



How-To Board4You

Codename: BERGAMOTTE

Autor

Roman Gassmann

Hochschule für Technik Rapperswil

Electronics4You

CH-8640 Rapperswil, 22. Juni 2016

Dieses Dokument wurde mit L^AT_EX gesetzt.

© by Roman Gassmann.

Die Arbeit wurde ausschliesslich mit freierhältlicher Software erstellt.

Bilder wurden mit Tikz (3.0.1a), Gimp oder InkScape erstellt/editiert.

1 Toolchain

Wie bereits bekannt ist, muss jedes Programm das geschrieben wurde, von einem Compiler in einen Maschinen-Code übersetzt werden. In vielen Fällen wird dabei ein Compiler nicht alleine installiert, sondern noch eine Vielzahl zusätzlicher Tools wie z.B. das «Make» zur Automatisierung des Build- und Kompilervorgangs etc.. Dabei wird das Paket mit all diesen zusätzlichen Tools als Toolchain bezeichnet. Eine der gängigsten Toolchain ist dabei die GNU-Toolchain welche frei erhältlich ist. Des Weiteren ist auch deren Source offen gelegt, und kann somit mit einem kleinen Aufwand auf neue / andere Systeme angepasst werden.

Die GNU Toolchain beinhaltet dabei folgende Pakete:

Make

Automatisierung des Build- und Kompilervorgangs

Compiler Collection (GCC)

Compiler für verschiedene Programmiersprachen

GNU Binutils

Sammlung von Programmierwerkzeugen darunter sind der Assembler (as), der Archiv ersteller (ar), der Linker (ld), der Objektdatei-Kopierer (objcopy) und der Objektdatei-Dumper (objdump) die bekanntesten.

GNU Debugger (GDB)

Wie der Name bereits andeutet handelt es sich hier um den Debugger.

GNU Build System (GNU Autotools)

Tools zur Portierung von Quellcode-Paketen auf Unix-Systemen. Es sind folgende tools: Autoconf, Autoheader, Automake, Libtool.

Für sämtliche ARM-Prozessoren existiert dabei ein GNU ARM Toolchain. Diese ist unter dem Namen: gcc-arm-none-eabi bekannt. Sie kann unter <https://launchpad.net/gcc-arm-embedded> heruntergeladen werden wobei natürlich auf das vom Benutzer eingesetzte Betriebssystem zu achten ist.

1.1 Installation

1.1.1 Windows

Download von gcc-arm-none-eabi-....exe ist diese auszuführen. Es erscheint der Installations-Assistent welcher mit Weiter zu bestätigen ist. Das Anschliessende Lizenzabkommen muss Angenommen werden. Es wird anschliessend nach dem Installationspfad (C:\Program Files (x86)\GNU Tools ARM Embedded\) gefragt. Dieser kann zwar verändert werden, es wird jedoch nicht empfohlen. Mit Installieren kann die Installation nun gestartet werden. Beim Abschluss der Installation sollten nur die Flags `Add path to environment variable` und `Add registry information` aktiviert werden.



1.1.2 Linux

Es sind die Pakete:

- gcc-arm-none-eabi
- libstdc++-arm-none-eabi-newlib
- binutils-arm-none-eabi
- libnewlib-arm-none-eabi

zu installieren.

1.1.3 MAC

2 Codeblocks

Als IDE (Integrated Development Environment, deutsch Entwicklungsumgebung) wird Codeblocks empfohlen, da diese eine einfache und überschaubare Grösse hat. Dies ist bei vielen Anderen IDE's wie etwa Eclipse oder auch das von Freescale empfohlene KDS (auf Eclipse basierend) nicht der Fall, was zu einer unnötigen Belastung des Programmierers führt. Codeblocks ist ebenfalls frei erhältlich und kann unter <http://www.codeblocks.org> heruntergeladen werden.

