

---

## Ein hochgenauer Analog-Digital-Umsetzer mit ICL7

Gunther Zielosko

### 1. Grundlagen

Der BASIC-Tiger® verfügt mit seinen 4 Analogeingängen über eine praxisorientierte Ausstattung. Auflösungen von 8 oder 10 Bit sind Standard, 12 Bit sind durch mathematische Interpolation zu erreichen. Die folgende Tabelle zeigt die Verhältnisse bei den einzelnen Auflösungen:

Auflösung in Bit	Anzahl der Meßwerte	Auflösung bei 5V-Meßbereich ca.
8	256	20mV
10	1024	5mV
12	4096	1mV

Sollte jedoch für ein Projekt 16 Bit erforderlich sein, bietet sich der hochgenaue A-D-Umsetzer ICL 7135 an, welcher optimale Realisierungen ermöglicht.

Wie jeder andere hat auch der im BASIC-Tiger® eingebaute AD-Wandler Schwankungen, die im Analog- und Digitalisierungsteil liegen. So muß man z.B. prinzipiell mit einer Abweichung von plus/minus einem Digit rechnen, d.h. bei einem angelegten Spannungswert von 1 V kann der angezeigte Wert bei 8 Bit-Einstellung 0,98 V oder 1,02 V betragen.

Bei den genannten Bit-Auflösungen sind die jeweiligen Quantelungen aus diesem Grunde nicht exakt, bei der 8-Bit-Auflösung und 5 V Meßbereich springt der angezeigte Meßwert um genau 19,53...mV. Diesen Effekt versucht man dadurch auszugleichen, daß man den Meßbereich entsprechend anpaßt, deshalb wählt man z.B. bei der 12-Bit-Auflösung den Meßbereich von 0 bis 4,096 V. Damit ist die Auflösung genau 1 mV. Bei den klassischen digitalen Meßgeräten löst man das Problem jedoch andersherum. Der eingebaute AD-Wandler ist so konfiguriert, daß man die Quantelung und den Meßbereich auf die Anzeige optimiert hat. Ein typisches Digitalvoltmeter hat z.B. einen Anzeigebereich von +2,000 V bis -2,000 V. Dabei werden dann "echte" 1 mV-Schritte erreicht.

Die vorliegende Applikation nutzt diese Eigenschaften bestimmter AD-Wandler, um mit dem BASIC-Tiger® ein genaues und bedienungsfreundliches Meßsystem zu bekommen.

Die beschriebenen AD-Wandler für anzeigende Meßgeräte arbeiten meist nach dem Dual-Slope-Verfahren. Dieses Verfahren bietet Vorteile, wie z.B. eine sehr genaue Messung, eine günstige Darstellung der Meßwerte, eine unkomplizierte Stromversorgung, gute Möglichkeiten des Nullpunktgleiches, eine sehr hochohmige Eingangsimpedanz und vieles anderes mehr. Jedoch werden wegen des integrierenden Prinzips nur wenige Messungen pro Sekunde durchgeführt. Für die Anwendung in Rechnersystemen ist bei vielen solchen DVM-Schaltkreisen zudem zu vermerken, daß sie für die direkt Ansteuerung von Displays (LED, LCD) ausgelegt sind. Für die vorliegende Applikation wurde deshalb der ICL7135 gewählt, der in idealer Weise alle unsere Wünsche erfüllt.

## 2. Das Schaltungskonzept mit dem ICL7135 AD-Wandler-IC

Dieser Wandlerbaustein ist preiswert und wird von verschiedenen Herstellern und Distributoren angeboten. Mit wenigen externen Bauteilen ist der Analogteil betriebsfähig und arbeitet selbstständig. Er bietet im +/- 2 V-Bereich eine Auflösung von 40000 Schritten (-1,9999 V bis +1,9999V). Würde man die obige Tabelle mit den Daten des ICL7135 ergänzen, käme folgendes heraus:

Auflösung in Bit	Anzahl der Meßwerte	Auflösung bei 2V-Meßbereich genau
15 bis 16	39999	0,1mV

Ein Datenblatt von MAXIM ist über folgenden Link direkt einzusehen:

Datenblatt ICL7135 <http://209.1.238.250/arpdf/1496.pdf>

Der 28-polige Wandler hat einen für viele Multimeteranwendungen typischen Analogteil (die komplette äußere Beschaltung zeigt das Bild am Ende). Die Komponenten sind unkritisch, wichtig bei erhöhten Genauigkeitsansprüchen sind die Kondensatoren und die Referenzspannungserzeugung. Man kann die Spannung von genau 1,0000 V mit Widerstandsteiler entweder aus der Betriebsspannung (weniger genau: Variante 1) oder aus einer Bandgap-Referenz (hochgenau: Variante 2) ableiten.

