

**AVR-8-bit-Mikrocontroller**  
**Gruppe 200 - Einsetzen von AVR-Tools**  
**Teil 201 - Experimentierboards**



## **Teil 201 - Experimentierboards**

### **1 Experimentierboards zum Testen und Programmieren von AVR-Mikrocontroller**

#### **1.1 Mit welchen Mitteln AVR-Mikrocontroller programmiert werden**

#### **1.2 Starterkit STK500**

#### **1.3 Entwicklungs-Tool AVR Dragon (nicht mehr im Handel!)**

#### **1.4 ATM18-Controllermodul und ATM18-Testboard (nicht mehr im Handel!)**

#### **1.5 AVR-ALE-Testboard**

## **Teil 202 - ISP-Programmieradapter**

### **2 ISP-Programmieradapter**

#### **2.1 ISP Bezogen auf die verschiedenen Schnittstellen**

##### **2.1.1 Serielle Schnittstelle**

##### **2.1.2 Parallele Schnittstelle**

##### **2.1.3 USB-Schnittstelle**

#### **2.2 CC2-AVR-Programmer alias USBprog**

##### **2.2.1 Aufbau**

##### **2.2.2 Arbeitsweise**

##### **2.2.3 Firmware-Änderung**

## **Teil 203 - AVR-ALE-Testboard**

### **3 Beschreibung des AVR-ALE-Testboard**

#### **3.1 Schaltungsaufbau**

#### **3.2 Stromversorgung**

#### **3.3 Einsatz verschiedener AVR-Mikrocontroller**

#### **3.4 Anzahl LEDs und Tasten**

#### **3.5 LCD-Interface und 20x4-LCD**

##### **3.5.1 Erzeugung des Enable-Signals für das LCD**

##### **3.5.2 LCD-Backlight**

#### **3.6 Ansteuerung von Relais**

#### **3.7 RS-232-Schnittstelle**

#### **3.8 USART-Testboard-Schnittstelle**

## **Teil 204 - AVR Studio**

### **4 Einsatz des AVR Studio**

#### **4.1 AVR Studio installieren**

#### **4.2 Testboard und Programmer zusammenschalten**

##### **4.2.1 Treiber AVRISP mkII neu installieren**

#### **4.3 Starten von AVR Studio**

##### **4.3.1 AVR Studio und CC2-AVR-Programmer**

##### **4.3.2 Mikrocontroller-Einstellungen im AVR Studio**

## **Teil 205 - Assembler und AVR Studio**

### **5 Assembler und AVR Studio**

#### **5.1 Der Übersetzer (Assembler)**

#### **5.2 Ein neues Projekt erzeugen**

##### **5.2.1 Der Projekt-Bereich**

##### **5.2.2 Bearbeiten der Assemblerdatei**

##### **5.2.3 Assemblieren des Quell-Codes**

#### **5.3 Simulation des Codes**

##### **5.3.1 Programmausführung im Einzelschrittverfahren**

##### **5.3.2 Debugger-Stopp-Punkte**

#### **5.4 Verändern des Programmtextes**

# AVR-8-bit-Mikrocontroller

## Gruppe 200 - Einsetzen von AVR-Tools

### Teil 201 - Experimentierboards

- 5.4.1 Überwachen von Variablen
- 5.4.2 Anzeigen der Prozessordetails
- 5.4.3 Speichern des Projekts
- 5.5 Erzeugen eines weiteren ASM-Projektes im Schnelldurchgang
- 5.6 Flashen eines ASM-Programms in ein Mikrocontroller ATmega88

### Teil 206 - C-Compiler und AVR Studio

- 6 CodeVisionAVR C-Compiler und AVR Studio
  - 6.1 CodeVisionAVR C-Compiler installieren
  - 6.2 Erzeugen eines C-Projektes
    - 6.2.1 Ein neues Projekt beginnen
    - 6.2.2 Ein C-Projekt generieren
  - 6.3 Einbinden von AVR Studio in den CVAVR
  - 6.4 AVR Studio Debugger für CVAVR
  - 6.5 Flashen eines C-Programms in ein Mikrocontroller ATmega88

