

M16C-Flasher – First Steps

General

The M16C-Flasher is a tool for the programming of Renesas-microcontroller. It uses the Standard Serial Mode 2 (async), this means you only need a serial cable between pc and microcontroller. There is no additional hardware required.

Keep care for the different voltages of the pc-comport (+12V & -12V) and the microcontroller (5V & 0V). Use a MAX232 or something similar.

The NON-PROFIT version of the M16C-Flasher is free for private, non-commercial use only. Otherwise you have to purchase a license.

Help

Most of the function are selfexplaining. There are tooltips for most buttons and options, so it should be easy to use.

This manual shows only the first steps with the M16C-Flasher. It explains some basic functions and problems.

First Start

You have to set the comport and baudrate. ('Settings'). The maximum baudrate depends on the microcontroller and the quartz-frequency. For the first test, I suggest 9600 baud.

Select protocol 'Revision 2' and 'Standard Connect'. (Like suggested in actual datasheets.)

If you use the Control-Lines for Reset and CNVSS, you have to keep care for the correct logic. (see also 'Control-Lines' in this manual)

Select the microcontroller...

Connect

Before you can do any function (programming, erasing, reading,...), you have to connect with the bootloader of the microcontroller. If you don't use the Control-Lines, you have to set CNVSS and make a reset of the mikrocontroller, to enter the bootloader mode.

Press 'Connect' and the M16C-Flasher tries to establish a communication to the bootloader program of the microcontroller. The M16C-Flasher displays the version of the bootloader (VERx.xx), if it was successful.

Now you can perform the function (e.g. programming).

M16C-Flasher – First Steps

ID of microcontroller

The microcontroller is protected by a 7 byte ID. Without the correct ID, it is not possible to read, write or erase the flash memory.

The NON-PROFIT version of the M16C-Flasher only writes the ID 0x00 0x00 0x00 0x00 0x00 0x00 0x00, so there is no protection.

You have also the option to log all used IDs into a file (ID.LOG)

By default the M16C-Flasher uses the last working ID. This is useful under most conditions. But if you are working at different devices with different IDs, you should select 'Extract ID from File'.

'ID mismatch' - Error

If the ID of M16C-Flasher doesn't match the ID of the program in the microcontroller, you will get a 'ID mismatch'-error. Insert the correct ID, turn off and on the power supply of the microcontroller and try again.

If you don't switch off the power supply, the ID-test will fail again. This is a protection against the 'bruce force' method to hack the ID.

Option 'Load File once'

You want to program to same file more often and you don't want to select the same file again and again in the 'Open'-dialog?

Just select the 'Load File once' option and you have to select the file only one time. The next time you press on 'Prog', it won't ask again...

Control-Lines

You want to make the programming more easy, you don't like the manual reset and setting of CNVSS?

The M16C-Flasher support the use of the control-lines RTS & DTR. With these lines you can reset the microcontroller and set it into the bootloader mode.

You find an example in the 'Setting' of M16C-Flasher.

To make the control-lines independent of your used logic, you can invert each signal.

Keep care of RDY & HLD, if you use the external bus.

AutoFlash-Function

To make programming as easy as possible the M16C-Flasher has the AutoFlash-function. Just program a MOT file once and the M16C-Flasher looks for changes of the file in the background. So if you recompile your project, the M16C-Flasher starts programming automatically.