

Version vom: 22.7.2012

## Transistortester Hardware

Passend zu den folgenden Projekten:

Projektbeschreibung von Markus Frejek

<http://www.mikrocontroller.net/articles/AVR-Transistortester>

und die Weiterführung von Karl-Heinz Kübbeler

[http://www.mikrocontroller.net/articles/AVR\\_Transistortester](http://www.mikrocontroller.net/articles/AVR_Transistortester)

und Beiträge dazu im Forum:

Markus Frejek: Transistortester mit AVR

<http://www.mikrocontroller.net/topic/transistortester-mit-avr#1244222>

Bernhard S.: einfacher Transistortester (Assembler) ATmega8

<http://www.mikrocontroller.net/topic/31386#new>

Karl-Heinz Kübbeler: Transistortester AVR

<http://www.mikrocontroller.net/topic/248078#2540547>

werde ich hier meine Aufbau-Variante beschreiben. Bei der Gelegenheit auch allen Mitarbeitern bei der Programmerstellung DANKE! für das tolle Messgerät!

### **Anstoß für diesen Nachbau:**

Meine erste Aufbau-Variante entstand schon 2009 nach der Beschreibung von Markus Frejek mit dem ATmega 8 und der damaligen Software auf Lochrasterplatte und hat immer brav ihren Dienst erfüllt.

Beim der Internet-Suche nach möglichen Software-Updates stelle ich vor einiger Zeit mit Freude fest, dass durch Karl-Heinz Kübbeler das Projekt mit einem ATmega168... zu neuen Ufern unterwegs war.

Deshalb sollte ein neuer Nachbau vom Aufbau „edler“ werden und natürlich mit dem größeren ATmega 168 noch Platz für mehr Funktionen bieten.

Dazu entstand dazu bei mir eine einseitige Leiterplatte 60x80mm in Toner-Transfer-Technik für ein preiswertes Gehäuse. Diese stelle ich mit den erforderlichen Unterlagen für eventuelle Nachbauer gerne zur Verfügung. Wenn Fragen auftauchen, versuche ich gerne zu helfen.

### **Selbstbau im Schnelldurchlauf:**

**Das LCD-Modul** hat im Originalzustand einen 16-pol. Folienanschluß im 1mm-Raster. Mangels passendem Stecker habe ich die Folie abgelötet (zuerst Klebung lösen, dann Lötstelle der Folie mit dem LötKolben erwärmen und ohne Kraft abrollen. Dazu Display einspannen. Vor dem Anlöten der verzinnten Litzen (D=0,7mm, 60mm lang) die Kontaktstellen mit Flussmittel benetzen.). Am andern Ende eine passende 14-pol. Buchse anlöten und die Kontakte mit Schrumpfschlauch isolieren.

**Die Leiterplatte** habe ich in Toner-Transfer-Technik hergestellt. Die ersten Versuche in dieser Technik ergaben, als ich damit anfang, stark wechselnde Qualität: Manchmal waren die Leiterbahnen unterätzt und zu schmal, bei größeren Flächen reichte der Toner Auftrag nicht aus und die Flächen und breiteren Leiterbahnen waren durchlöchert. Die Verwendung von Tonerverdichter LF-A ergab reproduzierbare einwandfreie Leiterbahnen nach dem Ätzen. Dabei den Spray vor dem Ätzen (Papier ist komplett entfernt!) 1x dick auf die waagrecht liegende Leiterplatte sprühen und eintrocknen lassen. Dabei passiert nach meiner Beobachtung folgendes: Das Lösungsmittel löst die groben Toner-Teilchen auf und verteilt sie in den Leiterbahnen gleichmäßig. Beim Abtrocknen bildet sich dann ein glatter, gleichmäßiger Film mit guter Haftung auf der Kupferoberfläche. Wenn der Toner auf den größten Flächen nicht gereicht hat, kann man diese nachträglich mit einem wasserfeste Stift großflächig nachschwärzen (versucht man eine „unverdichtete“ Fläche nachzuschwärzen, merkt man den Unterschied).

