

TransistorTester mit AVR-Mikrocontroller
und ein wenig mehr
Version 1.02k

Karl-Heinz Kübbeler
kh_kuebbeler@web.de

22. November 2012

Inhaltsverzeichnis

1	Eigenschaften	3
2	Hardware	6
2.1	Die Schaltung des TransistorTesters	6
2.2	Hinweise für den Aufbau des TransistorTesters	7
2.3	Programmierung des Mikrocontrollers	8
2.4	Fehlersuche	8
3	Bedienungshinweise	10
4	Konfigurieren des TransistorTesters	12
5	Beschreibung des Messverfahrens	17
5.1	Messung von Halbleitern	19
5.1.1	Messung eines PNP-Transistors oder eines P-Kanal MOSFETs	20
5.1.2	Messung eines NPN-Transistors oder eines N-Kanal-MOSFET	21
5.1.3	Messung von Dioden	23
5.1.4	Ergebnisse der verschiedenen Messungen	23
5.2	Widerstands-Messung	26
5.2.1	Widerstandsmessung mit den 680-Ohm-Widerständen	26
5.2.2	Widerstandsmessung mit den 470-kOhm-Widerständen	28
5.2.3	Ergebnisse der Widerstands-Messung	29
5.3	Messen von Kondensatoren	33
5.3.1	Entladen der Kondensatoren	33
5.3.2	Messung von großen Kapazitäten	33
5.3.3	Messen von kleinen Kapazitäten	35
5.3.4	Messen des äquivalenten Serienwiderstandes ESR	37
5.3.5	Ergebnisse der Kondensator-Messung	39
5.3.6	Automatischer Abgleich der Kondensator-Messung	44
5.4	Messen von Induktivitäten	48
5.4.1	Ergebnisse der Induktivitäts-Messungen	49
5.5	Selbsttest-Funktion	50
6	Bekannte Fehler und ungelöste Probleme	57
7	Spezielle Softwareteile	58
8	Arbeitsliste und neue Ideen	59

Vorwort

Grundsätzliches

Jeder Bastler kennt das folgende Problem: Man baut einen Transistor aus oder man nimmt einen aus einer Bastelkiste. Wenn man die Typenbezeichnung erkennen kann und man bereits ein Datenblatt hat oder eins bekommen kann, ist alles in Ordnung. Aber wenn man keine Datenblätter findet, hat man keine Idee, was das für ein Bauteil sein kann. Mit den konventionellen Meßmethoden ist es schwierig und zeitaufwändig den Typ des Bauteils und dessen Parameter herauszufinden. Es könnte ein NPN, PNP, N- oder P-Kanal-MOSFET usw. sein. Es war die Idee von Markus F., diese Arbeit von einem AVR-Mikrocontroller erledigen zu lassen.



Wie meine Arbeit begann



Meine Beschäftigung mit der Software des TransistorTesters von Markus F. [1] hat begonnen, als ich Probleme mit meinem Programmer hatte. Ich hatte eine Platine und Komponenten gekauft, aber ich war mit dem Windows-Treiber nicht in der Lage den EEprom-Speicher des ATmega8 ohne Fehlermeldungen zu beschreiben. Deshalb habe ich die Software von Markus F. genommen und habe alle Zugriffe auf den EEprom-Speicher durch Zugriffe auf den Programm-Speicher (Flash) ersetzt. Bei der Durchsicht der Software, um an anderer Stelle Programmspeicher (Flash) einzusparen, hatte ich die Idee, das Ergebnis der ReadADC Funktion von ADC-Einheiten in eine Millivolt (mV) Auflösung zu ändern. Die mV Auflösung wird für die Ausgabe von Spannungswerten gebraucht. Wenn die ReadADC-Funktion direkt die mV Auflösung liefert, kann man die Umwandlung für jeden Ausgabewert einsparen. Diese mV-Auflösung kann man erhalten, wenn man zuerst die Ergebnisse von 22 ADC-Einlesungen addiert. Die Summe muß mit zwei multipliziert und durch neun geteilt werden. Das ergibt einen Maximalwert von $\frac{1023 \cdot 22 \cdot 2}{9} = 5001$, welcher hervorragend zu der gewünschten mV Auflösung passt. So hatte ich zusätzlich die Hoffnung, dass die Erhöhung der ADC-Auflösung durch Überabtastung helfen könnte, die Spannungs-Einlesung zu verbessern, wie es in dem Atmel Report AVR121 [5] beschrieben ist. Die Original-Version von ReadADC hat die Ergebnisse von 20 ADC-Einlesungen addiert und danach durch 20 dividiert, so dass das Ergebnis wieder die Original-Auflösung des ADC hat. Deshalb konnte niemals eine Erhöhung der ADC-Auflösung durch Überabtastung stattfinden. So hatte ich wenig Arbeit, die ReadADC Funktion zu ändern, aber dies erforderte die Analyse des kompletten Programms und Änderung aller „if“-Abfragen im Programm, wo Spannungswerte geprüft wurden. Aber dies war nur der Beginn meiner Arbeit!

Mehr und mehr Ideen wurden eingebaut, um die Messung schneller und genauer zu machen. Zusätzlich wurde der Bereich der Widerstands- und Kondensator-Messung erweitert. Das Ausgabe-Format für das LC-Display wurde verändert, so wurden Symbole für die Darstellung von Dioden, Widerständen und Kondensatoren verwendet. Für weitere Einzelheiten schauen Sie in das aktuelle Eigenschaften-Kapitel 1. Geplante Arbeiten und neue Ideen wurden im Arbeitslisten-Kapitel 8 gesammelt. Inzwischen kann ich unter dem Linux-Betriebssystem auch den EEprom-Speicher des ATmega8 einwandfrei beschreiben.

Kapitel 1

Eigenschaften

1. Arbeitet mit Mikrocontrollern vom Typ ATmega8, ATmega88, ATmega168 oder ATmega328.
2. Anzeige der Messergebnisse auf einem 2x16 Zeichen großen LC-Display.
3. Ein-Tasten-Bedienung mit automatischer Abschaltfunktion.
4. Batterie-Betrieb ist möglich, weil der Strom im abgeschalteten Zustand nur etwa 20nA beträgt.
5. Preisgünstige Lösung ist möglich ohne Quarz und ohne automatische Abschaltung.
6. Automatische Erkennung von NPN und PNP bipolaren Transistoren, N- und P-Channel MOSFETs, JFETs, Dioden, Doppeldioden, Thyristors und Triacs.
7. Darstellung der Pin-Belegung der erkannten Bauteile.
8. Messung des Stromverstärkungsfaktors und der Basis-Emitter-Schwellspannung für bipolare Transistoren.
9. Darlington-Transistoren können durch die höhere Schwellspannung und durch den hohen Stromverstärkungsfaktor erkannt werden.
10. Automatische Erkennung einer Schutzdiode bei bipolaren Transistoren und bei MOSFETs.
11. Messung der Schwellwert-Spannungen und der Gate-Kapazitätswerte von MOSFETs.
12. Bis zu zwei Widerstände werden gemessen und mit Symbolen  und den Widerstandswerten mit bis zu vier Dezimalstellen in der richtigen Dimension angezeigt. Alle Symbole werden eingerahmt mit den gefundenen Testpin Nummern des Testers (1-3). Deshalb können auch Potentiometer gemessen werden. Wenn der Schleifer eines Potentiometers auf eine Endposition gestellt ist, kann der Tester nicht mehr zwischen mittlerem Anschluss und Endanschluss unterscheiden.
13. Die Auflösung der Widerstandsmessung ist jetzt 0,1Ω, Werte von bis zu 50MΩ werden erkannt.
14. Ein Kondensator kann erkannt und gemessen werden. Der wird mit dem Symbol  und Kapazitätswert mit bis zu vier Dezimalstellen in der richtigen Dimension angezeigt. Der Wert kann zwischen 25pF (bei 8MHz Takt, 50pF bei 1MHz Takt) bis 100mF mit einer Auflösung von bis zu 1pF (bei 8MHz Takt).
15. Bei Kondensatoren mit einer Kapazität über 2μF wird zusätzlich der äquivalente Serienwiderstand (ESR) des Kondensators mit einer Auflösung von 0.01Ω gemessen und mit zwei Dezimalstellen angezeigt. Diese Fähigkeit steht nur zur Verfügung, wenn der ATmega mindestens 16K Flashspeicher besitzt.

16. Bis zu zwei Dioden werden mit dem Symbol  oder dem Symbol  in der richtigen Reihenfolge angezeigt. Zusätzlich werden die Schwellspannungen angezeigt.
17. Eine LED wird als Diode erkannt, die Schwellspannung ist viel höher als bei einer normalen Diode. Doppeldioden werden als zwei Dioden erkannt.
18. Zener-Dioden können erkannt werden, wenn die Zener-Spannung unter 4,5V ist. Sie werden als zwei Dioden angezeigt, man kann das Bauelement nur mit den Spannungen erkennen. Die äußeren Testpin-Nummern, welche die Dioden Symbole umgeben, sind in diesem Fall identisch. Man kann die wirkliche Anode der Diode nur durch diejenige Diode herausfinden, deren Schwellwert-Spannung nahe bei 700mV liegt!
19. Wenn mehr als 3 Dioden erkannt werden, wird die gefundene Anzahl der Dioden zusammen mit der Fehlermeldung angezeigt. Das kann nur passieren, wenn Dioden an alle drei Test-Pins angeschlossen sind und wenigstens eine Zener-Diode ist. In diesem Fall sollte man nur zwei Test-Pins anschließen und die Messung erneut starten, eine nach der anderen.
20. Der Kapazitätswert einer einzelnen Diode in Sperr-Richtung wird automatisch ermittelt. Bipolare Transistoren können auch untersucht werden, wenn nur die Basis und entweder Kollektor oder Emitter angeschlossen wird.
21. Die Anschlüsse einer Gleichrichter-Brücke können mit nur einer Messung herausgefunden werden.
22. Kondensatoren mit Kapazitätswerten von unter 25pF werden normalerweise nicht erkannt, aber sie können zusammen mit einer parallel geschalteten Diode oder mit einem parallel geschaltetem Kondensator mit wenigstens 25pF gemessen werden. In diesem Fall muß der Kapazitätswert des parallel geschalteten Bauteils vom Meßergebnis abgezogen werden.
23. Bei Widerständen unter 2100 Ω wird auch eine Induktivitätsmessung durchgeführt, wenn der ATmega mehr als 8K Flashspeicher besitzt. Der Anzeigebereich ist etwa 0.01 mH bis über 20 H , die Genauigkeit ist allerdings nicht hoch. Das Ergebnis wird nur bei einem Einzelwiderstand zusammen mit dem Widerstandswert angezeigt.
24. Die Messzeit beträgt ungefähr zwei Sekunden, nur Kapazitätsmessungen und Induktivitätsmessungen können länger dauern.
25. Die Software kann für Messserien mit vorgebbarer Wiederhol-Zahl konfiguriert werden, bevor die automatische Abschaltung ausschaltet.
26. Eingebaute Selbsttest-Funktion inklusive einem optionalen 50Hz Frequenz-Generator um die Genauigkeit der Taktfrequenz und der Verzögerungszeiten zu überprüfen.
27. Wählbare Möglichkeit für ATmega168 oder ATmega328, den Nullabgleich für die Kondensatormessung und die Innenwiderstände für die Portausgänge beim Selbsttest automatisch zu bestimmen. Ein externer Kondensator mit einer Kapazität zwischen 100 nF und 20 μF an Pin 1 und Pin 3 ist notwendig, um die Offset Spannung des analogen Komparators zu kompensieren. Dies kann den Messfehler bei Kapazitätsmessungen bis zu 40 μF reduzieren. Mit dem gleichen Kondensator wird eine Korrekturspannung zum Einstellen der richtigen Verstärkung für die ADC Messung mit der internen Referenzspannung berechnet.

Thyristoren und Triacs können nur erkannt werden, wenn der Test-Strom über dem Halte-Strom liegt. Einige Thyristoren und Triacs brauchen auch einen höheren Zündstrom als dieser Tester liefern kann. Der verfügbare Teststrom ist nur ungefähr 6mA! Es ist möglich, dass nicht alle Optionen in

zukünftigen Versionen erhalten bleiben, weil diese Software immer noch im Teststadium ist. Alle Möglichkeiten sind nur bei Mikrocontrollern mit genug Programmspeicher (wie dem ATmega168) konfigurierbar.

Achtung: Stellen Sie immer sicher, dass Kondensatoren vor dem Anschluss an den Tester entladen sind! Der Tester könnte sonst beschädigt werden bevor er eingeschaltet ist. Es gibt nur wenig Schutzfunktion der ATmega Anschlüsse.

Kapitel 2

Hardware

2.1 Die Schaltung des TransistorTesters

Die Schaltung des TransistorTesters in Abbildung 2.1 basiert auf der Schaltung von Markus F., die er in Abb. 1 des AVR-Transistortester Reports [1] veröffentlicht hat. Geänderte oder verschobene Bauteile sind mit **grüner Farbe** markiert, optionale Teile sind mit **roter Farbe** gekennzeichnet. Einige Änderungen wurden gemacht, weil die Strom-Abschaltung in einigen Nachbauten Probleme bereitet hatte. Deshalb ist der Widerstand R7 auf $3.3k\Omega$ reduziert. Der Kondensator C2 ist auf 10nF verkleinert und der Widerstand R8 ist verschoben, sodass der Ausgang PD6 nicht versucht, den C2 Kondensator direkt zu laden. Zusätzliche Abblock-Kondensatoren wurden hinzugefügt und sollten nahe den Versorgungs-Anschlüssen des ATmega und nahe bei dem Spannungsregler platziert werden. Weil der PD7-Eingang und der PC6-Anschluss (RESET) die einzigen Anschlüsse sind, wo „pull-up“ Widerstände gebraucht werden, wurde ein zusätzlicher $27k\Omega$ Widerstand am PD7 (Pin 13) vorgesehen. Mit dieser Änderung können die internen „pull-up“ Widerstände des ATmega abgeschaltet werden. Ein Quarz mit seinen 22pF Kondensatoren wurde zusätzlich vorgesehen. Ein Quarz hat Vorteile für die Kapazitätsmessung wegen der genaueren Zeitmessung. Die neue Software kann den Bereich für den ADC umschalten. Die Umschalt-Geschwindigkeit wird durch den externen Kondensator C1 am AREF-Pin (21) des ATmega reduziert. Um die Messung nicht langsamer als notwendig machen zu müssen, sollte der Kondensator auf 1nF reduziert werden. Ein Entfernen des Kondensators ist ebenfalls möglich. Zum Anpassen der Software an die jeweilige Schaltung schauen Sie bitte in dem Konfigurations-Kapitel 4 nach. Einige unterschiedliche Kombinationen von R11 / R12 zirkulieren im Internet. Ich habe die Software an den Original-Entwurf von Markus F. [1] mit $10k\Omega$ und $3,3k\Omega$ angepasst. Die zusätzliche 2,5V Präzisions-Spannungsreferenz, die an Pin PC4 (ADC4) angeschlossen ist, wird für die Überprüfung und Kalibration der VCC Versorgungsspannung benutzt, ist aber nicht erforderlich. Ein zusätzlicher ISP-Anschluss wurde hinzugefügt, um leichter neue Software-Versionen laden zu können.

bestücken. Es wird als IC2 ein moderner „low voltage drop“ Spannungsregler wie MCP1702-5002 empfohlen, weil dieser nur $2\mu A$ Ruhestrom benötigt und auch noch 5V liefern kann, wenn die Eingangsspannung nur 5,4V beträgt. Aber dieser Regler ist leider nicht Pin-kompatibel zum bekannteren 78L05 Regler im TO-92 Gehäuse!

Nachdem alle benötigten Bauteile bestückt sind, sollte zuerst die Batterie oder das Netzteil angeschlossen werden. Das LCD sollte dabei nicht angeschlossen sein, und der Mikrocontroller sich noch nicht im Sockel befinden. Man sollte die Betriebsspannung des Mikrocontrollers und der LCD-Anzeige überprüfen während der Start-Taster gedrückt wird. Die Betriebsspannung sollte verschwinden, wenn man den Start-Taster loslässt. Wenn die Betriebsspannung die richtige Polarität und Grösse hatte, sollte man die Strom-Versorgung entfernen und den Mikrocontroller richtig herum einstecken. Sind Sie bitte vorsichtig und stellen Sie sicher, dass alle Pinne des Mikrocontrollers im Sockel stecken. Danach können Sie das LCD anschliessen. Prüfen Sie, dass die GND- und VCC-Anschlüsse des LCD richtig mit der Baugruppe verbunden sind.

Wenn Sie sicher sind, dass alles richtig angeschlossen ist, schliessen Sie die Spannungsversorgung wieder an. Wenn Sie den ATmega schon programmiert haben, können Sie den Start-Taster drücken. Durch das Drücken des Start-Tasters sollte die Hintergrundbeleuchtung der LCD-Anzeige angehen. Wenn Sie den Taster loslassen, sollte die LED auf der Platine schwach leuchten. Beachte, dass die Software für den Mikrocontroller für den richtigen Prozessor-Typ übersetzt sein muss. Ein Programm für den ATmega8 läuft nicht auf einem ATmega168!

2.3 Programmierung des Mikrocontrollers

Ich gebe die Software für den Mikrocontroller in Quelltext heraus. Die Entwicklung wurde mit dem Linux Betriebssystem (Ubuntu) gemacht und wird gesteuert mit einer Makefile. Die Makefile stellt sicher, dass die Software entsprechend der vorher in der Makefile eingestellten Optionen übersetzt wird. Schauen Sie bitte in die LiesMich.txt Datei im Verzeichnis Sourcecode/default und in das Konfigurations-Kapitel 4. Das Ergebnis der Übersetzung hat die Dateierweiterung .hex und .eep. Üblicherweise heißen die Dateien TransistorTester.hex und TransistorTester.eep. Die .hex Datei enthält die Daten für den Programmspeicher (Flash) des ATmega-Prozessors. Die .eep Datei enthält die Daten für den EEPROM-Speicher des ATmega. Beide Dateien müssen in den richtigen Speicher geladen werden.

Zusätzlich muss der ATmega mit den „fuses“ richtig konfiguriert werden. Wenn Sie meine Makefile zusammen mit dem Programm avrdude benutzen, brauchen Sie keine genaue Kenntnis über die Einzelheiten der fuses. Sie brauchen nur „make fuses“ aufrufen, wenn Sie keinen Quarz benutzen oder Sie müssen „make fuses-crystal“ aufrufen, wenn Sie einen 8MHz Quarz auf der Baugruppe installiert haben. Bei der ATmega168 Serie der Mikrocontroller können Sie alternativ auch „make fuses-crystal-lp“ aufrufen für den low power Quarz Betrieb. Benutzen Sie niemals die Quarz-Variante, wenn Sie keinen 8MHz Quarz installiert haben. Wenn Sie sich nicht sicher mit den fuses sind, lassen Sie diese erst einmal wie vom Werk gesetzt und bringen Sie den Tester in diesem Zustand zum Laufen. Es kann sein, dass das Programm zu langsam läuft, wenn Sie die für den 8MHz Betrieb erzeugten Programmdateien benutzen, aber das kann man später korrigieren! Aber falsch gesetzte fuses können die spätere ISP-Programmierung verhindern.

2.4 Fehlersuche

Bei den meisten Problemen werden Sie den Text auf dem LCD-Display vermissen. Zuerst sollten Sie prüfen, ob die LED auf der Platine schwach leuchtet, wenn Sie den Start-Taster loslassen.

Gerät schaltet nicht ein Wenn die LED nicht leuchtet, aber die VCC-Spannung den richtigen Wert hat, wenn man den Start-Taster gedrückt hält, schaltet der Mikrocontroller nicht richtig

ein. Der Mikrocontroller sollte die Spannung behalten indem der Ausgang PD6 auf 5V geschaltet wird, was üblicherweise als eine der ersten Aktionen getan wird. Wenn man die Start-Taste gedrückt hält, bleibt die Spannung ohnehin eingeschaltet. So können Sie mit gedrücktem Taster den Wert der VCC-Spannung und zusätzlich den Spannungswert am Ausgang PD6 prüfen. Wenn die VCC-Spannung den richtigen Wert (5V) hat, aber die Spannung am Ausgang PD6 unter 4V ist, startet der Mikrocontroller nicht richtig. Für diesen Fall sollten Sie prüfen, ob die Programmdatei für den Flashspeicher für den richtigen Prozessor-Typ ist und ob der Prozessor richtig konfiguriert ist (fuses). Wenn der ATmega den Ausgang PD6 auf 5V schaltet und die Betriebsspannung trotzdem nicht eingeschaltet bleibt, wenn man den Start-Taster loslässt, ist der Grund schwieriger zu finden. Zuerst kann man die LED kurzschliessen und es noch einmal versuchen. Wenn der Tester jetzt startet, ist die LED möglicherweise falsch herum eingebaut. Wenn das nicht die Ursache ist, könnte der Grund ein unzureichender Stromverstärkungsfaktor des Transistors T3 (BC557C) sein. Der Strom in die Basis von T3 ist niedriger, wenn der Mikrocontroller mit der LED einschaltet wie im „Taster gedrückt“-Zustand.

Nichts ist lesbar auf der LCD-Anzeige Prüfen Sie die Spannung am Kontrast-Pin der LCD-Anzeige (Pin 3). Stellen Sie mit dem Trimmer den Wert auf einen im Datenblatt angegebenen Wert und optimieren Sie durch Sichtkontrolle. Wenn Sie ein Hochtemperatur-Display haben, brauchen Sie eine negative Kontrast-Spannung für den Betrieb. In diesem Fall kann man den ICL 7660 Baustein zum Erzeugen der negativen Spannung aus der positiven 5V verwenden. Wenn keine Anzeige auf dem LCD erkannt wird und wenn die Hintergrundbeleuchtung an ist, sollten Sie die Spannungsversorgung trennen und alle vier Datenverbindungen sowie die beiden Steuersignale überprüfen. Wenn alle Verbindungen in Ordnung sind, sehe ich als Ursache nur noch die Möglichkeit einer falschen Zeitabfolge der Steuersignale. Die Ursache hierfür kann sein, dass der LCD-Controller langsamer ist als es die Software des ATmega erwartet. Es könnte auch sein, dass der ATmega auf der falschen Taktrate läuft. Bitte überprüfen Sie für welche Taktrate die Software übersetzt ist und ob die fuses des ATmega für diese Geschwindigkeit richtig gesetzt sind. Sie finden die eingestellte Taktrate in der betreffenden Makefile.

Einiges, aber nicht alles ist auf der LCD-Anzeige lesbar Überprüfen Sie ob die .eep Daten in den EEPROM Speicher des ATmega geladen wurden. Wenn alle Programmdatei richtig geladen wurden, sollten Sie die Taktrate ihrer Programmdatei (Makefile) und die ATmega-Einstellungen prüfen (fuses).

Messung ist zu langsam und Kapazitäten werden um Faktor 8 zu klein gemessen. Sie betreiben die Software, die für 8MHz übersetzt wurde mit einer Taktrate von 1MHz. Bitte konfigurieren Sie den ATmega mit den fuses richtig.

Die Messung ergibt seltsame Ergebnisse Überprüfe ob der ISP-Programmierstecker noch verbunden ist. Der ISP-Stecker sollte nicht während einer Messung eingesteckt bleiben. Sehr oft ist der Grund für falsche Messergebnisse, dass die Software mit der AUTOSCALE_ADC Option und der NO_REF_CAP Option übersetzt wurde, aber der Kondensator am AREF-Pin hat immer noch einen Wert von 100nF. Falsche Bestückung von Bauteilen können auch eine Ursache für Messfehler sein oder zurückgebliebene Flussmittelreste können die Messung stören. Bitte prüfen Sie nach Möglichkeit mit der Selbsttest-Funktion der TransistorTester Software. Zu den Einzelheiten schauen Sie in das Selbsttest Kapitel 5.5. Anderenfalls prüfen sie Ihre Platine visuell und prüfen Sie die Widerstandswerte mit einem Ohmmeter. Sie können die Pins des ATmega für diese Prüfung benutzen, zum Beispiel können Sie der Widerstand R1 zwischen Pin 23 und Pin 14 messen. Schauen Sie in das Schaltbild 2.1 für die Einzelheiten. Man braucht den Mikrocontroller nicht zu entfernen, nur die Stromversorgung sollte vorher getrennt werden.

