

```

100 ' [DEB0]
110 'Sortierroutine fuer Fließkommazahlen [E18C]
    auf CPC 464 [E0B4]
120 ' [B76C]
130 '(C) 1986 by Dirk Haltermann [31C0]
140 '   Lueftlbergstr.1 [3C7C]
150 '   8390 Passau [048C]
160 '
170 'Basiclader: Aufruf mit CALL &A00A,@a(0 [0C34]
    ),Anzahl der Elemente-1 [06C0]
180 ' [54DE]
190 DATA A00A,FE,02,C0,ED,53,00,A0,ED,6A65 [4560]
200 DATA A012,53,02,A0,DD,6E,02,DD,66,3274 [9EE0]
210 DATA A01A,03,22,04,A0,22,00,A0,11,03E1 [8F18]
220 DATA A022,05,00,19,22,06,A0,ED,5B,00B1 [C6FC]
230 DATA A02A,08,A0,E5,DD,E1,D5,FD,E1,3837 [12BE]
240 DATA A032,DD,7E,04,FD,8E,04,38,40,7B80 [D0B2]
250 DATA A03A,20,38,B7,28,42,DD,7E,03,0BF8 [9B16]
260 DATA A042,FD,AE,03,FA,74,A0,DD,7E,5A24 [39B8]
270 DATA A04A,03,FD,96,03,20,10,DD,7E,2C80 [0772]
280 DATA A052,02,FD,96,02,20,15,DD,7E,2C30 [4ED0]
290 DATA A05A,01,FD,96,01,20,0D,DD,7E,2DE0 [F196]
300 DATA A062,00,FD,96,00,28,19,9F,FD,2D67 [70EE]
310 DATA A06A,AE,03,87,9F,38,11,3C,C3,4FEF [5CD0]
320 DATA A072,81,A0,DD,7E,03,C3,6C,A0,77AC [E212]
330 DATA A07A,FD,7E,03,2F,C3,6C,A0,DA,65A2 [3C52]
340 DATA A082,87,A0,22,08,A0,2A,02,A0,6A4C [2142]
350 DATA A08A,11,01,00,97,ED,52,28,09,07C9 [2306]
360 DATA A092,22,02,A0,2A,06,A0,C3,21,0437 [1FC8]
370 DATA A09A,A0,2A,06,A0,ED,5B,08,A0,56F4 [CB68]
380 DATA A0A2,46,1A,EB,70,12,13,23,46,3F3C [D160]
390 DATA A0AA,1A,EB,70,12,13,23,46,1A,3862 [F396]
400 DATA A0B2,EB,70,12,13,23,46,1A,EB,6A2F [2478]
410 DATA A0BA,70,12,13,23,46,1A,EB,70,3F2E [8014]
420 DATA A0C2,12,97,2A,00,A0,11,01,00,2CC6 [CA98]
430 DATA A0CA,ED,52,C8,22,00,A0,22,02,7BE6 [8D52]
440 DATA A0D2,A0,2A,04,A0,C3,1E,A0,00,5720 [48C6]
450 DATA *ENDE* [6E88]
460 adr=&A00A:zeile=190:MEMORY &9FFF [068E]
470 READ d$:IF d$="*ENDE*"THEN 610 [8E16]
480 pr=0 [9C6C]
490 FOR i=1 TO 8 [0138]
500 READ a$:a=VAL("&"+a$) [AD14]
510 POKE adr,a:adr=adr+1 [AF94]
520 pr=pr*2:IF pr>65535 THEN pr=pr-65535 [3FAC]
530 pr=UNT(pr)XOR a:IF pr<0 THEN pr=pr+65535 [4602]
540 NEXT i [AD8E]
550 READ pr$:pr2=VAL("&"+pr$):IF pr2<0 THEN [9616]
    pr2=pr2+65536 [196A]
560 IF pr<>pr2 THEN PRINT"Pruefsummenfehler [AEC8]
    in Zeile";zeile:STOP
570 zeile=zeile+1:GOTO 470 [1E80]
580 ' [0CBA]
590 'Demo: Sortieren von 100 Fließkommazah [A274]
    len [EC5C]
600 ' [0FC0]
610 DIM a(100):FOR i=0 TO 99:a(i)=RND*1000: [B138]
    NEXT [B1C4]
620 FOR i=0 TO 99:PRINT a(i);:NEXT:PRINT:PR [B844]
    INT [B0DA]
630 ' [2B8C]
640 'Sortieren [C26A]
650 ' [9EE4]
660 z=TIME
670 CALL &A00A,@a(0),99
680 z=(TIME-z)/300
690 FOR i=0 TO 99:PRINT a(i);:NEXT:PRINT:PR
    INT
700 PRINT"Zeit =";z;"Sekunden"

