 AF xra a ;
0180 29A1 CD D3 29 call piord ;
0181 29A4 FE 00 cpi 00h ;
0182 29A6 C2 B8 29 jnz cferr ; CF nicht bereit -> Fehler
0183 29A9 C9 ret ; CF-Interface OK
0184 29AA
0185 29AA
0186 29AA
0187 29AA
0188 29AA 3E F0 avr_stat: mvi a,avrstat ; Status AVR ermitteln
0189 29AC CD DD 29 call piowr ;
0190 29AF CD D3 29 call piord ;
0191 29B2 FE 00 cpi 00h ;
0192 29B4 C2 C1 29 jnz cferr1 ; Driver Error
0193 29B7 C9 ret ;
0194 29B8
0195 29B8
0196 29B8 21 E7 2B cferr: lxi h,errmsg1 ; Fehlermeldung
0197 29BB CD 21 10 call string ; CF nicht bereit
0198 29BE 3E 01 mvi a,01 ;
0199 29C0 C9 ret ;
0200 29C1
0201 29C1 21 DB 2B cferr1: lxi h,errmsg ; Lese/schreibfehler
0202 29C4 3E 27 mvi a,27h ;
0203 29C6 CD 27 10 call error ;
0204 29C9 C3 15 29 jmp ende ;
0205 29CC
0206 29CC
0207 29CC
0208 29CC
0209 29CC
0210 29CC
0211 29CC
0212 29CC
0213 29CC F5 pioinit: push psw ; PIO Betriebsart einstellen
0214 29CD 3E C1 mvi a,PIO_BA ; PIO_BA equ 0c1h
0215 29CF D3 FA out PIO_Ctl ;
0216 29D1 F1 pop psw ;
0217 29D2 C9 ret ;
0218 29D3
0219 29D3
0220 29D3
0221 29D3
0222 29D3
0223 29D3
0224 29D3 DB EA piord: in PIO_C ; Status ermitteln
0225 29D5 E6 20 ani 00100000b ; IBF maskieren
0226 29D7 CA D3 29 jz piord ; warte auf Daten
0227 29DA DB CA in PIO_A ; daten lesen
0228 29DC C9 ret ; (A) Daten
0229 29DD
0230 29DD
0231 29DD
0232 29DD
0233 29DD F5 piowr: push psw ;
0234 29DE DB EA piosts: in PIO_C ; Status ermitteln
0235 29E0 E6 80 ani 10000000b ; /OBF maskieren
0236 29E2 CA DE 29 jz piosts ; warten auf Quittung /ACK
0237 29E5 F1 pop psw ;
0238 29E6 D3 CA out PIO_A ; Daten schreiben
0239 29E8 C9 ret ;
0240 29E9
0241 29E9
0242 29E9
0243 29E9
0244 29E9 3E 53 bytrx: mvi a,v24brx ; Lesebefehl
0245 29EB CD DD 29 call piowr ;
0246 29EE CD D3 29 call piord ; Daten holen
0247 29F1 C9 ret ;
0248 29F2
0249 29F2 F5 byttx: push psw ;
0250 29F3 3E 52 mvi a,v24bttx ;
0251 29F5 CD DD 29 call piowr ; Schreibbefehl
0252 29F8 F1 pop psw ;
0253 29F9 CD DD 29 call piowr ;
0254 29FC C9 ret ;
0255 29FD ;

```



```

0341 2A20 0E 0F          mvi c,0fh          ; Fehlerzaehler setzen
0342 2A22 7E             hxi_sl: mov a,m          ; Startmarke suchen
0343 2A23 FE 3A          cpi 3ah              ;
0344 2A25 CA 35 2A          jz hxi_ln          ; weiter mit Laenge holen
0345 2A28                ; Zeilenende(abspann) behandeln
0346 2A28 FE 0D          cpi 0Dh              ; test auf CR
0347 2A2A CA 31 2A          jz hxi_mt          ;
0348 2A2D 0D             dcr c                ; Fehlerzaehler testen
0349 2A2E CA 87 2A          jz hxi_erm          ; keine Startmarke gefunden
0350 2A31 23             hxi_mt: inx h          ; nachstes Byte holen
0351 2A32 C3 22 2A          jmp hxi_sl          ;
0352 2A35                ;
0353 2A35                ; Laenge holen
0354 2A35                ;
0355 2A35 C1             hxi_ln: pop b          ;