### Teil 207 - Editor - UltraEdit

- 7 Editor - UltraEdit
  - 7.1 Kopf- und Fuß-Zeile
    - 7.1.1 Einstellungen für Assembler-Programme
    - 7.1.2 Einstellungen für C-Compiler-Programme
  - 7.2. Syntaxhervorhebung (Syntax Highlighting)
    - 7.2.1 AVR-Assembler
      - 7.2.1.1 Syntaxbefehle für den AVR-Assembler
      - 7.2.1.2 Farben und Schriftstile der Gruppen für den AVR-Assembler
    - 7.2.2 CodeVisionAVR C-Compiler
      - 7.2.2.1 Syntaxbefehle für den CVAVR C-Compiler
      - 7.2.2.2 Farben und Schriftstile der Gruppen für den CVAVR C-Compiler
  - 7.3 Wortsammlung für AVR-Assembler
  - 7.4 Wortsammlung für CodeVisionAVR C-Compiler

### Hinweis

Externe Anschaltungen und Hardware-Erweiterungen werden in der **Gruppe 400 - ASM-Projekte** und in der **Gruppe 600 - C-Projekte** detailliert beschrieben.

**HINWEIS: Seit 2016 ist die Firma ATMEL von der Firma MICROCHIP übernommen worden, die dadurch zum drittgrößten Mikrocontroller-Anbieter geworden ist. Damit sind alle in diesem Tutorial erwähnten ATMEL-Erzeugnisse und Handbücher zu finden bei:**



Zu den bekanntesten Produktserien gehören die [PIC-Mikrocontroller](#) und [AVR-Mikrocontroller](#), die auch bei Hobby-Elektronikern sehr beliebt sind. Hierfür liefert Microchip auch notwendige Entwicklungsumgebungen (Hardware und Software).

# AVR-8-bit-Mikrocontroller

## Gruppe 200 - Einsetzen von AVR-Tools

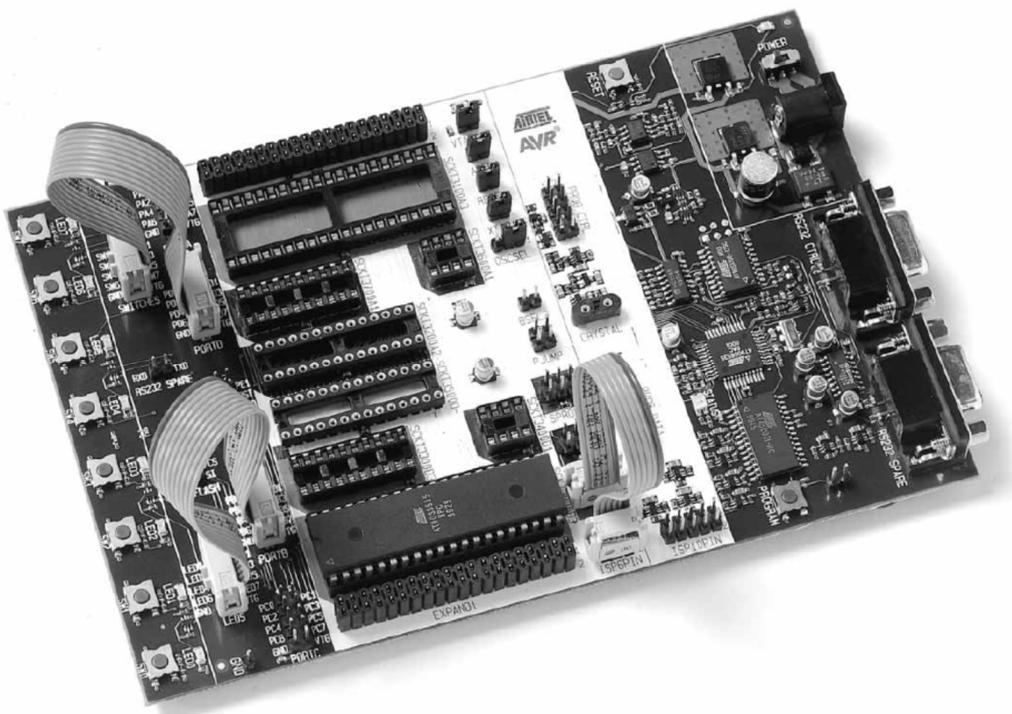
### Teil 201 - Experimentierboards

## 1. Experimentierboards zum Testen und Programmieren

### 1.1 Mit welchen Mitteln AVR-Mikrocontroller programmiert werden

Ein Mikrocontroller alleine ist noch zu nichts nützlich!