Der Digitalteil ist auf den Anschluß von Mikroprozessoren zugeschnitten. Die Ausgabe der Meßwerte erfolgt gemultiplext im BCD-Format an den 4 BCD-Ausgängen B1, B2, B4, B8, die Zuordnung der Ziffern zu den Stellen wird über die Digit-Ausgänge D1, D2, D3, D4, D5 erreicht. Intern bildet der ICL7135 also den kompletten Meßwert über 5 Digits und gibt die Ziffern von 0-9, die über vier Bit kodiert werden, nacheinander aus:

0 = 0000  
1 = 0001  
2 = 0020  
:  
9 = 1001

Wann welche Stelle dran ist, wird mit High-Signal an dem jeweiligen Digitausgang signalisiert. Mit diesen Informationen könnte der BASIC-Tiger® bereits problemlos arbeiten, allerdings wären 4 BCD- und 5 Digitleitungen sowie eine weitere für die Overrange-Anzeige nötig. Dazu käme noch eine weitere Leitung und zwar die Anzeige der Polarität. Insgesamt sind also 11 Leitungen vorhanden, d.h. mehr als ein kompletter Port des BASIC-Tiger®s. Deshalb wird ein Trick angewandt, um mit dem Port 8 allein auszukommen. Die Digitleitungen werden so geschickt kodiert, daß Port 8 ausreicht. Dies ist günstig, wenn die anderen Komponenten des BASIC-Tiger® wie Display usw. ohne Einschränkungen weiterbenutzt werden sollen. Der Trick wird mit dem Prioritäts-Encoder CD4532 geschafft, der alle wichtigen Informationen des ICL7135 "einsammelt" und in geeigneter Form an den BASIC-Tiger® weitergibt. Zur Funktion dieses IC sei darauf verwiesen, daß es an insgesamt 8 Eingängen I0 bis I7 überprüft, welches das höchstwertige Bit mit H-Pegel ist. In unserer Schaltung ist dies der Overrange-Ausgang des AD-Wandlers, der an I7 liegt. Ist Overrange

aktiv (H-Pegel), macht es daraus dann unabhängig von sonstigen Aufgaben die Ausgangsinformation HHH der drei Ausgänge O2 bis O0. Ist Overrange inaktiv, hat Digit D5 höchste Priorität, ist es high, kommt die Ausgabe HHL an O2 bis O0 usw. So erfolgt für die 5 Digits und Overrange (6 Bit) eine Umsetzung auf 3 Bit.

Unter dem folgenden Link erhalten Sie das Datenblatt:

Datenblatt CD4532 <http://www-s.ti.com/sc/psheets/schs082/schs082.pdf>

Der Vorteil der Reduzierung der notwendigen Portleitungen wird durch eine aufwendigere Digitalbeschaltung und mit einer erweiterten Software ermöglicht. Es ist nun notwendig, die Zuordnung der Informationen auf den Port 8 des BASIC-Tigers® ein wenig zu verschlüsseln. Zum besseren Verständnis soll die folgende Tabelle dienen. Am Beispiel des Meßwertes +13479 wird gezeigt, wie sich die Informationen verteilen:

Portleitung BASIC-Tiger®	87	86	85	84	83	82	81	80	Wertigkeit der oberen	Vorzeichen	Dezimalwert, untere 4 Bits
Pinname	O2	O1	O0	POL	B8	B4	B2	B1	3 Bits dezimal		
Ausgang von	4532	4532	4532	7135	7135	7135	7135	7135			
	Digit-Kodierung			Vorz.	BCD-Wert						
OVRG aktiv	H	H	H	H	L	L	L	L	224	+	*
Digit 5 aktiv	H	H	L	H	L	L	H	H	192	+	1
Digit 4 aktiv	H	L	H	H	L	H	L	L	160	+	3
Digit 3 aktiv	H	L	L	H	L	H	H	H	128	+	4
Digit 2 aktiv	L	H	H	H	H	L	L	H	96	+	7
Digit 1 aktiv	L	H	L	H	H	L	L	H	96	+	9

