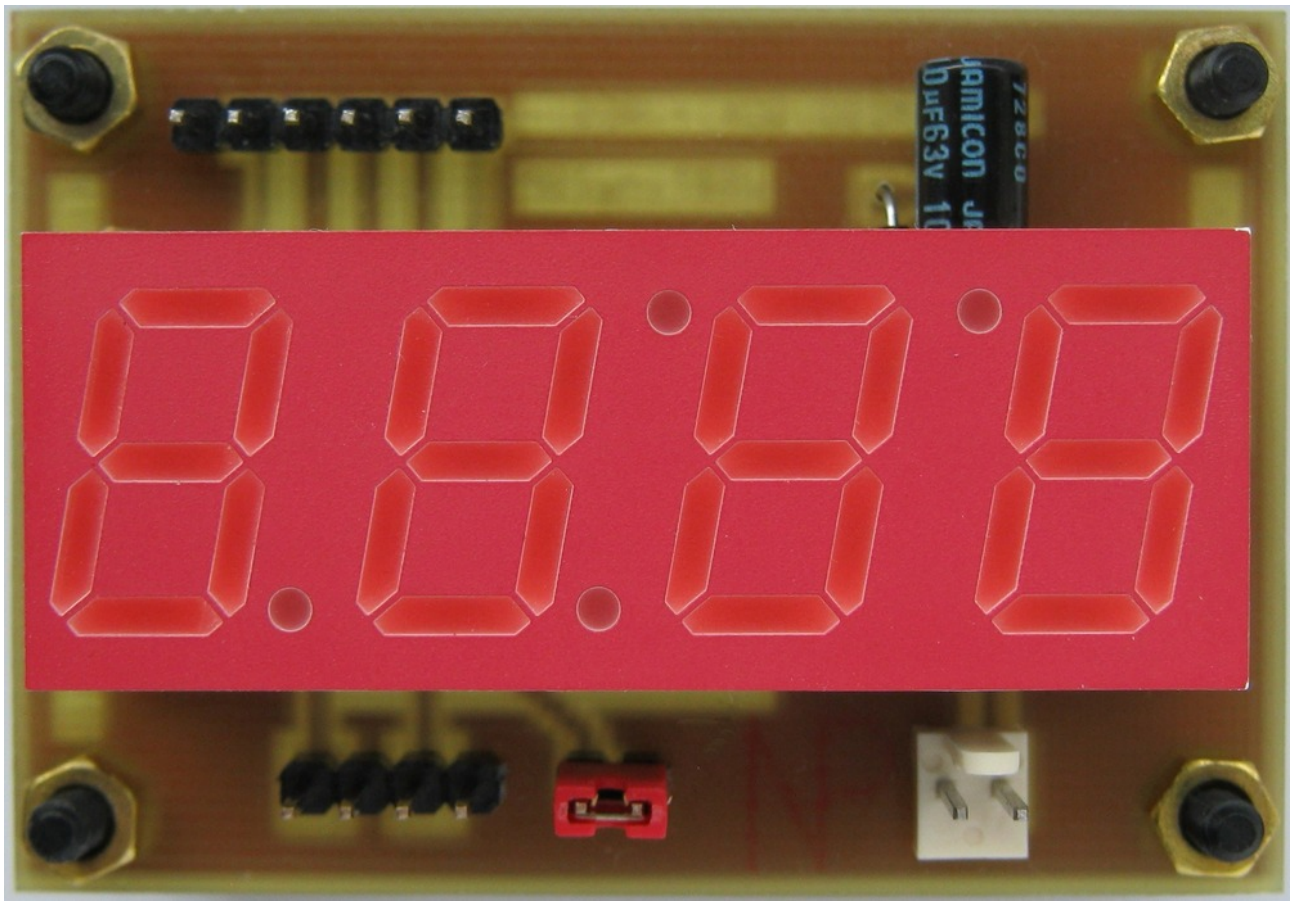
7 Segment LED-Anzeige

© Uwe, de0508

Technische Beschreibung und Baumappe
Version 1.02
Datum 03.09.2014
Edit 16.12.2014