Damit man etwas damit anfangen kann, braucht man eine Schaltung, in die der Controller eingesetzt wird. Aber bevor man fertige Platinen mit Mikrocontroller für Anwendungen entwirft, ist es ratsam, den Mikrocontroller zu programmieren und unter Wirkbedingungen zu testen. Dazu werden bei Elektronik-händlern Platinen angeboten, die alles nötige (Taster, LEDs, Steckverbinder usw.) enthalten, wie z.B. das **STK500** (offizielle ATMEL-Bezeichnung: ATSTK500). Mit solchen Testboards werden die entworfenen Schaltungen - oder funktionelle Teile davon - simuliert und emuliert. Dazu müssen die Test-boards 1. den einzusetzenden Mikrocontroller, 2. die entsprechenden Schnittstellen zur "Außenwelt" und 3. eine Schnittstelle zum Programmiersystem enthalten.



**Bild 1.1-01: Experimentierboard [ATSTK500](#) der Firma ATMEL ([Bildvergrößerung](#))**

Das STK500 ist dazu in der Lage und **ist vereinzelt noch für ca. 130 € erhältlich**. Wenn man nicht so viel ausgeben möchte, dann kann man auch selber eine Schaltung aufbauen, die das Programmieren (Flashen) von AVR-Mikrocontrollern und das Testen von Anwendungen ermöglicht. So umfangreich wie das STK500 muss es nicht sein, es reichen ggf. eine Hand voll Bauteile.

Dann braucht man noch einen Adapter, der den PC, in dem man seine Anwendungs-Programme entwirft, mit dem Experimentierboard bzw. selbst hergestellten Testboard verbindet, um das gerade erstellte Programm in den Mikrocontroller zu laden - auf Neudeutsch: zu flashen. Zum Flashen kommen verschiedene Programmier-Protokolle zur Anwendung:

- In System Programming (**ISP**)
- High Voltage Serial Programming (**HVSP**)
- Parallel Programming (**PP**)
- JTAG Programming (**JTAG**)

# AVR-8-bit-Mikrocontroller

## Gruppe 200 - Einsetzen von AVR-Tools

### Teil 201 - Experimentierboards

Im Weiteren wird nur das Programmier-Protokoll **ISP** behandelt, d.h. es wird ein **ISP-Programmieradapter** (In-System-Programmer) beschrieben, über die man die Programme vom PC in den Mikrocontroller (genauer: in den Flash-Speicher des Mikrocontrollers) übertragen kann.

So sieht die Anordnung zum Testen von Mikrocontroller-Anwendungen aus:

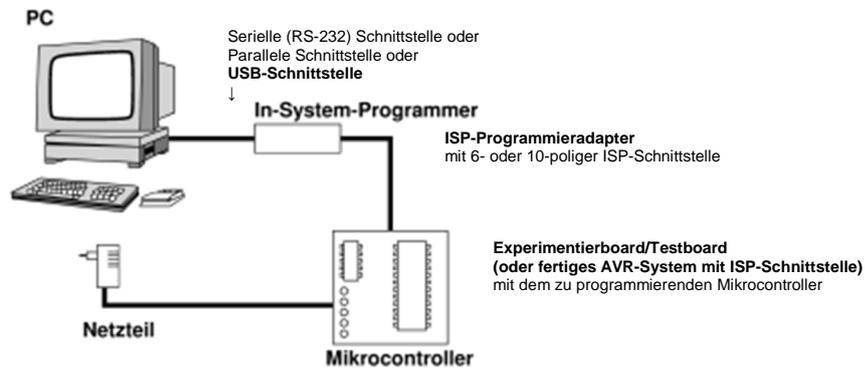


Bild 1.1-02: Zusammenschaltung von PC, ISP und Experimentierboard

## 1.2 Starterkit STK500

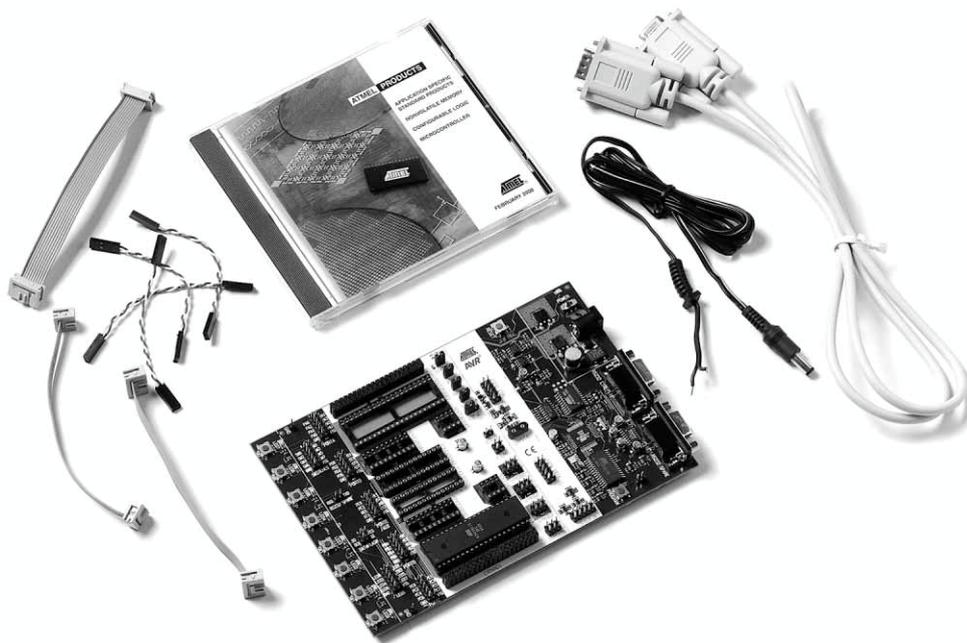


Bild 1.2-01: Das STK500 mit Zubehör ([Bildvergrößerung](#))

Das Starterkit [STK500](#) ist teuer und für den Hausgebrauch sehr umfangreich! Ein reduzierter und den eigenen Bedürfnissen angepasster Nachbau ist wahrscheinlich die bessere Lösung.

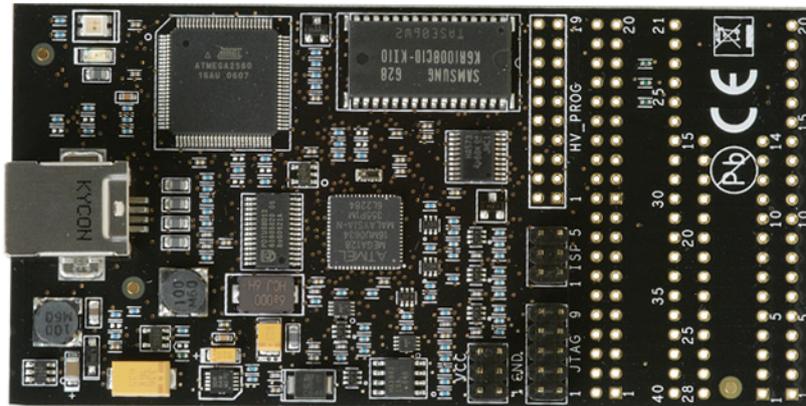
## 1.3 Entwicklungs-Tool AVR Dragon (nicht mehr im Handel!)

Die Firma ATMEL stellt mit dem Entwicklungs-Tool [AVR Dragon](#) auch ein preiswerteres Board (ca. 50 €) zur Verfügung. Abgesehen von einer Emulationsunterstützung ist es allerdings nur zum Flashen der AVR-Mikrocontroller bis zu einer Flash-Speicher-Größe von 32 KByte geeignet. Möglichkeiten zur Anschaltung von Ein-/Ausgabe-Modulen zur Simulation von Schaltungsvarianten fehlen.