\* bei Overrange wird immer 00000 angezeigt

Das dem BASIC-Tiger® angebotene Byte für jede Stelle besteht damit aus 3 Informationsteilen:

1. dem 4-Bit-BCDWert für die jeweilige Ziffer (niederwertiger Teil, Bits 0-3)
2. dem Bit für das Vorzeichen, H für positiv, L für negativ (Bit 4)
3. dem kodierten Digitalwert des jeweiligen Digits unter Berücksichtigung einer eventuellen Meßbereichsüberschreitung (höherwertiger Teil, Bits 5-7)

Dieses komplizierte „Datengemisch“ muß die Software dann wieder entkoppeln.

---

Neben einigen passiven Bauteilen benutzen wir noch einen 7660 zur Erzeugung der negativen Spannung V- von -5 V für den ICL7135 sowie einen Timer NE555 zur Takterzeugung. Für beide Teile hier die "Direktverbindung" zum Datenblatt:

Datenblatt ICL7660

<http://209.1.238.250/arpdf/1017.pdf>

Datenblatt NE555

<http://www-s.ti.com/sc/psheets/slfs022/slfs022.pdf>

### **3. Software**

Das kleine Programm BT7135B1.TIG macht die Schaltung nun zum Digitalvoltmeter. Am Anfang stehen die Definitionen und Installationen, die zum Betreiben des alphanumerischen Displays, für eine eventuelle Tastatur usw. notwendig sind. Danach erfolgt die Deklaration der Variablen und es werden Printformate festgelegt. Da wir nur den Port 8 einlesen müssen, genügt eine Eingangsdefinition für diesen Port.

Im eigentlichen Programm wird zunächst abgefragt, ob ein Meßwertüberlauf vorliegt. Das ist immer dann der Fall, wenn der AD-Wandler an ORNG High-Pegel anliegt. Daraus macht der Prioritäts-Encoder an seinen Ausgängen immer HHH. Da dies die 3 höchstwertigen Bits am Port 8 des BASIC-Tiger<sup>®</sup> sind, errechnet sich daraus dezimal der Wert 224 (128+64+32). Die Konzentration auf diese Bits wird mit dem Befehl BITAND erreicht, bei dem die nicht interessierenden Stellen einfach ausgeblendet werden. Auf ähnliche Weise wird mit den 5 Digits verfahren. Ist ein Digit erkannt, wird wiederum mit dem BITAND-Befehl der BCD-Wert dieser Ziffer ausgelesen. Zum Schluß erfolgt die dezimale Zusammensetzung des Meßwertes und, als Anregung für weitere Experimente, eine Maximumbildung.

### **4. Fazit**

Wir haben mit dem hier beschriebenen Verfahren einen Digitalvoltmeter gebaut und können nun überprüfen, ob sich die in dieses Projekt investierte Arbeit gelohnt hat.

Sicher könnte für diese Aufgabenstellung auf andere Multimeter mit vielen Meßgeräten zurückgegriffen werden. Die Funktion der Maximumbildung deutet jedoch an, was mit diesem Meßgerät noch alles ermöglicht werden kann. Es kann, wenn man will, bei einer programmierbaren Grenzwertüberschreitung hupen, seine Ergebnisse seriell ausgeben, Datum und Uhrzeit der Messung festhalten, aus dem Meßwert Steuerungsfunktionen ableiten und vieles andere mehr. Die hier vorgestellte Grundfunktion eines genauen Digital-Voltmeters ist die Basis für Meßwerterfassungssysteme, Ladegeräte-Konzepte, Steuerungen usw.