Kapitel 3

Bedienungshinweise

Die Bedienung des Transistortesters ist einfach. Trotzdem sind einige Hinweise erforderlich. Meistens sind an die drei Testports über Stecker-Leitungen mit Krokodilklemmen oder anderen Klemmen angeschlossen. Es können auch Fassungen für Transistoren angeschlossen sein. In jedem Fall können Sie Bauteile mit drei Anschlüssen mit den drei Testports in beliebiger Reihenfolge verbinden. Bei zweipoligen Bauteilen können Sie die beiden Anschlüsse mit beliebigen Testports verbinden. Normalerweise spielt die Polarität keine Rolle, auch Elektrolytkondensatoren können beliebig angeschlossen werden. Die Messung der Kapazität wird aber so durchgeführt, dass der Minuspol am Testport mit der kleineren Nummer liegt. Da die Messspannung aber zwischen 0,3 V und maximal 1,3 V liegt, spielt auch hier die Polarität keine wichtige Rolle. Wenn das Bauteil angeschlossen ist, sollte es während der Messung nicht berührt werden. Legen Sie es auf einen isolierenden Untergrund ab, wenn es nicht in einem Sockel steckt. Berühren Sie auch nicht die Isolation der Messkabel, das Messergebnis kann beeinflusst werden. Dann sollte der Starttaster gedrückt werden. Nach einer Startmeldung erscheint nach circa zwei Sekunden das Messergebnis. Bei einer Kondensatormessung kann es abhängig von der Kapazität auch deutlich länger dauern.

Was dann weiter geschieht, hängt von der Softwarekonfiguration des Testers ab.

Einzelmessung Wenn der Tester für Einzelmessung konfiguriert ist, schaltet der Tester nach einer Anzeigezeit von 10 Sekunden (konfigurierbar) wieder automatisch aus, um die Batterie zu schonen. Während der Anzeigezeit kann aber auch vorzeitig eine neue Messung gestartet werden. Nach der Abschaltung kann natürlich auch wieder eine neue Messung gestartet werden, entweder mit dem gleichen Bauteil oder mit einem anderen Bauteil.

Dauermessung Einen Sonderfall stellt die Konfiguration ohne die automatische Abschaltfunktion dar. Diese Konfiguration wird normalerweise nur ohne die Transistoren für die Abschaltung benutzt. Es wird stattdessen ein externer Ein-/Aus-Schalter benötigt. Hierbei wiederholt der Tester die Messungen solange, bis ausgeschaltet wird.

Serienmessung In diesem Konfigurationsfall wird der Tester nicht nach einer Messung sondern erst nach einer konfigurierbaren Zahl von Messungen abgeschaltet. Im Standardfall wird der Tester nach fünf Messungen ohne erkanntes Bauteil abgeschaltet. Wird ein angeschlossenes Bauteil erkannt, wird erst bei der doppelten Anzahl, also zehn Messungen abgeschaltet. Eine einzige Messung mit nicht erkanntem Bauteil setzt die Zählung für erkannte Bauteile auf Null zurück. Ebenso setzt eine einzige Messung mit erkanntem Bauteil die Zählung für die nicht erkannten Bauteile auf Null zurück. Dies hat zur Folge, dass auch ohne Betätigung des Starttasters immer weiter gemessen werden kann, wenn Bauteile regelmässig gewechselt werden. Ein Bauteilwechsel führt in der Regel durch die zwischenzeitlich leeren Klemmen zu einer Messung ohne

erkanntes Bauteil.

Eine Besonderheit gibt es in diesem Betriebsmodus für die Anzeigezeit. Wenn beim Einschalten der Starttaster nur kurz gedrückt wurde, beträgt die Anzeigezeit der Messergebnisse nur drei Sekunden. Wenn der Starttaster bis zum Erscheinen der ersten Meldung festgehalten wurde, beträgt die Anzeigezeit wie bei der Einzelmessung zehn Sekunden. Ein vorzeitiger neuer Messbeginn ist aber während der Anzeigezeit durch erneutes Drücken des Starttasters möglich.

Wenn die Software mit der Selbsttestfunktion konfiguriert ist, kann der Selbsttest durch einen Kurzschluss aller drei Testports und drücken der Starttaste gestartet werden. Hier werden die im Selbsttest-Kapitel 5.5 beschriebenen Tests ausgeführt. Die viermalige Testwiederholung kann vermieden werden, wenn der Starttaster gedrückt gehalten wird. So kann man uninteressante Tests schnell beenden und sich durch Loslassen des Starttasters interessante Tests viermal wiederholen lassen. Wenn die Funktion `AUTO_CAL` in der Makefile gewählt ist, wird beim Selbsttest eine Kalibration des Innenwiderstandes der Port-Ausgänge und eine Kalibration der Nullwertes für die Kondensatormessung durchgeführt. Sie sollten während der Kalibration weder die Testports noch angeschlossene Kabel berühren. Die Ausrüstung sollte aber die gleiche sein, die später zum Messen verwendet wird. Anderenfalls wird der Nullwert der Kondensatormessung nicht richtig bestimmt. Wenn Sie während Test Nummer 10 (nach der Ausgabe der Nullwerte der Kondensatormessung) einen Kondensator mit einer beliebiger Kapazität zwischen $100nF$ und $20\mu F$ an Pin 1 und Pin 3 anschliessen, wird die Offset Spannung des analogen Komparators kompensiert, um genauere Kapazitätswerte ermitteln zu können. Die Verstärkung für ADC Messungen mit der internen Referenz-Spannung wird dann ebenfalls abgeglichen, um bessere Widerstands-Messergebnisse mit der `AUTOSCALE_ADC` Option zu erreichen.

Der Nullwert für die ESR-Messung wird als Option `ESR_ZERO` in der Makefile vorbesetzt. Dieser Nullwert, der im Normalfall zu hoch gesetzt ist, wird von der Software im EEPROM vorbesetzt und wird bei jedem Selbsttest wieder auf diesen Wert zurückgesetzt. Nach jeder ESR-Messung wird geprüft, ob das Resultat negativ ist. In diesem Fall (Ausgabe von „ESR=0?“) wird der Nullwert so weit reduziert, dass sich bei der nächsten Messung ein ESR-Wert von Null ergibt. Damit kann ein Nullabgleich mit einem Elektrolytkondensator hoher Kapazität und mit niedrigem ESR durchgeführt werden. Dieser ESR-Nullabgleich muss aber nach jedem Selbsttest einige Male wiederholt werden.

Kapitel 4

Konfigurieren des TransistorTesters

Die ganze Software des TransistorTesters ist im Quellcode verfügbar. Die Übersetzung der Module wird mit einer Makefile gesteuert. Die Entwicklung wurde auf einen Ubuntu Linux Betriebssystem mit den GNU-Werkzeugen (GNU toolchain, gcc version 4.5.3) durchgeführt. Es sollte möglich sein, ohne Schwierigkeiten andere Linux-Betriebssysteme zu benutzen. Um die übersetzten Daten in den Flash-Speicher oder den EEPROM-Speicher zu laden, wird das Programm avrdude (Version 5.11svn) von der Makefile benutzt, wenn man „make upload“ aufruft. Das Programm avrdude ist für Linux und Windows verfügbar. Der GNU C-Kompiler wird auch von der AVR-studio-Software unter Windows benutzt. Sie können die Programmdateien (.hex und .eep) auch mit anderen Programmen in den ATmega laden, aber nur meine Makefile Version stellt sicher, dass die richtigen Daten in den gewählten Prozessor gelangen. Avrdude lädt Daten nur in den ATmega, wenn die Signaturbytes des angeschlossenen ATmega gleich mit dem ausgewählten sind. Wenn Sie die Makefile ändern, wird die Software komplett neu übersetzt, wenn man „make“ oder „make upload“ aufruft. Die Software, die für einen ATmega8 übersetzt wurde, läuft nicht auf einem ATmega88. Die Software, die für einen ATmega168 übersetzt wurde, läuft nicht auf einem ATmega88, sogar wenn der ATmega88 genug Speicher hat! Sind Sie vorsichtig, wenn Sie nicht das mitgelieferte Makefile benutzen. Mit den entsprechenden Optionen ist die Software auch auf dem unveränderten Hardware-Entwurf von Markus F. lauffähig (PARTNO=m8 , keine NO_AREF_CAP und keine PULLUP_DISABLE Option). Die Taktrate kann mit den fuses auch auf 8MHz gestellt werden, dazu ist kein Quarz erforderlich!

Die folgenden Optionen der Makefile sind verfügbar, um die Software für den Tester zu konfigurieren:

PARTNO beschreibt den Ziel-Prozessor:

m8 = ATmega8
m88 or m88p = ATmega88
m168 or m168p = ATmega168
m328 or m328p = ATmega328
Beispiel: PARTNO = m168

UI_LANGUAGE gibt die Sprache für den Tester an:

LANG_ENGLISH, LANG_GERMAN, LANG_POLISH, LANG_CZECH, LANG_SLOVAK
und LANG_SLOVENE sind derzeit verfügbar.
Beispiel: UI_LANGUAGE = LANG_ENGLISH

LCD_CYRILLIC wird nur gebraucht, wenn man ein LCD-Display mit kyrillischem Zeichensatz benutzt. Die Zeichen μ und Ω sind im kyrillischen Zeichensatz nicht enthalten. Wenn Sie diese Option angeben, werden beide Zeichen von der Software in das LCD geladen.

Beispiel: CFLAGS += -DLCD_CYRILLIC

STRIP_GRID_BOARD Diese Option passt die Software an eine andere Pinbelegung von Port D für Streifenleiterplatten an. Die Einzelheiten findet man im Hardwarekapitel 2.1.

WITH_SELFTEST Wenn Sie diese Option angeben, baut die Software eine Selbsttest-Funktion ein, die gestartet wird wenn Sie alle drei Prüfspitzen verbinden und eine Messung starten.
Beispiel: CFLAGS += -DWITH_SELFTEST

AUTO_CAL Der Nullabgleich für die Kondensatormessung und der Innenwiderstand der Port-Ausgänge wird beim Selbsttest zusätzlich ins EEprom geschrieben und ist damit für die weiteren Messungen abgeglichen. Wenn nach dem Nullabgleich der Kondensatormessung ein Kondensator mit einer Kapazität zwischen $100nF$ und $20\mu F$ an Pin 1 und Pin 3 angeschlossen wird, wird auch der Offset des analogen Komparators und die Skalierung für die AUTOSCALE_ADC Umschaltung auf die interne Spannungsreferenz ermittelt und ins EEprom geschrieben.
Beispiel: CFLAGS += -DAUTO_CAL

FREQUENCY_50HZ Zum Ende des Selbsttests wird bis zu einer Minute lang ein 50 Hz Signal auf Port 2 und Port 3 erzeugt.
Beispiel: CFLAGS += -DFREQUENCY_50HZ

CAP_EMPTY_LEVEL Diese Option legt die Spannung (mV) für einen entladenen Kondensator fest. Der Wert kann höher als 3mV gesetzt werden, wenn die Entladung nicht zum Ende kommt. In diesen Fall meldet der Tester nach längerer Zeit „Cell!“.
Beispiel: CFLAGS += -DCAP_EMPTY_LEVEL=3

WITH_AUTO_REF Mit dieser Option wird die Referenzspannung gemessen, um den aktuellen Faktor für die Kapazitätsmessung von kleineren Kapazitäten (unter $40\mu F$) zu ermitteln.
Beispiel: CFLAGS += -DWITH_AUTO_REF

REF_C_KORR gibt einen Offset für die gelesene Referenz-Spannung in mV-Einheiten an. Das kann benutzt werden, um die Kapazitätsmessung kleiner Kondensatoren abzugleichen. Wenn zusätzlich die AUTO_CAL Option gewählt wurde, ist diese Angabe nur ein zusätzlicher Offset für den gefundenen Komparator-Offset. Ein Wert von 10 ergibt etwa 1 Prozent kleinere Messergebnisse.
Beispiel: CFLAGS += -DREF_C_KORR=14

C_H_KORR gibt eine Korrektur der Messergebnisse für grosse Kondensatoren an. Eine Eingabe von 10 führt zu 1 Prozent kleineren Messergebnissen.
Beispiel: CFLAGS += -DC_H_KORR=10

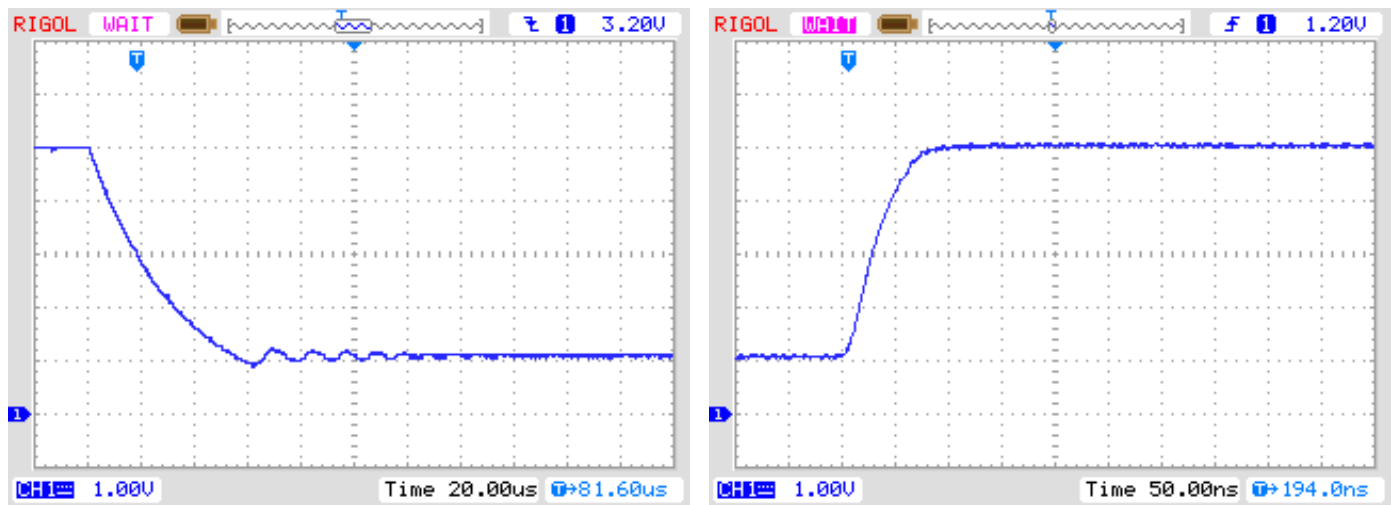
AUTOSCALE_ADC schaltet die automatische Bereichswahl des ADC (entweder VCC oder interne Referenz) ein. Die interne Referenz hat 2,56V für den ATmega8 und 1,1V für die anderen Prozessoren.
Beispiel: CFLAGS += -DAUTOSCALE_ADC

ESR_ZERO gibt einen Nullwert für die ESR-Messung von Kondensatoren vor. Dieser Nullwert wird von den ermittelten Messwerten abgezogen. Bei negativen ESR-Werten wird die Ausgabe „ESR=0?“ angezeigt und der Nullwert im EEprom korrigiert. Dieser korrigierte Wert wird bei den folgenden Messungen wirksam, bis er wieder beim nächsten Selbsttest zurückgesetzt wird. Falls Sie immer zu hohe ESR-Werte messen, sollten Sie diesen Vorgabewert erhöhen.
Beispiel: CFLAGS += -DESR_ZERO=29

NO_AREF_CAP teilt der Software mit, dass Sie keinen Kondensator am AREF Pin (Pin 21) angeschlossen haben. Dies ermöglicht kürzere Wartezeiten für die AUTOSCALE_ADC Umschaltung des ADC. Ein 1nF Kondensator wurde in diesem Modus ohne Fehler getestet. Die

Abbildungen 4.1a und 4.1b zeigen die Schaltzeiten mit einem $1nF$ Kondensator. Wie Sie sehen können ist das Schalten von 5V auf 1,1V viel langsamer als das Zurückschalten auf 5V. Wenn Sie noch einen $100nF$ installiert haben, ist die Schaltzeit etwa Faktor 100 länger!

Beispiel: `CFLAGS += -DNO_AREF_CAP`



(a) from 5V to 1.1V

(b) from 1.1V to 5V

Abbildung 4.1: Umschalten von AREF mit einem $1nF$ Kondensator

REF_R_KORR gibt einen Offset für die interne Referenz-Spannung in mV-Einheiten an. Mit diesem Offset kann eine Differenz bei der Umschaltung der Referenzspannung für die Widerstandsmessung abgeglichen werden. Wenn die `AUTO_CAL` Option gewählt wurde, ist dieser Wert nur ein Offset zu der gefundenen Spannungs-Differenz in der `AUTO_CAL` Funktion.

Beispiel: `CFLAGS += -DREF_R_KORR=10`

OP_MHZ gibt der Software an, mit welcher Taktfrequenz in MHz der Tester arbeiten wird. Die Software ist nur mit 1MHz, 8MHz und zusätzlich auch 16MHz getestet. Der Betrieb mit 8MHz wird wegen der besseren Auflösung der Kondensator- und Spulen-Messung empfohlen.

Beispiel: `OP_MHZ = 8`

USE_EEPROM gibt an, ob feste Texte und Tabellen im EEPROM-Speicher abgelegt werden sollen. Anderenfalls wird der Programmspeicher (Flash) benutzt. Es wird empfohlen, den EEPROM zu benutzen (Option gesetzt).

Beispiel: `CFLAGS += -DUSE_EEPROM`

PULLUP_DISABLE gibt an, dass man die internen „pull-up“ Widerstände nicht benötigt. Sie müssen einen externen „pull-up“ Widerstand an Pin 13 (PD7) und VCC angeschlossen haben, um diese Option benutzen zu können. Mit dieser Option wird ein möglicher Einfluss der „pull-up“ Widerstände auf die Mess-Ports (Port B und Port C) verhindert.

Beispiel: `CFLAGS += -DPULLUP_DISABLE`

ANZ_MESS diese Option gibt an, wie oft der ADC-Wert eingelesen und addiert werden soll. Sie können einen Wert zwischen 5 und 200 wählen um einen Mittelwert für eine ADC-Messung zu bilden. Höhere Werte ergeben eine bessere Genauigkeit, aber brauchen längere Messzeit. Eine ADC-Messung mit dem Wert 44 braucht etwa 5ms.

Beispiel: `CFLAGS += -DANZ_MESS=44`

POWER_OFF Diese Option schaltet die automatische Abschaltfunktion ein. Wenn Sie diese Option weglassen, werden die Messungen in einer Schleife endlos wiederholt, bis die Betriebsspannung unterbrochen wird (Ein/Aus Schalter). Wenn Sie einen Tester ohne die Schalttransistoren haben, können Sie diese Option weglassen. Wenn Sie mit den installierten Schalttransistoren die Option **POWER_OFF** weglassen haben, können Sie die Mess-Wiederholung dennoch abbrechen, wenn Sie den Start-Knopf einige Sekunden gedrückt halten während das Messergebnis angezeigt wird bis die „Timeout“ Meldung erscheint. // unschöner Satz... Wenn Sie den Knopf loslassen, schaltet der Tester den Strom ab. Sie können auch angeben, nach wie vielen Messungen ohne gefundenes Bauteil der Tester ausschaltet. Bei doppelt so viel aufeinanderfolgenden Messungen mit gefundenem Bauteil schaltet der Tester auch ab, wenn nicht zwischendurch eine Messung ohne gefundenes Bauteil war. Wenn Sie vergessen haben, ein angeschlossenes Bauteil abzuklemmen, wird so eine vollständige Batterie-Entladung verhindert. Bei einer Options-Angabe in der Form von `CFLAGS += -DPOWER_OFF=5` wird nach 5 aufeinanderfolgenden Messungen ohne gefundenes Bauteil abschaltet. Aufeinanderfolgende 10 Messungen mit gefundenem Bauteil schalten ebenfalls aus. Nur wenn die jeweilige Mess-Serie durch den anderen Typ unterbrochen wird, wird die Messung fortgesetzt. Die Messresultate für eine Einzelmessung werden 10 Sekunden angezeigt, bei der Mehrfachmessung wird die Anzeigezeit auf 3 Sekunden reduziert (wird in `config.h` gesetzt). Wenn der Startknopf beim ersten Einschalten lange gedrückt wird, wird das Messergebnis auch bei der Mehrfachmessung 10 Sekunden angezeigt. Der Maximalwert für die Wiederholungen ist 255 (`CFLAGS += -DPOWER_OFF=255`).

Beispiel 1: `CFLAGS += -DPOWER_OFF=5`

Beispiel 2: `CFLAGS += -DPOWER_OFF`

BAT_CHECK schaltet die Batterie Spannungsprüfung ein. Wenn Sie diese Option nicht angeben, wird die Versions-Nummer der Software angezeigt. Diese Option ist hilfreich um bei Batterie betriebenen Tester Versionen an den Batterie Wechsel zu erinnern.

Beispiel: `CFLAGS += -DBAT_CHECK`

BAT_OUT schaltet die Batterie-Spannungsanzeige auf dem LCD ein, wenn **BAT_CHECK** gewählt wurde. Wenn Ihre 9V-Versorgung eine Diode wegen des Verpolungs-Schutzes installiert hat, können Sie die Form `BAT_OUT=600` angeben, um die Dioden-Schwellspannung bei der Spannungsanzeige zu berücksichtigen. Auch der Spannungsverlust am Transistor T3 kann so mit dieser Option berücksichtigt werden. Die Angabe der Schwellspannung in mV beeinflusst nicht die Prüfspannungs Werte (**BAT_POOR**).

Beispiel 1: `CFLAGS += -DBAT_OUT=300`

Beispiel 2: `CFLAGS += -DBAT_OUT`

BAT_POOR setzt die Leer-Spannung für die Batteriespannungs-Prüfung auf den angegebenen Wert in Einheiten von 100mV (0,1V). Die Warn-Spannung ist immer 1V höher als die angegebene Leer-Spannung. Das Setzen der Leer-Spannung auf Werte wie 5,4V wird für wiederaufladbare 9V Batterien nicht empfohlen, weil das die Gefahr von Batterie-Schäden wegen der Tief-Entladung erhöht! Wenn Sie wiederaufladbare 9V Batterien einsetzen, werden „Ready to Use“ Typen wegen der geringeren Selbstentladung empfohlen.

Beispiel für low drop Regler (5.4V): `CFLAGS += -DBAT_POOR=54`

Beispiel für 7805 type Regler (6.4V): `CFLAGS += -DBAT_POOR=64`

PROGRAMMER stellt den Programmer Typ für das avrdude Schnittstellenprogramm ein. Eine richtige Einstellung des Programmer Typs (und Ports) ist notwendig, wenn Sie den „make upload“ oder „make fuses“ Aufruf dieser Makefile benutzen. Für weitere Informationen schauen Sie bitte in das Handbuch von avrdude oder in die Online-Dokumentation [12].

Beispiel: `PROGRAMMER=avrisp2`

PORT stellt die verwendete Schnittstelle ein, wo avrdude den Mikrocontroller (ATmega) erreichen kann. Für weitere Informationen schauen Sie bitte ins Handbuch von avrdude.

Beispiel: PORT=usb

Zusätzliche Parameter können in den Dateien Transistortester.h und config.h gesetzt werden. Die Datei Transistortester.h enthält globale Variablen und definiert die Port- / Pin-Konstellation sowie die Widerstandswerte, die für die Messung benutzt werden. Die Datei config.h setzt Parameter für die verschiedenen Prozessortypen, Wartezeiten und die Taktfrequenz für den ADC. Normalerweise brauchen diese Werte nicht ohne Grund geändert werden.

Beschreibung des Messverfahrens

Ein vereinfachtes Schaltbild eines Eingangs-/Ausgangs-Pin des ATmega wird in Abbildung 5.1 gezeigt. Der Schalter PUD schaltet die Versorgung für alle „Pull Up“-Widerstände des ATmega ab. Mit dem Schalter DD kann der Ausgang abgeschaltet werden, der Eingang funktioniert sowohl im Ausgabe- wie im Eingabe-Modus. Im Eingabe-Modus wird mit dem Ausgabewert (PORT) der „Pull Up“-Widerstand des Eingangs mit geschaltet. Die beiden Schalter PORT und DD können nicht gleichzeitig, sondern nur nacheinander geschaltet werden. Weil beim Umschalten der „Pull Up“ Widerstand die Messung stören könnte, bevorzuge ich die komplette Abschaltung aller „Pull Up“ Widerstände mit dem PUD-Schalter. Natürlich sind die Schalter elektronisch und die Widerstände 19Ω und 22Ω sind angenäherte Werte.

