

** BF'S LITTLE CHARSET SEARCH TOOL

** BF '88

** BEFEHLE:

** <START> SEKTOR VOR
** <SELECT> SEKTOR ZUREUCK
** <OPTION> SAVE CHARSET ALS DOS FILE
** <RESET> NEUSTART

** ORG \$A800
** JMP START

** ES FOLGEN MACROS FUER CIO

CIOV EQU \$E456

ICCOM EQU \$342
ICSTA EQU \$343
ICBAL EQU \$344
ICBAH EQU \$345
ICBLL EQU \$348
ICBLH EQU \$349
ICAX1 EQU \$34A
ICAX2 EQU \$34B

* CIO-Befehle

COPEN EQU 3
CCLSE EQU 12
CGTXT EQU 5
CPTXT EQU 9
CGBIN EQU 7
CPBIN EQU 11

EOL EQU \$9B

* MAKRO ZUR BERECHNUNG DER KANALNUMMER
* (hat nur interne Verwendung, ist
* Beispiel zur Verwendung von ver-
* schachtelten Makroaufrufen)

KANNUM MACRO KANAL
LDA #KANAL IOCB-Offset
ASL ;aus Kanalnr.
ASL ;(mal 16)
ASL
ASL
TAX ;ERGEBNIS IM X-REG
MEND

* Name : OPEN

* Zweck : oeffnen eines Files

* Aufruf:

* OPEN <Num>, <Aux1>, <Aux2>, <Filename>

* Beispiel: OPEN 1,4,0,"D:TEST.OBJ"

```

OPEN      MACRO KANAL,AUX1,AUX2,FILENAME
         JMP OP10
         ASC FILENAME
         DFB EOL
OP10     KANNUM KANAL
         LDA #AUX1
         STA ICAX1,X
         LDA #AUX2
         STA ICAX2,X
         LDA #COPEN
         STA ICCOM,X
         LDA #FNAME
         STA ICBAL,X
         LDA #FNAME/256
         STA ICBAH,X
         JSR CIOV
         MEND

*
* Name      : CLOSE
* Zweck     : File schliessen
* Aufruf    : CLOSE <Num>
* Beispiel: CLOSE 1
*
CLOSE    MACRO KANAL
         KANNUM KANAL
         LDA #CCLOSE
         STA ICCOM,X
         JSR CIOV
         MEND

*
* Name      : PRINT
* Zweck     : Ausgabe eines mit 'ASC'
*             definierten Textes, muss
*             mit EOL beendet werden
* Aufruf    : PRINT <kanal>,<Label>
* Beispiel: PRINT 0,TEXT1
*
PRINT    MACRO KANAL,LABEL
         KANNUM KANAL
         LDA #CPXTXT
         STA ICCOM,X
         LDA #LABEL
         STA ICBAL,X
         LDA #LABEL/256
         STA ICBAH,X
         LDA #127           max. Laenge
         STA ICBLL,X
         LDA #0
         STA ICBLH,X
         JSR CIOV
         MEND

*
* Name      : PRINTS
* Zweck     : direkte Ausgabe eines
*             Strings auf den Bildschirm
* Aufruf    : PRINT <String>
* Beispiel: PRINTS "HALLO"
*
PRINTS   MACRO STRING
         JMP PR20
         ASC STRING
         DFB EOL
PR10     PRINT 0,PR10           obiges Makro!
         MEND

```

```

* * Name : INPUT
* * Zweck : String einlesen
* * Aufruf : INPUT <Kanal>,<Label>
* * Beispiel: INPUT 0,TEXT1
*
INPUT      MACRO KANAL,LABEL
KANNUM KANAL
LDA #CGTXT
STA ICCOM,X
LDA #LABEL
STA ICBAL,X
LDA #LABEL/256
STA ICBAH,X
LDA #127           Max. Laenge
STA ICBLL,X
LDA #0
STA ICBLH,X
JSR CIOV
MEND

* * Name : BGET
* * Zweck : Einlesen eines Datenblockes
* *           der Laenge L ab Adresse A
* * Aufruf : BGET <NUM>,<L>,<A>
* * Beispiel: BGET 1,$B000,$100
*
BGET      MACRO KANAL,LAENGE,BUFFER
KANNUM KANAL
LDA #CGBIN
STA ICCOM,X
LDA #LAENGE
STA ICBLL,X
LDA #LAENGE/256
STA ICBLH,X
LDA #BUFFER
STA ICBAL,X
LDA #BUFFER/256
STA ICBAH,X
JSR CIOV
MEND

* * Name : BPUT
* * Zweck : Speichern eines Datenblockes
* *           der Laenge L ab Adresse A
* * Aufruf : BPUT<NUM>,<L>,<A>
* * Beispiel: BPUT 1,$B000,$100
*
BPUT      MACRO KANAL,LAENGE,BUFFER
KANNUM KANAL
LDA #CPBIN
STA ICCOM,X
LDA #LAENGE
STA ICBLL,X
LDA #LAENGE/256
STA ICBLH,X
LDA #BUFFER
STA ICBAL,X
LDA #BUFFER/256
STA ICBAH,X
JSR CIOV
MEND

* * DISPLAYLIST