2.1 Installation

2.1.1 Windows

Für Windows-User stehen mehrere Versionen zum Download zur Verfügung. Abhängig ob bereits ein Compiler installiert ist und Codeblocks zusätzlich für die Erstellung von Computer-Programmen eingesetzt wird, sind Folgende Downloads empfohlen:

Computer Programme erstellen gewünscht und kein Compiler `codeblocks-x.xmingw-setup.exe`

Gewünschte Compiler sind vorhanden `codeblocks-x.x-setup.exe` oder ggf. `codeblocks-x.x-setup-nonadmin.exe`

Es ist die Installation ganz normal vorzunehmen. Beim ersten Start von Code::Blocks wird nach dem default Compiler gefragt, wobei dieser meist automatisch gefunden wird.

2.1.2 Linux

Es sind die Pakete `gcc-arm-none-eabi` , `libstdc++-arm-none-eabi-newlib` , `binutils-arm-none-eabi` sowie `libnewlib-arm-none-eabi` zu installieren.

2.1.3 MAC

Ist noch nicht bekannt.

2.2 Konfiguration

Es muss nun der Compiler bzw. die ganze Toolchain abhängig vom Betriebssystem in Codeblocks konfiguriert werden. Dies kann im Menue **Settings** unter **Compiler** vorgenommen werden. Bevor irgendwelche Änderungen vorgenommen werden, ist unter **Selected Compiler** der **GNU GCC Compiler for ARM** auszuwählen. Anschliessend werden unter dem Reiter **Toolchain executables** folgende Einstellungen vorgenommen:

2.2.1 Windows

Compiler's Installation directory: Verweis auf den Installtionspath der Toolchain. Falls bei der Installation der default Path verwendet wurde muss hier

C:\Program Files (x86)\GNU Tools ARM Embedded\5.2 2015q4

angegeben werden (evt. wurde eine andere Toolchain Version installiert es ist dann natürlich der Pfad anzupassen).

C Compiler: arm-none-eabi-gcc.exe

C++ Compiler: arm-none-eabi-g++.exe

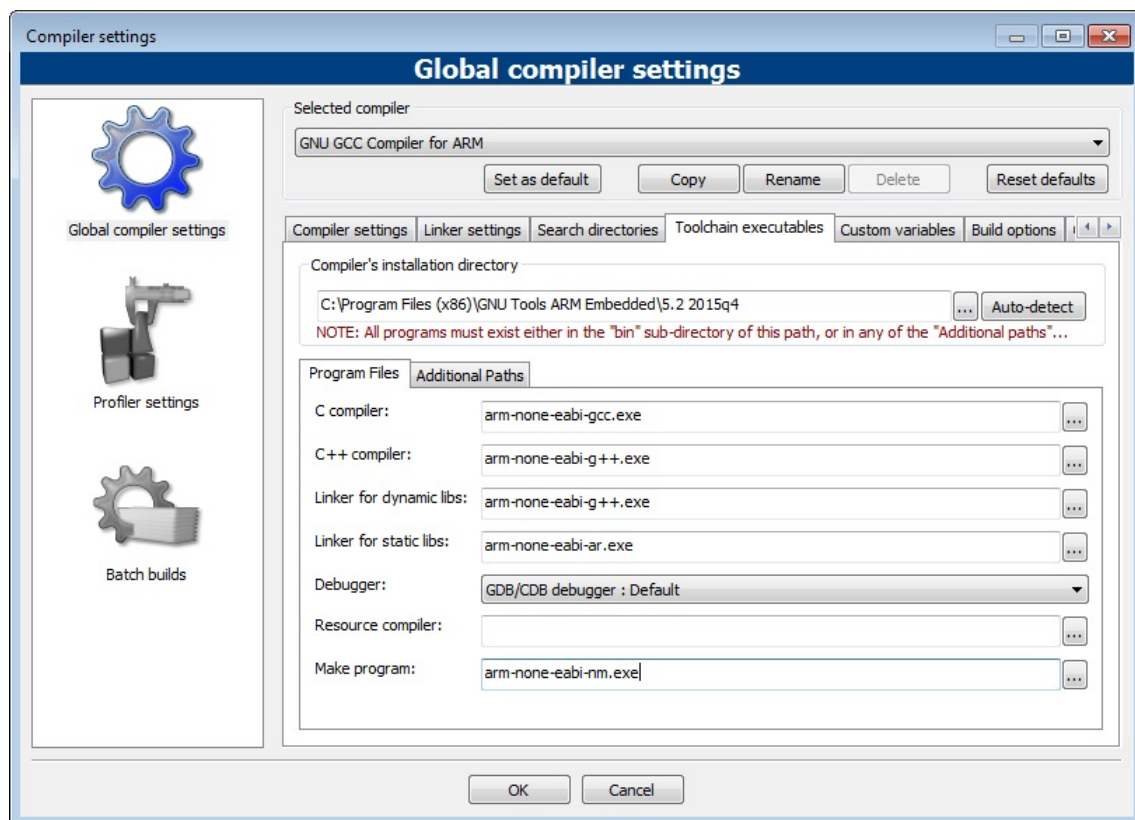
Linker for dynamic libs: arm-none-eabi-g++.exe

