Addition to the user manual for the AVR Transistor tester

K.-H. Kübbeler

is only valid for the test version 0.94k

Introduction:

Based to the Software of Markus Frejek I had started to modify the software.
At the beginning the reason for this was a problem, which I had with the programming of EEprom. Because writing to the flash memory was without problems, I believed that it is the quickest and best way to get a run capable tester, if I put the texts and parameters away from the EEprom and put them in the flash memory instead. Beginning from **version 0.92k you can select with option USE_EEPROM, if EEprom should be used or not.** By analysing the software I had an additional idea, which I have implemented as a test:
In order to display the voltage, values are needed in mV and not in steps of the ADC. This was done by original software for every value which is displayed. On the other side the original function ReadADC reads the ADC Value twenty times, adds every value and divide the sum by twenty. The resolution of the result is again 5000mV/1023 (same as ADC). If I read the ADC value instead of 20 times now 22 times, build the sum and double the sum and divide by nine, then my maximum value is 22*2*1023/9=5001, what matches nearly perfect to the wanted mV resolution. But with this idea real work started.
Now all "if statements" in the program must be adapted to the new resolution. So I learned more and more about the software of Markus Frejek. My actual version of the function ReadADC reads the ADC value 45 times, but adds only the last 44 values and divides the result by 9. Because I had the initial ambition to put the software of the reduced version (without resistor and capacitor measurement) in 4K flash, I did modifications which were not really necessary. So I replaced the wait loops by calls to a new written assembler routine, which uses only 66 bytes of flash, every call need only one instruction, but serves a total range from 10µs to 5s in steps of 1 2 3 4 5 10. The routine includes the Watch Dog Reset for all calls above 100ms.
Wait calls with interim value such as 8ms need two calls (5ms and 3ms). I don't know any implementation, which is more economical if you use many wait calls in your program. The wait calls matched the exactly delay time if the lowest time wait call does. Only the 100ms wait time calls are 1µs longer if your clock is 1MHz because of the additional Watch Dog Reset. The calls uses no registers, only the Stack Pointers for the return addresses in the RAM (at most 28 Byte stack space in current release) is used. The software version for 8MHz clock needs less memory than the 1MHz version, because the LCD-functions can then use this calls too. In this case the additional calls 1µs,2µs,3µs,4µs and 5µs are possible.

New features and modifications

1. Measuring of resistors are upgraded, so that potentiometers can be connected. The display format for this measurement is x-□-y-□-z in the first row, where x, y and z can be any of the pin numbers 1-3. The second row shows the value of the resistor which is connected to pin x and pin y followed by the value of the resistor connected to pin y and pin z. If you don't want to compute the total value yourself, disconnect pin y and start measurement again. If the potentiometer is adjusted to one of its ends, the Transistor tester cannot differ the middle pin and the end pin!

2. Capacitors are displayed with its symbol in the format:
    Pin number - symbol - Pin number.

3. The measurements of capacitor values are done by measurement of load time. The original software did this with a program loop, which reads the corresponding digital input pin until switch occurred and count the loop cycles. This has the handicap that the resolution is limited by the total time consumption of one loop cycle. I have replaced this loop for little capacitor values (about <50µF) by a technical feature of the AVR that the counter can save its counter value by a external event. The counter can operate at full clock rate (1MHz or 8MHz). The external event can be build by the output of the comparator. The comparator can operate with any ADC input pin and Band cap reference. So I discharge the capacitor, prepare the comparator to the proper pin input, start the counter at 0 and start charging of the capacitor with the 470kΩ resistor. Now I check in a program loop, if the counter flags signals a overflow event or a input capture (external) event. I count the overflow events until I detect the input capture event. In this case I stop the counter and check if I must count a additional overflow, because the counter can't be stopped by the input capture event. The input capture counter and the overflow counter built together the total time, from which I subtract a experimental find out constant to eliminate the measurement offset. I don't know, if this constant must be adapted to other boards or processors. Capacitors with bigger values as about 50µF are measured in an previous test. This is done by up to 500 load pulses with a length of 10ms, which is done with the 680Ω resistor. Load pulses are repeated until the load voltage reached more than 300mV, measured without any load current. The value of the capacity is then computed from the count of load pulses and the reached load voltage from a table. The table contains the factors to get the capacity in nF units from load time and the reached voltage with a spacing of 25mV. Interim value of voltage will be interpolated.
As a result of the lower load voltage the measurement time is much faster, because this advantage works also on discharging.
 Furthermore a diode, which is parallel connected to the capacitor don't disturb the measurement in most cases, because the flux voltage of most diodes is not reached.

4. Capacitors are measured only in three combinations: Pin 1 (-) and Pin 3 (+), Pin 1 and Pin 2, plus Pin 2 and Pin 3. The measurement with the other polarity are omitted.

5. Unfortunately the measurement of capacities had not the expected accuracy with respect to different AVR's, what I know from a single user's response. As a reason I assumed the difference of the internal 1,3V reference voltage, which was used with the comparator. Therefore the actual release can use a table with the theoretical dependency of the load time in respect to the comparator voltage. The table is spaced in 50mV steps and will be

interpolated according to the actual reference voltage. This function uses some of the limited flash memory. I hope that the reason for the different measurement values is found. Now the reference voltage is read by every power on if this function is selected with the option WITH_AUTO_REF. I noticed that the reference voltage is permanently somewhat to low, so that you can choose an offset with the Makefile option REF_KORR. The measured reference voltage will then be corrected (added) by your value (mV units).