```

```

* DLIST1    DFB 112
* DLIST1    DFB 7+64
* DLIST1    DFW TITLE
* DLIST1    DFB 7,7,7,6,2,2,2,2,6,6,65

* DLIST2    DFB 2+64
* DLIST2    DFW CHR
* DLIST2    DFB 2,2,112
* DLIST2    DFB 4+64
* DLIST2    DFW CHR
* DLIST2    DFB 4,4
* DLIST2    DFB 65
* DLIST2    DFW DLIST2

* SEKTORL   DFB 1 SNUMLOW
* SEKTORH   DFB 0 SNUM HI
* OLDDL     DFB 0,0 ADR ALTE DLIST.
* BUFF      DFB 0
* BUFFL     DFB 0
* BUFFH     DFB 0
*
* BILDSCREENHALTE
*
TITLE      ASC %           BF'S           %
TITLE      ASC $           LITTLE          $
TITLE      ASC %           ZEICHENSATZ    %
TITLE      ASC $           SCANNER         $
TITLE      ASC $-----$           %
TITLE      ASC $<START>  LESE NAECHSTE 3 SEKTOREN
$           ASC %<OPTION> SPEICHERE CHARSET ALS DO
$           ASC $<SELECT> SPEICHERE CHASET AB B00
$           ASC $           DIE ZULETZ GELESENEN SEK
$           ASC %-----%           %
$           ASC $DISK IN # 1 & START $
*
CHR        ORG *+120      (3 GR.2 ZEILEN)
*
* DLIST EIN
*
START     LDA 560      ADR ALTE DLIST
START     STA OLDDL    SICHERN.
START     LDA 561
START     STA OLDDL+1
*
*           LDA #$$5000:L BUFFER
*           STA BUFFL    IN LOW UND HI
*           LDA #$$5000:H
*           STA BUFFH
*
*           LDA #DLIST1
*           STA 560
*           LDA #DLIST1/256
*           STA 561
*
W4START   LDA 53279    START TASTE?
*           CMP #6
*           BNE W4START  NEIN.
*
* ALLE ZEICHEN IN BILDRAM

```

```
*          LDA #DLIST2
*          STA 560
*          LDA #DLIST2/256
*          STA 561
*
*          LDX #0
*          TXA
*          STA CHR,X
*          INX
*          CPX #128
*          BNE CHRLOOP
*
*          ** ZEICHENSATZ ZEIGER NACH $5000
*
*          LDA #$5000:H
*          STA 756
*
*          ** SEKTOR LESEN UND NACH $5000
*
*          LDA #1
*          STA 769      LW #1
*
*          LDA #'R
*          STA 770      READ DATA
*
*          READ      LDX #0
*          READ      STA BUFF
*          READNEXT INX
*          READNEXT STA BUFF
*          READNEXT STA BUFFL
*          READNEXT STA 772      BUFFER LOW
*          READNEXT STA BUFFH
*          READNEXT STA 773      BUFFER HI
*
*          LDA SEKTORL SNUM LOW
*          STA 778
*          LDA SEKTORH SNUM HI
*          STA 779
*
*          JSR ERH
*          JSR 58451      DISKINU
*
*          LDX BUFF
*          CPX #9
*          BNE READNEXT
*
*          LDA #$5000:L  BUFFER ZUREUCK
*          STA BUFFL
*          LDA #$5000:H
*          STA BUFFH
*
*          LOOP      LDA 53279
*          LOOP      CMP #6
*          LOOP      BNE NOT6
*          LOOP      JMP READ      LESE SEKTOR.
*
*          NOT6     CMP #5
*          NOT6     BNE NOT5
*          NOT6     JMP WRITE      $B000 SPEICHERN
*
*          NOT5     CMP #3
*          NOT5     BNE NOT4
*          NOT5     JSR DOSSAV
*
```

NOT4 JMP LOOP

*
* WRITE
*
WRITE LDA #\$B000:L
 STA BUFFL
 LDA #\$B000:H
 STA BUFFH

*
SEC LDA SEKTORL SEKTORNUMMER
 SBC #9 MINUS 3
 STA SEKTORL
 LDA SEKTORH
 SBC #0
 STA SEKTORH

*
LDA #'R
STA 770

*
LDY #0
STY BUFF

WR INY
 STY BUFF
 LDA BUFFL
 STA 772
 LDA BUFFH
 STA 773
 JSR ERH
 JSR 58451
 LDY BUFF
 CPY #9
 BNE WR

*
LDA #\$5000:L
STA BUFFL
LDA #\$5000:H
STA BUFFH
LDA #'R
STA 770
JMP READ

*
* SEKTOR ERNIEDRIGEN
*
ZUREUCK SEC
 LDA SEKTORL
 SBC #1
 STA SEKTORL
 LDA SEKTORH
 SBC #0
 STA SEKTORH
 RTS

*
* SEKTORNUMMER ERHOEHEN
*
ERH CLC
 LDA SEKTORL
 ADC #1
 STA SEKTORL
 LDA SEKTORH
 ADC #0
 STA SEKTORH

*
CLC
LDA BUFFL

```
ADC #128
STA BUFFL
LDA BUFFH
ADC #0
STA BUFFH
RTS

*
** SAVE ALS DOS FILE
*
DOSSAV    LDA OLDDL ALTE DISPLAY-
STA 560 LIST WIEDER
LDA OLDDL+1 HERSTELLEN.
STA 561
LDA #224 ORGINAL ZEICHEN-
STA 756 SATZ EIN.

*
PRINTS "Zeichensatz als DOS File speic
hern."
=====
PRINTS "====="
PRINTS "Name=D:CHR.FNT"
PRINTS "Bitte Disk in LW #1 und Taste
"
LDA #255
STA 764
LDA 764
CMP #255
BCS TASTE
LDA #255
STA 764

*
PRINTS "Speichere D:CHR.FNT."
*
OPEN 1,8,0,"D:CHR.FNT"
BPUT 1,1024,$5000
CLOSE 1

*
PRINTS "FERTIG !!! <TASTE>"

*
TASTE2    LDA 764
CMP #255
BCS TASTE2
JMP START
```