Linker for static libs: arm-none-eabi-ar.exe

Debugger: GDB/CDB debugger: Default

Resource compiler:

Make program: arm-none-eabi-nm.exe



2.2.2 Linux

Compiler's Installation directory: Verweis auf den Installtionspath der Toolchain. /usr/bin

C Compiler: arm-none-eabi-gcc

C++ Compiler: arm-none-eabi-g++

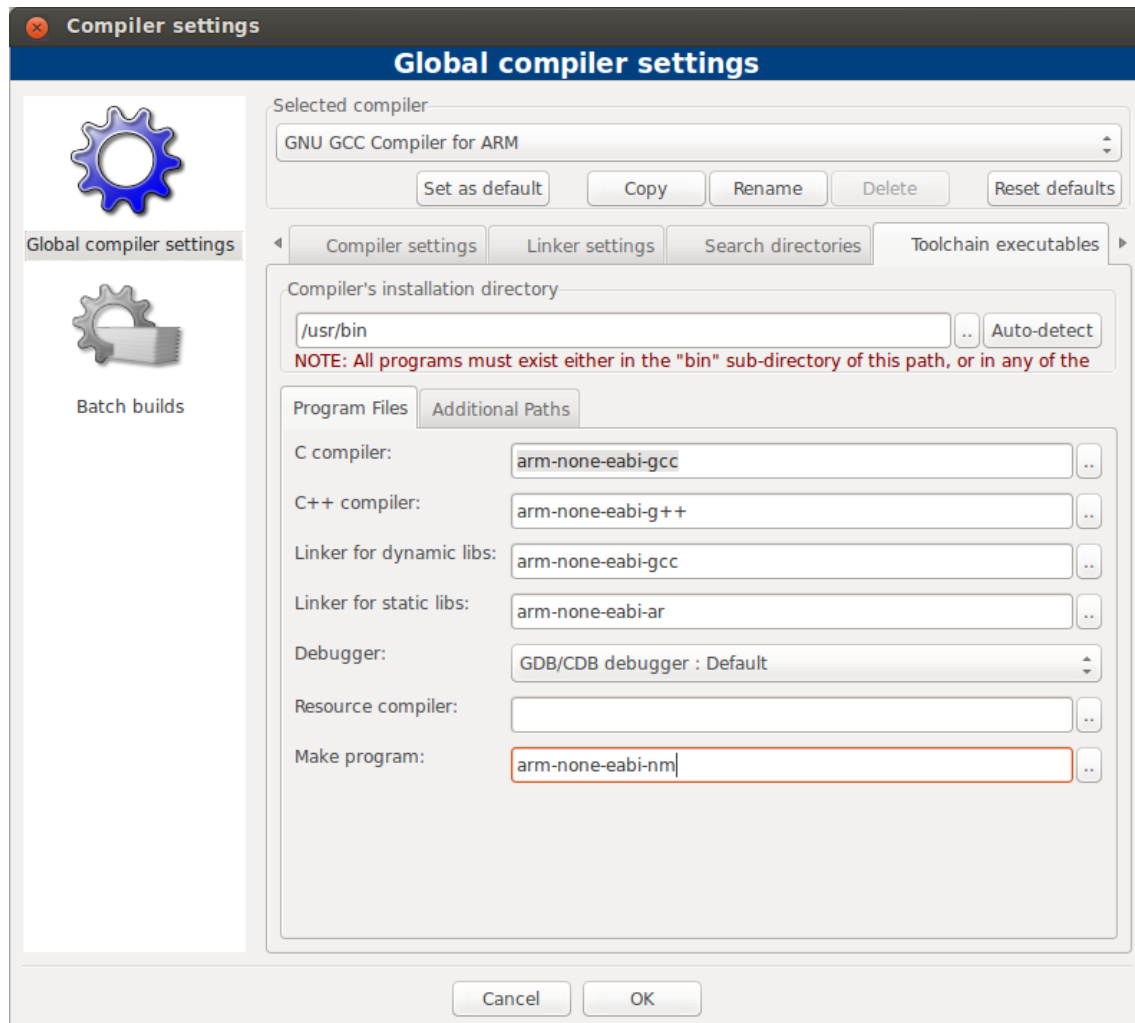
Linker for dynamic libs: arm-none-eabi-gcc

