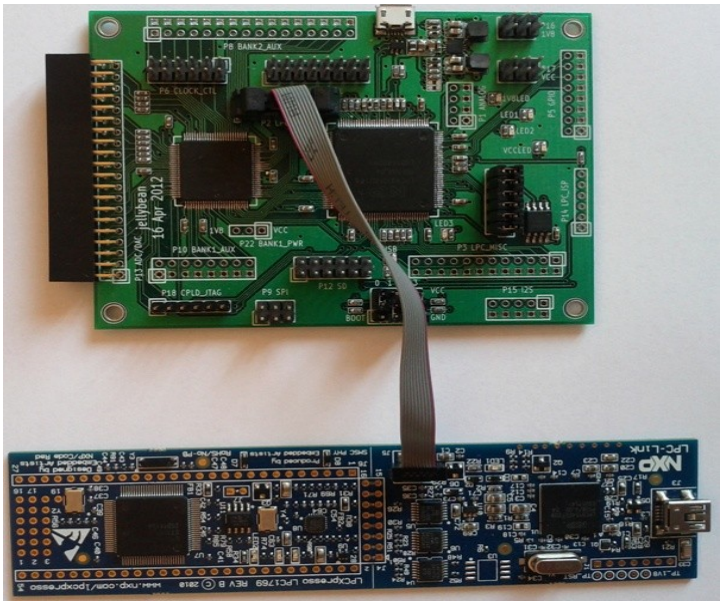
LPCXpresso Flash/Debug Tutorial

Hardware required

1) NXP LPC-Link board included with any LPCXPRESSO Board (for 20Euros) from Embedded Artists see http://www.embeddedartists.com/products/lpcxpresso
To separate the LPC-LINK side and the target side (required to use the debugger JTAG/SWD) on an other board), see document:
http://www.embeddedartists.com/sites/default/files/docs/HowToCreateAnLPC-LINK.pdf
The suggested LPCXpresso boards are LPC11U14, LPC1227 and LPC1769 which just need the use of desoldering braid to remove the solder bumps, for other board it is required to cut some wires.
For my Example I use a LPCXPresso LPC1769 board.

2) LPC43xx board in this tutorial I will use the HackRF board called JellyBean designed by Michael Ossmann (thanks for this amazing board ;-).
For this tutorial HackRF JellyBean board is used and example used is miniblink based on GitHub
https://github.com/mossmann/libopencm3/tree/master/examples/lpc43xx/hackrf-jellybean/miniblink
Main GitHub is https://github.com/mossmann/libopencm3



Nota: Pin1 is Red Pin on the 10-pin mini JTAG cable.
Nota I bought the mini JTAG cable at Embedded Artists here:
http://www.embeddedartists.com/products/acc/acc_idc_10.php

Software required

LPCXpresso v4.2.3 build 292 (size 213.4 MB):
Requires an account (Login/Password).
Windows version:
http://lpcxpresso.code-red-tech.com/LPCXpresso/system/files/LPCXpresso_4.2.3_292.exe
or
ftp://lpcxpresso.code-red-tech.com/LPCXpresso_4.2.3_292.exe

Linux version:
http://lpcxpresso.code-red-tech.com/LPCXpresso/system/files/Installer_LPCXpresso_4.2.3_255_Linux-x86.tar_.gz

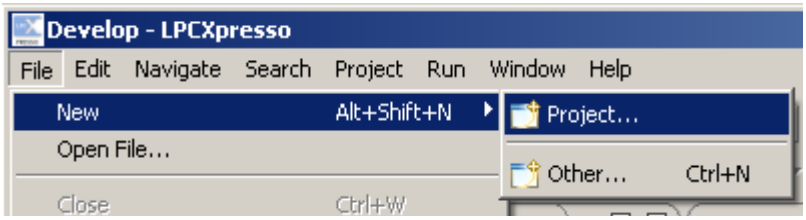
Nota: Tutorial is done with Windows XP SP3 and LPCXpresso v4.2.3 build 292 for windows.