6. Discharging of capacitors is changed. If the voltage is below 1300mV the capacitor is shortened by the output pins of the connected ADC port (Port C). I believe that this is legal because every output port has a built in resistance of about 20Ω. The data sheet Figure 149 (page 258) shows curves up to 2V. Of course I can not guaranty, that no damage can occur. I have tested the function with a 15mF Capacitor many times and I have never noticed any problem. The current should be below the specified limit of 40mA. Damage can occur if you don't discharge a (high voltage) capacitor before connecting it to your tester.

7. The format of displaying diodes is replaced by an diode symbol surrounded with the pin numbers. This should be known by every user with any first language. Unfortunately I see no way to show different symbols for special diodes like breakdown diode. Please notice the pin numbers instead. If the outside pin numbers of two diodes are identical, it can be a two-in-one LED, a breakdown diode or something else. Notice the flux voltages!

8. The measurement of single diodes is supplemented by a capacity measurement in inverse direction. Probably is is possible to select a diode for different purposes. I have measured values between some pF up to 5nF (base emitter diode of a BU508A Transistor). A experimental parallel connection of a diode and a capacitor with 330μF was detected correctly. If the capacitor has greater value, only the capacitor is detected. Of course this measurement can also be done for diodes of bipolar transistors if you connect the base and only one of the other terminals (collector or emitter).

9. To save program place, the format of displaying pin numbers of bipolar transistors is changed to the form EBC=xyz . Where EBC means Emitter, Base and Collector and xyz means the sequence of the corresponding pin numbers.

10. The measurement for bipolar transistors is also done with common collector (emitter follower). The hFE value of High Power transistors a more acceptable if measured by this way. In this kind of connection there is no risk to overload the base, even if the base is supplied with the 680Ω resistor. In case of Darlington transistor the voltage over the base resistor is too little, so this measurement is done also with the 470kΩ base resistor (release 0.92k). The normal measurement with common emitter is also done as before and the higher value of hFE is displayed. Because Darlington transistors can have very high hFE values (>20000), the display layout is changed from hFE= to B= to save space.

11. The differences of reference voltage of ATmega8, ATmega88, ATmega168 and ATmega328 are applied as noted in the data sheet. You can also use the feature WITH_AUTO_REF with a Makefile option, in this case the reference Voltage is read out with ADC. The factor to transform the measured load time to capacity is dependent of the reference voltage. If the curves in the data sheet are not only for one example, reading of actual reference voltage should not be required, but I'm in doubt about it.

12. The measurement of big capacitor values is expanded to values up to 100mF. Because I don't have such big capacitors, I could not yet test this. Capacity of Gold capacitors could not measured correctly by this method (too quickly).

13. Use of the AVR with 8MHz crystal is added to the Makefile. The correct fuses for the different ATmega's is selected with the „make fuses-crystal" call. If you prefer the operation with 1MHz, the clock divide by 8 fuse is selected automatically for the ATmega88 line. The ATmega8 must be connected with a 1MHz crystal, if you wish to use the 1MHz crystal mode (no clock divide).

14. Beginning from version 0.92k the program can be configured with Makefile options. You can select a supported language, if you wish to use measurement of resistors or capacities, if you want a serial output, if the battery voltage is displayed at the beginning, if you wish to use the self test function and if the program should use the EEprom. Additionally you can select your clock frequency (1MHz or 8MHz). You can also select which processor you have installed (m8, m48, m88, m168 and m328) and which programmers (avrisp2) you have connected if you use avrdude as programmer interface (call: make upload). The avrdude program checks signature before any load of program data is done. If the correct processor is not found, program terminates with a error message. You can select a clock generation with internal RC generator (make fuses) and a clock generation with a external crystal (make fuses-crystal). You can use the crystal version only if you have installed a 8MHz crystal between pin 9 and pin 10 of your ATmega. If you wish to use an 1MHz clock operation with crystal, you can do this. If you have a ATmega88 ATmega168 or ATmega328, the clock divide fuse is selected automatically by Makefile. Only for the ATmega8 you must install the 1MHz crystal if you wish to use a 1MHz clock operation with a crystal. Linux-user should be able to get an ready transistor tester with 4 steps:

    • Edit the Makefile, select your processor type, your programmer and options
    • call „make"
    • connect the ISP-plug and call „make upload"
    • if necessary, call „make fuses" or call „ make fuses-crystal" and disconnect the ISP-plug

15. Beginning from version 0.92k the ReadADC function adds 4 to the sum before dividing by 9 (round up to next integer).

16. Beginning from version 0.93k you can select the special option **AUTOSCALE_ADC** for the ReadADC function, so that not only the 5V reference is used for measurements, additionally the internal reference (1.1V for ATmega168) will be used if the input voltage enables that (Input voltage must be lower than 1V). Especially if the software switch the reference from 5V to 1.1V, the software must wait more than 5ms until the selected reference is stable. The reason for this is the external installed 100nF capacitor at the AREF pin (21) of the ATmega. Switch back to 5V is much quicker (300µs).
The necessary additional wait time results in a significant greater measurement period for capacitors with big values, because very often must be switched between 5V and 1.1V reference. The resolution of ReadADC function is always mV!

17. Version 0.94k has a new option in the Makefile „**NO_AREF_CAP**", which will reduce the wait time between switching the ADC reference between 1.1V and 5V. **You can use this option only if the 100nF capacitor connected to the AREF pin (21) is removed or replaced with a 1nF capacitor.** I have not noticed any degradation of measurements if this capacitor is removed. This option does have effect only, if the AUTOSCALE_ADC option is selected too. I have installed a 1nF capacitor after all.