0356 2A36 23             inx h                ; 2.Byte holen
0357 2A37 CD 93 2A          call asc_h          ; ASCII->HEX
0358 2A3A DA 86 2A          jc hxi_era          ; Fehler kein ASCII
0359 2A3D 4F             mov c,a              ; Bytezaehler laden
0360 2A3E 80             add b                ;
0361 2A3F 47             mov b,a              ; Pruefsumme schreiben
0362 2A40                ;
0363 2A40                ; Adresse holen
0364 2A40                ;
0365 2A40 23             inx h                ; HighByte
0366 2A41 CD 93 2A          call asc_h          ;
0367 2A44 DA 86 2A          jc hxi_era          ; Fehler kein ASCII
0368 2A47 57             mov d,a              ;
0369 2A48 80             add b                ;
0370 2A49 47             mov b,a              ; Pruefsumme schreiben
0371 2A4A                ;
0372 2A4A 23             inx h                ; LowByte
0373 2A4B CD 93 2A          call asc_h          ;
0374 2A4E DA 86 2A          jc hxi_era          ; Fehler kein ASCII
0375 2A51 5F             mov e,a              ;
0376 2A52 80             add b                ;
0377 2A53 47             mov b,a              ; Pruefsumme schreiben
0378 2A54                ;
0379 2A54                ; Type holen
0380 2A54                ;
0381 2A54 23             inx h                ;
0382 2A55 CD 93 2A          call asc_h          ;
0383 2A58 DA 86 2A          jc hxi_era          ; Fehler kein ASCII
0384 2A5B FE 00          cpi 00h              ; Nutzdaten folgen
0385 2A5D C2 8F 2A          jnz hxi_erf          ; EOF erreicht
0386 2A60 80             add b                ;
0387 2A61 47             mov b,a              ; Pruefsumme schreiben
0388 2A62                ;
0389 2A62                ; Daten holen
0390 2A62                ;
0391 2A62 23             hxi_dl: inx h          ;
0392 2A63 7E             mov a,m              ;
0393 2A64 CD 93 2A          call asc_h          ;
0394 2A67 DA 86 2A          jc hxi_era          ; Fehler kein ASCII
0395 2A6A F5             push psw             ;
0396 2A6B 80             add b                ;
0397 2A6C 47             mov b,a              ; Pruefsumme schreiben
0398 2A6D F1             pop psw              ;
0399 2A6E EB             xchg                 ;
0400 2A6F 77             mov m,a              ; Datenbyte in Zielbuffer
0401 2A70 EB             xchg                 ;
0402 2A71 13             inx d                ;
0403 2A72 0D             dcr c                ;
0404 2A73 C2 62 2A          jnz hxi_dl          ; naechstes Byte
0405 2A76                ;
0406 2A76                ; Pruefsumme holen
0407 2A76                ;
0408 2A76 23             inx h                ;
0409 2A77 7E             mov a,m              ;
0410 2A78 CD 93 2A          call asc_h          ;
0411 2A7B DA 86 2A          jc hxi_era          ; Fehler kein ASCII
