

CC65-Einsteigerkurs#

Seit einiger Zeit gibt es im Internet einen C-Compiler für Rechner mit dem 6502 Prozessor. Dank Christian Grössler, Mark Keates, Freddy Offenga und David Lloyd erzeugt dieser Compiler neben C64, Apple II, PET, Plus4, C128 auch Atari 8-Bit Programme.

Im ersten Teil dieses Kurse geht es um die Installation des CC65 Compilers und das Einrichten einer Entwicklungsumgebung. Diese wird mit dem "Hello World" Programm aus den Beispieldateien getestet.

Voraussetzungen:#

Der CC65 Compiler ist ein sogenannter "CrossCompiler", d.h. er läuft nicht auf Rechnern für die er später die Programme erzeugt. Man braucht neben einem Atari 8-Bit Rechner noch einen anderen Computer, auf dem der CC65 Compiler läuft. Dieser Rechner kann ein normaler Intel-kompatibler PC (386 und höher) mit den Betriebssystemen DOS, Windows 9x/NT/2000/XP, OS/2 oder Linux sein. Da sich der CC65 auf fast jedem Standard-Unix übersetzen lässt, kann als Übersetzungssystem auch ein Atari ST, Amiga oder Apple mit Unix (z. B. NetBSD) oder MacOS X benutzt werden. In diesem Kurs gehe ich von einem PC Betriebssystem mit DOS-ähnlicher Kommandozeile (DOS, Windows, OS/2) aus.

Weiterhin muss es möglich sein, die erstellten Atari Programme auch auszuprobieren. Dies kann mit einer Verbindung zwischen dem Übersetzungsrechner (PC) und dem Atari geschehen (z. B. per SIO2PC Kabel) oder mit einem Atari Emulator.

CC65 aus dem Internet laden#

Der CC65 ist ein "OpenSource" Programm und kann als solches zusammen mit dem Quelltext aus dem Internet geladen werden. Die Internetadresse ist <http://www.cc65.org>. Da der Haupt-FTP Server ein wenig eigen mit der korrekten Konfiguration des Webbrowsers und der Namensauflösung ist, sollte man bei Problemen die angegebenen Spiegel ("Mirror") Server benutzen. Der CC65 Compiler wird auf dem Server in verschiedenen Paketen angeboten:

- cc65-dos32-2.6.2.zip Compiler für DOS im 32Modus
- cc65-os2-2.6.2.zip Compiler für OS/2 mit EMX Bibliotheken
- cc65-win32-2.6.2.zip Compiler für Windows 32Bit Systeme (9x,ME,2000,XP)

Im Unterverzeichnis "RPM" des Servers finden sich Pakete für Linux Systeme mit RedHat Package Manager (RPM). Von diesen Compilerpaketen wird ein Paket für das verwendete Betriebssystem benötigt.

Zusätzlich zum Compiler gibt es Pakete für die verschiedenen Zielplattformen:

- cc65-apple2-2.6.2.zip Dateien für die Apple II Entwicklung
- cc65-atari-2.6.2.zip Dateien für die Atari 8-Bit (800, 800 XL, 130 XE) Entwicklung
- cc65-c128-2.6.2.zip Dateien für die Comodore C128 Entwicklung
- cc65-c64-2.6.2.zip Dateien für die Comodore C64 Entwicklung
- cc65-cbm610-2.6.2.zip Dateien für die Comodore CBM610 Entwicklung
- cc65-pet-2.6.2.zip Dateien für die Comodore PET Entwicklung
- cc65-plus4-2.6.2.zip Dateien für die Comodore Plus4 Entwicklung

- cc65-geos-2.6.2.zip Dateien für die GEOS Oberfläche (C64) Entwicklung

Zusätzlich finden sich in der Datei cc65-sources-2.6.2.tar.gz die Quelldateien des Compiler.

Installation#

Für die Entwicklung von Programmen von Atari 8-Bit Systeme wird ein Paket für den Compiler (je nach Betriebssystem) und das Atari 8-Bit Paket benötigt. Diese Dateien werden mit einem Entpacker (infoZip unzip oder WinZip) in ein Unterverzeichnis geschrieben. Im Verzeichnis CC65 sollte nun fünf Verzeichnisse (bin, doc, include, lib, samples) und eine Datei "announce.txt" enthalten.