Starting LPCXpresso IDE

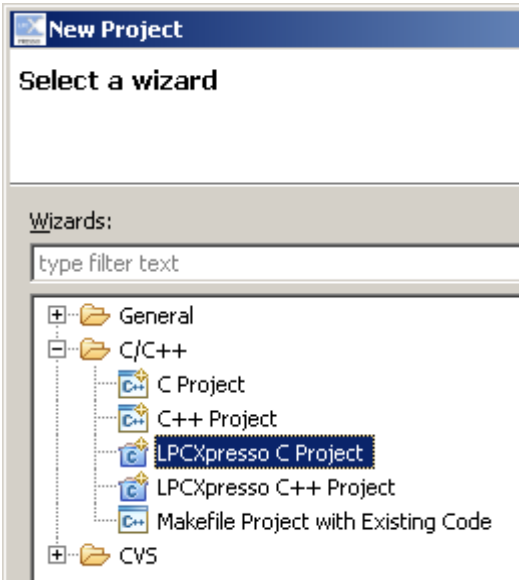
Before starting LPCXpresso you should register it (registration code shall be activated online with the card you will receive with your LPCXPRESSO Board), the registered version is only limited to debug code of max 128KB.

Create a project

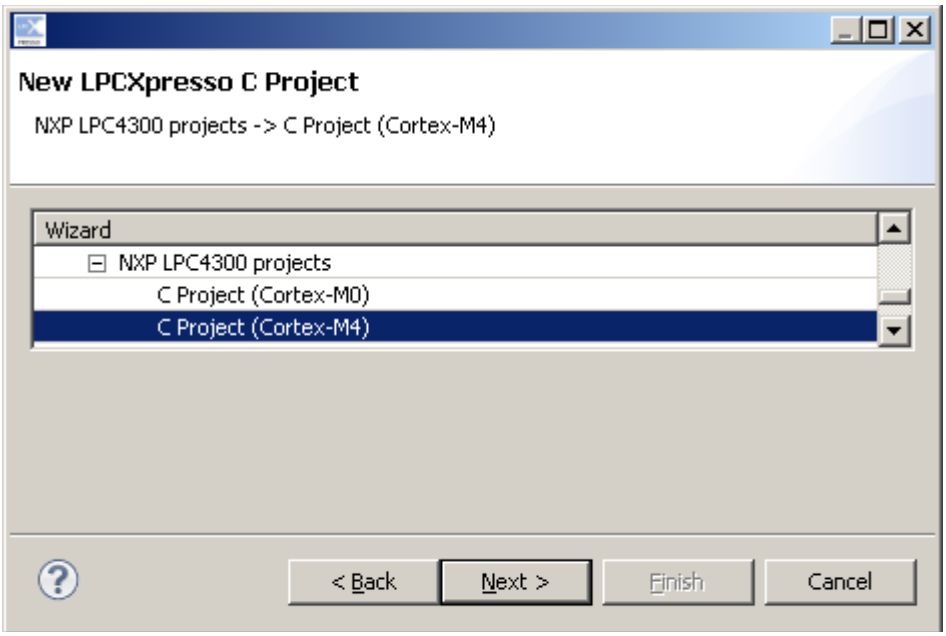
Choose Menu File->New->Project



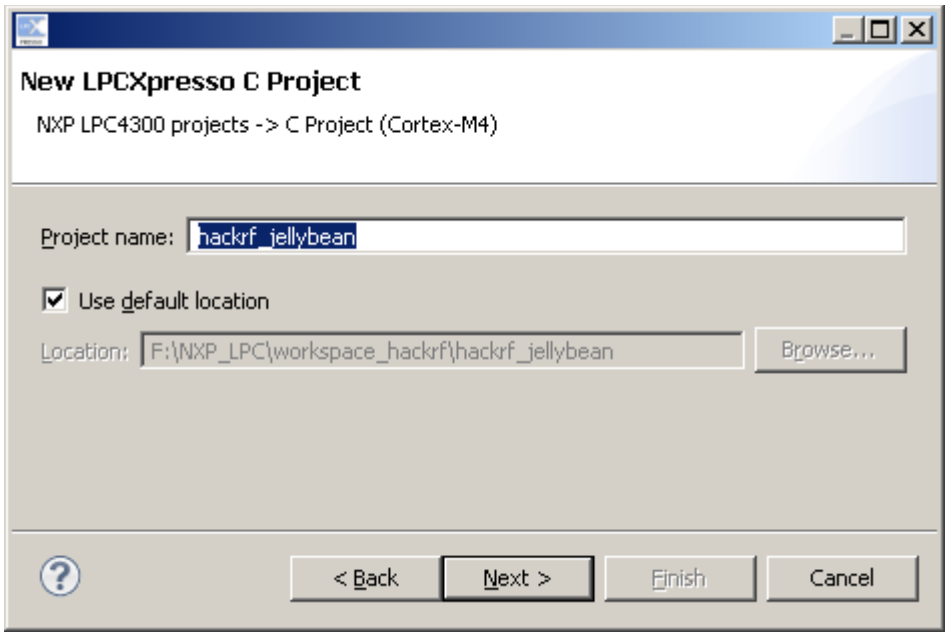
Select C/C++ & LPCXpresso C project and click **Next**



