

ATMEL Kurs mit BASCOM AVR ®

Vorweg

Für Einsteiger,
und solche die es werden wollen ...

Entwickelt von

Pierre Mai

Speziell für



Vorweg

Copyright

Dieser Kurs und die beschriebenen Schaltungen, Verfahren und Programme wurden sorgfältig erstellt und getestet. Trotzdem können Irrtümer und Fehler nicht ausgeschlossen werden.

Der Author ist dankbar auf Hinweise und Fehler, übernimmt jedoch keine Verantwortung für die uneingeschränkte Richtigkeit und Anwendbarkeit des Kursinhaltes. Jegliche Verantwortung und Haftung wird ausgeschlossen.

Die Information im vorliegenden Kurs werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht. Alle in diesem Kurs vorkommenden Marken- und Handelsnamen sind Eigentum ihrer jeweiligen Besitzer und unterliegen den einschlägigen gesetzlichen Bestimmungen. Das Fehlen einer gesonderten Kennzeichnung bedeutet nicht, das es sich dabei um einen freien Namen handelt.

Dieser Kurs ist als ganzes und in Teilen urheberrechtlich geschützt.

Jede Verwendung ohne schriftliche Genehmigung des Authors ist unzulässig und strafbar. Dies gilt insbesondere für Vervielfältigungen jeder Art, Übersetzungen, Mikroverfilmung und für die Einspeicherung in Elektronische Systeme.

Kursteilnehmer sind dem Autor per E-Mail bekannt, und haben die Genehmigung zum persönlichen Druck und Verwendung soweit es für den Kurs erforderlich ist. Eine Weitergabe an dritte wird hiermit explizit untersagt.

Der Autor behält sich vor diesen Kurs als gedruckte version zu veröffentlichen.

Alle Rechte vorbehalten. All rights reserved.

Piere Mai

Einleitung

Dieser Kurs soll eine Einführung in die ATMEL Mikrocontroller (μ C) Programmierung mit BASCOM auf einem Atmel Mega 16 darstellen.

Der Kurs behandelt die Allgemeinen Grundlagen der μ C Programmierung sowie Themen die speziell zugeschnitten sind auf den Raketenmodellbau.

Weiterhin werden auch die verwendeten Außenbeschaltungen des μ C behandelt, die zum Betrieb notwendig sind.

Der Kurs gliedert sich in mehrere Teile, die aus Didaktischen Gründen einzeln an die Kursteilnehmer geschickt werden, um einen möglichst hohen Lernerfolg zu erzielen.

Weiterhin sind in diesem Kurs Erfolgskontrollen eingebaut, wobei der Kursteilnehmer sein programmiertes Ergebnis per E-Mail an den Kursleiter zur Kontrolle schicken sollte, um den Lernerfolg zu überwachen, und eventuelle Nachschulungen zu ermöglichen.

Grundlage für diesen Kurs ist das Entwickler Board das in Zusammenarbeit des Raketenmodellbau Forums entstanden ist. Jedoch können die Programmbeispiele auch auf andere Systeme übertragen werden, bei Anpassung der Software an vorhandene Hardware.

Die Programmierung erfolgt in diesem Kurs mittels des Original ATMEL Programmers (ATAVRISP) mit Serieller Schnittstelle, der auch per USB zu Seriell Wandler betrieben werden kann, da einige PC's/ Laptops diese Schnittstelle nicht mehr besitzen.

Danksagung

Hiermit möchte ich mich schon einmal bei allen Helfern und Helfershelfern bedanken, die diesen Kurs mit ins Leben gerufen haben, und ihre Kommentare und Anmerkungen mit in den Kurs eingebracht haben.

Ein weiterer Dank geht an das Forum von www.raketenmodellbau.com mit all seinen Teilnehmern die immer für Anregungen und Kritik sorgen. Hier insbesondere an Reinhard Rath der erheblichen Aufwand in die Schaltungen investiert hat, sowie Stefan Wimmer und Ulrich Nieland für die Kommentare und Realisierung.

Mein spezieller Dank geht an meine Freundin Marilena Martella, die durch ihre Korrektur, und Versorgung meines leiblichen Wohls sowie ihr Verständnis für Stundenlanges „Computern“ in keiner vergleichbaren Weise zu diesem Kurs beigetragen hat.

ATMEL Kurs mit BASCOM AVR ® Grundlagen

Für Einsteiger,
und solche die es werden wollen ...