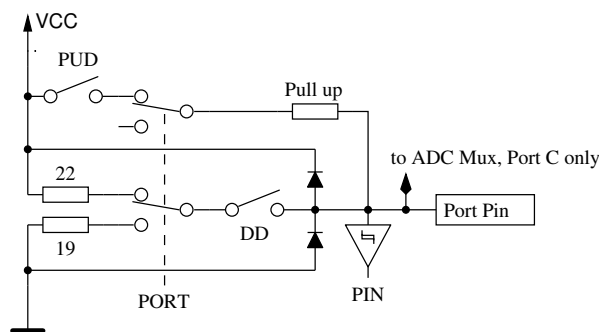


Abbildung 5.1: Vereinfachtes Schaltbild jedes ATmega-Portpins

Jeder der drei Testpins Ihres TransistorTesters wird aus drei ATmega-Portpins gebildet, was im vereinfachten Schaltbild des Testpins TP2 (mittlerer der drei Pins) in Abbildung 5.2 gezeigt wird.

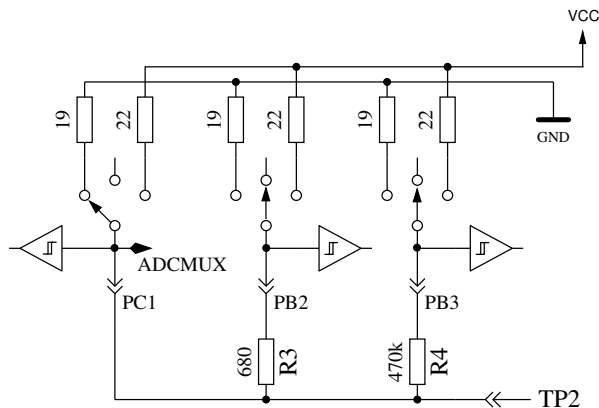


Abbildung 5.2: Vereinfachtes Schaltbild des Testpins TP2

Jeder Testpin (Messport) kann als digitaler oder analoger Eingang benutzt werden. Diese Messfähigkeit ist unabhängig von der Verwendung des Ports als Ausgang. Jeder Testpin kann als Ausgang verwendet werden und in diesem Zustand mit GND (0V) oder VCC (5V) verbunden werden, oder er kann über die Widerstände (680Ω oder $470k\Omega$) mit entweder GND oder VCC verbunden werden. Tabelle 5.1 zeigt alle denkbaren Messmöglichkeiten. Beachten Sie, dass der positive Zustand durch direktes Verbinden mit VCC (Port C) oder durch Verbinden mit dem 680Ω Widerstand mit VCC (Port B) erreicht werden kann. Die gleiche Möglichkeit hat der negative Zustand des Testpins zu der GND-Seite. Der Test-Zustand meint, dass der Pin offen sein kann (Eingang), verbunden über den $470k\Omega$ -Widerstand mit VCC oder GND, oder der Pin kann über den 680Ω -Widerstand mit VCC oder GND verbunden sein.

	Zustand Pin 1	Zustand Pin 2	Zustand Pin 3
1.	positiv	negativ	test
2.	positiv	test	negativ
3.	test	negativ	positiv
4.	test	positiv	negativ
5.	negativ	test	positiv
6.	negativ	positiv	test

Tabelle 5.1: alle Messmöglichkeiten

Wenn die Kondensatormessung des Testers konfiguriert ist, versucht der Tester vor allen Messungen erst einmal, die Kondensatoren an allen Anschlusspins zu entladen. Wenn das nicht gelingt, also die Restspannung zu hoch bleibt, wird das Entladen nach etwa 12 Sekunden mit der Meldung „Cell!“ abgebrochen. Dies kann auch dann vorkommen, wenn gar kein Kondensator angeschlossen ist. Die Ursache kann in diesem Fall sein, dass die Entlade-Grenzspannung für diesen ATmega zu niedrig gewählt ist. Man kann eine höhere Restspannung mit der Makefile Option CAP_EMPTY_LEVEL wählen.

5.1 Messung von Halbleitern

Ein Testpin wird als negative Seite des Bauteils angenommen. Ein anderer Pin wird als positive Seite des Bauteils angenommen. Als erster Test wird die positive Seite des Bauteils direkt mit VCC verbunden. Die negative Seite wird mit dem 680Ω Widerstand nach GND verbunden. Der Testpin (dritter Pin, auch TriStatePin genannt) wird zuerst mit dem 680Ω -Widerstand für 10ms mit GND verbunden. Die Spannung des negativen Testpins wird gemessen, während der TriStatePin auf Eingang geschaltet ist. Es wird angenommen, dass das getestete Bauteil ein P-Kanal-MOSFET sein kann und dass das Gate entladen sein sollte. Wenn die gemessene Spannung über 976mV ist, nimmt der nächste Test an, dass das getestete Bauteil auch ein P-Kanal-MOSFET sein könnte und dafür wird der 680Ω Widerstand für 10ms zur VCC-Seite geschaltet. Auch für diesen Fall wird die Spannung des negativen Pins mit stromlosen TriStatePin gemessen. Wenn die Spannung des negativen Pins grösser als 92mV ist, werden zusätzliche Tests gemacht, um N-Kanal JFET oder D-MOSFET (Verarmungs-Typ) und P-Kanal JFET oder P-MOSFET zu unterscheiden. Die MOSFET-Versionen können unterschieden werden durch den stromlosen Zustand des TriStatePins in jedem Zustand. Um Parameter dieser Verarmungstypen messen zu können, werden sie mit einem 680Ω -Widerstand am Source-Pin vermessen, wie in Abbildung 5.3 gezeigt wird. Diese Messung wird anstelle der üblichen Messung des Stromes bei einer Gate-Spannung auf Source-Potential gemacht, da wegen des relativ hohen 680Ω Widerstandes in vielen Fällen der Kennstrom I_{DSS} des FETs nicht erreicht würde.

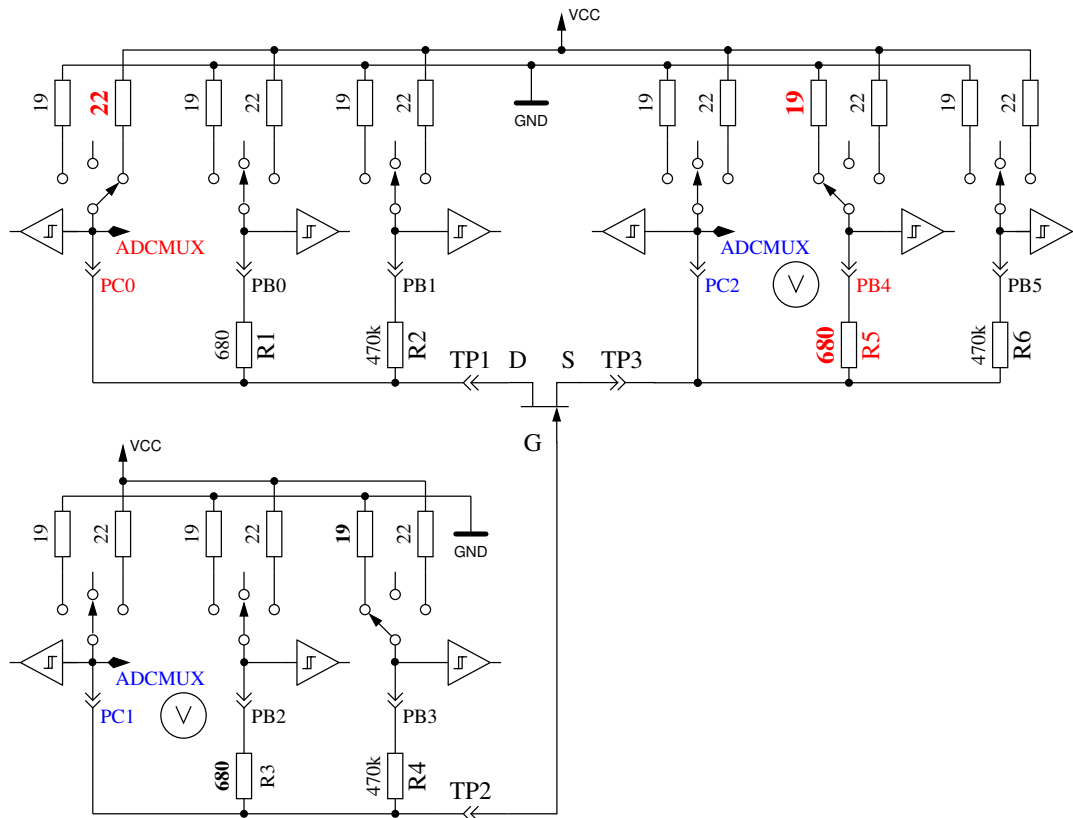


Abbildung 5.3: Messung von Gate-Source-Spannung und Source-Strom eines N-JFET-Transistors

Wenn das Bauteil keinen Strom zwischen dem positiven Pin und dem negativen Pin ohne ein Signal auf dem TristatePin hat, sind die nächsten Tests im nächsten Unterkapitel 5.1.1 beschrieben. Wenn Strom festgestellt wird, sind die nächsten Tests in dem Dioden-Unterkapitel 5.1.3 beschrieben.

5.1.1 Messung eines PNP-Transistors oder eines P-Kanal MOSFETs

Zuerst wird der Stromverstärkungsfaktor in der Kollektor-Schaltung (Emitter-Folger) für den angenommenen PNP-Transistor gemessen. Die Messsituation wird in Abbildung 5.4 gezeigt. Wenn die gemessene Basis-Spannung (U_B) über 9mV mit dem 680Ω Widerstand liegt, wird die Stromverstärkung hFE berechnet mit $hFE = \frac{U_E - U_B}{U_B}$. Die Spannung U_E ist die Differenz der Emitter-Spannung zu VCC. Die Differenz des 22Ω und 19Ω-Widerstandes wird nicht berücksichtigt. Wenn die Spannung U_B unter 10mV liegt, wird die Messung mit dem 470kΩ-Widerstand an der Basis gemacht. Für diesen Fall wird der Stromverstärkungsfaktor mit $hFE = \frac{U_E \cdot 470000}{U_B \cdot (680 + 22)}$ gebildet.



Abbildung 5.4: hFE -Messung eines PNP-Transistors in Kollektor-Schaltung

Als Nächstes werden die Tests in Emitter-Schaltung für den angenommenen PNP-Transistor gemacht. Die positive Seite wird jetzt direkt mit VCC verbunden, der 680Ω-Widerstand der negativen Seite wird mit GND verbunden, wie es in Abbildung 5.5 gezeigt wird. Wenn die negative Seite des Bauteils eine Spannung über 3,4V hat, wenn der 680Ω-Widerstand auf der Basis-Seite mit GND verbunden ist, muss es ein PNP-Transistor oder ein P-Kanal-FET sein. Das kann einfach unterschieden werden durch Prüfen der Basis-Spannung: Wenn sie grösser als 0,97V ist, muss es ein PNP sein. Für die Messung des Stromverstärkungsfaktors wird anstelle des 680Ω-Widerstandes der 470kΩ-Widerstand als Basis-Widerstand genommen. Der Stromverstärkungsfaktor wird berechnet mit $hFE = \frac{U_C \cdot 470000}{U_B \cdot (680 + 19)}$. Der höhere Stromverstärkungsfaktor wird als der richtige angenommen, dieser hier oder der mit der Kollektor-Schaltung bestimmte.

Die Werte, die für den PNP-Transistor herausgefunden wurden, sind nur gültig, wenn ein zweiter Satz von Messungen gemacht wurde. Um zu verhindern, dass der PNP-Transistor in der inversen Schaltung (Kollektor und Emitter vertauscht) erkannt wird, wird dann die Messung mit dem höheren Stromverstärkungsfaktor als richtige Messung genommen. Wenn die Basis-Spannung kleiner als 0,97V ist, muss es ein P-E-MOS sein. In diesem Fall wird die Gate-Schwellwertspannung dadurch bestimmt, dass die Spannung am Gate langsam mit dem 470kΩ-Widerstand rauf und runter gezogen wird bis die Drain-Seite schaltet und dann die Spannung am Gate gemessen wird.



Abbildung 5.5: Prüfung und h_{FE} -Messung eines PNP-Transistors in der Emitter-Schaltung

5.1.2 Messung eines NPN-Transistors oder eines N-Kanal-MOSFET

Die Messung eines NPN-Transistors beginnt auf gleiche Weise wie die PNP-Transistor-Messung, nämlich mit der Messung des Stromverstärkungsfaktors in der Kollektor-Schaltung. Zuerst wird die Messung mit einem nach VCC geschalteten 680Ω -Basiswiderstand gemacht. Wenn die Spannung am Basis-Widerstand zu klein ist, wird stattdessen der $470k\Omega$ -Widerstand genommen. Die Messungen werden dann in der Emitter-Schaltung fortgeführt, wie in Abbildung 5.6 gezeigt.



Abbildung 5.6: Prüfung und hFE-Messung eines NPN-Transistors in Emitter-Schaltung

Wenn weder Thyristor noch TRIAC bestätigt wurden, kann es ein NPN oder ein N-Kanal E-MOSFET sein. Die Basis-Spannung von einem NPN-Transistor wird nahe bei der Emitter-Spannung liegen, so dass dieser Typ sicher erkannt werden kann. Der Stromverstärkungsfaktor in der Emitter-Schaltung wird durch $hFE = \frac{(VCC-UC) \cdot 470000}{(VCC-UB) \cdot (680+22)}$ gebildet. Wenn die Spannung an der Basis zeigt, dass kein oder wenig Strom fließt, wird das Bauteil ein N-Kanal E-MOS (Anreicherungs-MOSFET) sein. In diesem Fall wird die Schwellspannung gemessen, indem die Spannung des Gates langsam mit dem $470k\Omega$ -Widerstand nach VCC und GND gezogen wird, darauf wartend, dass das digitale Eingangssignal auf der Drain Seite schaltet, wobei dann die Gate-Spannung gelesen wird. Die Messung wird elf Mal wiederholt wie in Abbildung 5.7 gezeigt und die Ergebnisse addiert. Diese Summe wird mit vier multipliziert und durch neun geteilt um eine Auflösung in mV zu erhalten.

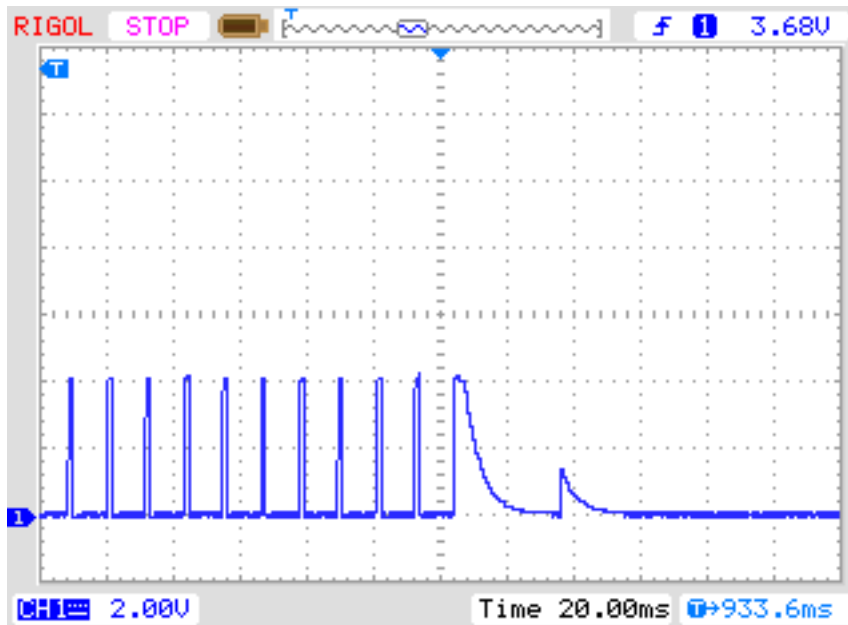


Abbildung 5.7: Messung der Schwellspannung eines N-Kanal-MOSFET

5.1.3 Messung von Dioden

Wenn Strom bei den Vortests festgestellt wurde, wird das Bauteil auf Diodenverhalten geprüft. Die Flussspannung mit dem 680Ω -Widerstand muss zwischen 0,15V und 4,64V liegen. Die Flussspannung mit dem 680Ω -Widerstand muss grösser als 1,125 Mal der Flussspannung mit dem $470k\Omega$ -Widerstand sein und acht Mal die Flussspannung mit dem $470k\Omega$ -Widerstand muss grösser als die Flussspannung mit dem 680Ω -Widerstand sein. Ich hoffe, dass ein Bauteil mit diesem Verhalten immer eine Diode ist.

5.1.4 Ergebnisse der verschiedenen Messungen

Die folgenden drei Tabelle zeigen die Ergebnisse verschiedener Bauteile mit verschiedenen Software-Konfigurationen von ATmega8- und ATmega168-Prozessoren.

Diode Type	Mega8@8MHz Signatur 1E9307 WITH_AUTO_REF	Mega168 @8MHz Signatur 1E9406	Mega168 @8MHz Signatur 1E9406 AUTO_CAL AUTOSCALE_ADC
1N4148	Diode, .71V, 1pF	Diode, 721mV, 0pF	Diode, .72V, 0pF
1N4150	Diode, .67V, 2pF	Diode, 672mV, 1pF	Diode, .67V, 1pF
BA157	Diode, .62V, 19pF	Diode, 625V, 20pF	Diode, .63V, 19pF
BY398	Diode, .54V, 17pF	Diode, 541mV, 16pF	Diode, .54V, 15pF
1N4007	Diode, .65V, 13pF	Diode, 660mV, 13pF	Diode, .66V, 12pF
LED green	Diode, 1.95V, 6pF	Diode, 1.95V, 6pF	Diode, 1.95V, 6pF
ZPD2,7	2xDi, .74V, 2.53V	2xDi, 743mV, 2.53V	2xDi, .74V, 2.52V
BU508A B+E	Diode, .61V, 5.20nF	Diode, 614mV, 5.30nF	Diode, .61V, 5.29nF
BU508A B+C	Diode, .58V, 257pF	Diode, 589mV, 264pF	Diode, .61V, 285pF
AC128 B+E	Diode, .27V, 0pF	Diode, 277mV, 0pF	Diode, .28V, 0pF
MBR20100CT	2xDi, .33V, .33V	2xDi, 341mV, 341mV	2xDi, .34V, .34V
MBR20100CT	Diode, .33V, 358pF	Diode, 341mV, 360pF	Diode, .34V, 358pF
MBR4045PT	Diode, .23V, 1.94nF	Diode, 239mV, 1.98nF	Diode, .24V, 1.94nF
SF38G	Diode, .52V, 107pF	Diode, 523mV, 108pF	Diode, .53V, 108pF

Tabelle 5.2: Messergebnisse der Dioden-Tests

Transistor Type	Mega8@8MHz Signatur 1E9307 WITH_AUTO_REF	Mega168 @8MHz Signatur 1E9406	Mega168 @8MHz Signatur 1E9406 AUTO_CAL AUTOSCALE_ADC
BU508A	NPN, B=10, .60V	NPN, B=9, 599mV	NPN, B=9, .60V
2N3055	NPN, B=21, .62V	NPN, B=21, 551mV	NPN, B=21, .55V
BC639	NPN, B=180, .72V	NPN, B=215, 633mV	NPN, B=196, .64V
BC640	PNP, B=216, .72V	PNP, B=226, 639mV	PNP, B=178, .61V
BC517	NPN, B=28296, 1.40V	NPN, B=25.7k, 1.21V	NPN, B=25.4k, 1.23V
BC516	PNP, B=65535, 1.41V	PNP, B=78.7k, 1.19V	PNP, B=73.9k, 1.20V
BC546B	NPN, B=385, .77V	NPN, B=428, 686mV	NPN, B=428, .69V
BC556B	PNP, B=306, .79V	PNP, B=426, 696mV	PNP, B=256, .67V
AC128 (Ge.)	PNP, B=64, .27V	PNP, B=63, 190mV	PNP, B=59, .19V
BRY55/200	Thyristor	Thyristor	Thyristor

Tabelle 5.3: Messergebnisse der Tests mit bipolaren Transistoren

FET Type	Mega8@8MHz Signatur 1E9307 WITH_AUTO_REF	Mega168 @8MHz Signatur 1E9406	Mega168 @8MHz Signatur 1E9406 AUTO_CAL AUTOSCALE_ADC
ZVNL120A	N-E-MOS,D, 1.5V 142pF	N-E-MOS,D, 1.5V 146pF	N-E-MOS,D, 1.5V 144pF
IRF530N	N-E-MOS,D, 3.6V 1.55nF	N-E-MOS,D, 3.6V 1.56nF	N-E-MOS,D, 3.6V 1.55nF
BS170	N-E-MOS,D, 2.6V 71pF	N-E-MOS,D, 2.6V 73pF	N-E-MOS,D, 2.6V 71pF
IRL3803	N-E-MOS,D, 2.3V 9.80nF	N-E-MOS,D, 2.3V 9.82nF	N-E-MOS,D, 2.3V 9.80nF
IRFU120N	N-E-MOS,D, 4.2V 916pF	N-E-MOS,D, 4.2V 924pF	N-E-MOS,D, 4.2V 911pF
BUZ71A	N-E-MOS,D, 3.2V 706pF	N-E-MOS,D, 3.2V 718pF	N-E-MOS,D, 3.2V 707pF
ZVP2106A	P-E-MOS,D, 3.2V 118pF	P-E-MOS,D, 3.2V 121pF	P-E-MOS,D, 3.2V 119pF
IRF5305	P-E-MOS,D, 3.6V 2.24nF	P-E-MOS,D, 3.6V 2.25nF	P-E-MOS,D, 3.6V 2.24nF
BS250	P-E-MOS,D, 2.6V 46pF	P-E-MOS,D, 2.6V 48pF	P-E-MOS,D, 2.6V 46pF
IRFU9024	P-E-MOS,D, 3.5V 956pF	P-E-MOS,D, 3.6V 954pF	P-E-MOS,D, 3.5V 943pF
J310 Idss=24-60mA	N-JFET I=3.1mA Vgs=2.2V	N-JFET I=3.1mA Vgs=2.2V	N-JFET I=3.1mA Vgs=2.2V
BF256C Idss=11-18mA	N-JFET I=3.1mA Vgs=2.2V	N-JFET I=3.4mA Vgs=2.3V	N-JFET I=3.3mA Vgs=2.3V
BF245A Idss=2-6mA	N-JFET I=3.1mA Vgs=2.2V	N-JFET I=1.1mA Vgs=0.74V	N-JFET I=1.1mA Vgs=0.75V
BF245B Idss=6-15mA	N-JFET I=3.1mA Vgs=2.2V	N-JFET I=2.5mA Vgs=1.7V	N-JFET I=2.5mA Vgs=1.7V
BF245C Idss=12-25mA	N-JFET I=3.1mA Vgs=2.2V	N-JFET I=3.9mA Vgs=2.7V	N-JFET I=3.9mA Vgs=2.7V

Tabelle 5.4: Messergebnisse der MOS-Transistor-Tests

5.2 Widerstands-Messung

Jeder Widerstand wird mit vier verschiedenen Messmethoden in einer Stromrichtung vermessen. Der gleiche Widerstand wird auch mit den gleichen vier Messmethoden in die andere Stromrichtung vermessen. Die Messung in die Gegenrichtung wird nur für die Erkennung auf Widerstand benutzt. Wenn die Abweichung dieser beiden Messungen zu groß ist, ist es kein Widerstand.

5.2.1 Widerstandsmessung mit den 680-Ohm-Widerständen

Die Messung des unbekannten Widerstandes R_x wird in zwei verschiedenen Wegen mit den 680 Ω -Präzisionswiderständen durchgeführt. Das Schaltbild dieser Messungen mit Testpin 1 (TP1) und Testpin 3 (TP3) werden vereinfacht in Abbildung 5.8 und Abbildung 5.9 als ein Beispiel von den sechs Kombinationsmöglichkeiten gezeigt.



Abbildung 5.8: Messung Type 1 mit 680 Ω



Abbildung 5.9: Messung Type 2 mit 680 Ω

Auf der linken Seite wird der Testpin 1 und auf der rechten Seite der Testpin 3 gezeigt. In beiden Schaltungen kann man erkennen, dass der Anschluss 3 (TP3) mit VCC und die linke Seite (TP1) mit GND verbunden ist. Die Stromrichtung durch den Widerstand R_x ist immer die gleiche. Die Werte für auf Ausgang geschaltete Ports werden mit roter Farbe dargestellt, die Werte für die Eingänge werden mit blauer Farbe dargestellt, inaktive Ports sind schwarz. In beiden gezeigten Messmethoden sollte der Strom den gleichen Wert haben, weil die Summe der Widerstände zwischen VCC und GND gleich ist, vorausgesetzt die eingebauten Widerstände sind gleich. Normalerweise sind die gemessenen Spannungen aber nicht gleich, weil die Reihenfolge der Widerstände vertauscht ist.