Das "bin" Verzeichnis enthält die ausführbaren Programmbestandteile des Compilers: den C-Compiler (cc65.exe), einen Linker (ld65.exe), eine Compiler-Shell (cl65.exe), einen Assembler (ca65.exe), Objekt-Daten Ausgabe (od65.exe), Disassembler (da65.exe). Der GEOS Resource Compiler wird auf dem C64 für die Programmierung der GEOS Oberfläche benötigt und ist für die Atari Programmierung nicht von Bedeutung. Es ist sinnvoll, das "bin" Verzeichnis mit in den Suchpfad für Programme aufzunehmen, z. B. mit dem Befehl "PATH=%PATH%;c:\cc65\bin" in der AUTOEXEC.BAT.

Der Compiler, Assembler und Linker arbeiten zusammen um ein ausführbares Programm zu erzeugen. Damit diese drei Programme nicht einzeln aufgerufen werden müssen, erledigt die CompilerShell (cl65.exe) dies automatisch. Der Compiler benötigt zum übersetzen die Header-Dateien (.h). Diese befinden sich im "INCLUDE" Verzeichnis. Damit der Compiler diese findet, muss die Systemvariable CC65_INC auf das "INCLUDE" Verzeichnis gesetzt werden, z. B. mit dem Befehl "SET CC65_INC=C:\cc65\include" in der AUTOEXEC.BAT.

Ein Programm übersetzen#

Im Verzeichnis "samples" befinden sich Beispiel-Quelldateien, an denen wir die Compilerinstallation ausprobieren können. Haben wir den Programmpfad und die Include-Variable gesetzt, wechseln wir in das Verzeichnis "samples". Dort liegt der Quellcode des "Hello-World" Programms ("hello.c"). Ein Aufruf von "cl65 -t atari hello.c" startet den Übersetzungsvorgang. Da der CC65 standardmäßig Programme für den C64 erzeugt, gibt der Parameter "-t atari" an, das wir ein Programm für den Atari 8-Bit Computer übersetzen wollen. Wem dies nicht behagt, der kann den CC65 aus den Quellen neu übersetzen und den Atari 8-Bit als Standardsystem einsetzen.

Alles geklappt? Nein? Ok, es gab möglicherweise die Fehlermeldung "Error: Cannot open `atari.o': No such file or directory". Dies ist eine unschöne Eigenheit der "nicht-Unix" Versionen des CC65. Der Linker benötigt die Dateien "atari.o" und "atari.lib" aus dem "lib" Verzeichnis, und er sucht sie im aktuellen Verzeichnis sowie im Verzeichnis "/usr/lib/cc65/lib/". Auf Unix Systemen führt das zum Erfolg, bei Dos, Windows etc leider nicht. Ich kenne derzeit drei Möglichkeiten, diesen Fehler zu umgehen:

1. die Dateien "atari.o" und "atari.lib" in das jeweils aktuelle Verzeichnis mit den Quelldateien kopieren (in diesem Fall "samples")
2. die Quelldatei des CC65 ändern und den CC65 aus den Quellen neu übersetzen
3. mit einem Hexeditor den Eintrag in ld65.exe ändern (Vorsicht, vorher eine Sicherheitskopie von ld65.exe anfertigen)

Nachdem wir nun diesen Fehler umgangen haben können wir die Übersetzung erneut starten. Wir führen wieder "cl65 -t atari hello.c" aus und finden, wenn alles klappt, zwei neue Dateien im Verzeichnis: "hello.o" und "hello". "hello.o" ist die Objektdatei vor der Bearbeitung durch den Linker, d.h. ohne den atarispezifischen Ladevorspann in der Datei, welche dem DOS angibt, an welcher Stelle im Speicher das Programm geladen wird und wo es gestartet wird. "hello" ist das fertige Programm, wir können es ohne weiteres in "hello.com" umbenennen.

Programm ausprobieren: <#>

Ich benutze hier zum Test des Programms den XFormer Atari Emulator. Es kann jedoch auch ein anderer Atari Emulator oder ein original Atari Rechner benutzt werden. Der Befehl "xformer dos25.xfd hello.com" startet den Atari mit einer DOS 2.5 Diskette in Laufwerk 1 und dem neuen Programm in Laufwerk 2. Nun mit "L" und der Eingabe "D2:HELLO.COM" das Programm gestartet. Es sollte ein schwarzer Bildschirm mit weissem Rand und dem Text "Hello world!" zu lesen sein.

Und weiter....

Soweit zum ersten Teil unseres Kurses. Im nächsten Teil werden wir uns näher mit der Programmiersprache "C" beschäftigen, die wichtigsten Unterschiede zu Basic lernen und ein Basic Programm nach "C" übersetzen. Bis dahin viel Spass mit CC65.

Carsten Strotmann, 2002

erschieden im ABBUC Magazin