

**Note on how to install
OpenBLP bootloader for**

OLIMEX's

EM32G880F128-STK

STM32-H103

STM32-P103

LPC-L2294

A friend of Olimex provides and supports a pretty nice bootloader released under GNU GPL altogether with a neat and tidy Wiki. The main page of this project is:

<http://feaser.com/openblt/doku.php>

There is a lot of info about the features of the project but I want to specifically point how well everything is organized and documented.

If you look the examples page there are a number of Olimex boards also supported:

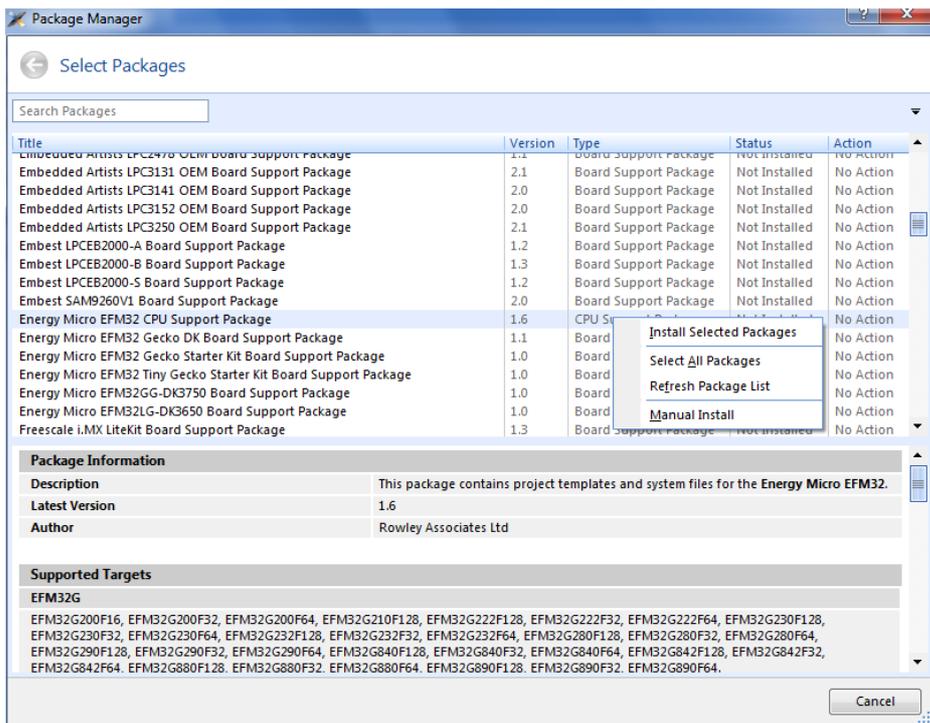
<http://feaser.com/openblt/doku.php?id=manual:demos>

Further more, there are very good guides on making everything work together using different ways of downloading the code on those boards.

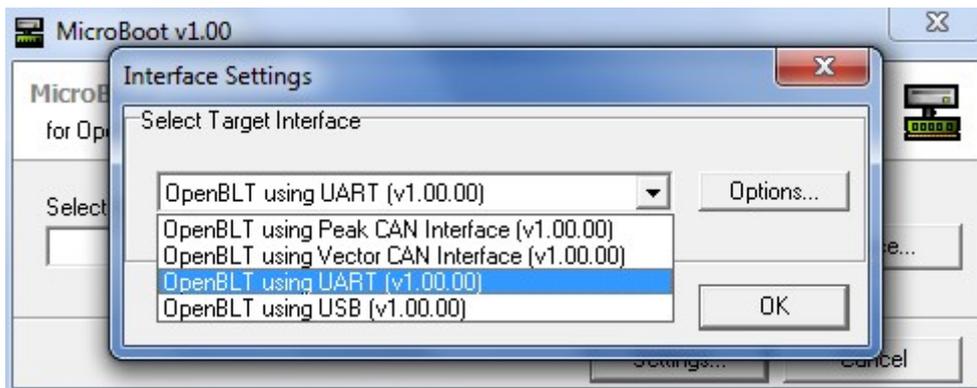
Having bootloader on EM32G880F128-STK is real treat, because that way a person doesn't need an expensive SWD debugger for the board. That being said the installation of the OpenBLT on the other boards doesn't differ much from the one I will present.