Inhaltsverzeichnis

Funktionsbeschreibung

Die 7 Segment LED-Anzeige basiert auf einem atTiny4313, dieser übernimmt die Anzeige der Daten im Multiplexverfahren[1] mit 1000Hz. D.h. jede Stelle leuchtet 4ms (250Hz) lang und dann erfolgt ein Wechsel auf die nächste Stelle. Über den eingebauten Hardware-Uart, mit 9600 Baud 8/N/1, gelangen die Steuerbefehle zum µC, der diese sofort ausführt. Das sind z.B. einzelne Zeichen darstellen oder auch Befehle wie „!“ Anzeige löschen, „$“ Cursor nach links oder „?“ die Ausgabe einer Hilfeseite 4.

Die Darstellung der Zeichen erfolgt von links nach rechts, so dass der Cursor nach dem letzten und 4ten Zeichen wieder an die erste Stelle springt.

Die eingesetzte 4-fach LED-Anzeige ist eine sogenannte „super bright“ Ausführung und benötigt deshalb nur sehr wenig Strom. Die bemessenen Vorwiderstände von 390Ω bis 270Ω reichen für eine Anzeige bei Sonnenlicht und in Räumen aus, deshalb liefert der atTiny4313 den erforderlichen gesamt Anodenstrom problemlos.

Im Normalbetrieb ist der Jumper **JP1** gesetzt und es erfolgt keine Ausgabe des Copyright Textes auf der LED-Anzeige. Für erste Tests lässt man **JP1** offen und hat so einen schnellen Funktionstest. **JP1** hat noch eine weitere Bedeutung beim Abgleich des RC Oszillators des atTiny4313, über ihn wird 1/800 der Oszillatorfrequenz, also ca. 10kHz, ausgegeben. Mehr dazu später.

Die Kommunikation mit dem AVR µC atTiny4313 erfolgt über eine serielle Verbindung mit 9600 Baud 8/N/1. Dazu verbindet man einen PC über einen **USB TTL-Wandler** mit dem atTiny4313.

USB TTL-Schnittstellenwandler

Ich verwende gerne USB TTL-Schnittstellenwandler auf Basis eines FTDI FT232RL[2] oder Microchip MCP2200[3]. Stabile Treiber habe ich für Linux und Win7 Pro im Einsatz. Für Windows gibt es diese bei den Herstellern.

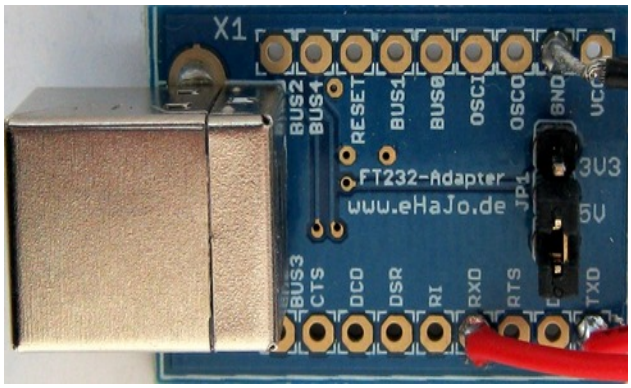


Abbildung 1: FT232RL USB-TTL-Wandler

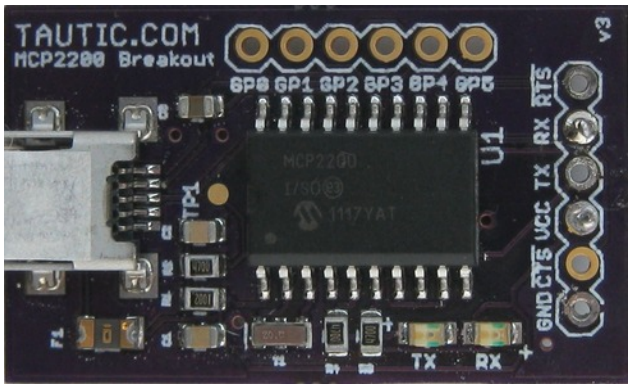


Abbildung 2: MCP2200 USB-TTL-Wandler

1 http://de.wikipedia.org/wiki/Multiplexverfahren
2 http://www.ftdichip.com/Products/ICs/FT232R.htm
3 http://www.microchip.com/wwwproducts/devices.aspx?dDocName=en546923