Der Aufbau kann entweder als Kompletgerät oder als Zusatzschaltung für das Plug-and-Play-Lab erfolgen. Im letzteren Falle wird der im Schaltplan umrahmte Teil allein aufgebaut und mit dem Port 8 sowie mit +5 V und der Masse des Systems kontaktiert.

```
-----  
' Name: BT7135B1.TIG mit ICL7135 und BCD-Umrechnung.  
-----  
  
#INCLUDE DEFINE_A.INC  
USER_EPORT ACT,NOACTIVE  
USER_VAR_STRICT  
#DEFINE LCD 1  
#INCLUDE KEYB_PP.INC  
#INCLUDE UFUNC.INC  
  
TASK MAIN  
  
INSTALL DEVICE #1, "LCD1.TDD"  
  
    BYTE A,B,C,D,E,F,G,H,L  
  
    LONG J  
    REAL I,K  
  
    DIR_PORT 8,255  
    PRINT #1, "<1Bh>c<0><F0h>";  
    K = -2  
    L = 0  
    USING "NF<1><1>0<4>V0.0.0.0.0.3.3,4.0.0.0.0.0.0" ' Format für Print-Ausgabe  
  
overflow:  
    IN 8, A  
    IF A BITAND 11100000b = 224 THEN  
        L = 1  
    ELSE  
        L = 0  
    ENDIF  
  
digit5:  
    IN 8, A  
    IN 8, B  
    IF B <> A THEN  
        GOTO digit5  
    ENDIF  
    IF A BITAND 11100000b = 192 THEN  
        C = A BITAND 00001111b  
        H = A BITAND 00010000b  
    ELSE  
        GOTO digit5  
    ENDIF  
  
digit4:  
    IN 8, A  
    IF A BITAND 11100000b = 160 THEN  
        D = A BITAND 00001111b  
    ELSE  
        GOTO digit4  
    ENDIF  
  
digit3:  
    IN 8, A  
    IF A BITAND 11100000b = 128 THEN  
        E = A BITAND 00001111b
```

```
ELSE                                     ' sonst weiter abfragen
  GOTO digit3                             ' bei Digit3 anfangen
ENDIF                                     ' Ende Abfrage Digit3

digit2:                                  ' Digit 2
  IN 8, A                                  ' Port einlesen
  IF A BITAND 11100000b = 96 THEN          ' 3 obere Bits von Wort A als Digit
    F = A BITAND 00001111b                ' BCD-Wert von Digit2 als F
  ELSE                                     ' sonst weiter abfragen