Mit Einschränkungen werden alle Programmier-Protokolle unterstützt. Der AVR Dragon wird über eine USB-Schnittstelle mit dem PC verbunden und erhält über diese seine Versorgungsspannung. Es ist

## AVR-8-bit-Mikrocontroller Gruppe 200 - Einsetzen von AVR-Tools Teil 201 - Experimentierboards

damit in der Lage, den zu programmierenden Mikrocontroller im Zielsystem zu flashen: auf dem Dragon-Board, auf einem Testboard oder in der Original-Schaltung einer Anwendung.



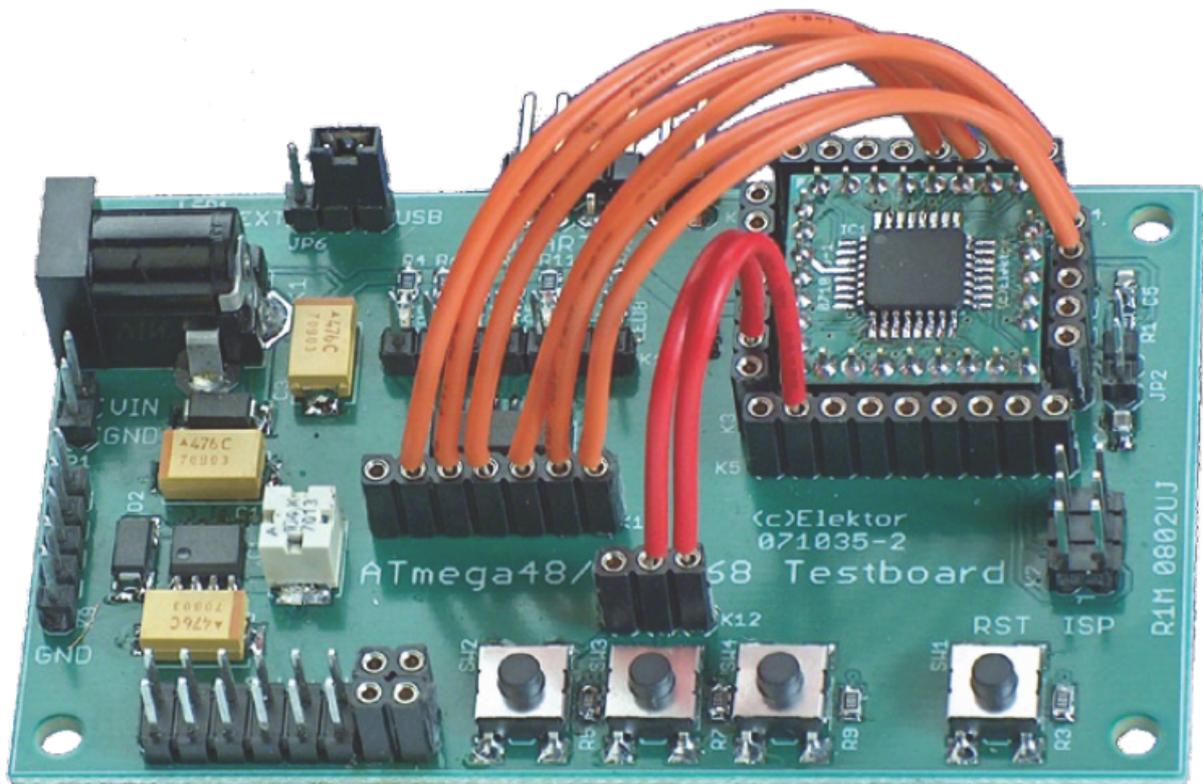
**Bild 1.3-01: Das Entwicklungs-Tool AVR Dragon ([Bildvergrößerung](#))**

Auf der rechten Seite der Abbildung werden die Pins je nach Bedarf mit IC-Sockeln bestückt. Die Pins sind so beschaltet, dass die verschiedenen AVR-Mikrocontroller nur die notwendigen Programmier-Signale erhalten.