Entwickelt von

Pierre Mai

Speziell für



Grundlagen

Symbole

In diesem Kurs verwende ich am rechten Rand Symbole, wenn Textabschnitte von bestimmter Wichtigkeit sind.

Die nicht Beachtung der Symbole kann im schlimmsten Fall ihren Prozessor schädigen, oder Kurzschlüsse verursachen.

Das Entwickler Board, oder das Flight Board muss zum Übernehmen der Einstellungen vom Netz getrennt werden.



Bei diesen Einstellungen bitte keine Änderungen vornehmen, ausser die im Text erwähnten. Es besteht die Gefahr das der Prozessor nicht mehr funktioniert.



Wichtige Hinweise die Unbedingt zu beachten sind.



Hier behandeln wir die Lock oder Fuse Bits des μ C. Bitte nur die explizit angegebenen Bits ändern!



Hier wird auf Angabe verwiesen, die man sich anschauen sollte (Link, Dokument, etc.).



Dieses Symbol sagt euch, das ihr die gestellte Aufgabe per E-Mail, die euch bei Anmeldung zum Kurs mitgeteilt wurde, dem Kursleiter zuzusenden. Bitte nicht vergessen im Betreff die Aufgabennummer anzugeben.



Zahlensysteme

Es werden verschiedene Zahlensysteme zum Einsatz kommen, deren Umrechnung bekannt sein sollte, bzw. mit einem Taschenrechner oder Tabellen möglich sein sollte.

Dezimal: Unser „normal“ gebräuchliches Zahlensystem 0,1,2 ...9,10,11...19,20,21... also mit einem Übertrag von 9 auf 0 auf der Basis 10.

Binär: Das „Digitale“ Zahlensystem 0,1,10,11,100,101... also mit einem Übertrag von 1 nach 0 auf der Basis von 2.

Hexadezimal: Das „Buchstaben“ Zahlensystem 0,1,2...9,A,B...F,10,11,12....1F,20,21 mit einem Übertrag immer von F nach 0 auf der basis 16.

Grundsätzlich gilt in diesem Kurs die „normale“ also Dezimale Schreibweise, wenn nicht der Zahlenwert anders angegeben wird. Ich lehne mich in diesem Kurs an die in Bascom verwendeten Syntax, wobei gilt:

- Dezimal: &DXXXXX bzw. keine Angabe (Beispiel: &D65535 oder 65535)
- Binär: &BXXXXXXXX (Beispiel: &B01010101)
- Hexadezimal: &HXXXX (Beispiel: &HA03F)

Gebraucht werden alle Schreibweisen bis maximal 16Bit (&D65535, &B10101010 10101010, &HFFFF) Für den normalen Gebrauch reicht jedoch eine tabelle von 0 bis 254.

Damit die Suche im Internet nicht zu viel von unserer Zeit nimmt, befindet sich eine solche Tabelle im Anhang I.



Register

Ein Mikroprozessor (μC) nutzt für Einstellungen und Ein/Ausgabeoptionen Register. Die einzelnen Register werden im Kurs nacheinander angesprochen.

In diesem Kurs werden die Register folgendermaßen dargestellt:

DDRA								
BIT	7	6	5	4	3	2	1	0
&H 1A	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
INIT.	0	0	0	0	0	0	0	0

Oben links wird die Bezeichnung des Registers angegeben, wie sie auch in dem Datenblatt und in Bascom verwendet wird. Hier DDRA = Data Direction Register port A.

&H gibt an, auf welcher Hexadezimalen Adresse das Register zu erreichen ist, dieses ist nur zur Übersicht, um den Vergleich zur Assembler bzw. C Programmierung zu vereinfachen.

In der grau unterlegten Zeile werden die einzelnen Bits des Registers dargestellt. Begonnen wird mit dem MSB (Most Significant Bit / Höchstwertiges Bit) bis hin zum LSB (Least Significant Bit / Niederwertiges Bit).

Unter dem jeweiligen Bit steht, ob dieses Bit vom Anwender verändert werden kann (W), und ob dieses gelesen werden kann (R).

Init. stellt die Grundeinstellung des Prozessors nach einem Reset oder einem Power Off dar.

Da dieses Register die Datenrichtung für den Port A behandelt und Port A auch 8 Anschlüsse besitzt, ist die Bedeutung der einzelnen Bits gleich. es gibt jedoch auch Register, wo die einzelnen Bits unterschiedliche Bedeutungen haben. Jedes einzelne Bit, wird jedoch Extra besprochen.