Die Konfiguration

Den Befehlssatz des integrierten Kommandointerpreters, erhält man durch die Eingabe eines „**?**“ am PC Terminal. Der USB TTL-Schnittstellenwandler, bzw. das PC Terminal, muss vorher auf 9600 Baud 8/N/1 eingestellt werden.

Anmerkung: das An- und Ausführungszeichen „ “ dient nur zur Begrenzung eines Zeichens oder einer Zeichenkette und wird natürlich nicht mit über das PC Terminal eingegeben.

Der Befehlssatz des Kommandointerpreters wird dann sogleich ausgeben:

```
-- LED Driver
(c) by -----, Uwe
Version 1.0
Release 29.07.2014

-- command
?                         help
!                         home display
$                         cursor to pos 0
[.|,]                     set LED DP
<{number};{string};       scroll {string} with {number} ms delay

-- character
[0,..,9]                  digit '0' to '9'
[a,..,z]                  char 'a' to 'z'
[A,..,Z]                  char 'A' to 'Z'
[ ,+,-,(,),|,_]           other character

-- system config
@E[0,1]*;                 dis-/ enable echo
@F{number};               10kHz frequency generator for {number} ms
@S{number};               set OSCCAL to {number}
@P;                       print OSCCAL
@W;                       write OSCCAL to EEprom
```

Die Kommandogruppe

```
-- command
?                         help
!                         home display
$                         cursor to pos 0
[.|,]                     set LED DP
<{number};{string};       scroll {string} with {number} ms delay
```

Das sind einfache Steuerbefehle:

Eine Hilfeseite wird durch Eingabe eines „**?**“ ausgegeben.

Durch „**!**“ wird der Cursor auf das 1 Zeichen gesetzt und die LED-Anzeige gelöscht.

Das „**$**“ setzt nur den Cursor auf das 1 Zeichen.

Diese LED-Anzeige hat ja 4 Dezimalpunkte, die man durch Eingabe eines „**.**“ oder „**,**“ setzen kann. Dabei ist zu anzumerken, dass die ersten beiden Dezimalpunkte an der „richtigen“ Stelle sitzen, und der 3 und 4 auf der Anzeige oben angeordnet sind.

Ein Beispiel:

Senden wir den Text „$1.234“, so sehen wir in der LED-Anzeige:

```
1.234
```

Der Text „$-2,56“ führt zur Darstellung von:

```
-2.56
```

Etwas komplexer ist die Befehlssequenz „**<{number};{string};**“. Sie gibt einen Text „**{string}**“ mit einer Anzeigenzeit von „**{number}**“ ms aus. Die „**;**“ dienen dabei als Trennzeichen zwischen den Parametern.

Zur Funktion: jedes Zeichen wird von rechts nach links „durch“ die LED-Anzeige geschoben – somit entsteht der Eindruck eines Lauftextes.

Die ASCII-Zeichen

Neben den Ziffern „0“ bis „9“ wurden auch alle Buchstaben „A“ bis „Z“ implementiert. Leider sind diese nicht immer einfach zu identifizieren, da z.B. ein „M“ drei Spalten benötigt, wir aber nur zwei nutzen können.

```
-- character
[0,..,9]                digit '0' to '9'
[a,..,z]                char 'a' to 'z'
[A,..,Z]                char 'A' to 'Z'
[ ,+,-,(,),|,_]         other character
```

Als weitere Zeichen können angezeigt werden

- „ “ das Leerzeichen,
- „**+**“ ein Plus Zeichen,
- „**-**“ ein Minus Zeichen,
- „**(**“ ein linke Klammer,
- „**)**“ ein rechte Klammer,
- „**|**“ das Pipe Zeichen und
- „**_**“ ein Unterstrich.