**Das Tool wird nicht mehr hergestellt!**

### [1.4 ATM18-Controllermodul und ATM18-Testboard \(nicht mehr im Handel!\)](#)

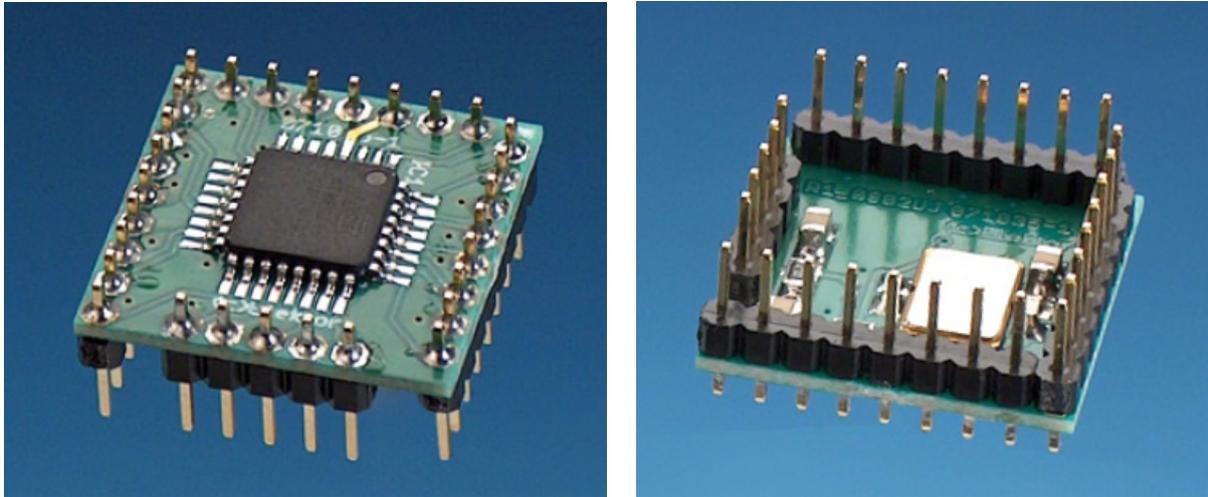
Zum Testen von Versuchs-Schaltungen, die ggf. später in konkrete Anwendungen überführt werden sollen, kann man von Elektor ein mit SMD-Bauteilen fertig bestücktes Testboard ([Artikel](#)) für 37,50 € erwerben.



**Bild 1.4-01: ATM18-Testboard (Elektor 071035-92 - EUR 37,50, [Bildvergrößerung](#))**

## AVR-8-bit-Mikrocontroller Gruppe 200 - Einsetzen von AVR-Tools Teil 201 - Experimentierboards

Zu diesem wird allerdings ein zusätzliches spezielles Minimodul für 12,50 € benötigt, auf dem ein **ATmega88** einschließlich des Quarzes für die Taktversorgung aufgelötet ist. Einsatz eigener Mikrocontroller oder Weiterverwendung des Minimoduls in eigenen Anwendungen wird dadurch erschwert und verteuert.



Bilder 1.4-02 und 1.4-03: ATM18-Controllermodul (Elektor 071035-91 - EUR 12,50)

Auf dieser Test-Ausstattung basiert das [ATM18-Projekt](#), das von **Elektor** und dem **Computer:club<sup>2</sup>** in Zusammenarbeit mit **Udo Jürß**, dem Chefentwickler von [www.microdrones.com](http://www.microdrones.com), ins Leben gerufen wurde.

**Diese Tools werden von Elektor nicht mehr angeboten!**

### [1.5 AVR-ALE-Testboard](#)

Der Autor bevorzugt - in Anlehnung an das ATM18-Projekt - ein selbst hergestelltes Testboard, das mit Chips der DIL-Gehäuseform (Dual In-Line) auskommt. Auch der Mikrocontroller wird in DIL-Ausführung mit einem sog. Nullkraft-Sockel eingesetzt. Das Testboard ist zum ATM18-Projekt voll kompatibel und beinhaltet bereits das ebenfalls vom ATM18-Projekt eingesetzte **2-Draht-LCD-Interface** und die zusätzliche Möglichkeit, 2 Relais anzusteuern.

**Da die hier dargestellten Tools entweder Auslaufmodelle oder nicht mehr im Handel sind, kann ich nur noch dieses Testboard empfehlen. Es ist immer noch mit Bauteilen, die in fast allen Elektronikmärkten erhältlich sind, aufzubauen.**

Im [Teil 203 - AVR-ALE-Testboard](#) wird das Board ausführlich beschrieben.