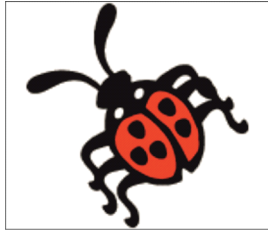


# AVR-8-bit-Mikrocontroller

## Gruppe 200 - Einsetzen von AVR-Tools

### Inhaltsverzeichnis



## Gruppe 200 - Einsetzen von AVR-Tools

### 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
  - 1.4 ATM18-Controllermodul und ATM18-Testboard
  - 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

# AVR-8-bit-Mikrocontroller

## Gruppe 200 - Einsetzen von AVR-Tools

### Inhaltsverzeichnis

- 5.4 Verändern des Programmtextes
  - 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. Einträge in das WORDFILE.TXT
    - 7.2.1 Grundsätzliches Format
    - 7.2.2 Syntax Befehle
  - 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.

# AVR-8-bit-Mikrocontroller

## Gruppe 200 - Einsetzen von AVR-Tools

### Inhaltsverzeichnis

#### Vorbemerkung

Nichts ist vollkommen - und nichts ist endgültig! So auch nicht dieses Tutorial! Deshalb bitte immer erst nach dem neuesten Datum schauen. Vielleicht gibt es wieder etwas Neues oder eine Fehlerbereinigung oder eine etwas bessere Erklärung. Wer Fehler findet oder Verbesserungen vorzuschlagen hat, bitte melden ([info@alenck.de](mailto:info@alenck.de)).

Immer nach dem Motto: Das Bessere ist Feind des Guten und nichts ist so gut, dass es nicht noch verbessert werden könnte.

Bild-, Beispiel-, Form- und Tabellen-Nummern sind nach folgendem Schema aufgebaut, damit bei Einfügungen/Löschungen nicht alle Nummern wieder geändert werden müssen (hier bunt dargestellt):

Darstellungsart	Abschnitt-LfdNummer: Beschreibung	allgemeines Schema
•	<b>Bild 5.1.4-02: Daten-Adress-Raum</b>	Benummerung eines Bildes
•	<b>Beispiel 5.1.4-03: EEPROM-Speicherung</b>	Benummerung eines Beispiels
•	<b>Form 5.1.3-01: Die main-Funktion</b>	Benummerung einer Formdarstellung
•	<b>Tabelle 5.1.4-01: Schlüsselwörter vom CVAVR</b>	Benummerung einer Tabelle

#### Gravierende Änderungen gegenüber der Vorversion

1.

Völlig neue Strukturierung in **Gruppen** und **Teile**, um das Tutorial umfassend ordnen zu können. Die **Abschnitte** in den **Teilen** sind weitgehend erhalten geblieben.

Gruppenbezeichnung	Kurzbezeichnung
Gruppe 100: Technologie der AVR-8-Bit-Mikrocontroller	Technologie
Gruppe 200: Einsetzen von AVR-Tools	Tools
Gruppe 300: Arbeiten mit AVR-Assembler 3xx_Programm_yyyy	ASM-Programmierung ASM-Programm-Beispiel
Gruppe 400: AVR-ASM-Projekte 4xx_Projekt_yyyy	ASM-Projekte ASM-Projekt-Bezeichnung
Gruppe 500: CodeVisionAVR C-Compiler 5xx_Programm_yyyy	C-Programmierung C-Programm-Beispiel
Gruppe 600: AVR-C-Projekte 6xx_Projekt_yyyy	C-Projekte C-Projekt-Bezeichnung

**xx** steht für die laufende Nummer innerhalb des **Teils**, in dem das Programm bzw. das Projekt erscheint und **yyyy** steht für die Programm- bzw. Projekt-Kurz-Bezeichnung.

2.

Notwendige Änderungen auf Grund Neuinstallation von **Windows 7**.

3.

**Windows 7** machte eine Installation von **CodeVisionAVR V2.60** als Vollversion notwendig. Daraus leiten sich auch viele Änderungen im Detail für die C-Programmierung (**Gruppe 500**) ab.

4.

Neu-Installation von **AVR Studio Vers. 4.19** unter **Windows 7**

5.

Zur Demonstration des Tools **AVR Studio** ist in **Gruppe 200** eine Trennung in **Teil 205 - Assembler und AVR Studio** und **Teil 206 - C-Compiler und AVR Studio** vorgenommen worden.

6.

**ASM-** und **C-Projekte** werden jeweils in eigenen Gruppen gesammelt (**Gruppe 400** für Assembler- und **Gruppe 600** für C-Projekte).