18. Version 0.93k changes the measurement of resistors. The measurement of voltage of the directly switched pin (ADC port) is removed, only the Pin which is connected across a resistor to the power (VCC or GND) is measured. The potential drop of the direct connected pin is computed with respect to the known internal resistance of the switched port.
    Four different measurements are made:
    Low-Pin connected to GND and High-Pin connected across 680Ω to VCC
    Low-Pin connected to GND and High-Pin connected across 470kΩ to VCC
    Low-Pin connected across 680Ω to GND and High-Pin connected to VCC
    Low-Pin connected across 470kΩ to GND and High-Pin connected to VCC
    Corresponding to the measurement results, two of the measurements are selected (680Ω or 470kΩ pair) and the results are averaged. If the AUTOSCALE_ADC option is selected and the voltage of one result is below 0,99V, an weighted average is build with factor 4 for this value, the other value is weighted with factor 1.
     In table B2 you can see the benefit (measurements around 22kΩ).

19. In version 0.93k have I added fill characters to the different texts to get the same length for every implemented language. In version 0.94k I have removed this fill characters again, because I had noticed, that there is no space left in EEprom for the additional data required to implement the option LCD_CYRILLIC. To make this option selectable again, the overrun area of the table RLtab was additionally removed.

20. By optimizing the program in version 0.94k it is possible to select the self test function together with all other options without AUTOSCALE_ADC for a ATmega8. Therefore the output of capacity value and the output of resistor value now uses a common function.

21. The computation of the current amplification factor for bipolar transistors is in version 0.94k done with „long int" (32 bit) , but result is limited to 65535 (match 16 bit). The overrun problem was reason for the implausible results of previous versions (BC516 measurements).

I think, that optimising of measurement results is nearly finished. Some more response from users would be helpful.
I hope, that I have not forgotten any important item and will introduce a last item, for which I will spend a separate chapter.

Self test Function

Beginning with release 0.9k I have implemented a self test function. Usage is very simple. If you have installed test terminal with clamps, put all clamps together to a piece of uninsulated wire and press the start button. The program notice the shorten probes and start the self test function. After finishing the self test the transistor tester will continue with normal measurement, if no equipment is connected, the program will end with „part unknown or damaged". The unhappy side of the self test function is that the 8K flash is used near the limit. The length of the ATmega8 version 0.9k is about 8000 bytes. The length of the ATmega88 version is with 8122 byte very near at the limit. Use of EEprom (with option USE_EEPROM) is one of the only possibility to save memory. Some functions like ReadADC are already implemented in assembler syntax.

The separate steps of the self test function is generally displayed on row 1 of the LCD display with the letter **T** followed by the step number. Every step is repeated 8 times, before the program continues with the next step. In every step only measurement results are displayed, no error analysis are done, you must interpret the results yourself.

At this place I will give you an additional important hint. Never do a measurement with connected ISP plug!

The ISP interface influences the measurement.

Here are the **Test steps**:

1. Measurement of the 1.3V (or 1.1V) reference Voltage (Band gap Reference). In row 1 the text „Ref=" and the measured Voltage in mV is displayed. The second row shows the resulting factors for capacity measurement.

2. Comparing of the 680Ω resistors.
    In row 1 the cryptic text „**L1+2- .3- 2+.**" is shown. Meaning of this is as follows: The **L** is the symbol for Low meaning the 680Ω resistors. The **1+** stand for: resistor at pin 1 is connected to VCC (+). The following **2-** means: A GND (-) connected resistor at pin 2. The result of this measurement is displayed in row 2 at the first place.
    In row 1 follows now a „ **.3-**" which means, that the first connection of measurement 1 continues(.), but that the GND (-) connected resistor at pin 3 is now in action. The result is displayed in the middle place of row 2.
    The last measurement of this test „**2+.**" means that now the resistor at pin 2 is connected to VCC (+) and the GND connection of measurement 2 continues (.). The result of measurement is displayed at the last place of LCR row 2.
    Please remember, that the resolution of the ADC is about 4.88mV!
    All these combinations with respect to the internal resistance of the pins should result to:
    $5001 / (18+680+680+22) * (18+680) = 2493$.

3. Comparing of the 470kΩ resistors.
    Now the display shows in row 1 „**H1+2- .3- 2+.**". The same procedure as done in step 2 is repeated with the 470kΩ resistors (symbol **H**).
    Result should be nearly $5001 / (18 + 470000 + 470000 + 22) * (18 + 470000) = 2500$ for all combinations.

4. In this step nothing is measured, but the order is displayed „ isolate Probe“,
    which means that it is time to separate the probes (release from wire).

5. This step tests the capability of GND (-) connected 470kΩ resistors (H) to pull the test pins
    to GND . Row 1 shows the text „**RH-**“ .
    Row 2 should display zero for all three pins.

6. This step tests the capability of VCC (+) connected 470kΩ resistors (H) to pull the test pins
    to VCC (+). Row 1 shows the text „**RH+**“.
    The best value for this three measurements is 5001.
     Great differences from the best value for test 5 and 6 are errors such as isolation problem,
    flux material or damaged port.

7. Measuring of internal resistance of pin output switched to the GND signal.
    The text in the 1st LCD row is **Ri_Lo = (mV)**. In the second row of the LCD three voltages
    were displayed. The internal resistance of the port C outputs switched to GND (-) are
    measured with the current of to VCC (+) switched 680Ω resistors. Only the three pins of the
    ADC port are measured, the resistor port B (PB0,PB2 and PB4) can not be measured
    without hardware modification. Is is assumed that the port resistance of the different ports
    are nearly identical. To get the resistor values, you must divide the displayed mV values by
    about 7 (see test 8).