**Gehäuse bearbeiten 1:** Nach dem Ätzen und Bohren der 4 Befestigungslöcher in der Leiterplatte wird diese mittig in das Unterteil des Gehäuses gehalten. Dabei einen Abstand von genau 37mm für die 9V-Batterie zum Rand einhalten und die Löcher mit Filzstift in dem Gehäuseunterteil anzeichnen. Die Nasen auf der Batterieseite und beim RS232 Stecker am besten mittels Stechbeitel entfernen. Gegenüber (falls gewünscht) eine Öffnung für eine 9-pol. Sub-D Buchse ausschneiden und festschrauben. Hier nur die beiden inneren Nasen entfernen. Oberkante der Buchse eben mit Gehäuse ausrichten. Auf dem passenden 9-pol- Stecker lassen sich dann weitere Messadapter anbringen.

**Bestückung der Leiterplatte:** An den vier Bohrungen (3mm) Distanzhülsen je zwei Distanzhülsen einschrauben. Dann lassen sich die Bauteile gut bestücken und es wackelt nicht beim Lötten. Zuerst die Drahtbrücken auf der Bestückungsseite einlöten (D=0,5 bis 0,6mm), dann die Widerstände und schließlich die höheren Teile.

**Gehäuse bearbeiten 2:** Die Platine wird mit 3mm Senkkopfschrauben von unten befestigt. Löcher nach dem Bohren so tief ansenken, dass der Schraubenkopf eben mit der Oberfläche abschließt. Nun 5mm Distanzhülsen einbauen. Nicht vergessen die Vertiefung für den seitlichen Taster mit einer runden Schlüsselfeile anzubringen.

**LCD-Display:** Das Display mit 4x 2,5mm Schrauben und 12x Muttern auf der Montageplatte befestigen. Vorher die 3mm Schrauben einsetzen. Die Litzen zwischen Display und Montageplatte verlegen. Die 3mm Schrauben so justieren, dass der schwarze Blech-Display-Rahmen genau 20mm über der Leiterplatte liegt. Jetzt Leiterplatte und Display festschrauben. Mit der Schieblehre die Maße auf das Gehäuseoberteil übertragen und die Durchbrüche grob ausbohren. Dann mit einer Feile passend machen. Zwischendurch immer wieder probieren (kleiner feilen geht nicht!)

**Inbetriebnahme:** Ohne ATmega und Display. Die Spannung anlegen und nachmessen. Schalter und Programmierbrücke testen. Display einstecken und Kontrast einstellen. Prozessor richtig herum einstecken (vorher Beine parallel ausrichten).

**Programmieren:** Die Software stammt von Karl-Heinz Kübbeler aus GNU tarball vom 16.7.2012 und zwar die Variante „trunk“. Hier tummeln sich verschiedene neue Programmteile aus Juni und Juli 2012 in Ergänzung zu „ttester\_097k“.

Mit **AVR Studio 4.18** habe ich zuerst ein neues C-Projekt mit dem Namen „Transistortester“ und main.c angelegt. Beim Kopieren aus dem Ordner „trunk“ nach „Transistortester“ wurde die leere Datei main.c überschrieben. Nach dem Öffnen des Projekts mit AVR Studio 4.18 zuerst

In Menue: Project/configuration options den Pfad auf das vorhandene externe makefile (default\Makefile) umstellen. Mit Build sollte jetzt im Ordner \default eine fehlerfreie \*.hex und eine \*.eep zum Test erzeugt werden.

### **Dann Makefile anpassen:**

die deutsche Sprache wählen :

```
#select your language
# available languages are: LANG_ENGLISH, LANG_GERMAN, LANG_POLISH,
LANG_CZECH, LANG_SLOVAK, LANG_SLOVENE
UI_LANGUAGE = LANG_GERMAN
```

Den Software UART freischalten:

```
# the WITH_UART option enables the software UART
CFLAGS += -DWITH_UART
```

....

Vor dem Programmieren über den Button „con“ den Programmer verbinden (hier ALL-AVR, der kompatibel zu AVRISP mkII ist).

Mit dem Programmer-Tool die Fuses setzen (siehe Info in Makefile):

```
ifeq ($(PARTNO),m168)
MCU = atmega168
# Operation with 8MHz crystal and /1 divider , low power
FUSES_CRY_L = -U lfuse:w:0xff:m -U hfuse:w:0xdc:m
# -U efuse:w:0xf9:m
```

Jetzt die Daten TransistorTester.hex und dann TransistorTester.eep in den ATmega programmieren.

Die serielle Schnittstelle gibt die Messwert mit 19200Baud (2400x8) aus. Anzeige z.B. mit Hyper-Terminal ging auf Anhieb!

Falls ich nicht vergessen habe, etwas wichtiges zu erwähnen, sollte jetzt das Messgerät funktionieren.