Das V-Symbol innerhalb eines Kreises markiert die Ports, die für die Spannungsmessung benutzt werden. In beiden Konfigurationen kann der Wert des Widerstandes Rx aus den bekannten Widerstandswerten und den gemessenen Spannungen berechnet werden, wenn das Verhältnis des Widerstands Rx und den 680Ω -Widerständen nicht zu hoch ist. Der theoretische Spannungsverlauf wird in Abbildung 5.10 gezeigt, wobei die Widerstandswerte in logarithmischer Skalierung dargestellt sind.



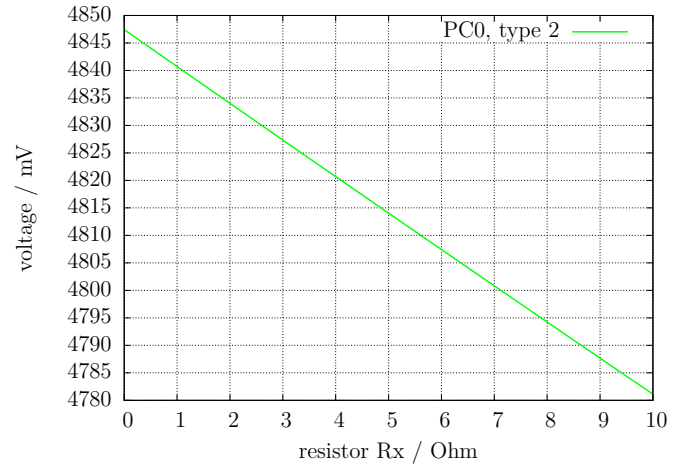
Abbildung 5.10: Spannung von Type 1 und Type 2 mit Messwiderstand 680Ω

Die Verlauf für Messung Type 1 wird in Abbildung 5.11a mit gespreizter Darstellung für die unteren Widerstandswerte gezeigt. Wie man sehen kann, braucht man eine bessere ADC-Auflösung als die möglichen $4,9\text{mV}$ bei der 5V ADC-Referenz, um richtige Widerstandswerte von der gemessenen Spannung unter 2Ω zu erhalten. Es gibt nur drei ADC-Stufen mit der 5V -Referenz zwischen 0Ω und 2Ω . Hier kann die Bereichsumschaltung mit der `AUTOSCALE_ADC` Option helfen. Der gleiche gespreizte Bereich für die Type 2 Messung wird in Abbildung 5.11b gezeigt. Unglücklicherweise kann man nicht die höhere ADC-Auflösung für Messmethode Type 2 benutzen, weil die Spannung zu hoch ist und unser ATmega keine differenziellen ADC-Eingänge besitzt. Die Messungen mit den 680Ω -Widerständen werden bis zu einem Widerstandswert von $20\text{k}\Omega$ (Spannung ist unter 169mV) zur Bestimmung des Messergebnisses verwendet.

Für höhere Widerstandswerte werden Messungen mit den $470\text{k}\Omega$ -Widerständen benutzt. Der Mittelwert von beiden Messungen wird für den angezeigten Widerstandwert benutzt, wenn alle Messungen ergeben, dass es kein anderes Bauteil ist. Wenn die `AUTOSCALE_ADC` Funktion benutzt wird und eine der gemessenen Spannungen für beide Versionen unter $0,98\text{V}$ liegt, wird ein gewichteter Mittelwert mit Faktor vier für die Messung mit der Spannung unter $0,98\text{V}$ benutzt. Der andere Wert wird mit Faktor eins bewertet. Das wird wegen der Faktor vier besseren Auflösung dieser Messung gemacht. Faktor vier wird nur für ATmega88, ATmega168 und ATmega328 Prozessoren verwendet, für ATmega8 wird ein Faktor zwei als Wichtung benutzt wenn die Spannung unter $0,98\text{V}$ ist, weil die ADC Referenzspannung hier $2,56\text{V}$ statt $1,1\text{V}$ beträgt. Wenn der ATmega mehr als 8 KByte Flashspeicher besitzt, wird die Spannungsmessung an den Widerständen so lange verzögert, bis keine Änderung mehr festgestellt wird oder eine Zeitgrenze überschritten wird. Durch diese Maßnahme werden auch große Kondensatoren nicht mehr irrtümlich als Widerstände erkannt und der Gleichstrom-Widerstand großer Induktivitäten wird richtig gemessen.



(a) Type 1 Messung



(b) Type 2 Messung

Abbildung 5.11: Ausschnitt des theoretischen Spannungsverlauf von 0Ω bis 10Ω

5.2.2 Widerstandsmessung mit den $470\text{-k}\Omega$ -Widerständen

Die nächsten Abbildungen 5.12 und 5.13 zeigen die gleichen Messmethoden für die Messungen mit $470\text{k}\Omega$ -Präzisionswiderständen. Weil $470\text{k}\Omega$ in Relation zu den Port-Widerständen 22Ω und 19Ω sehr groß ist, können die Port-Widerstände für die Berechnung des Widerstandswertes R_x vernachlässigt werden.

Für beide Messmethoden mit den $470\text{k}\Omega$ -Widerständen wird nur eine Spannung gemessen, weil der Strom so niedrig ist, dass keine Spannungsdifferenz an den internen Port Widerständen gemessen werden kann (wie zu erwarten). Der theoretische Spannungsverlauf wird in Abbildung 5.14 gezeigt, wobei die Widerstandswerte wieder in logarithmischer Skalierung gezeigt werden. Der theoretische Verlauf in diesem Diagramm endet bei $100\text{M}\Omega$, aber das Ergebnis des Testers wird auf $60\text{M}\Omega$ begrenzt, anderenfalls nimmt der Tester an, dass kein Widerstand angeschlossen ist. Als Ergebnis wird der gewichtete Mittelwert von beiden Messmethoden verwendet. Dies geschieht nach den gleichen Regeln, die schon bei den Messungen mit den 680Ω Widerständen beschrieben wurden. Ich habe beobachtet, dass die Messergebnisse für alle ATmega Typen näher am wahren Wert liegen, wenn zum Messergebnis ein konstanter Offset von 700Ω addiert wird. Dieser Offset kann mit der Konstante `RH_OFFSET` (define) in der Datei `config.h` angepasst werden.

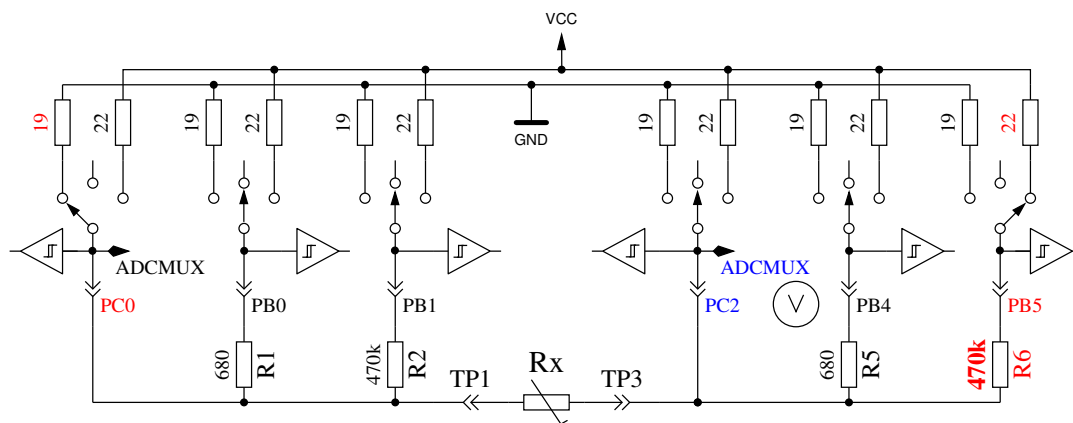


Abbildung 5.12: Messung Type 3 mit $470\text{k}\Omega$



Abbildung 5.13: Messung Type 4 mit $470k\Omega$



Abbildung 5.14: Spannungen bei Messungen von Type 3 und Type 4 mit $470k\Omega$

5.2.3 Ergebnisse der Widerstands-Messung

Abbildung 5.15 zeigt den relativen Fehler der Widerstandsmessung mit einem ATmega8. „Mega8“ sind die Ergebnisse ohne AUTOSCALE_ADC Option, „Mega8as“ sind die Ergebnisse mit der Option AUTOSCALE_ADC und „Mega8orig“ sind die gleichen Messungen mit der Originalsoftware von Markus F. ohne meine Verbesserungen. Die Ergebnisse für die AUTOSCALE_ADC Version zeigen hier kaum Verbesserungs-Potential. Abbildung 5.16 zeigt die gleichen Messungen mit einem ATmega168. „Mega168“ sind die Ergebnisse ohne die AUTOSCALE_ADC Option, „Mega168as“ die mit der AUTOSCALE_ADC Option.

Mit dem ATmega168 scheint es möglich zu sein, Messungen von Widerständen im Bereich von 20Ω bis $20M\Omega$ mit einem Messfehler von unter $\pm 1\%$ durchzuführen. Für Messungen unterhalb von 100Ω sollte man berücksichtigen, dass jede Prüfklemme mit Kabel ebenfalls einen Widerstandwert hat. Es ist besser, den Widerstand direkt mit den Anschlussklemmen zu verbinden. Wenn das nicht möglich ist, sollte man der Widerstandswert der kurzgeschlossenen Prüfklemmen vom Messergebnis abziehen. Zeigt beispielsweise der Tester einen Wert von $30,6\Omega$ an, wenn der Präzisionswiderstand

einen aufgedruckten Wert von 30Ω hat, und die im Kurzschluss gemessenen Prüfklemmen haben einen Wert von $0,5\Omega$, dann wird der Widerstand vom Tester mit $30,1\Omega$ gemessen. Unterhalb von einem Widerstandswert von 10Ω macht ein Auflösungs-Schritt von $0,1\Omega$ schon einen Fehler von mehr als 1%!

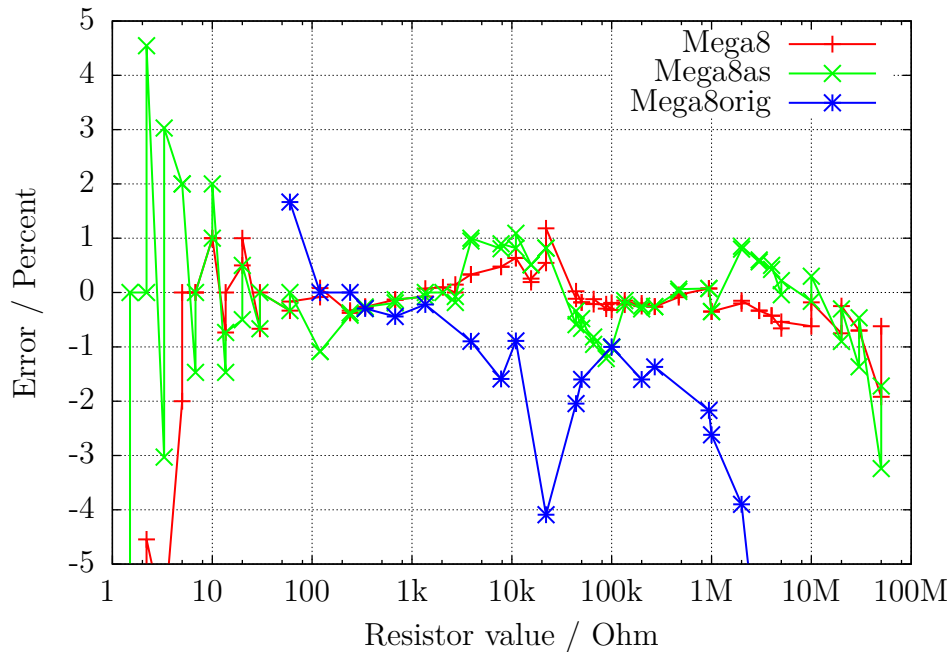


Abbildung 5.15: Relativer Fehler in % für Widerstands-Messungen mit ATmega8

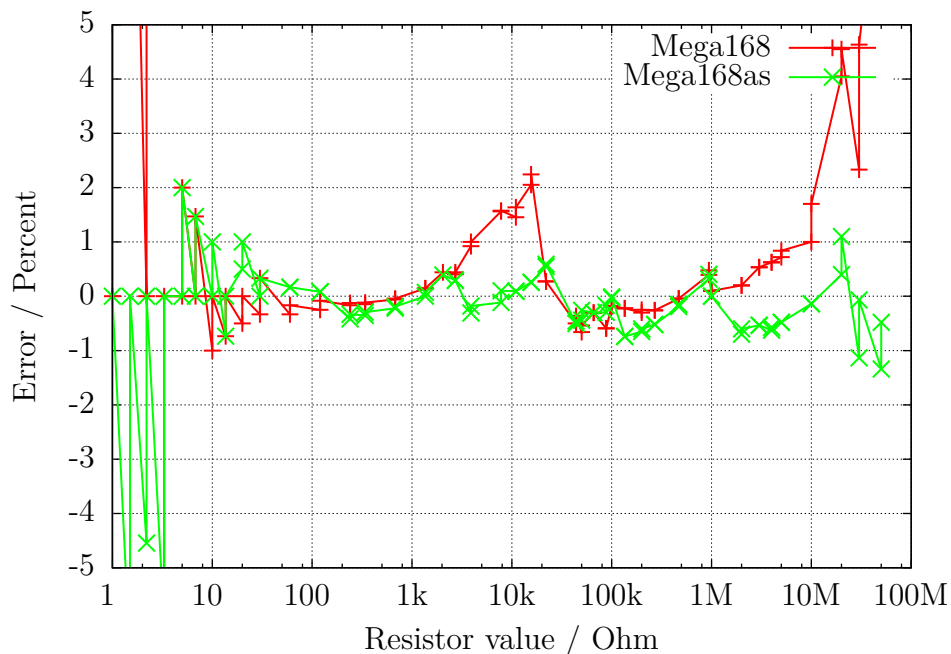


Abbildung 5.16: Relativer Fehler in % für Widerstand-Messungen mit ATmega168

Das Diagramm 5.17a zeigt die Messfehler von drei ATmega168-Prozessoren vor der Kalibration als Punkte, nach der Kalibration als Linie. Entsprechend werden die Messfehler von drei ATmega168A in Abbildung 5.17b und die Messfehler von drei ATmega328P in Abbildung 5.18 gezeigt. Die Messfehler der ATmega328 werden in den Abbildungen 5.19a und 5.19b gezeigt. Nach der automatischen Kalibration bleibt der Messfehler mit einer Ausnahme (ATmega328P-13, $22k\Omega$) im Widerstandsbereich

$10\ \Omega - 20\text{M}\Omega$ im Bereich $\pm 1\%$. Vor der Kalibration können die Messfehler bei einigen Prozessoren bis $\pm 3\%$ betragen. Der Fehler wird verursacht durch die AUTOSCALE_ADC-Umschaltung der ADC-Referenz. Durch den direkten Vergleich einer Kondensatorspannung von unter 1 V, einmal mit der VCC-Referenz und nochmal mit der internen ADC-Referenz gemessen, kann der Fehler ausgeglichen werden. In diesem Fall wird die Spannung mit dem gleichen Multiplexer-Kanal gemessen und die Bandgap-Referenz ist auf den AREF-Pin aufgeschaltet. Die direkte Vermessung der Bandgap-Referenz durch die direkte Wahl des Multiplexer-Eingangs führt leider zu diesem Offset, der entweder manuell mit der Option REF_R_KORR oder automatisch mit der Option AUTO_CAL des Selbsttestes beseitigt werden kann. Im AUTO_CAL-Modus ist REF_R_KORR ein zusätzlicher Offset zur automatisch gefundenen Spannungsdifferenz.

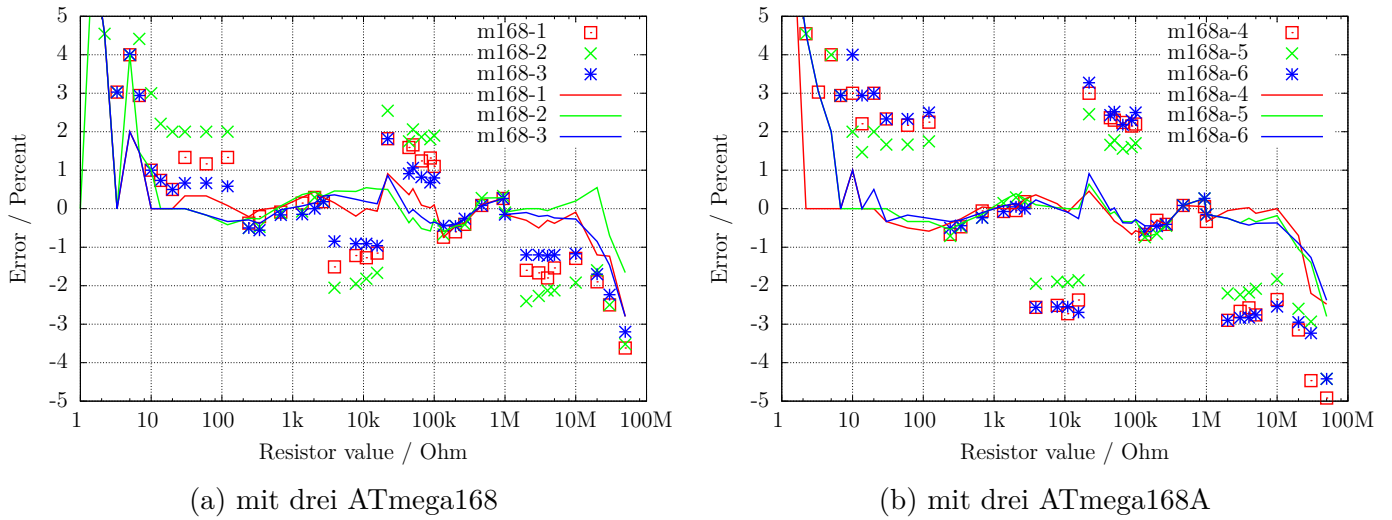


Abbildung 5.17: Relativer Fehler in % für Widerstands-Messungen

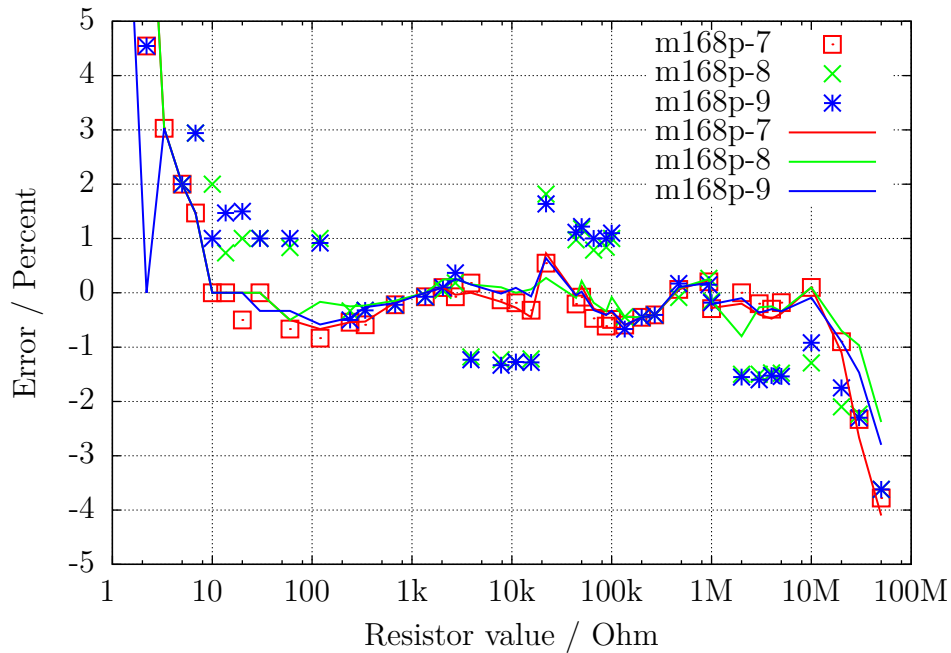


Abbildung 5.18: Relativer Fehler in % für Widerstands-Messungen mit drei ATmega168P



(a) mit drei ATmega328



(b) mit drei ATmega328P

Abbildung 5.19: Relativer Fehler in % für Widerstands-Messungen

5.3 Messen von Kondensatoren

Die Messung von Kapazitätswerten wird nach allen anderen Messungen als separater Teil mit einer Ladezeitmessung durchgeführt. Die Originalsoftware vom Markus F. hat das mit einer Programmschleife gemacht, die den betreffenden digitalen Eingangs-Pin bis zu einer Signaländerung gelesen hat und dabei die Schleifendurchläufe gezählt. Dies hat den Nachteil, dass die Auflösung der Zeitmessung begrenzt ist durch die Gesamtzeit eines Schleifendurchlaufs. Dies wurde üblicherweise in allen sechs Kombinationsmöglichkeiten für die drei Testpins durchgeführt. Die aktuelle Software benutzt zwei verschiedene Möglichkeiten, um die Ladezeit in nur drei Kombinationsmöglichkeiten für die drei Testpins zu erhalten. Die positive Seite ist nun immer die höhere Testpin-Nummer. Nur wenn die Kapazität zusammen mit einer parallel geschalteten Diode gemessen wird, kann die Polarität die andere Richtung haben.

5.3.1 Entladen der Kondensatoren

Sie sollten die Kapazitäten immer entladen, bevor sie mit dem Tester verbunden werden. Bevor irgendein Test gestartet wird, wird der Kondensator vom Tester immer noch einmal entladen. Wenn die Spannung unter 1300mV ist, wird der Kondensator dafür mit den angeschlossenen ADC-Ports (Port C) kurzgeschlossen. Ich glaube das ist in Ordnung, weil jeder Portausgang einen Innenwiderstand von ungefähr 20Ω hat. Die Abbildung 149 (Seite 258) im Atmega8-Datenblatt [2] zeigt einen Spannungsabfall an Ausgabe-Pins von bis zu 2V. Natürlich kann ich nicht garantieren, dass kein Schaden auftreten kann. Ich habe die Funktion mit Kondensatoren von mehr als $15mF$ sehr oft getestet und ich habe noch nie ein Problem bemerkt. Der Strom sollte unter der spezifizierten Grenze von 40mA bleiben und reduziert sich schnell durch die Entladung. Natürlich kann ein Schaden entstehen, wenn Sie einen (Hochvolt-) Kondensator nicht vollständig entladen, bevor Sie ihn an den Tester anschließen.

5.3.2 Messung von großen Kapazitäten

Eine Seite des Kondensators ist mit GND verbunden. Die andere Seite wird über den 680Ω -Widerstand für 10ms mit VCC verbunden. Danach wird dieser Messpin auf Eingang geschaltet (hochohmig). Nach diesem Strompuls wird die Spannung am Kondensator stromlos gemessen. Wenn die Spannung noch nicht den Minimalwert von 300mV erreicht hat, wird dieser Ladepuls bis zu weiteren 499 Mal wiederholt. Wenn nach 127 Pulsen (ungefähr 2s) noch nicht eine Minimalspannung von 75mV erreicht ist, wird der Ladevorgang abgebrochen, weil die 300mV mit den verbleibenden Ladepulsen nie erreicht werden wird. Abbildung 5.20 zeigt die drei Phasen der Kapazitätsbestimmung eines Kondensators. Der Kapazitätswert wird dann berechnet aus der Ladepuls-Anzahl und der erreichten Ladespannung über eine Tabelle. Die Tabelle enthält mit einem Spannungsabstand von 25mV die Faktoren, um aus der Ladezeit und der Spannung den Kapazitätswert zu bestimmen. Zwischenwerte der Spannung werden interpoliert.



Abbildung 5.20: Kondensator-Entladung und -Ladung mit 10ms Ladepulsen bis zur Spannung $>300\text{mV}$

Wegen der niedrigen Ladespannung wird die Messung viel schneller als bei der ursprünglichen Softwareversion, weil dieser Vorteil auch bei der Entladung wirkt. So können auch grössere Kondensatoren gemessen werden. Zusätzlich stört in den meisten Fällen eine parallel geschaltete Diode nicht die Messung, weil die Schwellspannung der Diode nicht erreicht wird. Abbildung 5.21a zeigt das Laden und Entladen eines $229\mu\text{F}$ großen Kondensators. Das flache Dach der Messkurve bis zum Entladebeginn ist durch die Messzeit und Berechnungszeit des ATmega verursacht. Abbildung 5.21b zeigt die gleiche Messung mit einem 5mF -Kondensator, beachte wie die Messzeit inklusive Entladung auf ungefähr 1,5 Sekunden angewachsen ist. Das letzte Beispiel in Abbildung 5.22 zeigt die Messung eines 15mF -Kondensators.