8. Measuring of internal resistance of port outputs switched to the VCC (+)signal.
    The needed current is generated with to GND connected 680Ω resistors .
    The text in the 1st LCD row is **Ri_Hi= (mV)**. In the second row of the LCD three voltages
    are displayed (in difference to VCC). It are the same measurements as those in test 7 to the
    other side.
    With the following steps you can get the resistance:
    To get the voltage of the 680Ω resistor: 5001 – (result of test 7) – (result of test 8)
    To get the current build: (voltage of 680Ω resistor) / 680
    Then you can get both resistor values by dividing the voltage (result of test 7 or 8) by the
    current.

9. A 50Hz rectangle signal is generated on Pin 2 and the same signal in opposite direction on
    Pin 3. Pin 1 is switched to GND . The current is limited with 680Ω resistors. This test is
    repeated 8 times with 5 seconds period each.
    You can check the time of the wait calls, if you have an oscilloscope or frequency counter.
    If you don't use the crystal clock version, the result may be inexactly.
    A exactly clock frequency and wait time are important for measurement of capacity values.

At the end of test function the text „**Auto Test End**“ is shown in row 1 and the version number of
software is shown in row 2. Then the program continues with the normal measurement task.
**Beginning from version 0.93k, a test step is not further repeated, if the start key is pressed. If
you leave the key pressed, every test is executed only once.** You can configure self test only
together with all other options for a ATmega168 (ATmega328 untested) . For the ATmega8 you
must at least omit the option AUTOSCALE_ADC because of the limited flash memory.

Appendix A
Result of self test
Software Version 0.94k

Microcontroller			1. Result	2. Result	3. Result
Mega8 @ 8MHz	Test 1 Reference Voltage should be 1298mV		Band gap Ref **1237**		
			RHfakt. 753	RLfakt. 4887	
Signature 1E 93 07 WITH_AUTO_REF	Test 2 comparison 680Ω best value: 2493		RL1+ RL2- **2488**	RL1+ RL3- **2488**	RL2+ RL3- **2484**
	Test 3 comparison 470kΩ best value: 2500		RH1+ RH2- **2493**	RH1+ RH3- **2493**	RH2+ RH3- **2493**
	isolate probe				
	Test 5 470kΩ (Isolation) best value: 0		RH1- **0**	RH2- **0**	RH3- **0**
	Test 6 470kΩ (Isolation) best value: 5001		RH1+ **4995**	RH2+ **4995**	RH3+ **4995**
	Test 7 Pin resistance Low		TP1- RL1+ **132**	TP2- RL2+ **132**	TP3- RL3+ **137**
	Test 8 Pin resistance High (VCC - voltage)=		TP2+ RL1- **151**	TP2- RL2+ **151**	TP3- TP3+ **151**
Mega168 @ 8MHz	Test 1 Reference voltage should be 1102mV		Band gap Ref **1090**mV		
			RHfakt. 865	RLfakt. 5649	
Signature 1E 94 06	Test 2 comparison 680Ω		RL1+ RL2- **2493**	RL1+ RL3- **2493**	RL2+ RL3- **2493**
	Test 3 comparison 470kΩ		RH1+ RH2- **2497**	RH1+ RH3- **2498**	RH2+ RH3- **2498**
	isolate probe				
	Test 5 470kΩ (Isolation)		RH1- **0**	RH2- **0**	RH3- **0**
	Test 6 470kΩ (Isolation)		RH1+ **4998**	RH2+ **4998**	RH3+ **4998**
	Test 7 Pin resistance Low		TP1- RL1+ **131**	TP2- RL2+ **132**	TP3- RL3+ **132**
	Test 8 Pin resistance High (VCC - Voltage)=		TP2+ RL1- **156**	TP2- RL2+ **156**	TP3- TP3+ **156**

Appendix B1
Results of resistor measurements
Software Version 0.92k (old)

resistor	Result of Mega8 @ 1MHz, original software Signature 1E 93 07		Result of Mega8 @ 8MHz, Signature 1E 93 07		Result of Mega88 @ 1MHz Signature 1E 93 0A	
60Ω 0.1%	**61Ω**	1.7%	**62Ω**	3.3%	**60Ω**	0%
120Ω 0.1%	**120Ω**	0%	**122Ω**	1.6%	**120Ω**	0%
240Ω 0.1%	**240Ω**	0%	**240Ω**	0%	**239Ω**	0.4%
340Ω 0.1%	**339Ω**	0.3%	**339Ω**	0.3%	**340Ω**	0%
680Ω 0.1%	**677Ω**	0.4%	**677Ω**	0.4%	**679Ω**	0.1%
1360Ω 0.1%	**1357kΩ**	0.2%	**1352Ω**	0.6%	**1361Ω**	0.1%
3.90kΩ 0.1%	**3865Ω**	0.9%	**3859Ω**	1.1%	**3896Ω**	0.1%
7.80kΩ 0.1%	**7676Ω**	1.6%	**7652Ω**	1.9%	**7776Ω**	0.3%
11.0kΩ 0.1%	**10902Ω**	0.9%	**10.70kΩ**	2.7%	**10.91kΩ**	0.8%
22.0kΩ 0.1%	**21.1kΩ**	4.1%	**21.1kΩ**	4.1%	**21.1kΩ**	4.1%
44.0kΩ 0.1%	**43.1kΩ**	2.0%	**43.0kΩ**	2.3%	**43.0kΩ**	2.3%
50kΩ 0.1%	**49.2kΩ**	1.6%	**49.2kΩ**	1.6%	**49.2kΩ**	1.6%
100kΩ 0.1%	**99.0kΩ**	1.0%	**98.9kΩ**	1.1%	**98.9kΩ**	1.1%
200kΩ 0.1%	**196.8kΩ**	1.6%	**197.7kΩ**	1.2%	**198.6kΩ**	0.7%
270kΩ 0.1%	**266.3kΩ**	1.4%	**267.5kΩ**	0.9%	**268.6kΩ**	0.5%
470kΩ 0.1%	**77.46µF ?**		**467.1kΩ**	0.6%	**469.0kΩ**	0.2%
940kΩ 0.1%	**919.6kΩ**	2.2%	**935.7kΩ**	0.5%	**940.0kΩ**	0%
1.00MΩ 0.1%	**973.8kΩ**	2.6%	**990.8kΩ**	0.9%	**996.2kΩ**	0.4%
2.00MΩ 0.1%	**1922.0kΩ**	3.9%	**1.975MΩ**	1.3%	**1.996MΩ**	0.2%
10MΩ 1%	**8433.8kΩ**	15.7%	**9.574MΩ**	4%	**10.11MΩ**	1%
additional: 50MΩ 1%	**----**		**43.87MΩ**	12%	**52.94MΩ**	6%