0412 2A7E 80             add b                ;
0413 2A7F FE 00          cpi 0h              ;
0414 2A81 C2 8B 2A          jnz hxi_ers          ; Fehler Pruefsumme
0415 2A84 23             inx h                ;
0416 2A85 C9             hxi_end: ret          ;
0417 2A86                ;
0418 2A86                ; Fehlerbehandlung
0419 2A86                ;
0420 2A86 C9             hxi_era: ret          ; (A)=01h CY=1 kein ASII-Zeichen
0421 2A87                ;
0422 2A87 AF             hxi_erm: xra a          ; (A)=02h Z=1 Startmarke nicht gefunden
0423 2A88 3E 02          mvi a,02h          ;
0424 2A8A C9             ret                  ;
0425 2A8B                ;

```

```

0426 2A8B AF      hxi_ers: xra a          ; (A)=03h Z=1 Pruefsumme falsch
0427 2A8C 3E 03      mvi a,03h
0428 2A8E C9          ret
0429 2A8F
0430 2A8F AF      hxi_erf: xra a          ; (A)=04h Z=1Dateiende (EOF)
0431 2A90 3E 04      mvi a,04h
0432 2A92 C9          ret
0433 2A93
0434 2A93          ;-----
0435 2A93
0436 2A93          ; ASCII -> HEX
0437 2A93          ; Teile aus Profimonitor rot entnommen (C)Rheinhard Goessler hms
0438 2A93
0439 2A93          ; (HL) Zeiger fuer Quellbuffer
0440 2A93          ; (A) Hexwert
0441 2A93
0442 2A93          ; Fehler: (A)=00h + CY=1 kein ASCII zeichen)
0443 2A93
0444 2A93 C5      asc_h:  push b
0445 2A94 D5          push d
0446 2A95 7E          mov a,m          ; 1. zeichen holen
0447 2A96 CD B1 2A      call hxtst
0448 2A99 DA AE 2A      jc asc_err      ; Fehler
0449 2A9C 07          rlc              ; Nible in obere Haelfte
0450 2A9D 07          rlc
0451 2A9E 07          rlc
0452 2A9F 07          rlc
0453 2AA0 E6 F0      ani 0F0h          ; unters Nible ausblenden
0454 2AA2 47          mov b,a
0455 2AA3
0456 2AA3 23          inx h            ; 2. Zeichen holen
0457 2AA4 7E          mov a,m
0458 2AA5 CD B1 2A      call hxtst
0459 2AA8 DA AE 2A      jc asc_err      ; Fehler
0460 2AAB E6 0F      ani 0Fh          ; oberes Nible ausblenden
0461 2AAD B0          ora b
0462 2AAE
0463 2AAE D1      asc_err: pop d
0464 2AAF C1          pop b
0465 2AB0 C9          ret              ; (A) Hex
0466 2AB1
0467 2AB1 C5      hxtst:  push b
0468 2AB2 E6 5F      ani 5fh          ;Grossbuchstabe
0469 2AB4 FE 10      cpi 10h          ;"0"
0470 2AB6 DA CA 2A      jc hexerr      ;Code zu klein:JMP
0471 2AB9 FE 1A      cpi 1ah          ;"9"+1
0472 2ABB DA D0 2A      jc hexcte      ;Code o.k.:JMP
0473 2ABE C6 C9      adi 0c9h          ;A...F adjust
0474 2AC0 FE 10      cpi 10h          ;"F"+1
0475 2AC2 D2 CA 2A      jnc hexerr      ;Code zu gross:JMP
0476 2AC5 FE 0A      cpi 0ah          ;korr. "A"
0477 2AC7 D2 D2 2A      jnc insft      ;A...F: JMP
0478 2ACA
0479 2ACA AF      hexerr: xra a          ;FLAGS setzen
0480 2ACB 3E 03      mvi a,03h          ;kein HEX-Digit
