

Floating Point Routines#

```
00010          .LI OFF
00020 -----
00030 *
00040 * FLOTINGPOINT  BERECHNUNGEN *
00050 *
00060 -----
00070 *
00080 *
00090 *
00100 *
00110 LBUFF      .EQ $580      WIRD VOM ROM FESTGELEGT
00120 *
00130 INBUFF     .EQ $F3       POINTER AUF LBUFF
00140 CIX        .EQ $F2       INDEXREGISTER. NORMAL =0
00150 FR0        .EQ $D4
00160 FR1        .EQ $E0
00170 FPPNTR     .EQ $FC       POINTER AUF FP ZAHL
00180 *
00190 -----
00200 AFP        .EQ $D800     ASCII=>FLOATING POINT (FP)
00210 FASC       .EQ $D8E6     FP=>ASCII IN LBUFF
00220 IFP        .EQ $D9AA     INTEGER FR0 ($D4)=> FP
00230 FPI        .EQ $D9D2     FP=>INTEGER IN FR0 ($D4)
00240 FSUB       .EQ $DA60     FR0<= FR0-FR1
00250 FADD       .EQ $DA66     FR0<= FR0+FR1
00260 FMUL       .EQ $DADB     FR0<= FR0*FR1
00270 FDIV       .EQ $DB28     FR0<= FR0/FR1
00280 -----
00290 FLDOR      .EQ $DD89     LADE FR0 MIT ZAHL
00300 * AUF DIE DIE REGISTER <X> UND <Y>
00310 * ZEIGEN. <X>=MSB, <Y>=LSB
00320 FLDOP      .EQ $DD8D     LADE FR0 MIT ZAHL
00330 * AUF DIE DER POINTER FPPNTR
00340 * ZEIGT.
00350 FLD1R      .EQ $DD98     WIE FLDOR DOCH MIT FR1
00360 FLD1P      .EQ $DD9C     WIE FLDOP DICH MIT FR1
00370 FSTOR      .EQ $DDA7     WIE FLDOR NUR IN ANDERER
00380 * RICHTUNG (SPEICHERN)
00390 FSTOP      .EQ $DDAB     WIE FSTOR NUR DURCH
00400 * FPPNTR ADRESSIERT.
00410 FMOVE      .EQ $DDB6     FR0=>FR1
00420 *
00430 -----
00440 INITFP     LDA #0        FP-ROUTINEN
00450           STA CIX        INITIALISIEREN
00460           LDA #LBUFF
00470           STA INBUFF
00480           LDA /LBUFF
00490           STA INBUFF+1
00500           RTS
00510 -----
00520 *
00530 -----
00540 PRINTFR0   JSR INITFP    INITIALISIERUNG
00550           JSR FASC       FP=>ASCII WANDELN
00560           LDY #0
00570 .1         STY WERT      ZEICHEN AUS
```

```

00580          LDA (INBUFF),Y PUFFER HOLEN
00590          BMI ENDE          INVERS? JA: LETZTES ZEICHEN
00600          JSR PUTCHAR        AUSGEBEN
00610          LDY WERT           NAECHSTES ZEICHEN
00620          INY
00630          BNE .1            HIER EIN JUMP
00640 ENDE      AND #$7F         INVERSES ZEICHEN ALS
00650          JMP PUTCHAR        NORMALZEICHEN AUSGEBEN
00660 *
00670 WERT      .HX 00           ZWISCHENSPEICHER
00680 *
00690 -----
00700 *
00710 -----
00720 PUTCHAR    TAX              VECTOR
00730          LDA $E407          JUMP
00740          PHA               UEBER
00750          LDA $E406          ROM
00760          PHA
00770          TXA
00780          RTS              JMP
00790 *
00800 -----
00810 CLRFR0     LDY #5           FR0
00820          LDA #0             LOESCHEN
00830 .1       STA FR0,Y
00840          DEY
00850          BPL .1
00860          RTS
00870 -----
00880 CLRFR1     LDY #5           FR1
00890          LDA #0             LOESCHEN
00900 .1       STA FR1,Y
00910          DEY
00920          BPL .1
00930          RTS
00940 -----

```

Floating Point Demo#

```

00010          .LI OFF
00020          .OR $4000
00030 *
00040 *
00050 * DAS FOLGENDE PROGRAMM
00060 * ENTSPRICHT DER BASIC-ZEILE:
00070 *
00080 * FOR I=0 TO 255:?I,I*I:NEXT I
00090 *
00100 *
00110 *
00120 *
00130 *
00140 *
00150 *
00160 *
00170 *
00180 START     JMP DEMO
00190 -----

```

```

00200 *
00210      .IN "D:FPOINT.INC"
00220 *
00230 -----
00240 DEMO      JSR CLRFR0      FR0 UND SCHLEIFEN-
00250           STA DWERT      WERT LOESCHEN
00260           STA DWERT
00270 *
00280 DLOOP     JSR IFP        INTEGER=> FP UMRECHNEN
00290           JSR PRINTFR0   ZAHL AUSGEBEN
00300           LDA #'#       TAB
00310           JSR PUTCHAR   AUSGEBEN
00320           JSR FMOVE     ZAHL NACH FR1 SCHIEBEN
00330           JSR FMUL      UND MIT FR0 MULTIPLIZIEREN
00340           BCS ERROR     IST DAS CAARY GESETZT SO LAG FEHLER VOR
00350           JSR PRINTFR0   ZAHL AUSGEBEN
00360           LDA #$9B      CR AUSGEBEN
00370           JSR PUTCHAR
00380           INC DWERT     INTEGERWERT
00390           LDA DWERT     HOCHZAEHLEN
00400           STA FR0      NACH FR0
00410           BEQ ERROR     BEI UEBERSCHLAG => ENDE
00420           LDA #0        MSB LOESCHEN
00430           STA FR0+1
00440 .1       JMP DLOOP     SCHLEIFE FORTFUEHREN
00450 ERROR     RTS
00460 *
00470 DWERT     .HX 00

```