Appendix B2
Results of resistor measurements
Software Version 0.94k

Resistor	Result of Mega8 @ 8MHz Signature 1E 93 07		Result of Mega8 @ 8MHz Signature 1E 93 07 AUTOSCALE_ADC		Result of Mega168@8MHz Signature 94 06		Result of Mega168@8MHz Signature1E 94 06 AUTOSCALE_ADC	
60Ω 0.1%	**60Ω**	0%	**61Ω**	1.7%	**60Ω**	0%	**60Ω**	0%
120Ω 0.1%	**120Ω**	0%	**120Ω**	0%	**120Ω**	0%	**120Ω**	0%
240Ω 0.1%	**240Ω**	0%	**240Ω**	0%	**240Ω**	0%	**239Ω**	0.4%
340Ω 0.1%	**339Ω**	0.3%	**339Ω**	0.3%	**340Ω**	0%	**340Ω**	0%
680Ω 0.1%	**678Ω**	0.3%	**678Ω**	0.3%	**681Ω**	0.1%	**679Ω**	0.1%
1360Ω 0.1%	**1359Ω**	0.1%	**1358Ω**	0.1%	**1359Ω**	0.1%	**1359Ω**	0.1%
3.90kΩ 0.1%	**3894Ω**	0.2%	**3895Ω**	0.1%	**3909Ω**	0.2%	**3903Ω**	0.1%
7.80kΩ 0.1%	**7813Ω**	0.2%	**7782Ω**	0.2%	**7828Ω**	0.4%	**7808Ω**	0.1%
11.0kΩ 0.1%	**11.04kΩ**	0.4%	**10.96kΩ**	0.4%	**11.08kΩ**	0.7%	**11.02kΩ**	0.2%
22.0kΩ 0.1%	**21.30kΩ**	3.2%	**21.60kΩ**	1.8%	**21.30kΩ**	3.2%	**21.7kΩ**	1.4%
44.0kΩ 0.1%	**43.10kΩ**	2.0%	**43.40kΩ**	1.4%	**43.30kΩ**	1.6%	**43.6kΩ**	0.9%
50kΩ 0.1%	**49.20kΩ**	1.6%	**49.30kΩ**	1.4%	**49.30kΩ**	1.4%	**49.60kΩ**	0.8%
100kΩ 0.1%	**98.90kΩ**	1.0%	**99.10kΩ**	0.9%	**99.10kΩ**	0.9%	**99.50kΩ**	0.5%
200kΩ 0.1%	**198.1kΩ**	1.0%	**198.1kΩ**	1.0%	**198.4kΩ**	0.8%	**198.4kΩ**	0.8%
270kΩ 0.1%	**267.7kΩ**	0.8%	**267.9kΩ**	0.8%	**267.9kΩ**	0.8%	**267.9kΩ**	0.8%
470kΩ 0.1%	**468.0kΩ**	0.4%	**468.0kΩ**	0.4%	**468.4kΩ**	0.3%	**468.6kΩ**	0.3%
940kΩ 0.1%	**938.6kΩ**	0.2%	**938.7kΩ**	0.2%	**940.8kΩ**	0.1%	**940.8kΩ**	0.1%
1.00MΩ 0.1%	**995.8kΩ**	0.4%	**995.3kΩ**	0.5%	**997.1kΩ**	0.3%	**998.0kΩ**	0.2%
2.00MΩ 0.1%	**1989kΩ**	0.6%	**1990kΩ**	0.5%	**1998kΩ**	0.1%	**1990kΩ**	0.5%
10MΩ 1%	**9894kΩ**	1.1%	**9857kΩ**	1.4%	**10.10MΩ**	1.0%	**9940kΩ**	0.6%
additional: 50MΩ 1%	**48MΩ - 51MΩ**	4%	**47MΩ - 50MΩ**	6%	**53.40MΩ/ not detected**	7%	**48.3MΩ – 50.2MΩ**	4%

The resistor measurement of many multimeter's ends at 20MΩ or 40MΩ!