0481 2ACD 1F          rar
0482 2ACE C1          pop b
0483 2ACF C9          ret              ;Fehler: CY=1
0484 2AD0
0485 2AD0 D6 10      hexcte: sui 10h          ;Rest der "3x"
0486 2AD2 47      insft:  mov b,a
0487 2AD3 AF          xra a              ;Flags l  schen
0488 2AD4 78          mov a,b
0489 2AD5 C1          pop b
0490 2AD6 C9          ret              ;REG A: HEX-Byte
0491 2AD7
0492 2AD7          ;-----
0493 2AD7          ; Texte f  r Bildschirmmeldungen
0494 2AD7
0495 2AD7 0D      meld_0: db 0dh          ; CR Startmeldung
0496 2AD8 0A          db 0ah          ; LF
0497 2AD9 49 6E 74 65      db "Intel-Hex "
0497 2ADD 6C 2D 48 65
0497 2AE1 78 20
0498 2AE3 4C 61 64 65      db "Ladeprogramm"
0498 2AE7 70 72 6F 67
0498 2AEB 72 61 6D 6D
0499 2AEF 20 56 33 2E      db " v3.0 (C)"
0499 2AF3 30 20 28 43
0499 2AF7 29
0500 2AF8 57 2E 52 6F      db "w.Roemer 2016"
0500 2AFC 65 6D 65 72
0500 2B00 20 32 30 31
0500 2B04 36
0501 2B05 00          db 00h          ; Ende

```

```

0502 2B06
0503 2B06 0D      meld_1: db 0dh      ; CR
0504 2B07 0A      db 0ah      ; LF
0505 2B08 53 65 6E 64      db "Sender starten"
0505 2B0C 65 72 20 73
0505 2B10 74 61 72 74
0505 2B14 65 6E
0506 2B16 0D      crlf:  db 0dh      ; CR
0507 2B17 0A      db 0ah      ; LF
0508 2B18 00      db 00h      ;
0509 2B19
0510 2B19 0D      meld_2: db 0dh      ; CR
0511 2B1A 0A      db 0ah      ; LF
0512 2B1B 45 6D 70 66      db "Empfang Ende"
0512 2B1F 61 6E 67 20
0512 2B23 45 6E 64 65
0513 2B27 0D      db 0dh      ; CR
0514 2B28 0A      db 0ah      ; LF
0515 2B29 00      db 00h      ;
0516 2B2A
0517 2B2A 0D      meld_3: db 0dh      ; CR
0518 2B2B 0A      db 0ah      ; LF
0519 2B2C 45 6D 70 66      db "Empfaenger"
0519 2B30 61 65 6E 67
0519 2B34 65 72
0520 2B36 20 73 74 61      db " starten"
0520 2B3A 72 74 65 6E
0521 2B3E 0D      db 0dh      ; CR
0522 2B3F 0A      db 0ah      ; LF
0523 2B40 00      db 00h      ;
0524 2B41
0525 2B41 0D      meld_4: db 0dh      ; CR
0526 2B42 0A      db 0ah      ; LF
0527 2B43 55 65 62 65      db "uebertragung"
0527 2B47 72 74 72 61
0527 2B4B 67 75 6E 67
0528 2B4F 20 62 65 65      db " beendet"
0528 2B53 6E 64 65 74
0529 2B57 0D      db 0dh      ; CR
0530 2B58 0A      db 0ah      ; LF
0531 2B59 00      db 00h      ;
0532 2B5A
0533 2B5A 0D      meld_5: db 0dh      ; CR
0534 2B5B 0A      db 0ah      ; LF
0535 2B5C 48 65 78 2D      db "Hex-Interpreter"
0535 2B60 49 6E 74 65
0535 2B64 72 70 72 65
0535 2B68 74 65 72
0536 2B6B 20 66 65 72      db " fertig"
0536 2B6F 74 69 67
0537 2B72 0D      db 0dh      ; CR
0538 2B73 0A      db 0ah      ; LF
0539 2B74 00      db 00h      ;
0540 2B75
0541 2B75      ; -----
0542 2B75      ; Fehlermeldungen
0543 2B75 0D      err_0: db 0dh      ;
0544 2B76 0A      db 0ah      ;
0545 2B77 20 55 65 62      db " uebertragungs"
0545 2B7B 65 72 74 72