```

Listing 1. Basic-Lader und Demonstration zu Sort-Flo

```

ds 10      ;Sort FLO CALL &A00A,@a(0),100
cp 2       ;Zwei Parameter ?
ret nz     ;Nein RETURN
ld (&a000),de ;Anzahl der Elemente
ld (&a002),de ;in &A000 u. &A002
ld 1,(ix+2) ;Startadr. in &A004
ld h,(ix+3) ;und &A008
ld (&a004),hl
s1 ld (&a008),hl
s2 ld de,5    ;Vergleichsadr. + 5
add hl,de     ;in &A006
ld (&a006),hl ;Vergleichsadr.
ld de,(&a008) ;Adr. groesster Wert
push hl      ;Vergleich
pop ix
push de
pop iy
ld a,(ix+4)
cp (iy+4)
jr c,s6
jr nz,s5
or a
jr z,s7
ld a,(ix+3)
xor (iy+3)
jp m,s5
ld a,(ix+3)
sub (iy+3)
jr nz,s4
ld a,(ix+2)
sub (iy+2)
jr nz,s4
ld a,(ix+1)
sub (iy+1)
jr nz,s4
ld a,(ix+0)
sub (iy+0)
jr z,s7
sbc a,a
xor (iy+3)
s3 add a,a
s4 sbc a,a
jr c,s7
inc a
jp s7
s5 ld a,(ix+3)
jp s3
s6 ld a,(iy+3)
cpl
jp s3
s7 jp c,s8      ;if de)hl zu s8
ld (&a008),hl ;Adr. vom groesstem Wert
s8 ld hl,(&a002) ;Alle Elemente
ld de,1       ;verglichen ?
sub a
sbc hl,de
jr z,s9       ;wenn ja zu s9
ld (&a002),hl ;Elementenanzahl-1
ld hl,(&a006) ;Vergleichsadr. in &A006
jp s2         ;Sprung zu s2
s9 ld hl,(&a006) ;Elementenaustausch
ld de,(&a008) ;Das Element mit
ld b,(hl)    ;dem groessten Wert wird

```

```

ld a,(de)    ;ans Arrayende gesetzt
ex de,hl     ;der Wert am Arrayende
ld (hl),b    ;kommt an die Adresse
ld (de),a    ;mit dem groessten Wert
inc de
inc hl
ld b,(hl)
ld a,(de)
ex de,hl
ld (hl),b
ld (de),a
inc de
inc hl
ld b,(hl)
ld a,(de)
ex de,hl
ld (hl),b
ld (de),a
inc de
inc hl
ld b,(hl)
ld a,(de)
ex de,hl
ld (hl),b
ld (de),a
sub a        ;Ist die Anzahl der
ld hl,(&a000) ;unsortierten Elemente
ld de,1      ;gleich 1, dann stop
sbc hl,de
ret z
ld (&a000),hl ;sonst Elementenanzahl-1
ld (&a002),hl
ld hl,(&a004)
jp s1        ;Sprung zu s1

```

Listing 2. Der Assembler-Quellcode enthält die veränderte Kopie einer Betriebssystemroutine