    GOTO digit2                             ' bei Digit2 anfangen
  ENDIF                                    ' Ende Abfrage Digit2

digit1:                                  ' Digit 1
  IN 8, A                                  ' Port einlesen
  IF A BITAND 11100000b = 64 THEN          ' 3 obere Bits von Wort A als Digit
    G = A BITAND 00001111b                ' BCD-Wert von Digit1 als G
  ELSE                                     ' sonst weiter abfragen
    GOTO digit1                             ' bei Digit1 anfangen
  ENDIF                                    ' Ende Abfrage Digit1

  IF H = 0 THEN                            ' Vorzeichenmultiplikator aus H
    J = -1                                  ' entweder zu -1
  ELSE                                     ' oder zu +1 machen
    J = 1
  ENDIF

I = J*((c*1)+(D*0.1)+(E*0.01)+(F*0.001)+(G*0.0001)) ' Dezimalzahl bilden
IF I > K THEN                              ' Abfrage, ob Wert > bisheriges Maximum
  K = I                                     ' Wenn ja, Maximum aktualisieren
ENDIF

IF L = 1 THEN                              ' Ausgabe bei Überlauf
  PRINT_USING #1, "<1>Overflow = ";I; " V"   ' Ausgabe auf LC-Display
  PRINT_USING #1, "Maximum = ";K; " V"     ' Ausgabe auf LC-Display
ELSE                                       ' Ausgabe bei normalem Meßwert
  PRINT_USING #1, "<1>Messwert = ";I; " V"   ' Ausgabe auf LC-Display
  PRINT_USING #1, "Maximum = ";K; " V"     ' Ausgabe auf LC-Display
ENDIF

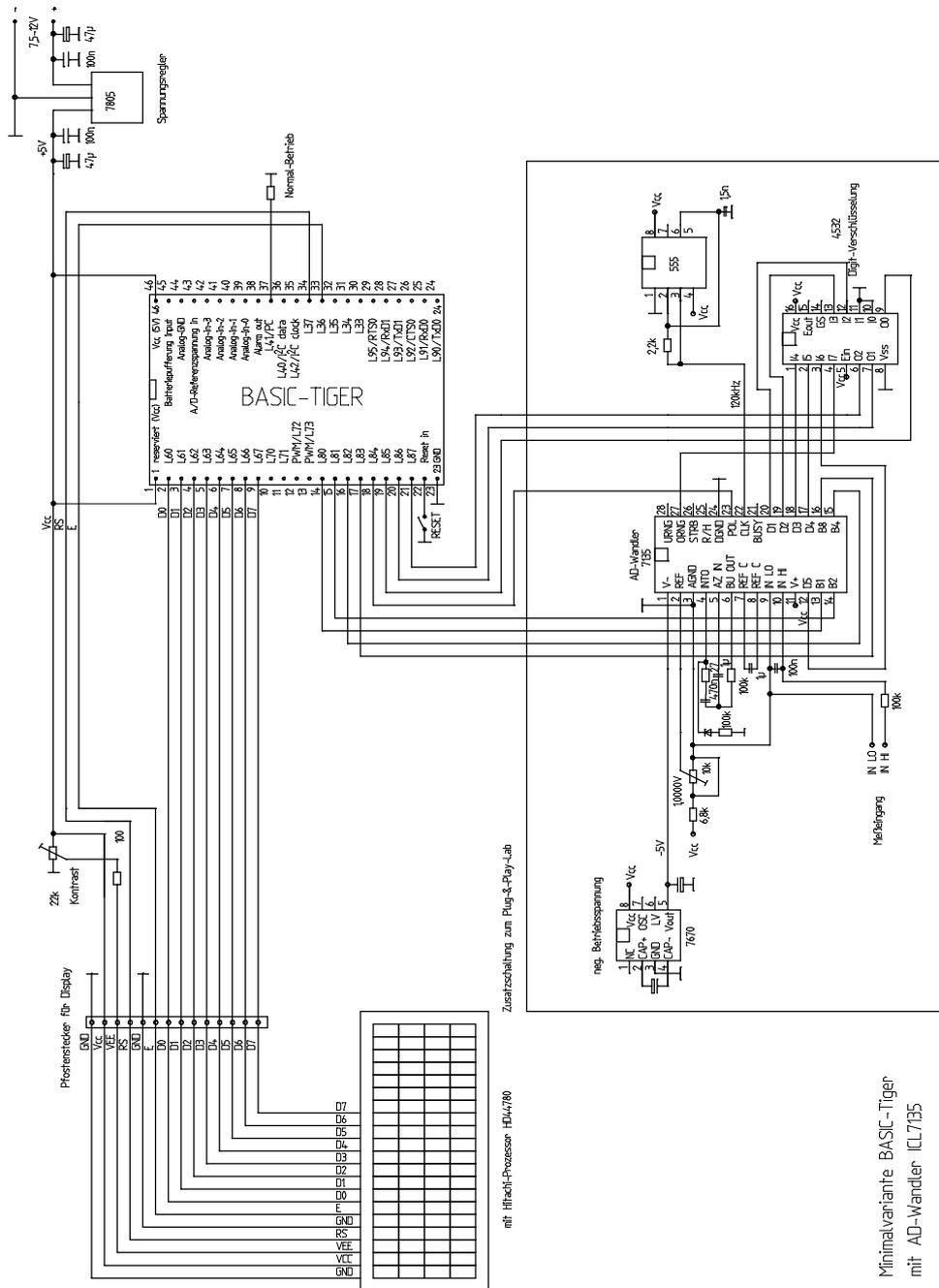
WAIT_DURATION 500                          ' 0,5 Sekunden warten
GOTO overflow                              ' Ende der kompletten Schleife

END                                         ' Ende der TASK MAIN
```

# BASIC-Tiger<sup>®</sup>

## Application Note No. 002

Rev. 1.0



Zusatzschaltung zum Plug-&-Play-Kit  
mit 16-Bit-Processor HD44780

Minimalvariante BASIC-Tiger  
mit AD-Wandler ICL7545