Appendix C1
results of capacity measurements
Software Version 0.92k (old)

Capacitor	Result of Multimeter PeakTech 3315	Result of Mega8 @1MHz signature 1E 93 07 Original software		Result of Mega8 @8MHz signature 1E 93 07 WITH_AUTO_REF		Result of Mega88 @1MHz signature 1E 93 0A		Result of Mega88 @8MHz signature 1E 93 0A	
56pF	58pF	----		58pF	0%	(42pF with a Diode)	28%	58pF	0%
110pF	114pF	----		114pF	0%	111pF	3%	117pF	3%
220pF	225pF	0.27nF	20%	226pF	0%	222pF	0%	228pF	0%
1000pF	1034pF	1.18nF	14%	988pF	4%	966pF	7%	981pF	5%
3.3nF	3.47nF	3.90nF	12%	3287pF	5%	3223pF	7%	3230pF	7%
7500pF	7.23nF	8.43nF	16%	7241pF	0%	7079pF	3%	7092pF	2%
10nF	10.45nF	11.93nF	14%	10.13nF	3%	9935pF	5%	9922pF	5%
33nF	33.2nF	37.27nF	12%	31.97nF	4%	31.26nF	6%	31.36nF	6%
100nF	97nF	112.36nF	16%	97.02nF	0%	94.98nF	3%	94.93nF	3%
330nF	333nF	385.41nF	14%	333.2nF	0%	325.9nF	2%	325.7nF	2%
1μF	955nF	1.10μF	15%	954nF	0%	934nF	2%	934nF	2%
2.2μF	2.2μF	2.54μF	15%	2198nF	0%	2150nF	2%	2149nF	2%
22μF	21.94μF	25.61μF	16%	21.92μF	0%	21.41μF	3%	21.41μF	3%
47μF	47.5μF	55.29μF	16%	47.30μF	0%	46.70μF	1%	46.9μF	1%
100μF	97.5μF	112.94μF	15%	91.30μF	6%	91.7μF	6%	91.3μF	6%
220μF	229μF	272.69μF	19%	219.2μF	5%	219.2μF	5%	219.2μF	5%
1000μF	1071μF	1371.95μF	28%	1076μF	0%	1063μF	1%	1066μF	1%
2200μF	2.231mF	3308.58μF	48%	2302μF	3%	2288μF	3%	2309μF	3%
4700μF	4.75mF	6103.00μF	28%	5042μF	6%	5042μF	6%	4982μF	5%
14.1mF	14.4mF	5Ω ??		15.13mF	5%	15.13mF	5%	15.31mF	6%

Appendix C2
Results of capacity measurements
Software Version 0.94k

| Capacitor	Result of Multimeter PeakTech3315	Result of Mega8 @8MHz Signature 1E 93 07 WITH_AUTO_REF		Result of Mega8 @8MHz Signature 1E 93 07 WITH_AUTO_REF AUTOSCALE_ADC		Result of Mega168 @8MHz Signature 1E 94 06		Result of Mega168 @8MHz Signature 1E 94 06 WITH_AUTO_REF AUTOSCALE_ADC REF_KORR=17	
56pF	58pF	58pF	0%	57pF	2%	57pF	2%	58pF	0%
110pF	114pF	114pF	0%	114pF	0%	113pF	1%	118pF	4%
220pF	225pF	225pF	0%	225pF	0%	223pF	1%	232pF	3%
1000pF	1034pF	987pF	5%	987pF	5%	968pF	6%	1007pF	3%
3.3nF	3.35nF	3294pF	2%	3287pF	2%	3214pF	4%	3324pF	1%
7500pF	7.23nF	7241pF	0%	7240pF	0%	7038pF	3%	7310pF	1%
10nF	10.45nF	10.06nF	4%	10.10nF	3%	9841pF	6%	10.20nF	2%
33nF	33.2nF	32.50nF	2%	32.50nF	2%	30.46nF	8%	32.69nF	2%
100nF	97nF	97.06nF	0%	96.87nF	0%	94.26nF	3%	97.71nF	1%
330nF	333nF	332.4nF	0%	332.0nF	0%	322.8nF	3%	334.8nF	1%
1µF	955nF	953.6nF	0%	952.4nF	0%	925.8nF	4%	960.3nF	1%
2.2µF	2.2µF	2195nF	0%	2195nF	0%	2134nF	3%	2214nF	1%
22µF	21.94µF	21.87µF	0%	21.92µF	0%	21.26µF	3%	22.07µF	1%
47µF	47.5µF	47.40µF	0%	47.49µF	0%	47.27µF	1%	47.31µF	1%
100µF	97.5µF	91.68µF	6%	91.54µF	6%	90.72µF	7%	91.68µF	6%
220µF	229µF	218.8µF	4%	220.9µF	4%	218.8µF	4%	216.6µF	5%
1000µF	1071µF	1076µF	1%	1076µF	0%	1069µF	0%	1062µF	1%
2200µF	2.231mF	2289µF	3%	2281µF	2%	2289µF	3%	2246µF	1%
4700µF	4.75mF	5032µF	6%	4999µF	5%	5032µF	6%	4923µF	4%
14.1mF	14.4mF	15.34mF	6%	15.29mF	6%	15.17µF	5%	14.99mF	4%