(a) $229\mu F$ -Kondensator



(b) $5mF$ -Kondensator

Abbildung 5.21: Laden und Entladen von großen Kondensatoren für die Messung



Abbildung 5.22: Laden und Entladen eines $15mF$ -Kondensators für die Messung

Nach dieser Kondensatormessung wird die Selbstentladung des Kondensators untersucht, indem der Spannungsverlust in einer der Ladezeit proportionalen Zeit untersucht wird. Der gemessene Kapazitätswert wird entsprechend korrigiert. Ein Test mit einer Parallelschaltung von einem $68\mu F$ -Kondensator mit einem $2.2k\Omega$ -Widerstand zeigt die Wirksamkeit der Methode. Der ermittelte Kapazitätswert ohne den Widerstand beträgt $66.5\mu F$, mit dem parallelen $2.2k\Omega$ Widerstand wird ein Kapazitätswert von $65.3\mu F$ gemessen. Zum Vergleich möchte ich die entsprechenden Ergebnisse mit einem Peaktech 3315 Multimeter angeben. Ohne den Widerstand wird eine Kapazität von $68.2\mu F$ gemessen, mit dem parallelen $2.2k\Omega$ Widerstand wird aber $192\mu F$ mit dem Multimeter gemessen.

5.3.3 Messen von kleinen Kapazitäten

Wenn der erste 10 ms Ladepuls den Kondensator zu hoch aufgeladen hat, wird eine andere Messtechnik benutzt. Der ATmega-Mikrocontroller hat einen eingebauten 16-Bit-Zähler, der bei voller

Taktfrequenz (1MHz oder 8MHz) arbeiten kann. Diese Zähler hat auch die Fähigkeit aufgrund eines externen Ereignisses den Zählerstand zu sichern. Dieses Ereignis kann auch das Ausgangs-Signal des Komparators sein. Der Komparator kann mit jedem ADC-Eingangspin und der Spannungsreferenz (Band Gap Reference) arbeiten. Das Schaltbild 5.23 zeigt ein vereinfachtes Diagramm der Messsituation. So entlade ich den Kondensator, bereite den Komparator für den richtigen Pin Eingang vor, starte den Zähler bei 0 und starte sofort das Laden des Kondensators. Dabei ist eine Seite des Kondensators mit GND, die andere Seite über den $470k\Omega$ -Widerstand mit VCC verbunden. Nun prüfe ich in einer Programm-Schleife, ob der Zähler ein Überlauf-Ereignis (overflow) oder ein externes Ereignis (input capture) meldet. Die Überlauf-Ereignisse zähle ich, bis ich ein externes Ereignis feststelle. In diesem Fall halte ich den Zähler an und prüfe, ob ich noch einen zusätzlichen Überlauf zählen muss, weil der Zähler nicht mit dem externen Ereignis (input capture) angehalten werden kann.

Der Zählerstand des externen Ereignisses bildet zusammen mit dem Überlaufzähler die Gesamtzeit, aus der die Kapazität mit einem Faktor bestimmt wird. Die aktuelle Software kann eine Tabelle mit der theoretischen Abhängigkeit der Ladezeit in Bezug auf die gemessene Komparator-Spannung berücksichtigen. Die Tabelle besitzt Einträge in Schritten von 50mV, die Ergebnisse werden entsprechend der aktuellen Referenz-Spannung interpoliert. Diese Tabelle wird nur mit der Makefile Option `WITH_AUTO_REF` aktiviert. Vom Kapazitätswert ziehe ich eine experimentell herausgefundene vordefinierte Konstante oder eine im Selbsttest herausgefundene Konstante ab, um den Messwerte-Offset zu beseitigen. Ich weiß nicht, ob die vordefinierte Konstante für andere Leiterplatten angepasst werden muss. Mit einem Selbsttest bei gesetzter `AUTO_CAL` Option wird diese Anpassung automatisch erledigt.

Ich habe bemerkt, dass die Referenz-Spannung meistens etwas zu klein gemessen wird, deshalb kann man einen Zusatz mit der Makefile Option `REF_C_KORR` angeben. Nach der Kalibration mit der `AUTO_CAL` option ist der `REF_C_KORR` Wert nur ein Zusatz zu der automatisch gefundenen Differenz zwischen geladener Kondensator-Spannung und der internen Referenz-Spannung. Die gemessene Referenz-Spannung wird dann mit dem Korrekturwert (in mV) korrigiert (addiert). Wenn die Option `WITH_AUTO_REF` nicht benutzt wird, werden die Referenz-Spannungen entsprechend den Angaben in den Datenblättern [2] [3] der ATmega8, ATmega88, ATmega168 und ATmega328 berücksichtigt. Eine Beispielmessung von diesem Typ ist in Abbildung 5.24 dargestellt. Die Messzeit für den $22\mu F$ -Kondensator beträgt ungefähr 2,6s weil der $470k\Omega$ -Widerstand für das Laden benutzt wird. Aber das Entladen geht in diesem Fall viel schneller als das Laden.

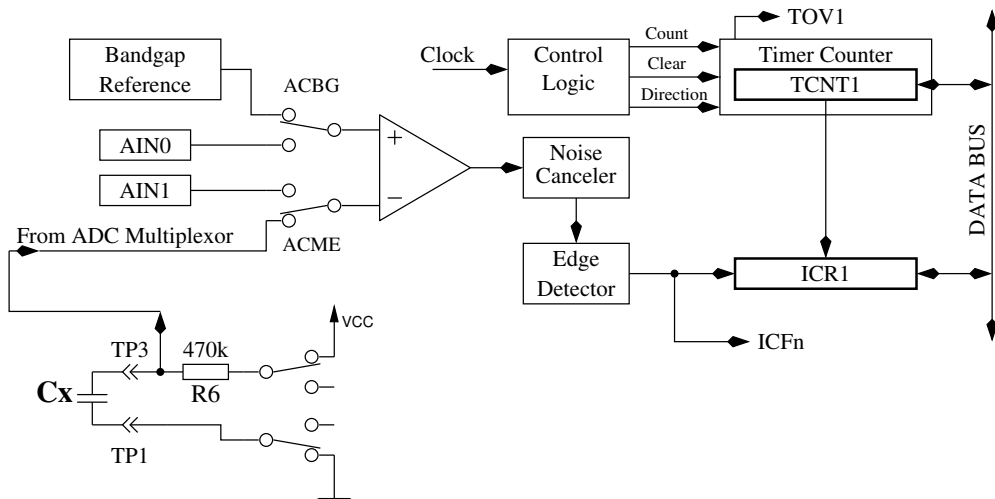


Abbildung 5.23: Messung kleiner Kapazitätswerte mit dem Komparator

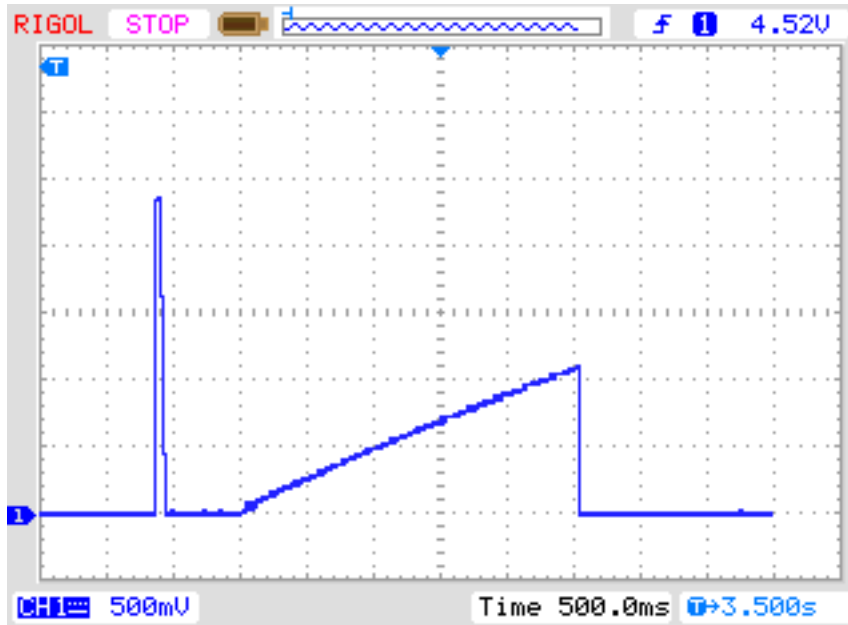


Abbildung 5.24: Laden und Entladen eines $22\mu F$ -Kondensators für die Messung

Im Prinzip kann diese Technik auch mit dem 680Ω -Widerstand benutzt werden, aber weil während dem Komparatorbetrieb der ADC nicht benutzt werden kann, besteht keine Möglichkeit die Ladespannung zu beobachten bis der Komparator angehalten wird. Wenn eine unentdeckte Diode mit dem Kondensator parallelgeschaltet ist, kann der Ladestrom von der Diode aufgenommen werden (Schwellspannung) und die Referenz-Spannung würde nie erreicht. Die Methode, die in der aktuellen Software für große Kondensatoren in Kapitel 5.3.2 verwendet wird, vermeidet diesen Konzeptfehler.

5.3.4 Messen des äquivalenten Serienwiderstandes ESR

Bei Kondensatoren mit mehr als etwa $2\mu F$ wird versucht, den Serienwiderstand von Kondensatoren zu messen. Der ESR [8] stellt zum Beispiel ein Maß für die Alterung von Elektrolyt-Kondensatoren dar. Die Abbildung 5.25 zeigt ein Ersatzschaltbild eines Kondensators. Der Widerstand R_p stellt den Isolationswiderstand des Kondensators dar, ESL die äquivalente Serieninduktivität und der Widerstand ESR stellt den äquivalenten Serienwiderstand dar.

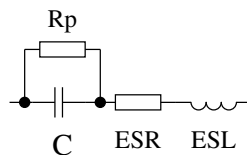


Abbildung 5.25: Ersatzschaltbild eines Kondensators

Genau genommen ist der ESR eines Kondensators abhängig von der Betriebsfrequenz und der Temperatur. Üblicherweise wird der mit einem sinusförmigen Signal bei $100kHz$ gemessene Wert in Datenblättern angegeben. Bei dieser Frequenz kann der ATmega ohne externe Beschaltung keine Messung durchführen. Das nachfolgend beschriebene Verfahren erreicht nur eine Messfrequenz von unter 680 Hz mit näherungsweise rechteckigem Signal. Um den äquivalenten Serienwiderstand zu bestimmen, wird der Kondensator zuerst in einer Richtung geladen und an beiden Anschlüssen die Spannung mit der internen Referenzspannung (1.1 V) gemessen. Nach der Messung wird der Ladestrom abgeschaltet und die Spannung am Kondensator ohne den Ladestrom gemessen. Wenn die Spannung am Kondensator ohne Ladestrom weniger als 3 mV beträgt, wird diese Messfolge wiederholt. Die Abbildung 5.26 zeigt die entsprechenden Schaltungen.

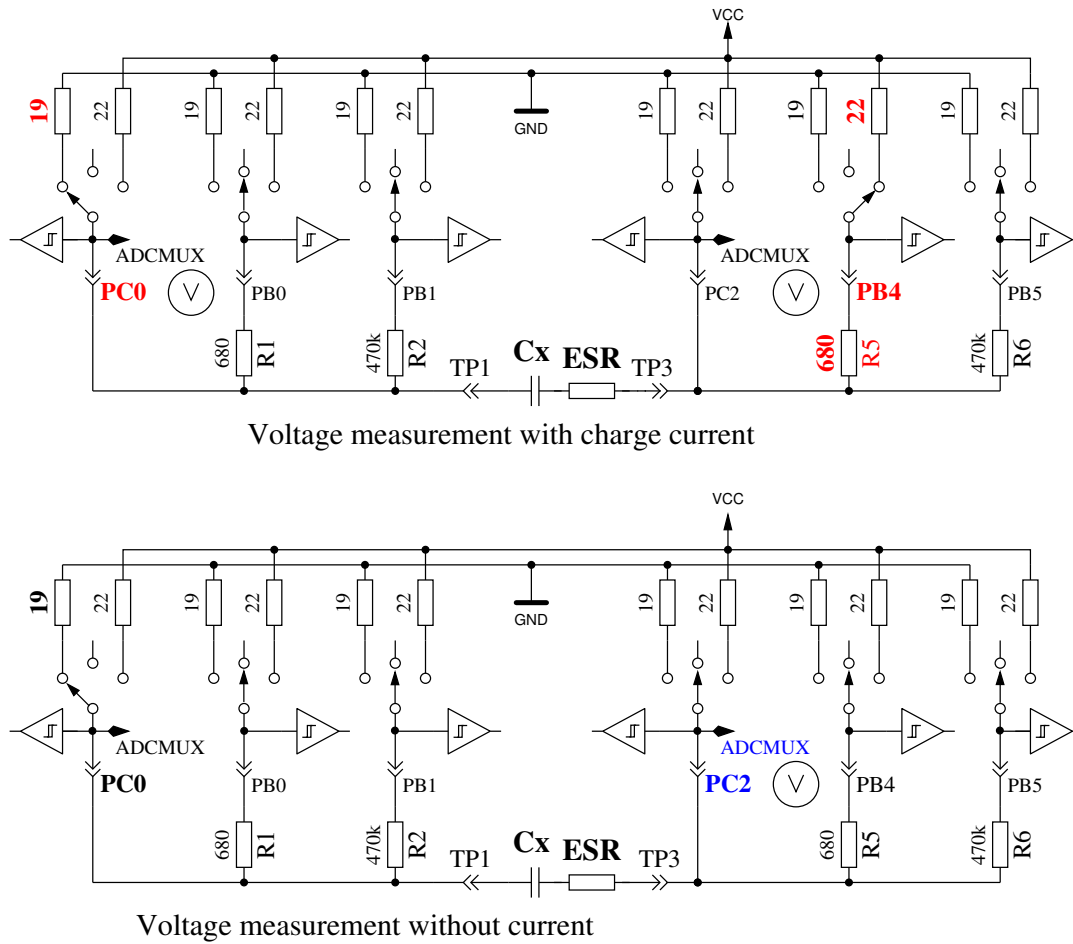


Abbildung 5.26: Schaltbild der ESR-Messung eines Kondensators

Die Differenz der Spannung am Kondensator mit und ohne Strom ist ein Maß für den internen Widerstand des Kondensators. Die zu erwartende Spannungsdifferenz ist allerdings so gering, dass mit einer Messung kein brauchbares Ergebnis erzielt werden kann. Aus diesem Grund wird danach der Strom umgekehrt und die Messung in der anderen Richtung wiederholt. Die gesamte Mess-Sequenz wird 128 Mal durchgeführt und die Ergebnisse der Spannungsmessungen addiert. Danach hat man drei Spannungssummen, die Spannung U_{lp} am Minuspol des Kondensators mit Strom, die Spannung U_{hp} am Pluspol des Kondensators mit Strom und die Spannung U_c am Pluspol des Kondensators ohne Strom. Die Spannungssumme am Minuspol des Kondensators repräsentiert den Spannungsabfall bei einem mittleren Ladestrom am Port-Ausgangswiderstand R_{port} . Aus der Differenz der Spannungssumme am Pluspol und Minuspol des Kondensators hat man ein Maß für die Spannung am Kondensator mit Ladestrom $U_{diff} = U_{hp} - U_{lp}$. Die Differenz $U_{esr} = U_{diff} - U_c$ soll jetzt den Spannungsabfall bei mittlerem Ladestrom am internen Widerstand des Kondensators repräsentieren. Der Widerstandswert wird aus dem Verhältnis dieser Spannung U_{esr} zu der Spannung U_{lp} skaliert mit dem bekannten Widerstandswert des Port-Ausgangs R_{port} . Dabei wird so skaliert, dass eine Widerstandsaufösung von 0.01Ω erreicht wird $R_{esr} = \frac{U_{esr} \cdot 10 \cdot R_{port}}{U_{lp}}$. Die Genauigkeit der ESR-Messung ist aus verschiedenen Gründen nicht sehr hoch:

1. Die Spannungsmessung an den beiden Kondensator-Anschlüssen kann nicht gleichzeitig sondern nur nacheinander durchgeführt werden. In der Zwischenzeit hat sich der Ladestrom durch den Ladevorgang des Kondensators geändert. Dies wird versucht mit einer kapazitätsabhängigen Korrektur der Minuspol-Spannung auszugleichen.
2. Der ADC nimmt den Spannungswert 1,5 Takte nach dem Start des Wandlungsvorgangs. Der Wandlungsvorgang beginnt mit der steigenden Flanke des ADC-Taktes, wenn das Startbit gesetzt

ist. Wenn der Ladestrom des Kondensators zu früh abgeschaltet wird, nimmt der ADC die falsche Spannung für die strombehaftete Messung auf. Wird der Ladestrom zu spät abgeschaltet, wird der Kondensator weiter geladen als es der strombehafteten Messung entspricht. Dann wird im stromlosen Zustand eine zu hohe Spannung gemessen. Es ist aber schwierig im Programm den genauen Zeitpunkt für die Stromabschaltung zu treffen. Um das Überladen auszugleichen, wird die im stromlosen Zustand gemessene Spannung um 2 % reduziert.

3. Der Port-Ausgangswiderstand wird bei dieser Messmethode als Referenz benutzt, dessen Widerstandwert ist aber auch nicht exakt bekannt.
4. Die Auflösung des ADC reicht nicht aus, um eine Widerstandsauflösung von 0.01Ω zu erreichen. Für alle Messungen wird die interne Spannungsreferenz (1.1 V) benutzt, um die best mögliche Auflösung zu verwenden. Zusätzlich wird versucht, den Auflösungs-Mangel durch eine große Zahl von Einzelmessungen abzumildern.
5. Mit der Abfrage des Fertig-Signals der ADC-Wandlung gelingt es nicht exakt, die Schaltzeiten der Port-Ausgänge mit dem ADC-Takt zu synchronisieren.

Trotz allen Schwierigkeiten scheinen die Ergebnisse brauchbar zu sein, wie die nachfolgende Abbildung 5.27 zeigt. Die ESR-Werte eines Bauteils gemessen mit dem Transistortester schwanken stärker als die Messungen des LCR-meters. Die Messergebnisse des LCR-Messgerätes wurden bei einer Frequenz von 1 kHz gemessen. Beim Transistortester muss auf die Qualität der Anschlüsse geachtet werden. Verwendete Anschlusskabel erhöhen den gemessenen Widerstandswert. Auch die Kontakte von Steckverbindern können die gemessenen Widerstandswerte erhöhen. Das LCR-Messgerät macht hier wegen den verwendeten Kelvin-Klemmen weniger Probleme.

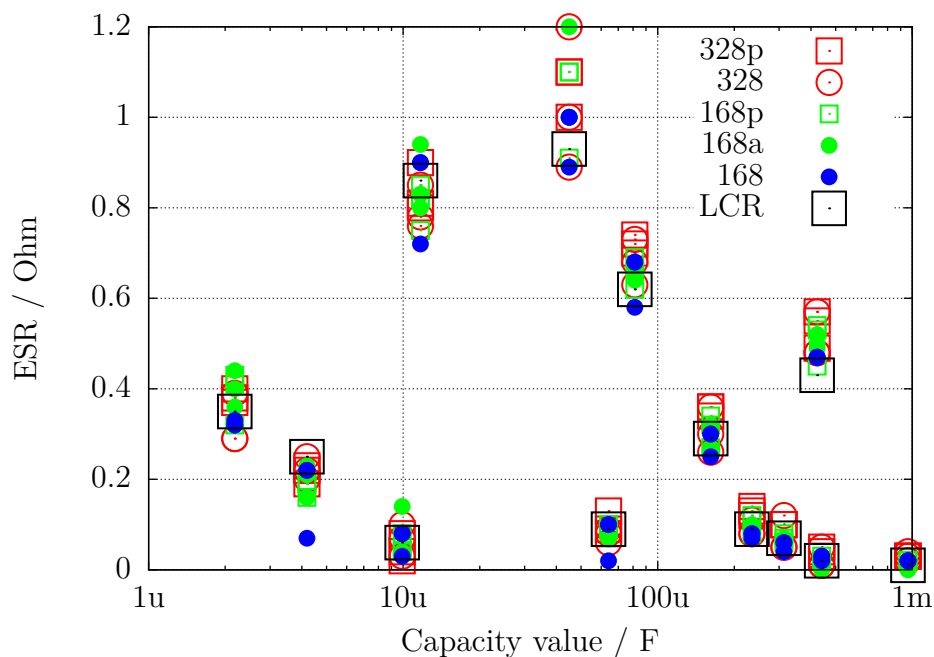


Abbildung 5.27: Messergebnisse der ESR-Messungen mit 15 verschiedenen ATmega

5.3.5 Ergebnisse der Kondensator-Messung

Die Ergebnisse meiner Messungen werden in Abbildung 5.28 für drei ATmega8 dargestellt. Zusätzlich sind einige Werte der Original-Software mit einem Korrekturfaktor von 0,88 (-12%) dargestellt. Weitere Ergebnisse von ATmega8 Versionen zeigen die Abbildungen 5.29a und 5.29b. Die Ergebnisse

der Messung der gleichen Kondensatoren mit einem ATmega168 werden in Abbildung 5.30 gezeigt. Die Referenz für die Fehlerberechnung sind die Messwerte eines PeakTech 2170 LCR-Messgerätes, nicht die aufgedruckten Werte der Bauteile. Die grösseren relativen Messfehler bei großen Kondensatoren liegen zum Teil an der zu hohen Messfrequenz (100 Hz) des LCR-Messgerätes für große Elektrolytkondensatoren, zum anderen spielt auch die schlechte Güte der Elektrolytkondensatoren eine Rolle.