0545 2B7F 61 67 75 6E
0545 2B83 67 73
0546 2B85 66 65 68 6C      db "fehler"
0546 2B89 65 72
0547 2B8B 0D      db 0dh      ; CR
0548 2B8C 0A      db 0ah      ; LF
0549 2B8D 00      db 00h      ;
0550 2B8E
0551 2B8E 0D      err_1: db 0dh      ; CR
0552 2B8F 0A      db 0ah      ; LF
0553 2B90 20 6B 65 69      db " kein ASCII"
0553 2B94 6E 20 41 53
0553 2B98 43 49 49
0554 2B9B 20 5A 65 69      db " zeichen"
0554 2B9F 63 68 65 6E
0555 2BA3 0D      db 0dh      ; CR
0556 2BA4 0A      db 0ah      ; LF
0557 2BA5 00      db 00h      ;
0558 2BA6
0559 2BA6 0D      err_2: db 0dh      ; CR
0560 2BA7 0A      db 0ah      ; LF
0561 2BA8 20 6B 65 69      db " kein Startmarke"
0561 2BAC 6E 20 53 74
0561 2BB0 61 72 74 6D
0561 2BB4 61 72 6B 65

```

```

0562 2BB8 20 67 65 66          db " gefunden"
0562 2BBC 75 6E 64 65
0562 2BC0 6E
0563 2BC1 0D          db 0dh          ; CR
0564 2BC2 0A          db 0ah          ; LF
0565 2BC3 00          db 00h          ;
0566 2BC4
0567 2BC4 0D          err_3: db 0dh          ; CR
0568 2BC5 0A          db 0ah          ; LF
0569 2BC6 20 50 72 75          db " Pruefsumme"
0569 2BCA 65 66 73 75
0569 2BCE 6D 6D 65
0570 2BD1 20 66 61 6C          db " falsch"
0570 2BD5 73 63 68
0571 2BD8 0D          db 0dh          ; CR
0572 2BD9 0A          db 0ah          ; LF
0573 2BDA 00          db 00h          ;
0574 2BDB
0575 2BDB 0D          errmsg: DB 0dh          ; CR
0576 2BDC 0A          DB 0Ah          ; LF
0577 2BDD 45 72 72 6F          DB "Error F0"
0577 2BE1 72 20 46 30
0578 2BE5 0D          DB 0dh          ;
0579 2BE6 00          DB 00h
0580 2BE7
0581 2BE7 0D          errmsg1: DB 0dh          ; CR
0582 2BE8 0A          DB 0Ah          ; LF
0583 2BE9 20 43 46 2D          DB " CF-Karte "
0583 2BED 4B 61 72 74
0583 2BF1 65 20
0584 2BF3 6E 69 63 68          db "nicht vorhanden"
0584 2BF7 74 20 76 6F
0584 2BFB 72 68 61 6E
0584 2BFF 64 65 6E
0585 2C02 0D          DB 0dh          ;
0586 2C03 00          DB 00h
0587 2C04
0588 2C04
0589 2C04          ;-----
0590 2C04          ;
0591 2C04          ; Ladeprogramm fuer Intel-Hex Lader
0592 2C04          ; Hex-Lader aus EPROM ab 4200h ins RAM 2900h
0593 2C04          ;
0594 2C04          ;
0595 2C04          org      4000h      ;
0596 4000
0597 4000 C3 03 40          start: jmp start-      ; "Bootlader"
0598 4003          ;
0599 4003 21 00 42          start-: lxi      h,4200h ; Quelle Startadr.
0600 4006 22 BF 2F          shld     srcbeg ;
0601 4009 21 FF 44          lxi      h,44ffh ; Quelle Endadr.
0602 400C 22 C1 2F          shld     srcend ;
0603 400F 21 00 29          lxi      h,2900h ; Ziel Startadr.
0604 4012 22 C3 2F          shld     dstbeg ;
0605 4015 CD 88 00          call     copy ;
0606 4018
0607 4018 C3 00 29          jmp      2900h      ;
0608 401B
0609 401B
0610 401B
0611 401B
0612 401B
0613 401B
Number of errors = 0

```