Below you will find the installation of OpenBLT on EM32G880F128-STK using OLIMEX ARM-USB-OCD combined with ARM-JTAG-SWD. The IDE I will be using is Rowley's Crossworks for ARM which supports our SWD interface out of the box.

- 1) First step is to install the Crossworks IDE and the driver for the ARM debugger. We also mount the SWD adapter between the debugger and the cable it is shipped with.
- 2) Open Crosswork IDE and make sure you right-click install the Energy Micro EFM32 support package. And also that you have selected ARM-USB-OCD..



- 3) Download the latest OpenBLT from <http://sourceforge.net/projects/openblt/> and then extract it.
- 4) Open \Target\Demo\ARMCM3_EFM32_Olimex_EM32G880F128STK_Crossworks\Boot\ide\ folder and locate the project file and open it. Clean and rebuild the project
- 5) Upload it to the board using the debugger with the SWD adapter.
- 6) Start the Microboot utility (found in \Host\MicroBoot.exe” needs to be configured to connect to the correct COM-port and to communication at the correct baudrate- click the “Settings”-button and select “OpenBLT using UART” from the dropdown menu. Click the “Options”-button to configure the communication parameters. Select the PC's COM-port that the board is connected to. Set the baudrate to 9600 bits/sec from the Device manager.

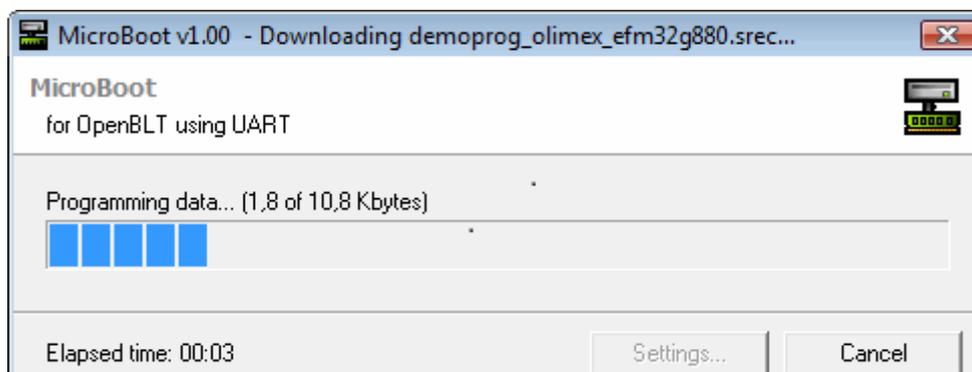


- 7) Once you saved the settings by clicking the “OK”-button twice, you can close Microboot by clicking the “Cancel”-button on the main screen. Microboot is now ready for action!

Now it is time to check how the bootloader works

- 8) We open, clean and rebuild the project file found in:
\Target\Demo\ARMCM3_EFM32_Olimex_EM32G880F128STK_Crossworks\Prog\ide\
- 9) The output file can be found in
Target\Demo\ARMCM3_EFM32_Olimex_EM32G880F128STK_Crossworks\Prog\bin\demoprogram_olimax_efm32g880.srec

To download this demoprogram_olimax_efm32g880.srec file to the remaining flash memory using the bootloader, start the Microboot program and click the “Browse”-button. Browse to the “\Target\Demo\ARMCM3_EFM32_Olimex_EM32G880F128STK_Crossworks\Prog\bin\” directory and select the demoprogram_olimax_efm32g880.srec file to start the download.



Once the download completed, the newly programmed software will be started by the bootloader. You can verify this by checking that the test “Olimex” blinks on the LCD. Congratulations! That's

all you have to do to use the bootloader.

If you like the idea of the OpenBLP you can check on how to help the project:

<http://feaser.com/openblt/doku.php?id=contributions>

Remember to follow the guides provided here in case you want to use GCC or IAR, or if you want to download a bootloader using STM32H103, STM32P103 or LPC-L2294:

<http://feaser.com/openblt/doku.php?id=manual:demos>