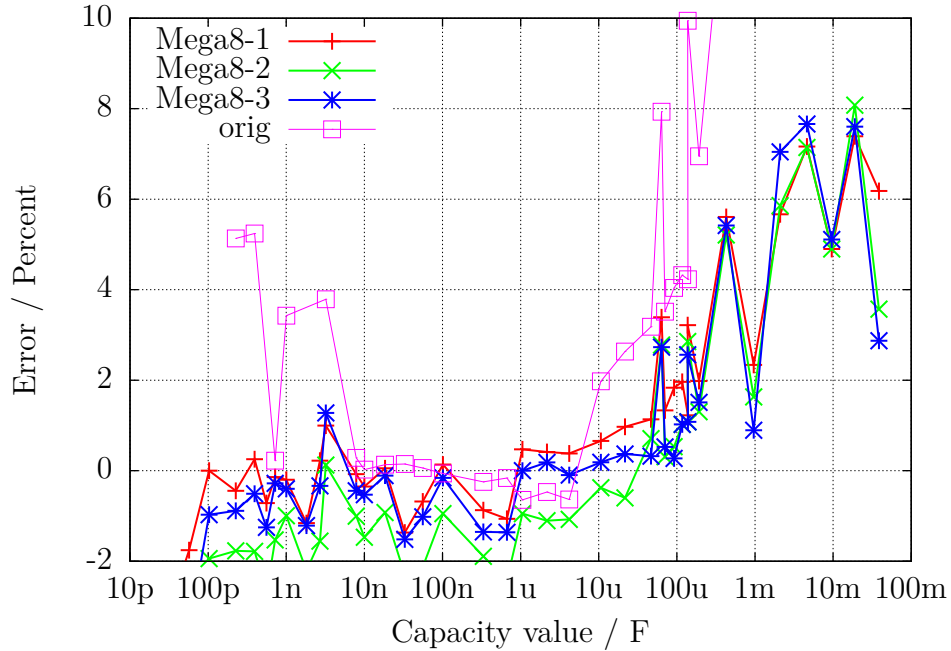
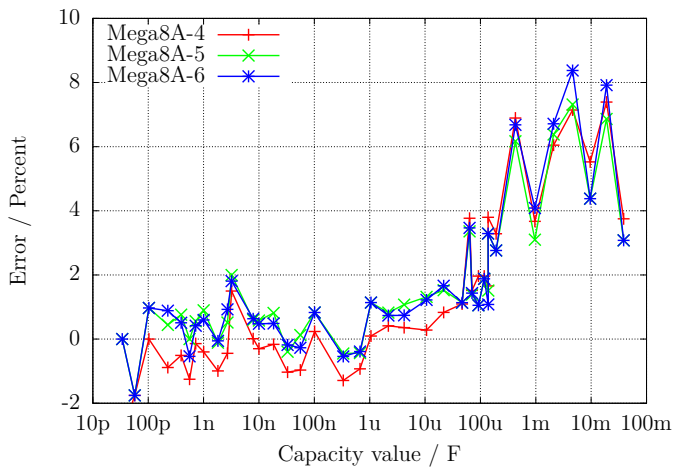
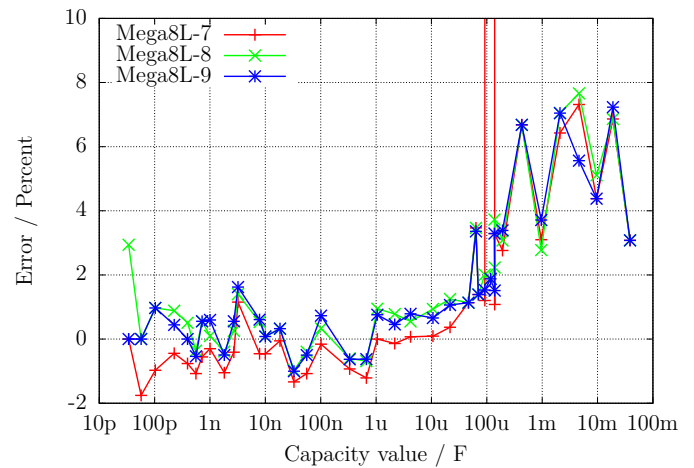


Abbildung 5.28: Prozentualer Fehler der Kondensator-Messungen mit drei ATmega8



(a) mit drei ATmega8A



(b) mit drei ATmega8L

Abbildung 5.29: Prozentualer Kondensator-Messfehler

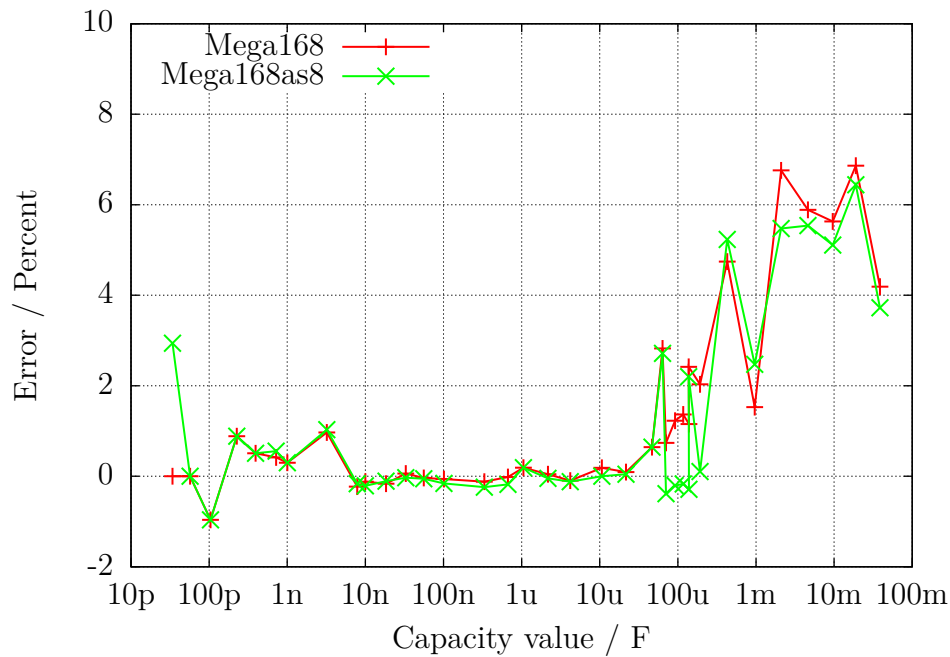


Abbildung 5.30: Prozentualer Fehler der Kondensator-Messungen mit dem ATmega168

Wie schwierig es ist, für Kondensatormessungen den richtigen Bezugswert zu finden, soll die Abbildung 5.31 zeigen. Als Bezugswert wurde hier eine beste Schätzung genommen. Die Kurve „Multimeter“ zeigt die Abweichungen, die mit einem Peaktech 3315 Multimeter gemessen wurden. Die nächste Kurve „LCR“ zeigt die Abweichungen, die mit einem Peaktech 2170 LCR-Meter in dem jeweils günstigsten Frequenzbereich gemessen wurden. Zum Vergleich werden mit der Kurve „ATmega168as“ auch noch die Messabweichungen eines ATmega168 bestücktem Transistor-Testers gezeigt. Ob die gezeigten Fehler aber tatsächliche Messfehler des jeweiligen Gerätes sind, muss bezweifelt werden, da auch die Schätzung des Kapazitätswert nicht der wirklichen Kapazität entspricht.

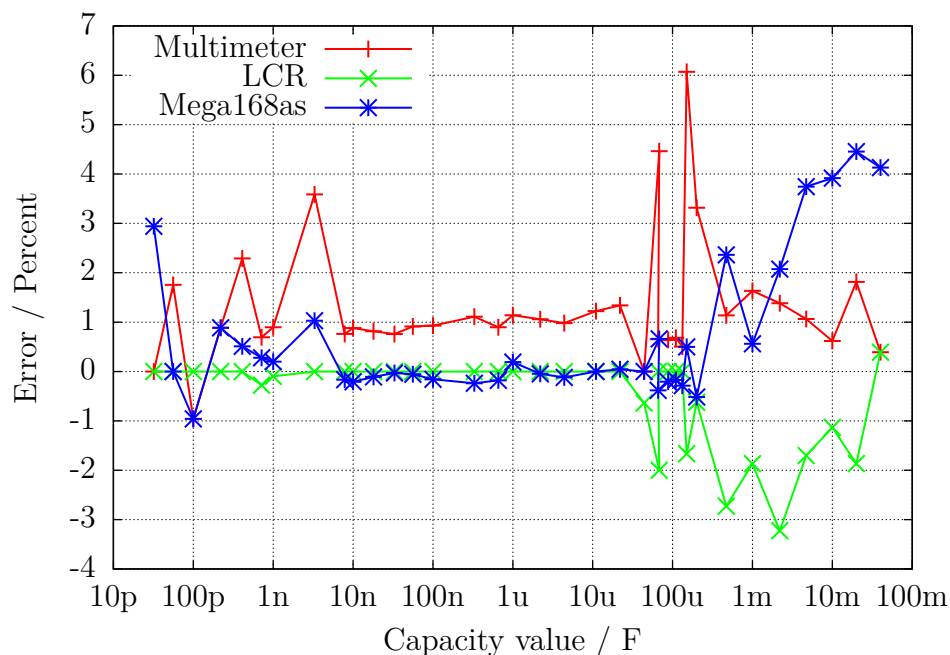


Abbildung 5.31: Vergleich der Kondensator-Messungen mit Multimeter, LCR-Meter und ATmega168

Die Abweichungen der Messergebnisse von drei verschiedenen ATmega168 werden in Abbil-

dung 5.32a dargestellt. Hier wurde die Messung des LCR-Meters als Vergleichsbasis genommen. Entsprechend werden die Messergebnisse von drei verschiedenen ATmega168A in Abbildung 5.32b, von drei verschiedenen ATmega168PA in Abbildung 5.33 und von drei verschiedenen ATmega328 in Abbildung 5.34a sowie von drei ATmega328P in Abbildung 5.34b gezeigt. Hierbei wurde nur der Nullwert der Kapazitätsmessung von 39pF berücksichtigt, alle anderen Korrekturmöglichkeiten wurden nicht benutzt. Dieser Nullwert beinhaltet schon die 2-3pF, die durch die etwa 12 cm langen Anschlussleitungen mit den Klemmen verursacht werden. Auch das Board-Layout hat Einfluss auf diesen Nullwert. Diesen Nullwert habe ich mit der Boardversion „DG2BRS V 5.2.1“ ermittelt.

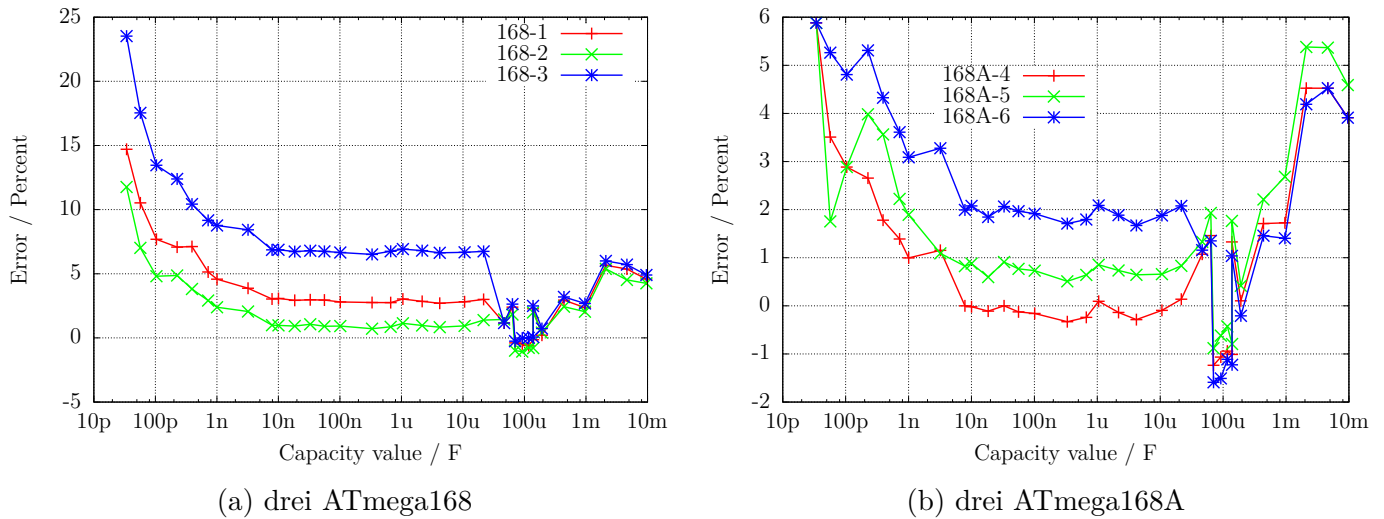


Abbildung 5.32: Kondensator-Messfehler, unkalibriert

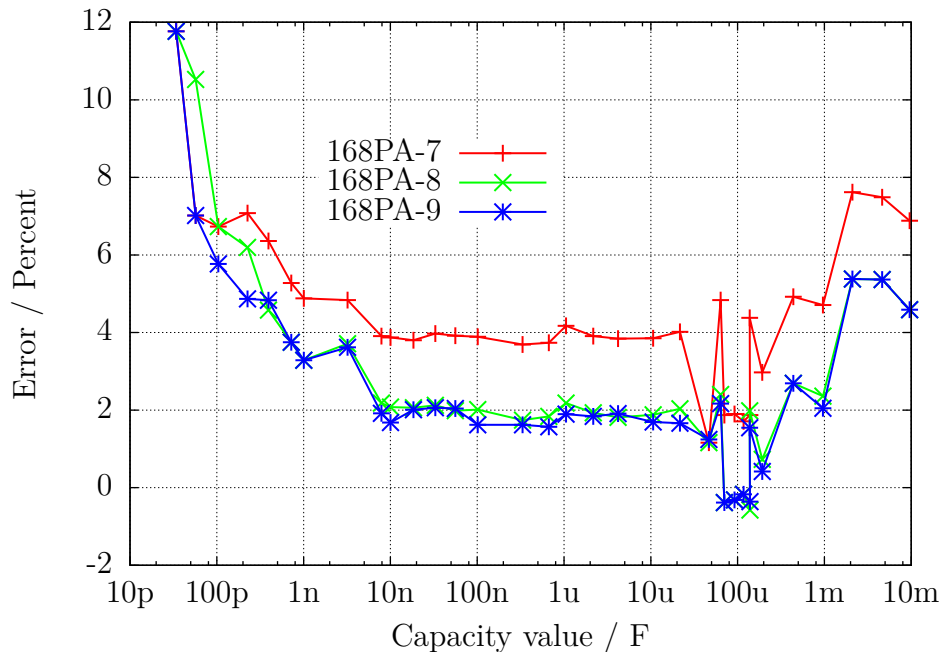
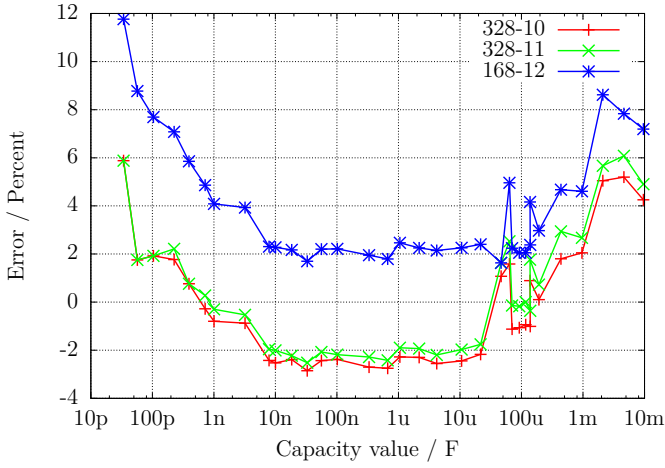
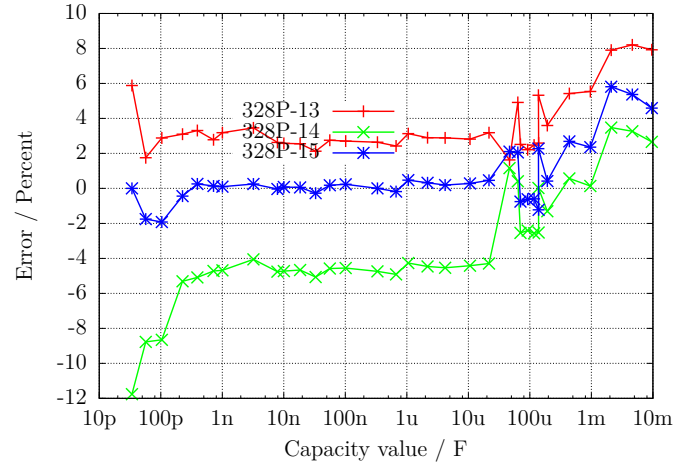


Abbildung 5.33: Kondensator-Messfehler von drei ATmega168PA, unkalibriert



(a) drei ATmega328



(b) drei ATmega328P

Abbildung 5.34: Kondensator-Messfehler, unkalibriert

Zum Erreichen der besten Messgenauigkeit muss die Software an die individuellen Eigenschaften des verwendeten ATmega angepasst werden. Dazu kann die Korrekturspannung REF_C_KORR für den Komparator angegeben werden, der für die Messung der kleinen Kapazitäten eingesetzt wird. Eine Korrekturspannung von 1 mV führt zu einer Verminderung der Kapazitätsanzeige von etwa 1,1 Promille. Bei einem automatischen Abgleich ist REF_C_KORR nur ein Offset zu der gemessenen Differenzspannung von geladenem Kondensator und der internen Referenz. Für die großen Kapazitäten kann der Promillewert C_H_KORR angegeben werden, um den die Messungen zu groß sind. Da die großen Kondensatoren meistens Elektrolytkondensatoren mit schlechter Güte sind, ist die Bestimmung des wahren Kapazitätswertes und damit auch des Messfehlers hier besonders schwierig. Besonders bei den ATmega168 habe ich bei kleinen Kapazitätswerten einen Messfehler beobachtet, der von der Anstiegsgeschwindigkeit der Spannung beim Ladevorgang abhängig ist. Die Abbildung 5.35 zeigt die Messabweichung bei Kondensatormessung nur mit der Berücksichtigung des Nullwertes (168-3-A), mit dem Korrekturfaktor für kleine Kondensatoren REF_C_KORR=66 sowie dem Korrekturfaktor für große Kondensatoren C_H_KORR=5 (168-3-B), sowie zusätzlich als Kurve 168-3-C mit der Berücksichtigung einer Spannungsanstiegskomponente (COMP_SLEW1=4000 und COMP_SLEW2=220) und des Selbstentladeverhaltens der großen Kondensatoren. Der Spannungsanstiegs-Korrekturfaktor berechnet sich nach $\frac{COMP_SLEW1}{cval+COMP_SLEW2} - \frac{COMP_SLEW1}{COMP_SLEW2}$, wobei cval der gemessene Kapazitätswert in pF ist.

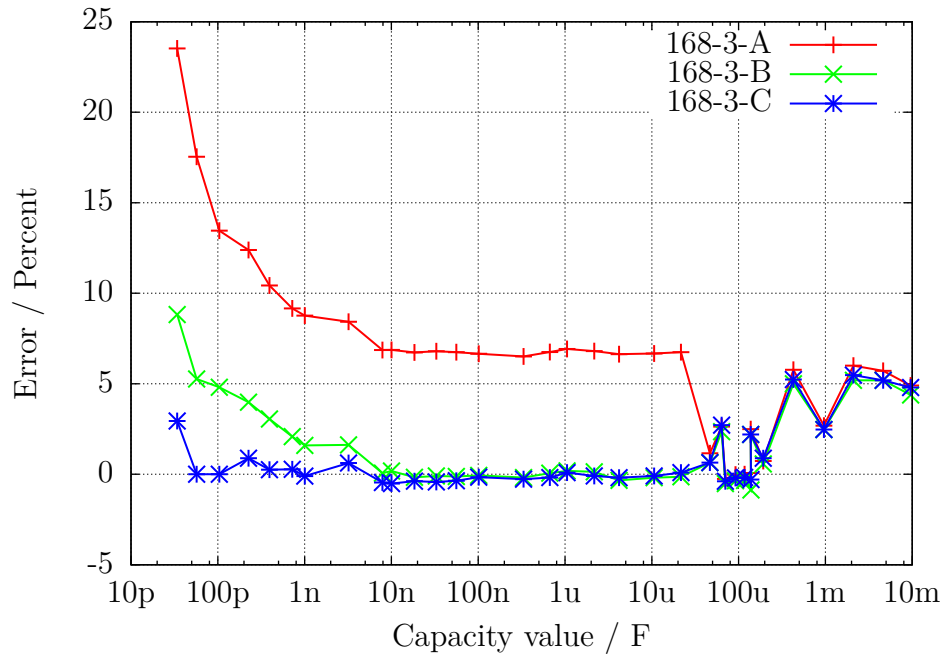


Abbildung 5.35: Optimierung der Kondensator-Messung eines ATmega168

5.3.6 Automatischer Abgleich der Kondensator-Messung

Der automatische Abgleich besteht aus zwei Teilen. Der erste Teil besteht darin, einen Nullabgleich der Kondensatormessung durchzuführen. Dazu wird in der Selbsttest-Routine der Mittelwert der gemessenen Kapazität bei nicht angeschlossenem Kondensator bestimmt. Es werden die Mittelwerte aus jeweils 8 Messungen für alle 6 Messkombinationen bestimmt. Nach erfolgreichem Abgleich werden die Korrekturwerte im EEprom festgehalten und für künftige Messungen verwendet. Schwieriger ist die Beseitigung der Exemplarstreuungen bei der Messung von Kondensatoren bis etwa $40\mu F$, wie sie in den Abbildungen 5.32a, 5.32b und 5.33 gezeigt wurden. Als wesentliche Ursache wurde das unterschiedliche Verhalten (Offset-Spannung) des analogen Komparators herausgefunden.

In Abbildung 5.36 werden die Daten von den neun untersuchten Prozessoren gezeigt. Die Punkte „diff2ref“ zeigen die Spannungsdifferenz, die sich nach dem Laden eines Kondensators mit $660nF$ zu der jeweiligen internen Referenzspannung ergibt. Idealerweise wäre diese Spannung immer Null, wenn der analoge Komparator rechtzeitig das Signal zum Beenden des Ladevorganges gegeben hätte. Die kurze Verwaltungszeit des ATmega sollte bei der relativ großen Kapazität zu keiner messbaren Spannungserhöhung geführt haben. Die Punkte „CapErr“ zeigen die aus den Abbildungen 5.32a, 5.32b und 5.33 geschätzten Messfehler der einzelnen ATmega-Exemplare in Promille. Auffällig ist, wie die „CapErr“ Punkte den „diff2ref“ Punkten folgen. Deshalb zeigen die Punkte „diff“ die Differenz zwischen den jeweiligen „CapErr“ und „diff2ref“ Punkten. Mit einem mittleren Wert für diff kann man also einen guten Schätzwert für die Korrektur der Kondensatormessung aus der Differenz der Kondensatorspannung nach dem Laden zur internen Referenzspannung berechnen. Im zweiten Teil der Abgleichprozedur muss also ein Kondensator hoher Güte zwischen Pin 1 und Pin 3 mit einer Kapazität zwischen $100nF$ und $20\mu F$ angeschlossen werden. Es sollte ein Folienkondensator, nach Möglichkeit kein keramischer Kondensator und schon gar kein Elektrolyt-Kondensator sein. Der Kapazitätswert braucht aber für diesen Abgleich nicht genau bekannt zu sein.

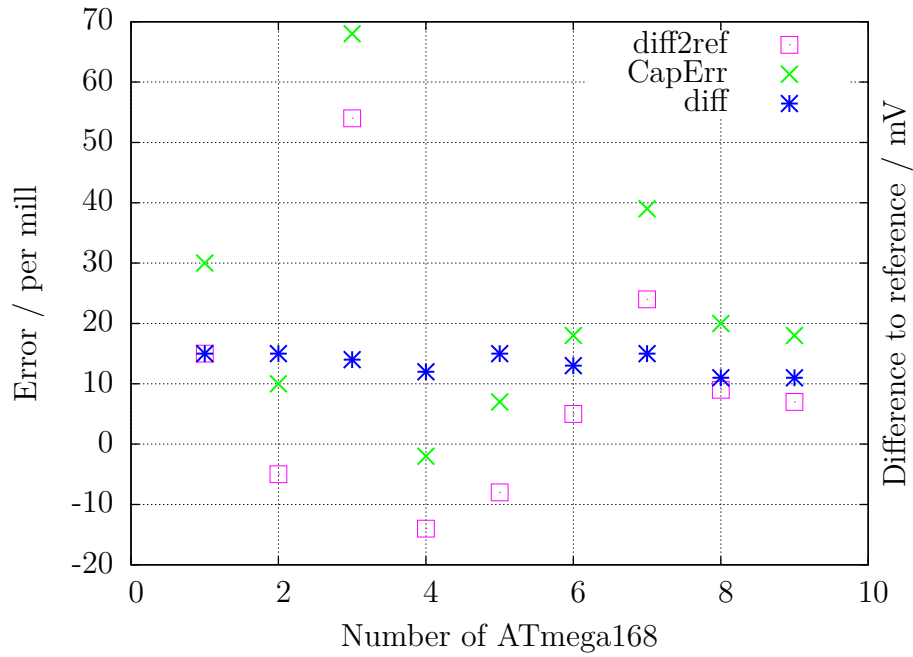
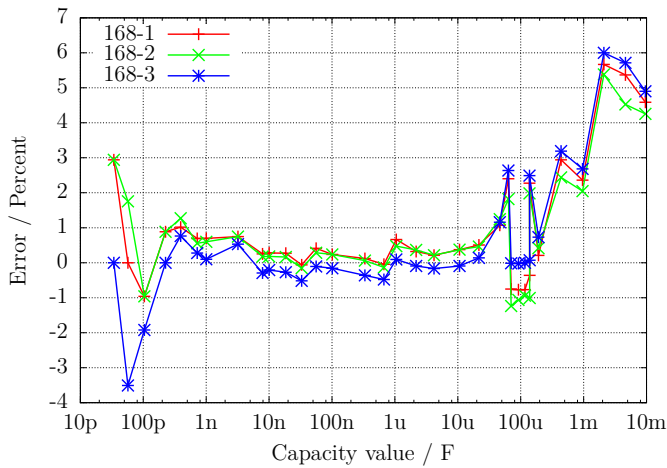
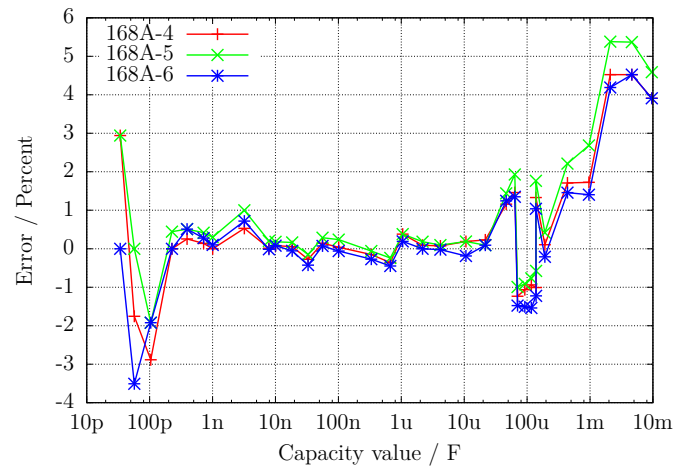


Abbildung 5.36: Daten von neun ATmega168-Prozessoren

Die Diagramme 5.37a, 5.37b, 5.38, 5.39a und 5.39b zeigen die Messergebnisse der gleichen Prozessoren in einer Standard-Softwarekonfiguration nach der Autokalibration. Die Prozessoren wurden alle mit der selben Software programmiert, lediglich für die ATmega168PA musste die Makefile wegen „PARTNO = m168p“ angepasst werden. Nach der Programmierung wurde bei jedem ATmega ein Selbsttest gestartet und bei Test 10 ein Kondensator mit $330nF$ an Pin 1 und Pin 3 angeschlossen.