Linker for static libs: arm-none-eabi-ar

Debugger: GDB/CDB debugger: Default

Resource compiler:

Make program: arm-none-eabi-nm



2.2.3 MAC

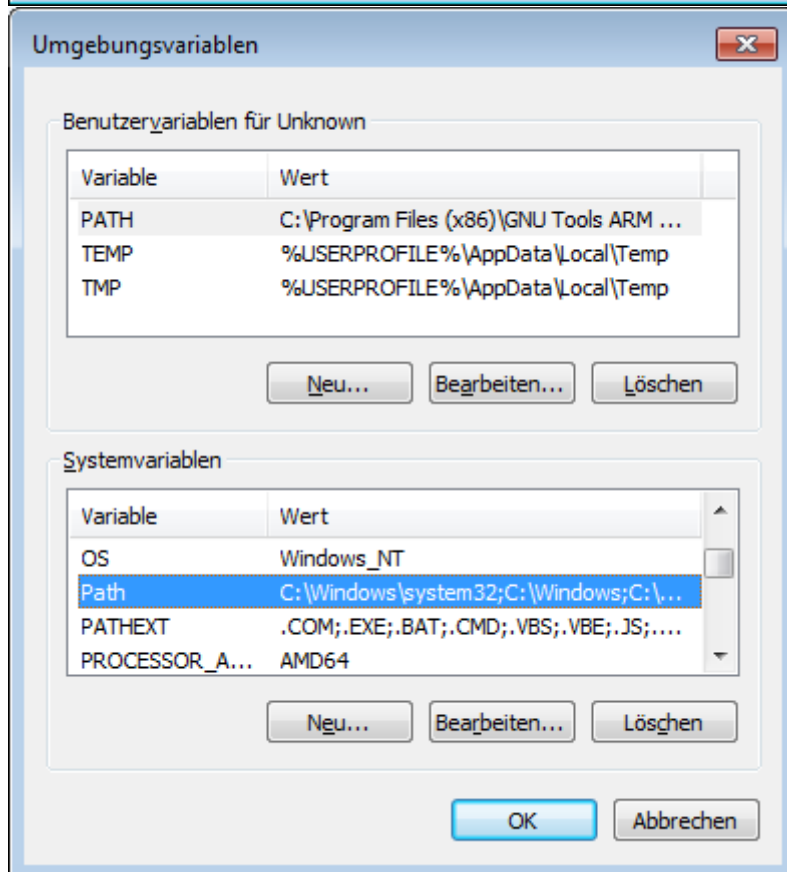
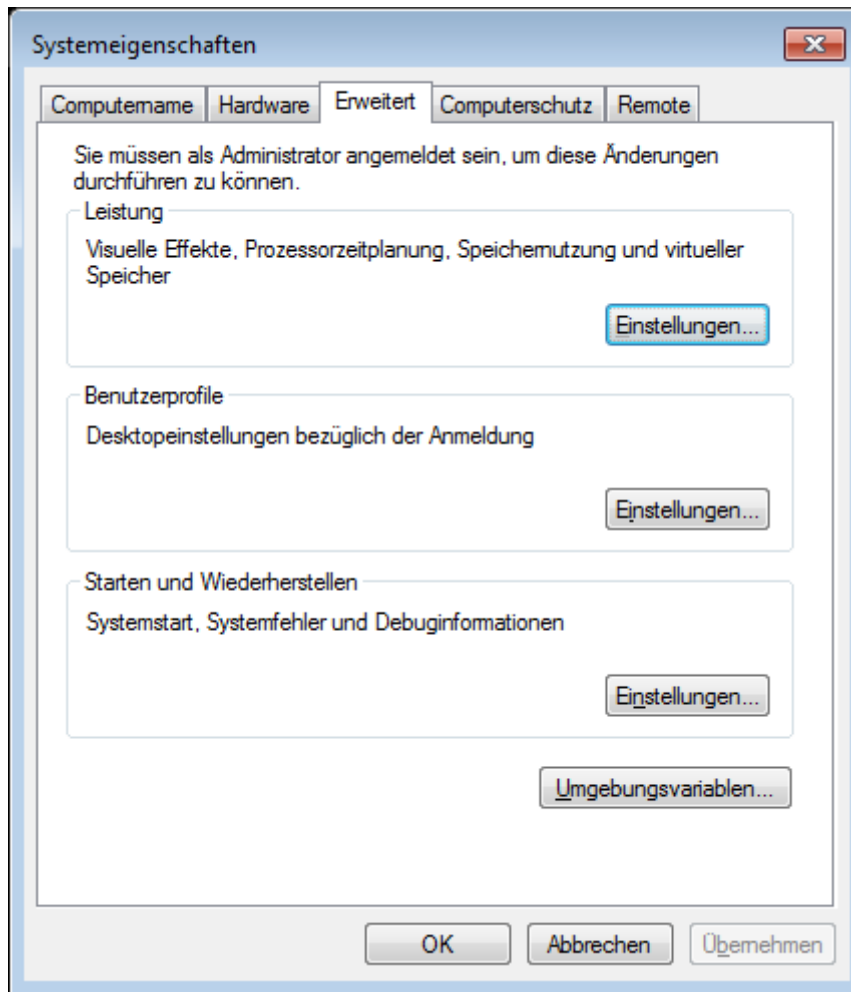
3 Board4You

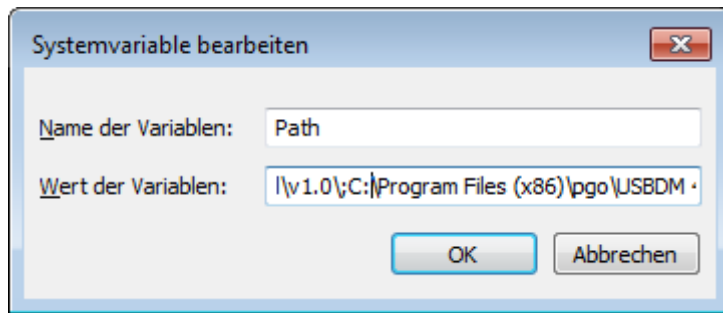
Alle FRDM-Boards haben einen HW-Downloader integriert. Dabei ist standard mässig das OpenSDA installiert. Dies ist zwar ein einfach bedienbares Interface ist jedoch für eine häufige Programmierung aus dem Codeblocks nicht gerade ideal. Es wird deshalb empfohlen, das USBDM einzusetzen. Dies ist aber zur Zeit leider nur für Windows und Linux System nicht aber für MAC vorhanden.