Appendix D
## Results of semiconductor tests
Software Version 094k

semiconductor	Result of Mega8@8MHz signature 1E 93 07 WITH_AUTO_REF	Result of Mega168 @8MHz signature 1E 93 07	Result of Mega168 @8MHz signature 1E 94 06 WITH_AUTO_REF AUTOSCALE_ADC
1N4148	Diode, 721mV, 0pF	Diode, 729mV, 0pF	Diode, 725mV, 0pF
1N4150	Diode, 678mV, 0pF	Diode, 681mV, 0pF	Diode, 682mV, 0pF
BA157	Diode,623mV, 17pF	Diode, 631mV, 16pF	Diode, 620mV, 15pF
BY398	Diode, 541mV, 0pF	Diode, 553mV, 0pF	Diode, 542mV, 0pF
1N4007	Diode, 654mV, 13pF	Diode, 665mV, 9pF	Diode, 658mV, 11pF
LED green	Diode, 1954mV, 6pF	Diode, 1970mV, 6pF	Diode, 1951mV, 4pF
ZPD2,7	2xDi, 729mV, 2659mV	2xDi, 738mV, 2674mV	2xDi, 730mV, 2656mV
BU508A	NPN, B=9, 613mV	NPN, B=9, 621mV	NPN, B=9, 615mV
BU508A B+E	Diode, 613mV, 5201pF	Diode, 621mV, 5285pF	Diode, 611mV, 5344pF
BU508A B+C	Diode, 595mV, 261pF	Diode, 597mV, 267pF	Diode, 591mV, 272pF
2N3055	NPN, B=21, 617mV	NPN, B=21, 626mV	NPN, B=21, 625mV
BC546B	NPN, B=381,780mV	NPN, B=376, 777mV	NPN, B=387, 771mV
BC556B	PNP, B=266, 790mV	PNP, B=429, 787mV	PNP, B=266, 790mV
BC639	NPN, B=180, 722mV	NPN, B=180, 733mV	NPN, B=188, 724mV
BC640	PNP, B=185, 716mV	PNP, B=227, 725mV	PNP, B=187, 719mV
AC128 (Ge.)	PNP, B=68, 270mV	PNP, B=64, 269mV	PNP, B=66, 271mV
BC517	NPN, B=26996, 1419mV	NPN, B=28220, 1413mV	NPN, B=28250, 1404mV
BC516	PNP, B=65535, 1430mV	PNP, B=65535, 1420mV	PNP, B=65535, 1417mV
BS170	N-E-MOS,D,2616mV,66pF	N-E-MOS,D,2562mV, 67pF	N-E-MOS,D,2564mV, 68pF
J310	N-JFET	N-JFET	N-JFET
BRY55/200	Thyristor	Thyristor	Thyristor
IRFU120N	N-E-MOS,D,4151mV, 922pF	N-E-MOS,D,4156mV,894pF	N-E-MOS,D,4153mV, 933pF
IRFU9024	P-E-MOS, D,3525mV, 960pF	P-E-MOS, D,3525mV,926pF	P-E-MOS, D,3534mV, 965pF
ZVP2106A	P-E-MOS,D,3217mV, 115pF	P-E-MOS,D,3220mV,114pF	P-E-MOS,D,3217mV, 113pF
ZVNL120A	N-E-MOS,D,1560mV, 140pF	N-E-MOS,D,1535mV,138pF	N-E-MOS,D,1535mV 138pF

Appendix E
Pictures
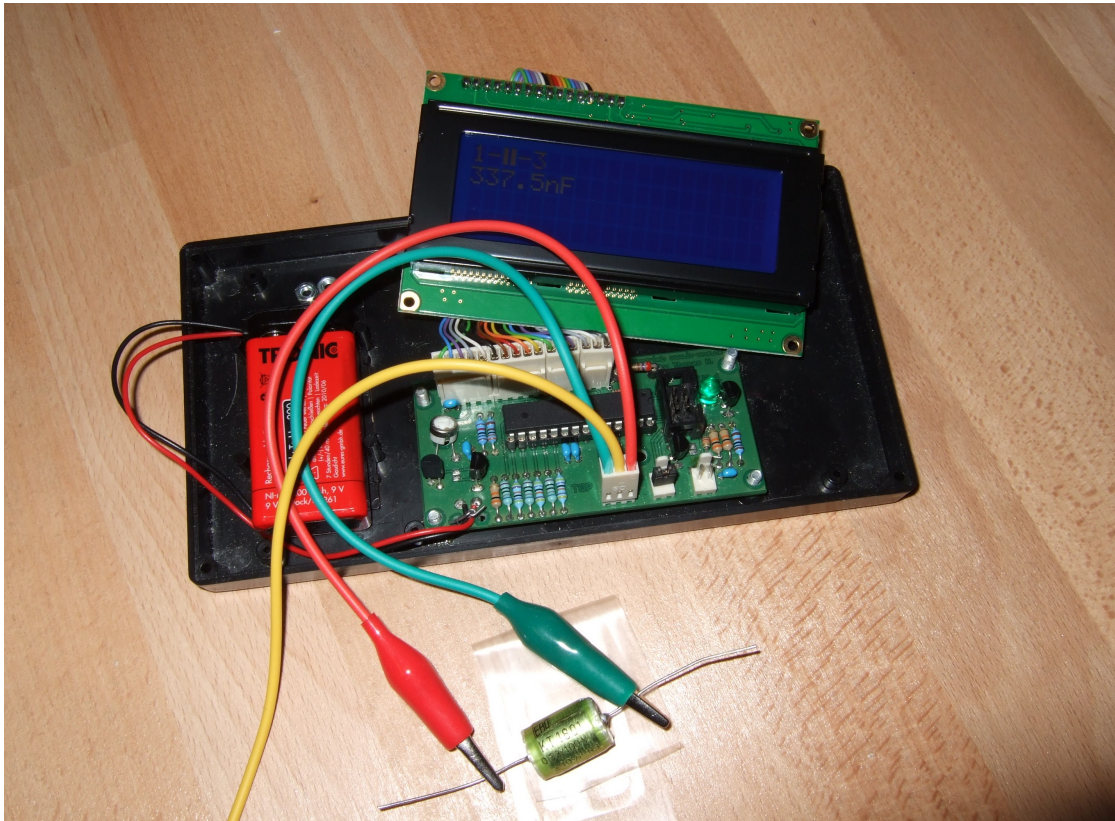
Test version with unmounted 4x20 Display
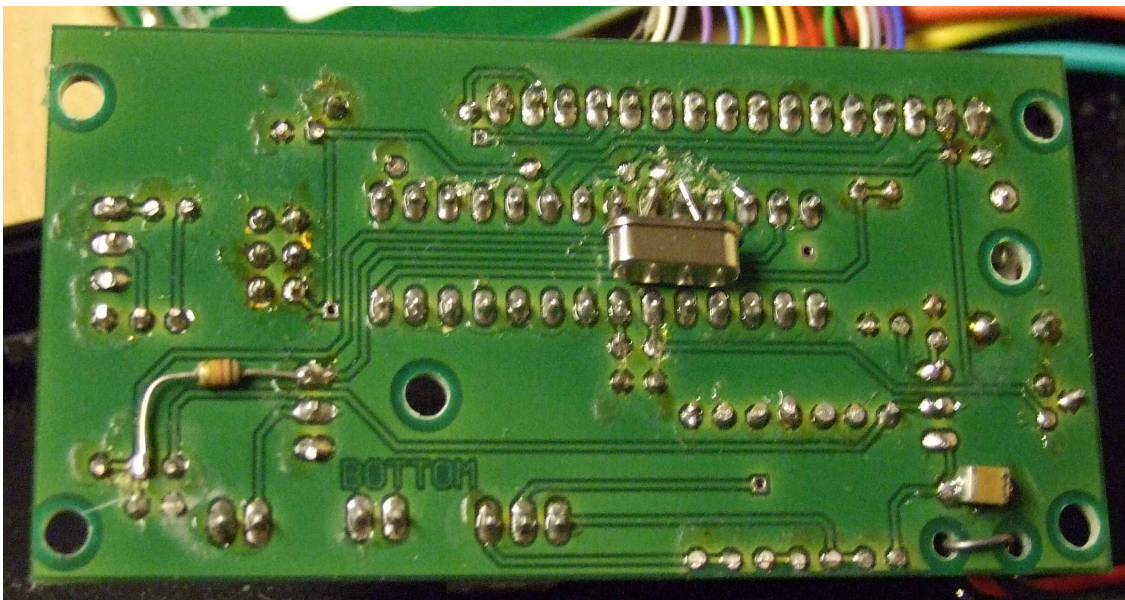
Modifications: 8MHz Crystal (Pin9+10) and
Pull-up resistor (Pin12+13), better (Pin13 and VCC)