(a) drei ATmega168



(b) drei ATmega168A

Abbildung 5.37: Kondensator-Messfehler, kalibriert

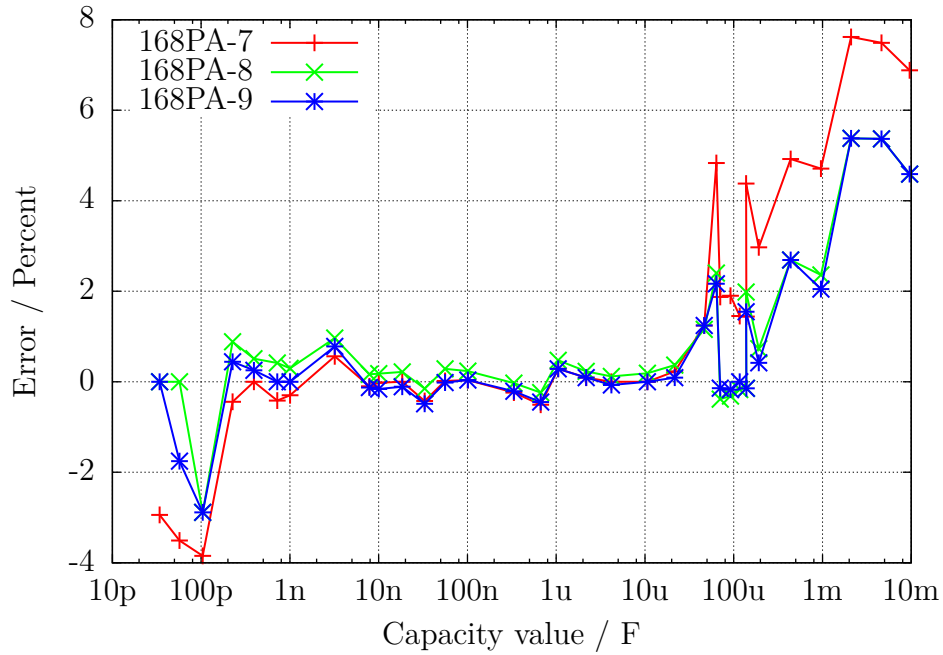
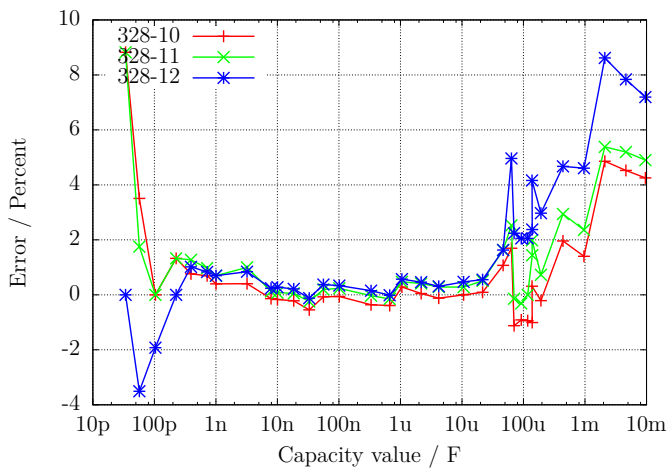
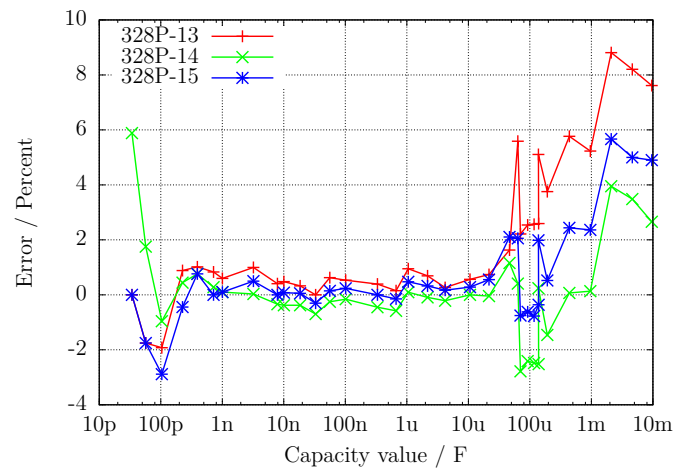


Abbildung 5.38: Kondensator-Messfehler von drei ATmega168PA, kalibriert



(a) drei ATmega328



(b) drei ATmega328P

Abbildung 5.39: Kondensator-Messfehler, kalibriert

Zuletzt will ich die Wirkung der AUTO_CAL Option im Selbsttest noch einmal verdeutlichen. Die folgende Diagramm 5.40 zeigt die Ergebnisse von drei ATmega Prozessoren mit der grössten Messabweichung noch einmal vor und nach der Kalibration. Die Punkte mit der Endung „unc“ zeigen die Messabweichungen ohne Kalibration. Die Linien mit der Endung „cal“ zeigen die Messabweichungen der gleichen Prozessoren mit der gleichen Software nach der Kalibration im Selbsttest-Zweig. Die Ursache der Messabweichungen für große Kondensatoren ($> 40\mu F$) ist noch nicht bekannt. Alle verwendeten Kondensatoren für diese Messreihe waren Folienkondensatoren oder keramische Kondensatoren ($56pF$, $100pF$ und $3,3nF$), keine Elkos.

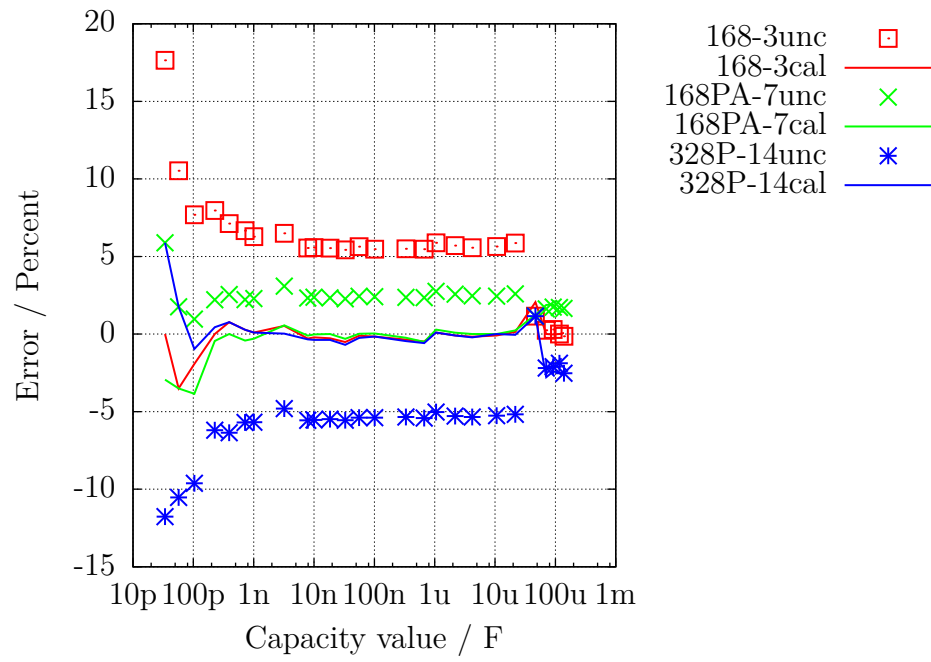


Abbildung 5.40: Kondensator-Messfehler von drei ATmega, vor und nach der Kalibration

5.4 Messen von Induktivitäten

Die Messung von Induktivitätswerten wird nach allen anderen Messungen als separater Teil mit allen gefundenen Widerständen mit weniger als $2100\ \Omega$ durchgeführt. Das Messverfahren beruht auf dem Prinzip, dass beim Schliessen des Stromkreises der Strom nach der Formel $I(t) = I_{max} \cdot (1 - \exp(-\frac{t}{\tau}))$ ansteigt. Die Zeitkonstante $\tau = \frac{L}{R}$ ist proportional zu der Induktivität L , aber umgekehrt proportional zum Widerstand R . Der Strom kann hier nur indirekt über den Spannungsabfall an einem Widerstand gemessen werden. Leider wird durch den relativ hohen Widerstand $680\ \Omega$ die Zeitkonstante zusätzlich verringert, was wiederum die Messung von kleinen Induktivitäten mit dem Takt von $8\ \text{MHz}$ zusätzlich erschwert. Um die Zeitkonstante zu bestimmen, wird die Spannung am $680\ \Omega$ -Widerstand als Stromsensor mit dem analogen Komparator überwacht. Wenn der Spannungsabfall am $680\ \Omega$ -Widerstand grösser als die Vergleichs-Spannung der internen Spannungsreferenz wird, meldet der Komparator dies an den beim Stromeinschalten gestarteten 16-Bit-Zähler weiter, der daraufhin den Zählerstand dieses Ereignisses festhält. Eventuelle Überläufe des Zählers werden vom Programm mitgezählt. Wenn die Spannung grösser ist, wird der Zähler sofort angehalten und aus dem festgehaltenen Zählerstand und dem Überlaufzähler die Gesamtzeit bestimmt. Der Anschluss der Spule wird wieder von VCC auf GND geschaltet, und über eine Spannungsüberwachung beider Anschlüsse gewartet, bis kein Strom mehr festgestellt wird. Das Schaltbild 5.41 zeigt ein vereinfachtes Diagramm der Messsituation.

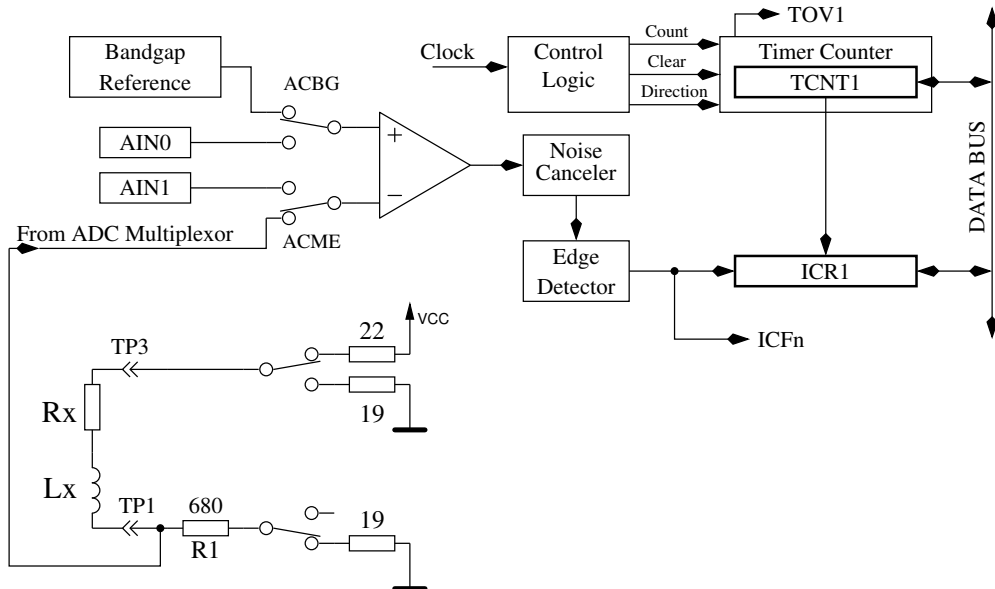


Abbildung 5.41: Messung von Induktivitäten mit dem Komparator

Aus der Versorgungsspannung VCC und der Summe aller Widerstände im Stromkreis kann der Maximalstrom I_{max} und daraus der Anteil der Vergleichsspannung im Verhältnis zur Maximalspannung am $680\ \Omega$ -Widerstand $U_{max} = I_{max} \cdot (680 + 19)$ bestimmt werden. Mit der Formel $L = -\frac{t \cdot R_{ges}}{\log(1 - \frac{U_{ref}}{U_{max}})}$ kann die Induktivität bestimmt werden. Der natürliche Logarithmus wird im Programm mit einer Tabelle ermittelt. Um auch kleinere Induktivitäten messen zu können, wird der $680\ \Omega$ -Widerstand im Stromkreis weggelassen, wenn der Widerstandswert der Spule kleiner $24\ \Omega$ gemessen wurde. Als Messwiderstand für die Strom-Messung dient in diesem Fall der Ausgangswiderstand der Ausgabeports ($19\ \Omega$). In diesem Fall wird der Spitzenstrom grösser als es die Spezifikation des ATmega erlaubt. Da das nur für eine sehr kurze Zeit passiert, erwarte ich keine Schäden. Um eine längere Zeitdauer mit überhöhtem Strom auszuschliessen, wird die zusätzliche Messung mit verzögertem Zählerstart immer mit $680\ \Omega$ -Widerstand durchgeführt. Um die Meßergebnisse an den tatsächlichen Induktivitätswert anzugleichen, wird vom Zählerstand ein Nulloffset von 6 abgezogen,

wenn ohne 680Ω gemessen wurde. Sonst wird ein Nulloffset von 8 berücksichtigt.

Bei großen Induktivitäten können parasitäre Kapazitäten den Strom so schnell ansteigen lassen, dass die Spannungsüberwachung mit dem Komparator sofort anspricht. Um dennoch die Induktivität bestimmen zu können, wird die gleiche Messung noch einmal gemacht, aber der Zähler etwas später gestartet, damit der Spannungsanstieg durch den Stromzuwachs der Induktivität und nicht die Stromspitze durch die Steukapazität gemessen wird. Die Messungen werden in beiden Stromrichtungen durchgeführt. Von den beiden Messungen in gleicher Stromrichtung wird das höhere Messergebnis verwendet. Von den Messungen in verschiedenen Stromrichtungen wird der kleinere Wert als Resultat der Induktivitätsmessung genommen.

5.4.1 Ergebnisse der Induktivitäts-Messungen

Die Abbildung 5.42 zeigt die Messergebnisse verschiedener Induktivitäten. Die Induktivitäten über $1H$ sind Relays und Primärwindungen von Netztrafos, die wegen der Remanenz des Eisenkerns schwierig zu messen sind.

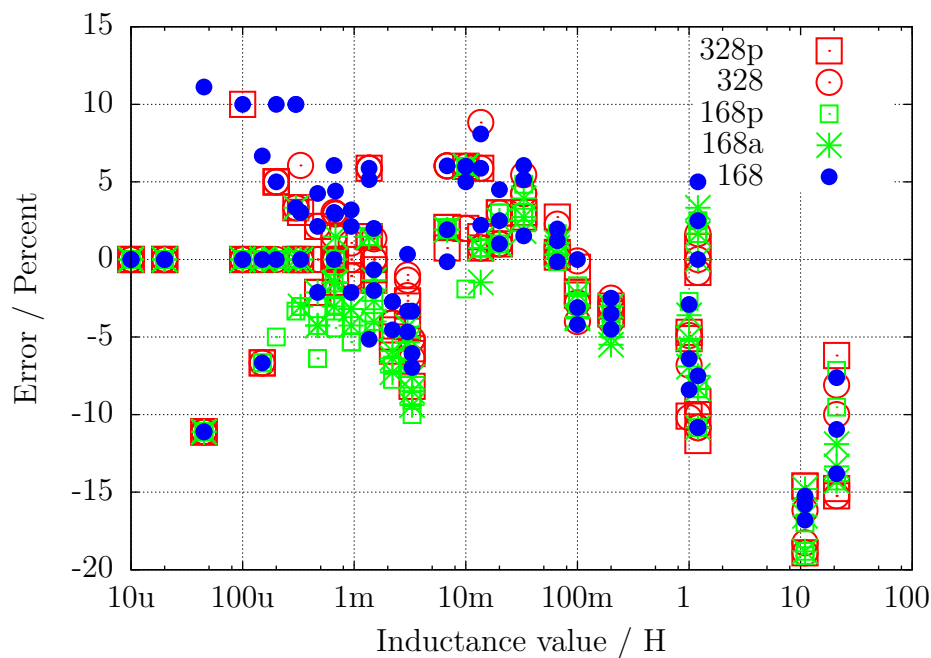


Abbildung 5.42: Induktivitäts-Messfehler von 15 verschiedenen ATmega

5.5 Selbsttest-Funktion

Beginnend mit Version 0.9k habe ich eine Selbsttest-Funktion eingebaut. Die Benutzung ist sehr einfach. Wenn Sie Prüfspitzen mit Klemmen verwenden, klemmen Sie alle Klemmen auf ein unisoliertes Stück Draht und drücken Sie den Start-Taster. Das Programm bemerkt die kurzgeschlossenen Klemmen und startet den Selbsttest. Nach Beenden des Selbsttestes wird der Tester mit normaler Messung fortfahren. Wenn kein Bauteil angeschlossen ist, wird der TransistorTester mit der Meldung „kein,unbek. oder defektes Bauteil“ enden. Man kann die Selbsttest-Funktion zusammen mit allen anderen Optionen nur für einen ATmega168 oder ATmega328 wählen. Die einzelnen Schritte der Selbsttestfunktion werden für Test 1 bis Test 6 in Zeile 1 des LC-Displays mit dem Buchstaben T gefolgt von der Schritt-Nummer dargestellt. Die Schritte 1 bis 6 wird viermal wiederholt, bevor das Programm zum nächsten Schritt geht. Aber wenn man den Start Taster gedrückt hält, wenn ein Durchlauf beendet wird, wird dieser Test nicht mehr wiederholt. Wenn man den Taster während dem gesamten Selbsttest gedrückt hält, wird jeder Schritt nur einmal ausgeführt.

Ohne die AUTO_CAL Option werden in jedem Schritt nur Messergebnisse dargestellt, es wird keine Fehleranalyse durchgeführt. Man muss die Ergebnisse selbst interpretieren. An dieser Stelle möchte ich noch einen wichtigen Hinweis geben. Führen Sie niemals Messungen mit eingestecktem ISP-Stecker durch! Die ISP-Schnittstelle stört die Messungen.

Hier ist die Liste der derzeit eingebauten Tests:

1. **Messung der 1,3V (oder 1,1V) Referenzspannung (band gap Reference).** In Zeile 1 wird der Text „Ref=“ und die gemessene Spannung in mV angezeigt. Für den ATmega8 sollte die gemessene Spannung nahe bei 1,3V liegen, für die anderen Prozessoren liegt die Referenzspannung normal um die 1,1V. Die zweite Zeile zeigt den resultierenden Faktor für die Kapazitäts-Messung mit dem $470k\Omega$ Widerstand.
2. **Vergleich der 680Ω Widerstände.** In Zeile 1 wird der kryptische Text „+RL- 12 13 23“ angezeigt. Das bedeutet: RL ist eine Kurzform für Resistor Low, was die 680Ω Widerstände meint. Die „12“ steht für: Widerstand am Pin 1 ist verbunden mit VCC (+) und Widerstand am Pin 2 ist verbunden mit GND (-). Das Ergebnis dieser Messung steht in Zeile 2 an erster Stelle als Differenz zum theoretischen Wert. In Zeile 1 folgt nun „13“, was bedeutet, dass der Widerstand von Pin 1 wieder nach VCC verbunden ist, aber dass jetzt der 680Ω Widerstand von Pin 3 mit GND verbunden ist. Das Ergebnis steht in Zeile 2 an der mittleren Stelle als Differenz zum theoretischen Wert.

Die letzte Messung dieses Tests „23“ bedeutet, dass nun der Widerstand von Pin 2 mit VCC verbunden ist und der Widerstand von Pin 3 mit GND verbunden ist. Das Ergebnis steht in der zweiten LCD-Zeile an letzter Stelle als Differenz zum theoretischen Wert. Ich möchte daran erinnern, dass die Auflösung des ADC etwa 4,88mV beträgt! Die Messsituation wird auch in Abbildung 5.43 dargestellt. Der theoretische Wert ergibt sich im Hinblick auf die internen Port-Widerstände wie folgt: $\frac{5001 \cdot (19+680)}{(19+680+680+22)} = 2493$.

8. **Messung des Innenwiderstandes von auf VCC geschalteten Port-Ausgängen.** Der benötigte Strom wird von auf GND geschalteten 680Ω-Widerständen geliefert. Es ist die gleiche Messung wie die in Test 7 zur anderen Seite, wie in Abbildung 5.45 gezeigt wird. Der Innenwiderstand wird wie folgt berechnet: $(5001 - (ErgebnisvonTest7) - (ErgebnisvonTest8))/680$. Die Widerstandswerte ergeben sich, wenn die gemessene Spannung durch diesen Strom dividiert wird. Der gefundene Mittelwert der Innenwiderstände der Ausgabe-Ports wird im EEprom festgehalten. Das Ergebnis für diesen Test wird dann in Zeile 1 mit dem Text „RI_Hi=“ in Ω angezeigt, der Innenwiderstand zur GND-Seite wird in Zeile 2 mit dem Text „RI_Lo=“ angezeigt. Diese Werte stehen dann für weitere Messungen zur Verfügung.

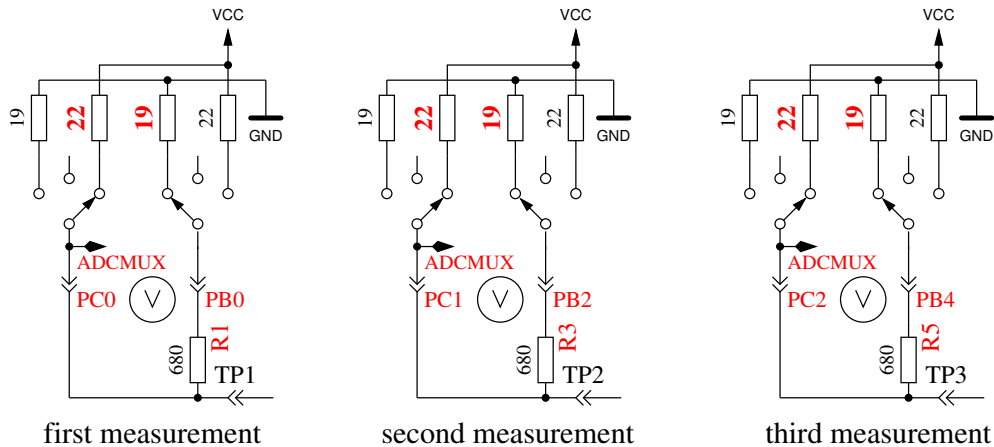


Abbildung 5.45: Messung des Innenwiderstandes des auf VCC geschalteten Port-C-Ausgängen.

9. **Messung des Nulloffsets der Kondensatormessung.** Für die Pin-Kombinationen 1:3, 2:3 und 1:2 wird der Nullwert der Kondensatormessung in pF ausgegeben. In der Software wird ein Vorgabewert von circa 39 pF für die normale Messwertausgabe berücksichtigt. Für die Ausgabe dieses Tests ist keine Korrektur berücksichtigt, es wird auch kein Nulloffset abgezogen. Die gefundenen Nulloffsets werden im EEprom festgehalten, wenn alle Null Offsets kleiner als 70pF sind. Die gefundene Null Offsets werden für die weiteren Kapazitäts-Messungen Pin abhängig berücksichtigt. Angezeigt wird die Ausgabe „C0“ in Zeile 1, gefolgt von den drei Null-Offsets für die Pin-Kombinationen 1:3, 2:3 und 1:2. Bitte beachten Sie, dass Veränderungen des Messaufbaus einen Neuabgleich sinnvoll machen. So kann sich der Nulloffset durch Verwenden von Kabeln mit Klemmen um circa 3 pF gegenüber einem leeren Sockel erhöhen.
10. **Warten auf den Anschluss eines Kondensators an Pin 1 und Pin 3.** Zur Vorbereitung der Messung des Spannungs-Offsets des analogen Komparators muss ein ausreichend großer Kondensator zwischen Pin 1 und Pin 3 angeschlossen werden. Es sollte ein Kondensator hoher Güte mit einer Kapazität zwischen 100nF und 20μF sein. Auf keinen Fall sollte man Elektrolyt-Kondensatoren verwenden.
11. **Messung des Komparatoroffsets für den Abgleich der Kondensatormessung.** Um den Offset des analogen Komparators zu bestimmen, muss ein Kondensator an Pin 1 und Pin 3 angeschlossen sein. Der Kondensator wird für die Pufferung der Ladespannung bei der Kondensatormessung benötigt, um den Unterschied der Ladespannung zur internen Referenzspannung bestimmen zu können. Bei erfolgreicher Messung wird der Korrekturwert kurz in Zeile 1 mit dem Text „REF_C=“ angezeigt und in den EEprom-Speicher geschrieben. Wenn Sie die AUTOSCALE_ADC-Option gewählt haben, wird die Verstärkung der ADC-Lesefunktion mit der internen Referenzspannung abgeglichen. Das geschieht durch den Vergleich der Kondensatorspannung von unter 1 V, einmal mit der VCC-Referenz gemessen und einmal mit der

internen Referenz. Die gefundene Differenz wird in Zeile 2 mit dem Text „REF_R=” angezeigt und ebenfalls im EEPROM festgehalten. Der REF_R_KORR-Wert ist dann nur ein zusätzlicher Offset für diese automatisch gefundene Differenz.