3.1 Installation

3.1.1 Windows

Unter <http://sourceforge.net/projects/usbdm/> steht USBDM zur Verfügung. Zusätzlich steht unter dem selben Link auch ein Treiber zur Verfügung welcher ebenfalls benötigt wird. Zu finden ist er unter *Files* → *Version 4.xx.x* → *Drivers* → *entsprechender Treiber*. Es kann anschliessend die Installation durchgeführt werden. **Wichtig:** Der Installationspfad ist zu merken, da dieser noch benötigt wird. Nach abgeschlossener Installation ist in den Systemeinstellungen Systemsteuerung → System) unter Erweiterte Systemeinstellungen im Reiter Erweitert unter Umgebungsvariablen bei der Systemvariablen **Path** der Installationspfad des USBDM hinzu zufügen (Zur Trennung zu anderen Pfaden wird hier das Semikolon (;) benötigt).





Nach der Installation der Software müssen noch die Treiber installiert werden. Jetzt fehlt noch der letzte Schritt, die Software auf dem Downloader. Hierzu wird auf dem Board die RST Taste gedrückt und gehalten. Es wird anschliessend das Board mit dem PC verbunden. Nun kann die Taste wieder losgelassen werden. Es sollte sich jetzt ein neues Wechselmedium mit dem Namen BOOTLOADER einhängen. Auf dieses Laufwerk muss nun die Datei OpenSDAv1_Unique.ID.sx, welche sich unter dem Installtionspfad im Ordner **FlashImages\MKxx_new** befindet, kopiert werden. Es blinkt dabei die grüne LED D4 auf dem Board. Nach dem Kopiervorgang kann das Board vom PC getrennt werden und erneut verbunden werden (ohne das Drücken der Taste). Windows sollte nun neue Treiber installieren.

3.1.2 Linux

Unter <http://sourceforge.net/projects/usbdm/> ist das entsprechende Packet zu hohlen und zu installieren. Jetzt fehlt noch der letzte Schritt, die Software auf dem Downloader. Hierzu wird auf dem Board die RST Taste gedrückt und gehalten. Es wird anschliessend das Board mit dem PC verbunden. Nun kann die Taste wieder losgelassen werden. Es sollte sich jetzt ein neues Wechselmedium mit dem Namen BOOTLOADER einhängen. Auf dieses Laufwerk muss nun die Datei OpenSDAv1_Unique.ID.sx, welche sich im Ordner **/usr/share/usbdm/FlashImages/MKxx** befindet, kopiert werden. Es blinkt dabei die grüne LED D4 auf dem Board. Nach dem Kopiervorgang kann das Board vom PC getrennt werden und erneut verbunden werden (ohne das Drücken der Taste). Windows sollte nun neue Treiber installieren.

3.1.3 Mac

Es muss über mbed bzw. OpenSDA gefahren werden.

3.2 Benutzung

Unter <https://github.com/RGassmann/FRDM> stehen einige Beispielprojekte zur Verfügung. Diese dienen als Grundlage und haben bereits einige Konfigurationen enthalten. Für neue Projekte ist es empfehlenswert, das BASE-Projekt zu kopieren und umzubenennen. Zur entwicklung neuer Programme stehen folgende Dokumente auf dem Internet zur Verfügung:

- FRDM-KL25z-sch.pdf - Schema des EvalBoards
- KL25P80M48SF0RM.pdf - Reference Manual