

Read Decimal, Hex and Binary values#

from BiboAssembler Tooldisk 1

```
00010          .LI OFF
00020 *
00030 *
00040 -----
00050 * Eingaberoutinen fuer      *
00060 * Dezimalzahlen,            *
00070 * Hexadezimalzahlen und    *
00080 * Binaerzahlen              *
00090 * Ohne Korrekturmoeglichkeit *
00100 *                          *
00110 * ACHTUNG: Routinen benoeti- *
00120 * gen die Ein- und          *
00130 * Ausgaberoutinen          *
00140 * aus dem File              *
00150 * INOUT.INC                 *
00160 * Sollten diese            *
00170 * nicht schon an an-      *
00180 * derer Stelle              *
00190 * stehen muessen Sie*
00200 * sie hier mit ein-      *
00210 * binden.                    *
00220 -----
00230 *
00240 *
00250 *
00260 *
00270 *
00280 GETDEZ   LDA #0           Wert
00290          STA WERT         loeschen
00300          STA WERT+1
00310 .1      JSR GETKEY       Taste holen
00320          CMP #$9B         Nur RETURN
00330          BEQ OUT          und
00340          CMP #'0         Ziffern
00350          BCC .1          von 0-9
00360          CMP #'9+1       zulassen
00370          BCS .1
00380          PHA
00390          JSR PUTCHAR     ausgeben
00400          PLA
00410          AND #$F         Nur untere
00420          PHA             4 Bits werden gebraucht
00430 *
00440          LDA WERT         Wert
00450          STA HOLD        wird
00460          LDA WERT+1      mit 10
00470          STA HOLD+1     multi-
00480          ASL HOLD        pliziert
00490          ROL HOLD+1     plus
00500          ASL HOLD        Ziffern
00510          ROL HOLD+1     wert
00520          CLC
00530          LDA HOLD
00540          ADC WERT
00550          STA WERT
```

```

00560      LDA HOLD+1
00570      ADC WERT+1
00580      STA WERT+1
00590      ASL WERT
00600      ROL WERT+1
00610      CLC
00620      PLA
00630      ADC WERT
00640      STA WERT
00650      BCC .2
00660      INC WERT+1
00670 .2    JMP .1          Zurueck in Schleife
00680 *
00690 OUT    JMP PUTCHAR   Ausstieg mit RETURN
00700 *
00710 WERT   .HX 0000     Hilfsregister
00720 HOLD   .HX 0000
00730 *
00740 -----
00750 GETHEX  LDA #0       Wert
00760         STA WERT     loeschen
00770         STA WERT+1
00780         LDA #3       Nur 4
00790         STA HOLD     Zeichen
00800 *
00810 .1     JSR GETKEY   Taste holen
00820         CMP #$9B     Nur
00830         BEQ OUT      RETURN
00840         CMP #'0      sowie
00850         BCC .1
00860         CMP #'G      0-9
00870         BCS .1       und
00880         CMP #'9+1    A-F
00890         BCC .2
00900         CMP #'A      zulassen
00910         BCC .1
00920 *
00930         PHA
00940         JSR PUTCHAR  Ausgeben
00950         PLA
00960 .2     CMP #'9+1    Wert
00970         BCC .3       auf 0-15
00980         SEC          umrechnen
00990         SBC #7
01000 .3    SEC
01010         SBC #$30
01020         LDX #3      Zahl*16
01030 .4    ASL WERT
01040         ROL WERT+1
01050         DEX
01060         BPL .4
01070         ORA WERT     Wert+Zahl
01080         STA WERT     Zahl
01090         DEC HOLD    naechtes
01100         BPL .1      Zeichen
01110 OUT2   LDA #$9B   Ausstieg mit
01120         JMP OUT     Return
01130 *
01140 -----

```

01150	*		
01160	GETBIN	LDA #0	Zahl
01170		STA WERT	loeschen
01180		STA WERT+1	
01190		LDA #7	8 Zeichen
01200		STA HOLD	
01210	*		
01220	.1	JSR GETKEY	Taste holen
01230		CMP #\$9B	Nur
01240		BEQ OUT2	RETURN
01250		CMP #'0	und
01260		BCC .1	Ziffern
01270		CMP #'2	0 und 1
01280		BCS .1	zulassen
01290		PHA	
01300		JSR PUTCHAR	Ausgeben
01310		PLA	
01320		AND #1	Nur ein
01330		LSR	Bit wird
01340		ROL WERT	gebraucht
01350		DEC HOLD	naechtes Zeichen
01360		BPL .1	
01370		JMP OUT2	Ausstieg