Ich möchte hier einmal auf das Datenblatt des Mega16 hinweisen, welches unbedingt als Grundlage für jede Programmierung zur Hand genommen werden sollte. Das Datenblatt findet ihr unter http://www.atmel.com/dyn/products/devices.asp?family_id=607 , und da natürlich unter mega AVR den Mega16 auswählen.



ATMEL Kurs mit BASCOM AVR ®

Einführung

Für Einsteiger,
und solche die es werden wollen ...

Entwickelt von

Pierre Mai

Speziell für



Einführung

Benötigtes Material

Natürlich brauchen wir für den Kurs auch ein wenig Material, ohne das es nicht sinnvoll ist überhaupt anzufangen.

- Computer mit einem Windows ® Betriebssystem (200/XP).
- Bascom AVR ® als Demo oder registrierte Vollversion (Build 1.11.7.7)
- ATMEL ® Programmiermaus (ATAVRISP).
- Eventuell einen USB/ Seriell Wandler für die Programmiermaus.
- Entwickler Board oder Flight Board von www.raketenmodellbau.com
- Ein (Stecker) Netzteil passend zum AVR Board
- Ein LCD mit mindestens 2*16 Zeichen (besser 4*16 Zeichen) mit einem Controller HD44780 oder 100% kompatibel.
- Eventuell weitere Hard/ Software die dann aber in den Übungsblättern extra aufgeführt werden.
- Das Mega16 Datenblatt

Die Bascom Demo (Build 1.11.7.7) ist bis zu einer Codegröße von 4kB mit allen Funktionen unter http://www.mcselec.com/download_avr.htm zum Download verfügbar.

Die in diesem Kurs erstellten Programme werden die Codegrößenbeschränkung der Demo Version nicht überschreiten.



Der Betrieb von Bascom unter Pseudo-Oberflächen oder Emulatoren für andere Betriebssysteme wurde von mir nicht getestet, und eine einwandfreie Funktion kann nicht garantiert werden. Dieses gilt auch für alle anderen in diesem Kurs verwendeten Programme.

Installation

Nachdem wir nun alle Programme und die Hardware organisiert haben, gehen wir nun den Schritt der Installation.

- Bascom AVR gemäß Installationsanleitung installieren.
- Falls benötigt : USB/Seriell Wandler installieren (Treiber)
- Programmiermaus mit der Seriellen Schnittstelle oder dem USB/ Seriell Wandler verbinden
- Programmiermaus mit dem Entwickler/ Flight Board verbinden
- Netzteile anschließen

Auf die Spannung und Leistung des verwendeten Netzteils achten!



Funktionstest

Damit wir im Laufe des Kurses nicht in irgendwelche Schwierigkeiten kommen, ist in dem Neuen AVR vom Raketenmodellbau.org schon ein Programm eingebaut, welches die Funktionen des Entwickler bzw. Flight Boards testet.

Der Funktionstest ist der Reihe nach aufgebaut, bitte den bestandenen Test abhacken.

- 1 Einschalten der Betriebsspannung
 - LED 1 blinkt im 1 Sekunden Takt
 - LED 1 Okay
- 2 Drücken von Taster 1
 - LED 1 fängt an schnell zu Blinken
 - Taster 1 Okay
- 3 Drücken von Taster 2
 - LED 1 leuchtet permanent
 - Taster 2 Okay
- 4 Drücken von Taster 1
 - LED 2 blinkt im 1 Sekunden takt
 - LED 2 Okay
- 5 Beide Potis auf Mittelstellung drehen
- 6 Drücken von Taster 2
 - LED 1 und LED 2 leuchten auf
- 7 Poti 1 langsam nach links bis Anschlag drehen, dann nach rechts bis Anschlag drehen
 - LED 1 erlischt auf einer Position (nahe Mittelstellung)

Bei vorhandenen Fehlern bitte noch einmal ALLE betreffenden Bauteile auf richtigen Einbau überprüfen, besonders die Position bei LED's Dioden und Elkos sind wichtig! Sind auch alle Lötverbindungen korrekt, und keine eventuellen Lötbrücken vorhanden? Sollte auch nach einer Gründlichen Überprüfung der entsprechende Teil nicht funktionieren, bitte eine E-Mail mit den bis dahin ausgeführten Schritten an den Kursleiter, er wird versuchen euch per Mail oder bei komplexeren Teilen Telefon zu kontaktieren. Dazu die Telefonnummer und die Erreichbarkeit bitte in der Mail mit angeben!

