



# Application Note 008: Tool chains for own stepRocker software development

TRINAMIC® Motion Control GmbH & Co. KG  
GERMANY

[www.trinamic.com](http://www.trinamic.com)

The stepRocker comes with a firmware that fully implements TMCL, the Trinamic Motion Control Language. With TMCL it is possible to solve many motion control tasks, either by controlling the stepRocker from a host computer or by programming the stepRocker with a TMCL program and using it in stand alone mode.

But sometimes it is also desirable to be able to develop an own stepRocker firmware in C, for example for motion control tasks that cannot be easily solved using standard TMCL. An open source C library for use with the GNU C compiler is provided for this case.

This application note provides an overview of the tool chain needed for stepRocker firmware development in C.

## 1 Table of contents

<b>1</b>	<b>TABLE OF CONTENTS</b> .....	<b>1</b>
<b>2</b>	<b>THE C LIBRARY</b> .....	<b>2</b>
<b>3</b>	<b>THE C COMPILER</b> .....	<b>2</b>
3.1	THE GNU C COMPILER (GCC).....	2
3.2	COMMERCIAL C COMPILERS .....	2
<b>4</b>	<b>GETTING YOUR OWN FIRMWARE ONTO THE STEPROCKER</b> .....	<b>2</b>
4.1	DOWNLOAD VIA USB.....	2
4.2	USING A PROGRAMMER .....	2
<b>5</b>	<b>REFERENCES</b> .....	<b>3</b>
<b>6</b>	<b>REVISION HISTORY</b> .....	<b>3</b>
6.1	DOCUMENTATION REVISION .....	3

[TRINAMIC Disclaimer, Info on TRINAMIC, German Quality Logo]

## 2 The C library

First of all, a C library is provided by Trinamic as an open source software project. It can be downloaded from Github. A doxygen documentation and a special manual for the library are also included [TMCL-LITE].

This C library provides functions for initializing the hardware and for easy access to the TMC429 motion controller, the TMC262 stepper driver and the EEPROM. It also provides functions for RS485 communication. USB and CAN are not supported by the current version of the library but will be added in a later revision.

It is strongly recommended to make use of this library (although it would also be possible to develop stepRocker firmware without using this library). Using this library will make firmware development faster and easier, and it will also be easier to share code with other stepRocker users.

## 3 The C compiler

### 3.1 The GNU C compiler (GCC)

As a C compiler we recommend using the GNU C compiler (GCC) for Cortex processors. All existing firmware has been developed using GCC and thus the library has also been developed for use with GCC. We recommend using the Codesourcery (now called CodeBench) distribution of GCC. It can be downloaded from [www.codesourcery.com](http://www.codesourcery.com) in different editions, equipped with different features.

The so called LITE Edition of the Codesourcery G++ (CodeBench) distribution is completely free of charge, and only contains the GNU command line tools. But this is completely sufficient for developing firmware for the stepRocker (the TMCL firmware for the stepRocker has been developed using only the command line tools). The only additional part of software you will need is a good source code editor.

For installing the command-line version the following steps are necessary:

- Download the ARM/EABI release of the CodeBench LITE distribution
- Unpack it on your computer
- Add the path to the PATH variable on your system
- Now you can compile your firmware by typing “make” on the command line

As an alternative you can also use Eclipse and integrate all the tools into it.

There are also many other free of charge distributions of the GCC for ARM/Cortex microcontrollers which can be used just the same way.

### 3.2 Commercial C compilers

There are also commercial C compilers for ARM/Cortex microcontrollers available. One example is the IAR C compiler. Some changes to the stepRocker C library have to be made to be able to use it with this compiler.

Another example for a commercial C compiler is the Keil C compiler. It is based on GCC.

## 4 Getting your own firmware onto the stepRocker

There are mainly two choices for downloading your own firmware to the stepRocker: either via the USB interface of the stepRocker, using the stepRocker boot loader and either the stepRocker boot PC application or the “Install OS” function of the TMCL-IDE, or via the programming interface on the bottom side of the stepRocker PCB. Both choices have their own advantages and disadvantages.

### 4.1 Download via USB

The main advantages of this method are its ease of use and that no extra hardware will be needed. Just the PC boot application or the TMCL-IDE (V2.04 or higher) and a USB cable will be needed. The PC boot application and also the TMCL-IDE are only available for Windows (but can be run on Linux using WINE), but one stepRocker user has already developed and own stepRocker boot tool for Linux (please see the motion control community forum for details). There is also a manual [TMCM-1110-BL] available that describes the firmware update protocol, so it is possible to develop download tools also for other operating systems.

Another advantage is that this can be done by every stepRocker user, so this method can also be used to deploy your own firmware to other users. It is just the same process as a normal firmware update (please see also the stepRocker C library manual [TMCL-LITE] on how to use this method).

The disadvantage of this method is that source level debugging will not be possible. Downloading a program via the USB interface also takes longer than downloading via a programmer. So the developing process may take longer.

### 4.2 Using a programmer

The main advantages of this method are that source level debugging will be possible and the slightly shorter download time. These two things can shorten the development process. For using this method some extra hardware will be needed:

- A programmer that supports SWD (single wire debug) programming
- The debugRocker (a PCB that allows connecting the stepRocker to the programmer easily)

The following two programmers have already been tested by Trinamic (but for sure there are many other programmers that will work, too):

- Segger J-Link: this programmer is also being used for the mass production of the stepRockers (in conjunction with the Segger J-Flash software). The software of this programmer also includes a GDB server that can be used for source level debugging together with GDB (GNU Debugger). The debug features of many IDEs for GCC (including Eclipse) are also based on GDB so that this programmer offers a fast and convenient way of firmware downloading and source level debugging. The J-Link programmer is quite expensive, but can be used with many compilers and IDEs. It can be purchased directly from Segger and also from many distributors.
- SM-Link: this programmer can be purchased via Samsung distributors, but it does not support debugging via GDB nor programming of firmware generated using GCC. Instead, it is supported by the IAR C compiler and IDE and cannot be used with other compilers and IDEs. It is cheap, but a rather expensive C compiler is needed to make use of it.

Please check the following two things if you plan to use other programmers:

- The programmer must support SWD programming at 5V level
- The software provided for the programmer must support the Samsung S3FN41 processor

There are several other (also open source) programmers that support SWD programming. But as the S3FN41 microcontroller is quite new on the market there is not much programming software around that supports this microcontroller. For example, the very common open source programming software "OpenOCD" does not support it in its current version (but does support SWD programming, so one could think of adding this feature himself).

Please note that when downloading firmware to the stepRocker via a programmer the boot loader will be overwritten. That means after doing so one will not be able to download a firmware via USB (as described in section 4.1) any more. But the boot loader code can also be downloaded from the motion control community website for programming it into the stepRocker again via a programmer. This will make USB firmware downloading possible again.

The disadvantage of this method is the cost of extra hardware and sometimes also extra software.

## 5 References

[TMCL-LITE]                      TMCL-LITE firmware manual  
 [TMCM-1110-BL]                stepRocker boot loader protocol manual

## 6 Revision history

### 6.1 Documentation revision

Version	Date	Author	Description
1.00	2012-Apr-03	OK	First version

Table 1: Documentation revisions