An Ende der Selbsttest-Funktion wird der Text „Auto Test End” in Zeile 1 und die Versionsnummer der Software in Zeile 2 angezeigt. Wenn gewählt, wird ein **50 Hz Rechtecksignal** auf Testpin 2 generiert und das gegenphasige Signal auf Testpin 3. Testpin 1 wird auf GND geschaltet. Der Strom für Testpin 2 und 3 wird mit 680Ω Widerständen begrenzt. Angezeigt wird das durch die Ausgabe von ”50Hz” am Ende der Zeile 1. Diese Ausgabe wird dreißig Mal mit jeweils 2 Sekunden Dauer durchgeführt. Man kann die Zeit der Verzögerungs-Aufrufe prüfen, wenn man ein Oszilloskop oder einen Frequenz-Zähler besitzt. Wenn man für die Takterzeugung keinen Quarz benutzt, können die Ergebnisse der Kondensatormessung ungenau sein. Eine genaue Taktfrequenz und genaue Wartezeiten sind wichtig für die Bestimmung von Kapazitätswerten. Die Ausgabe des 50Hz-Signals kann durch Gedrückthalten des Start-Tasters vorzeitig abgebrochen werden. Dann fährt das Programm mit der normalen Messfunktion fort.

Die Ergebnisse des Selbsttests der 9 verschiedenen ATmega168-Prozessoren und der 6 ATmega328-Prozessoren sind in den folgenden Abbildungen dargestellt.

Test No.	Messtyp	Idealwert	Abbildung
Test 1	band gap Ref	1100	5.46
Test 2	RL Mitte	0	5.47a
Test 3	RH Mitte	0	5.47b
Test 5	RH Low	0	5.48a
Test 6	RH High	0	5.48b
Test 7	R out Lo	131	5.49a
Test 8	R out Hi	151	5.49b
Test 9	Cap0	30	5.50
Test 9	Referenz-Korrektur	0	5.51

Tabelle 5.5: Verzeichnis der Selbsttest-Diagramme

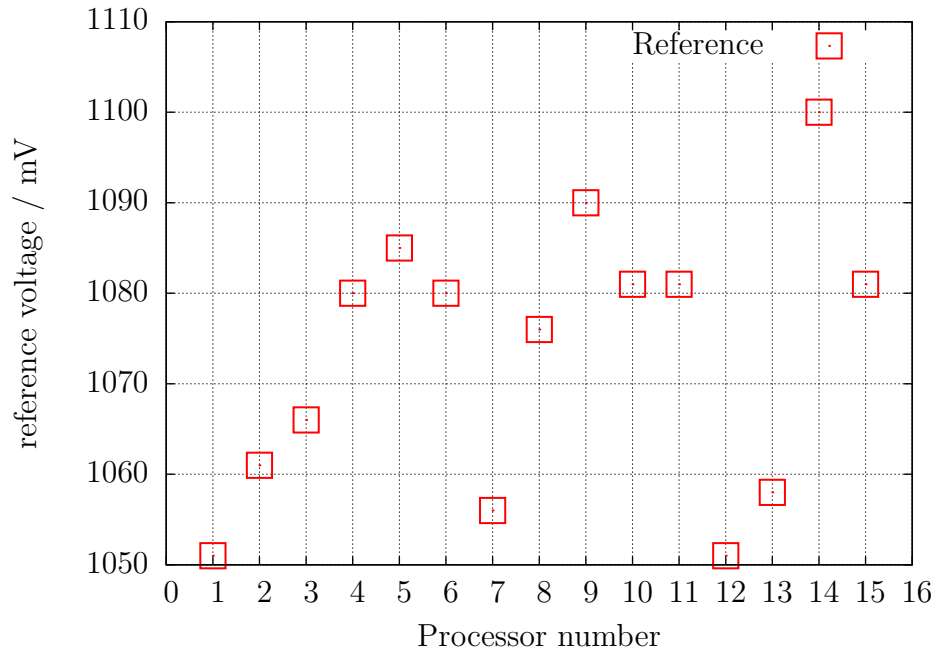
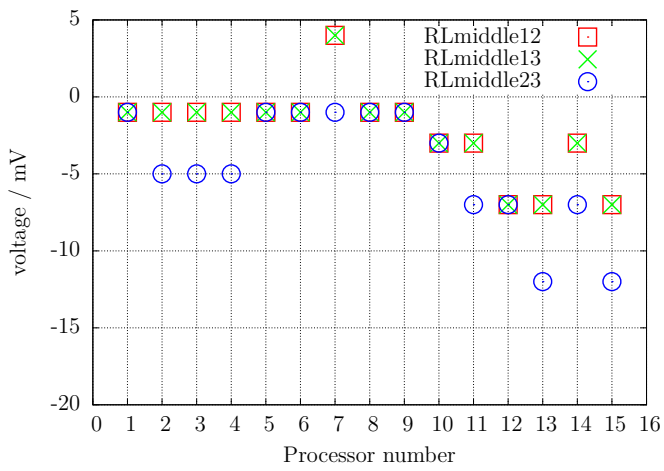
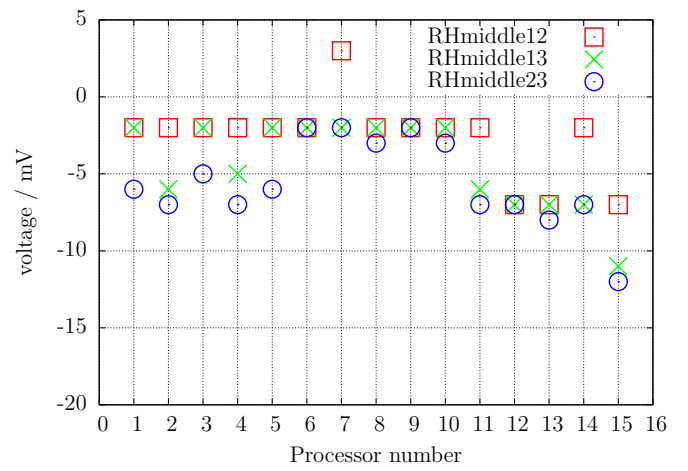


Abbildung 5.46: Selbsttest: Referenz-Spannungen

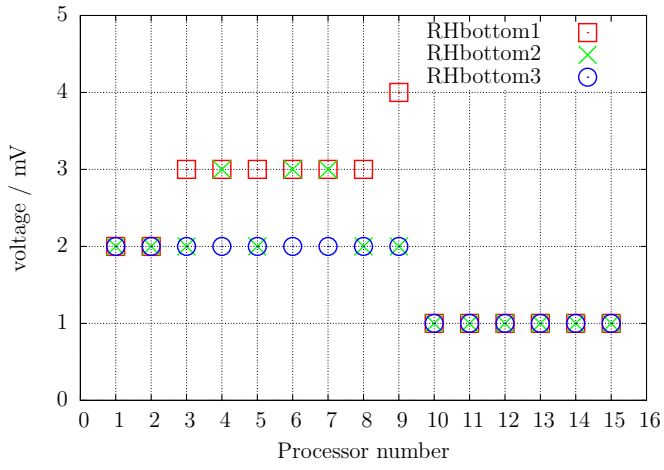


(a) mit 680Ω

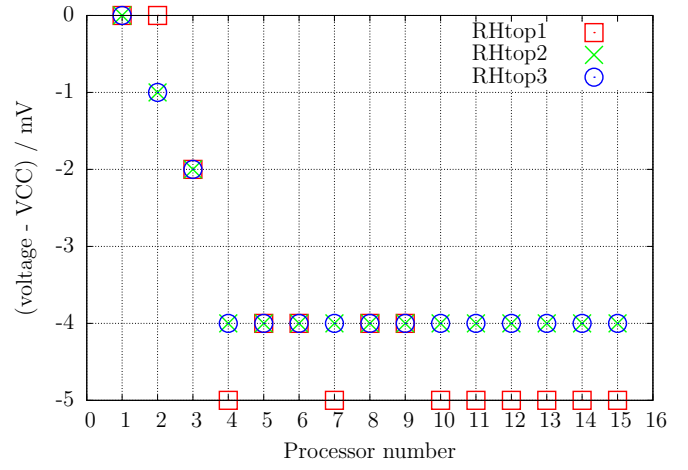


(b) mit 470kΩ

Abbildung 5.47: Selbsttest: Abweichung der Mittenspannung

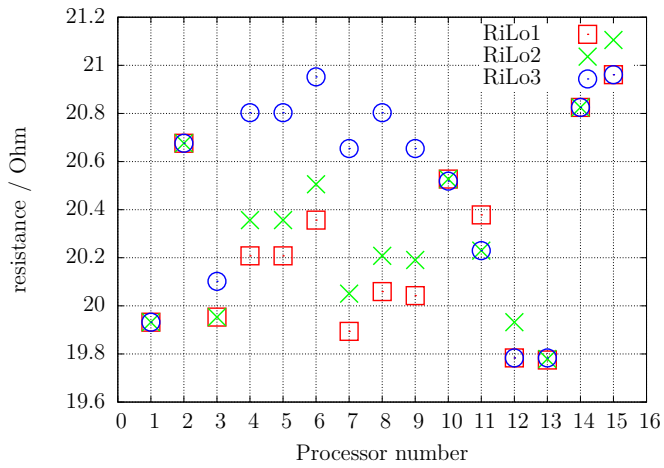


(a) mit $470k\Omega$ auf 0V

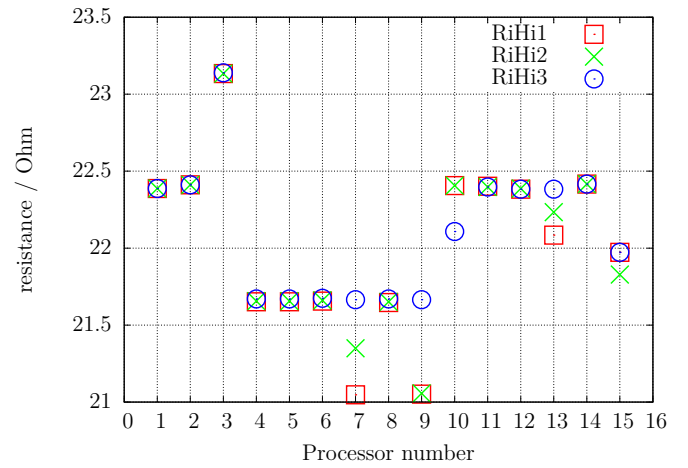


(b) mit $470k\Omega$ auf 5V

Abbildung 5.48: Selbsttest: Eingangsspannung



(a) mit 680Ω auf 5V



(b) mit 680Ω auf 0V

Abbildung 5.49: Selbsttest: Ausgangswiderstand

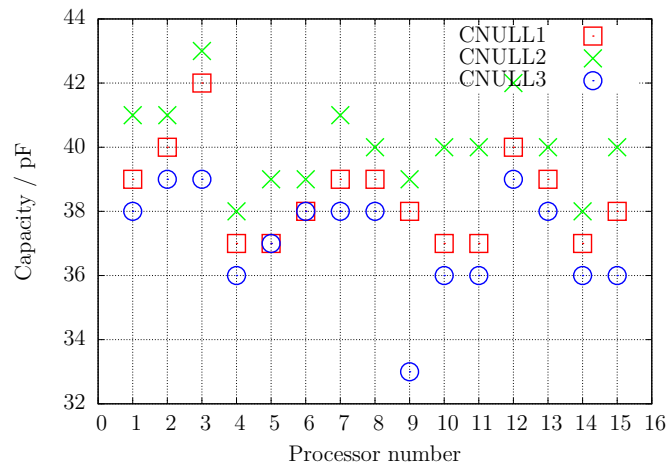


Abbildung 5.50: Selbsttest: Nullwert der Kapazitätsmessung

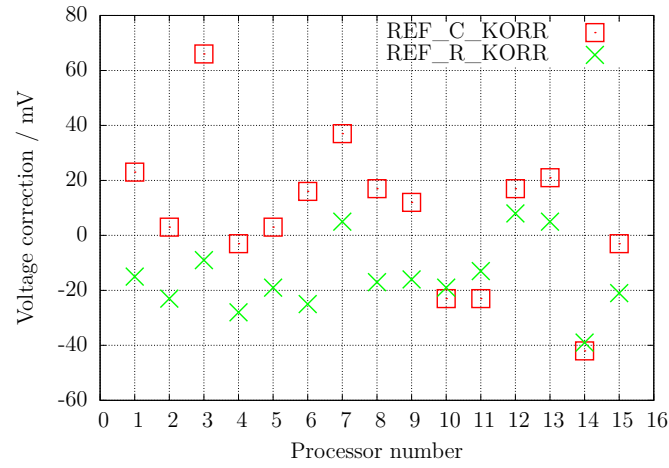


Abbildung 5.51: Selbsttest: Korrekturwerte nach automatischer Kalibration

Zuletzt möchte ich die Differenzen der am AREF-Pin mit einem Multimeter gemessenen Spannungen und der intern mit dem ADC gemessenen Referenzspannungen der 15 verschiedenen ATmega und die mit dem automatischen Abgleich gefundenen Korrekturspannung (REF_R_KORR) in Abbildung 5.52 gegenüberstellen. Sie können sehen, dass die Werte der automatischen Kalibration nahezu den extern gemessenen Referenz-Spannungswerten folgen.

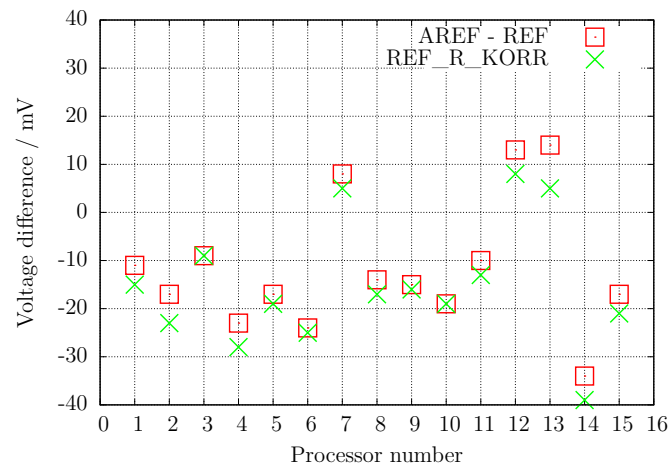


Abbildung 5.52: Selbsttest: Spannungsdifferenz der internen Referenz

Kapitel 6

Bekannte Fehler und ungelöste Probleme

Software-Version 1.02k

1. Germanium-Dioden (AC128) werden nicht in allen Fällen entdeckt.
2. Bei Leistungs-Doppeldioden vom Schottky-Typ wie MBR3045PT kann bei Anschluß einer Einzeldiode keine Kapazitätsmessung in Sperr-Richtung durchgeführt werden. Der Grund ist ein zu hoher Reststrom. Der Fehler kann manchmal durch Kühlen (Kältespray) umgangen werden.

Kapitel 7

Spezielle Softwareteile

Es wurden zahlreiche Veränderungen durchgeführt um Flashspeicher zu sparen. Die LCD-Ausgabe von Test-Anschlußnummern wurde in der Form „lcd_data('1'+Pin)“ durchgeführt. Um den Platz für die zusätzliche Addition für jeden Aufruf zu sparen, wurde die Funktion „lcd_testpin(uint8_t pin)“ in die Datei lcd_routines.c eingefügt.

Die Pseudofunktionen in der Form `_delay_ms(200)` sind keine Bibliotheksfunktionen sondern es werden für jede Aufrufstelle Warteschleifen in das Programm eingebaut. Das verbraucht viel Speicher, wenn man viele Aufrufe an unterschiedlichen Programmstellen hat. Alle solche Aufrufe wurden durch Aufrufe einer speziellen in Assembler-Sprache geschriebenen Bibliothek ersetzt, welche nur 74 Bytes des Flashspeichers (bei 8MHz) verbraucht, aber Aufrufe von `wait1us()` bis `wait5s()` in den Stufen 1,2,3,4,5,10,20... zur Verfügung stellt. Die Routinen enthalten den "Watch Dog Reset"-Befehl für alle Aufrufe über 50ms. Jeder Aufruf benötigt nur eine Anweisung (2 Byte). Warte-Aufrufe mit Zwischenwerten wie 8ms benötigen zwei Aufrufe (`wait5ms()` und `wait3ms()` oder zwei mal `wait4ms()` Aufrufe). Ich kenne keine Lösung, die ökonomischer wäre, wenn man viele Warteaufrufe im Programm benutzt. Die Aufrufe benutzen keine Register, nur der Stapelzeiger (Stack Pointer) wird für die Rückkehr-Adressen im RAM (maximal 28 Byte Stack-Platz) benutzt.

Die vollständige Liste der Funktionen ist:

`wait1us()`, `wait2us()`, `wait3us()`, `wait4us()`, `wait5us()`, `wait10us()`,
`wait20us()`, `wait30us()`, `wait30us()`, `wait40us()`, `wait50us()`, `wait100us()`,
`wait200us()`, `wait300us()`, `wait400us()`, `wait500us()`, `wait1ms()`,
`wait2ms()`, `wait3ms()`, `wait4ms()`, `wait5ms()`, `wait10ms()`,
`wait20ms()`, `wait30ms()`, `wait40ms()`, `wait50ms()`, `wait100ms()`,
`wait200ms()`, `wait300ms()`, `wait400ms`, `wait500ms()`, `wait1s()`,
`wait2s()`, `wait3s()`, `wait4s()` und `wait5s()`;

Das sind 36 Funktionen, die nur 37 Anweisungen inklusive dem Watch Dog Reset benutzen! Da gibt es keinen Weg diese Bibliothek zu verkürzen. Zu guter Letzt halten die Aufrufe die exakte Zeit ein, wenn der unterste Aufruf (`wait1us`) dies tut. Nur die Warte-Aufrufe über 50ms sind einen Takt pro 100ms zu lang wegen des zusätzlich eingebauten Watch Dog Reset.

Zusätzlich wurde die oft benutzte Folge „`wait5ms()`; `ReadADC(...)`;“ durch einen einzelnen Aufruf „`W5msReadADC(...)`;“ ersetzt. Dasselbe wurde auch für die Folge „`wait20ms()`; `ReadADC(...)`;“ gemacht, die durch den Aufruf „`W20msReadADC(...)`;“ ersetzt wurde. Die Funktion `ReadADC` wurde zusätzlich in Assembler portiert, so daß diese Erweiterung sehr effektiv integriert werden konnte. Eine funktionell identische C-Version der `ReadADC`-Funktion ist auch beigelegt.

Kapitel 8

Arbeitsliste und neue Ideen

1. Ergänze mehr und bessere Dokumentation.
2. Prüfe, ob der TransistorTester besser „interpolierte“ ADC-Werte erhalten kann, wenn zusätzliches Rauschen zum Signal oder zur Referenz hinzugefügt wird (siehe ATMEL-Dokument AVR121 [5]: Enhancing ADC resolution by oversampling). Wenn alle Werte gleich sind, kann keine Verbesserung der Auflösung durch Überabtasten stattfinden. Kann genügend Rauschen mit den ATmega-Zählern erzeugt werden? Natürlich kann diese Methode nicht alle ADC-Fehler beseitigen.
3. Diese Methode kann getestet werden, indem ein langsam ansteigendes Eingangssignal erzeugt wird und dieses Signal beobachtet wird. Das ansteigende Signal kann erzeugt werden indem ein großer Kondensator mit dem $470k\Omega$ -Widerstand geladen wird. Das Anwachsen der Spannung kann dann mit dem LC-Display in einem speziellen Teil des Selbsttests beobachtet werden.
4. Darüber nachdenken, wie man den wirklichen Innenwiderstand der Port-B-Ausgänge (Widerstands-Schalter) bestimmen kann, anstatt anzunehmen, dass die Ports gleich sind.
5. Kann das Entladen von Kondensatoren beschleunigt werden, wenn der Minus-Pol zusätzlich mit dem 680Ω - Widerstand nach VCC (+) geschaltet wird?
6. Prüfe, ob der Tester Fließkommadarstellung von Werten gebrauchen kann. Das Überlaufisiko (overflow) ist geringer. Man braucht keine Konstruktion mit Multiplikation oder Division, um einen Faktor mit einer gebrochenen Zahl nachzubilden. Aber ich weiß nicht, wieviel Platz für die Bibliothek gebraucht wird.
7. Schreibe eine Gebrauchsanweisung zum Konfigurieren des Testers mit den Makefile-Optionen und beschreibe den Ablauf bis zum fertigen Prozessor.
8. Wenn der Haltestrom eines Thyristors nicht mit dem 680Ω Widerstand erreicht werden kann, ist es ungefährlich für eine sehr kurze Zeit die Kathode direkt auf GND und die Anode direkt auf VCC zu schalten? Der Strom kann mehr als 100mA erreichen. Wird der Port beschädigt? Was ist mit der Spannungsversorgung (Spannungsregler)?
9. Prüfe die Ports nach dieser Aktion mit der Selbsttest-Funktion!
10. Können Spannungsregler geprüft werden? (Eingang, Ausgang, GND)
11. Können Optokoppler getestet werden?
12. Ausgabe einer Warnung, wenn die relativ zu VCC gemessene Referenz-Spannung nicht plausibel im Hinblick auf das ATmega-Modell ist.

13. Wie wäre es mit einem neuen Tester mit einem größeren ATmega, der differenzielle ADC-Ports hat, mehr Programmspeicher, ...? Es gibt keinen ATxmega, der eine Versorgungs-Spannung von 5V hat, nur die ATmega Linie ist möglich.
14. Idee für ein neues Projekt: USB-Version ohne LC-Display, Power vom USB-Port, Kommunikation zum PC über eine USB-Serial-Brücke.

Literaturverzeichnis

- [1] Markus Frejek *AVR-Transistortester*,. Embedded Projects Journal, 11. Ausgabe, 2011
- [2] Atmel Corporation *8-bit AVR with 8KBytes In-System Programmable Flash - ATmega8(L)*,. Manual, 2486Z-AVR-02/11, 2011
- [3] Atmel Corporation *8-bit AVR with 4/8/16/32KBytes In-System Programmable Flash - ATmega48 - ATmega328*,. Manual, 8271D-AVR-05/11, 2011
- [4] Atmel Corporation *Atmel AVR126: ADC of megaAVR in Single Ended Mode*,. Application Note, 8444A-AVR-10/11, 2011
- [5] Atmel Corporation *Atmel AVR121: Enhancing ADC resolution by oversampling*,. Application Note, 8003A-AVR-09/05, 2005
- [6] http://en.wikibooks.org/wiki/LaTeX_LaTeX_documentation,. Guide to the LaTeX markup language, 2012
- [7] http://en.wikibooks.org/wiki/Gnuplot_Gnuplot_documentation,. Documentation for the plotting tool gnuplot, 2012
- [8] Wikipedia http://de.wikipedia.org/wiki/Equivalent_Series_Resistance *Erläuterung zum ESR*. Normung und Ersatzschaltbild eines Kondensators, 2012
- [9] <http://www.xfig.org/userman> *Xfig documentation*,. Documentation of the interactive drawing tool xfig, 2009
- [10] http://docs.gimp.org/2.6/de_gimp_documentation. Documentation of the GNU Image Manipulation Program, 2010
- [11] <http://www.mikrocontroller.net/articles/AVR-Transistortester> *Online Dokumentation des Transistortester*, Online Article, 2009-2011
- [12] <http://www.mikrocontroller.net/articles/AVRDUDE> *Online Dokumentation des avrdude programmer interface*, Online Article, 2004-2011
- [13] <http://www.mikrocontroller.net/topic/131804> *Thread von Markus F.*, Forum thread, 2009
- [14] http://www.mikrocontroller.net/articles/AVR_Transistortester *Kurze Eigenschaftenbeschreibung des TransistorTester von Karl-Heinz K.*, Online Article, 2012
- [15] <http://www.mikrocontroller.net/topic/248078> *Thread von Karl-Heinz K.*, Thread und Software Versionen, 2012