Appendix F
Known errors
Software Version 0.94k

- The measurement results of little capacity values vary with the Pin combinations. Combination 1:2 values are about 3pF less than the values of the other pin combinations (1:3 and 2:3). This effect is equal on any tested AVR processor.
- Germanium Diodes (AC128) are not detected in all cases.
- Does program work correctly without the automatic shut off?

Appendix T
to do list (not sorted)
Software Version 0.93k

- Add more and better documentation.
- By my tests I have noticed that the measured voltages of the internal Band cap reference is lower than the data sheets let me expect. The reason is unknown. VCC?, ADC-error?
- Check if transistor tester could get better "interpolated" ADC values if additional noise is added to the signal or to the ADC reference (see ATMEL document AVR121: Enhancing ADC resolution by oversampling). If all items are identical, there can't be any enhancement of resolution by oversampling. Can enough noise be generated with the ATmega counter? How additional noised affects the upper and lower limit values?
  Of course this method can not eliminate all of the ADC errors.
- Test of the effectivity of this method can be done by building a ramp input signal and monitoring this signal. The ramp signal can be build by slowly charging a big capacitor with the 470kΩ resistor. The growing of the voltage can then be monitored with the LCD display in a special part of self test. The difference of the ReadADC function alternatives 44/9, 22*2/9 or 11*4/9 can be monitored too.
- Think about how we can get the real internal resistance of port B output (resistor swiching port) instead of assuming, that ports are equal.
- Can discharging of capacitors be made more quickly, if the minus pin is additionally raised with the 680Ω resistor to VCC (+)?
- Who is using the serial port? I did not test this function and even I don't know how.
- Can inductance be tested?
- How measurement results changes by variation of the supply voltage between 4,5V and 5V?
- Find better organisation of directory structure.
- Check if the tester can use floatingpoint representation of values. The risk of overflow is lower. There is no need to use multiplication and division together to build a multiplication with a non integer factor. But I don't know how much flash memory must be spend for the library.
- Write User's guide for configuring the tester with the Makefile options and description of the build chain.
- Development of a new board with crystal or crystal generator clock for ATmega.
- If the holding current of a thyristor can not be reached with the 680Ω resistor, is it harmless to switch the cathode directly to GND and the anode directly to VCC for a very short time? The current could reach more than 100ms. Will the port be damaged? What is with the power supply (voltage regulator)?
  Check the Port afterwards with self test function!
- Can voltage regulators be checked? (Input, Output, GND)
- Can optoelectronic couplers be checked?
- Can the transistor tester act as clock generator for AVR devices, which can't be programed because of wrong fuse configuration?
- Test if 0,1Ω can be displayed with the AUTOSCALE_ADC option of ATmega168. In this case probably the ESR measurement of electrolytical capacitors is possible.

Additional:

As you can see, the list of ideas and "to do" tasks is still rather long. My purpose is to make the transistor tester more precise, faster and multifunctional. Probably I purchase a second board for easier checking of different processors (mega8 and mega168).

I can not promise you, that the task list will ever be empty. Also I can not promise you, that my software is free of errors. I don't give you any warrenty! Using my software is your own risk. During my tests never a ATmega or other part was damaged. But I give you the promise that I never will publish a release without source code!.

 By now I develop my software with the GNU toolchain under a Linux (Ubuntu) operating system. I had got a useful hint from another thread on www.mikrocontroller.net to use my programmer (Diamex ALL-AVR) without any problems. The same programmer was not able to program the EEprom of my ATmega8's with the Windows driver. Now I can also use the original software of Markus Frejek for my tester. I still hope, that my software become a official version. But I still wait on answer to my email to 5volt and to linuxgeek from February 2012.