Die Befehlsgruppe „@“

```
-- system config
@E[0,1]*;               dis-/ enable echo
@F{number};             10kHz frequency generator for {number} ms
@S{number};             set OSCCAL to {number}
@P;                     print OSCCAL
@W;                     write OSCCAL to EEprom
```

Über die Eingabe von „**@E1;**“ lässt sich das zurücksenden eines eingegeben Zeichens ein-, bzw. über „**@E;**“ oder „**@E0;**“ wieder ausschalten.

Einem CR(13) wird dabei automatisch auch ein LF(10) nachgesendet, so dass wir auf dem PC Terminal einen Zeilenumbruch CR-LF erhalten.

Der Abgleich des internen 8MHz R/C Oszillator auf genau 7,9872MHz ist für die genaue Baudrate von 9600 notwendig.

Wer etwas Mathematik mag, sieht das in dieser Formel:

```
CPU-Frequenz = Baudrate * Timer-Frequenzteiler * Timer-Teiler
```

Mit den Werten Baudrate= 9600, Timer-Frequenzteiler= 16 und Timer-Teiler= 52 ergibt sich:

```
CPU-Frequenz = 9600 * 16 * 52 = 7.987.200Hz
```

„**@P;**“ gibt den OSCCAL-Wert des internen RC Oszillator aus, er liegt im Bereich 1 bis 255.

Diesen Wert kann man lokal auch durch den folgenden Befehl ändern.

„**@S{number};**“ die Anpassungen des OSCCAL-Werts sollten immer nur in kleinen Schrittweiten von +/- 1 erfolgen, um das System auch weiter per Uart ansprechen zu können.

Aber wohin ändert man nun den Wert?

Das ermitteln wir durch eine Frequenzmessung mit einem Oszilloskop oder einem Frequenzzähler. Wir schließen den Zähler an den Anschluss mit der Beschriftung **JP1** an und verwenden ihn nun als Ausgang des 10kHz Frequenzgenerators.

„**@F{number};**“ erzeugt an **JP1** eine TTL-Rechtecksignal mit ca. 10kHz für die Zeitdauer von „{number}“ ms.

Ein Beispiel erläutert die Abfolge der Befehle bestimmt anschaulicher:

- eine Eingabe von „**@F10000;**“ erzeugt ein Rechtecksignal mit ca. 10kHz für 10.000ms = 10Sek.
  Unsere **Soll-Frequenz** liegt bei **9,984kHz** = 7,9872MHz /800.
- In dieser Zeit ermitteln wir die Frequenz und sehen in welche Richtung wir den OSCCAL-Wert verändern müssen.
- Den aktuellen OSCCAL-Wert lesen wir mit „**@P;**“ aus, in unserem Bsp.; **85**.
- Liegt die Frequenz zu hoch, so verringern wir den OSCCAL-Wert um 1 auf **84** und setzen ihn durch den Befehl „**@S84;**“ neu.
- Liegt sie zu niedrig, erhöhen wir den OSCCAL-Wert um 1 auf **86** und setzen ihn durch „**@S86;**“ neu.
- Iterativ bestimmen wir das weitere vorgehen, indem wir eine erneute Frequenzmessung durch Eingabe von „**@F10000;**“ einleiten.
- Sind wir mit dem Ergebnis zufrieden, d.h. die Frequenz liegt nahe an 9,984kHz, so speichern wir den aktuellen OSCCAL-Wert in das Eeprom des atTiny4313 permanent ab. Dazu geben wir am Terminal den Befehl „**@W;**“ ein. Nach dem Einschalten wird er so automatisch verwendet.

Man kann natürlich auch die gesamte Baugruppe stromlos schalten, um einen Reset durchzuführen. Dadurch werden keine Daten gespeichert.

Ansichten der Platine