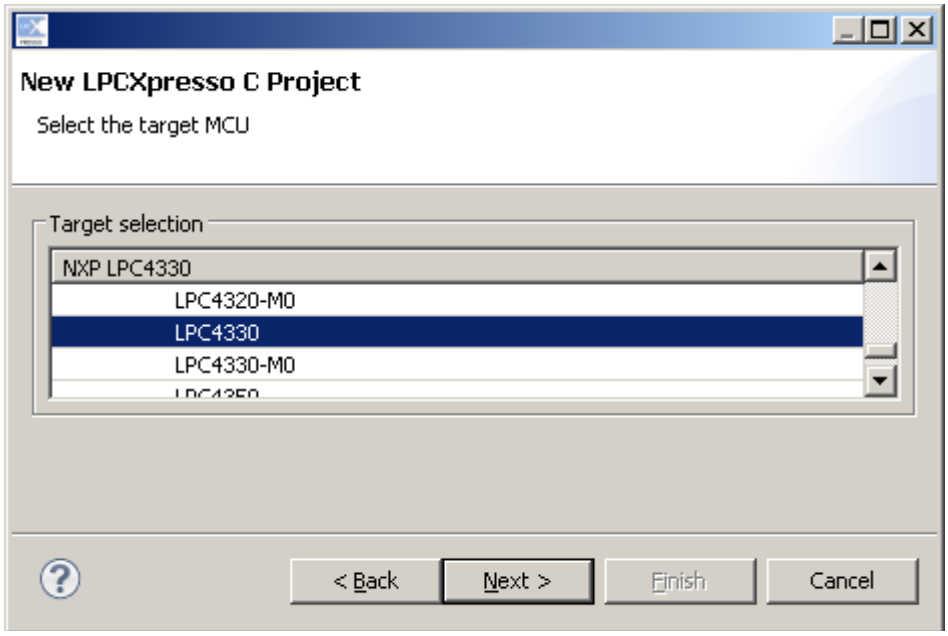
Select NXP LPC4300 -> C Project (Cortex M4) and click **Next**



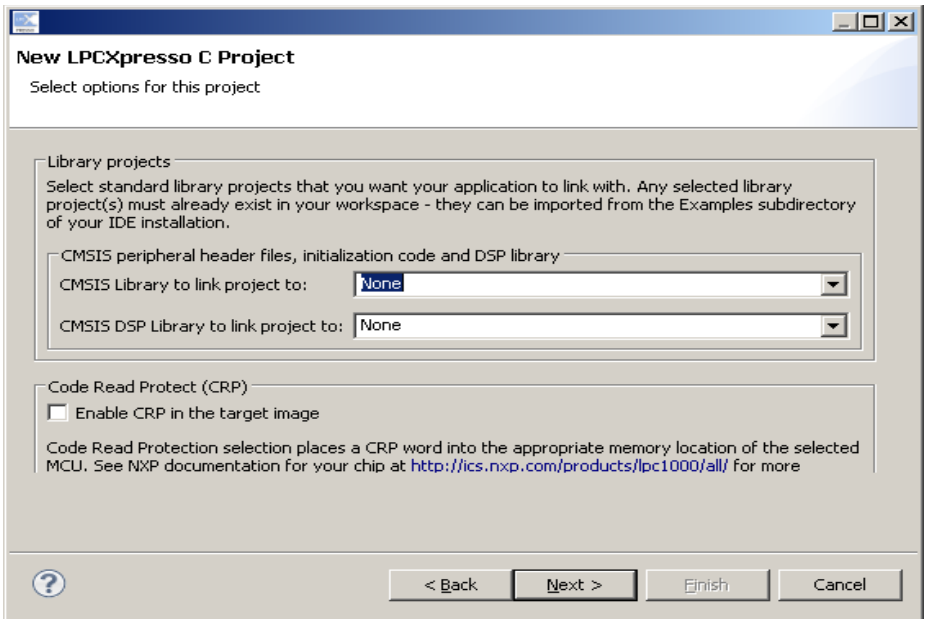
Choose project Name in the example hackrf_jellybean and click **Next**



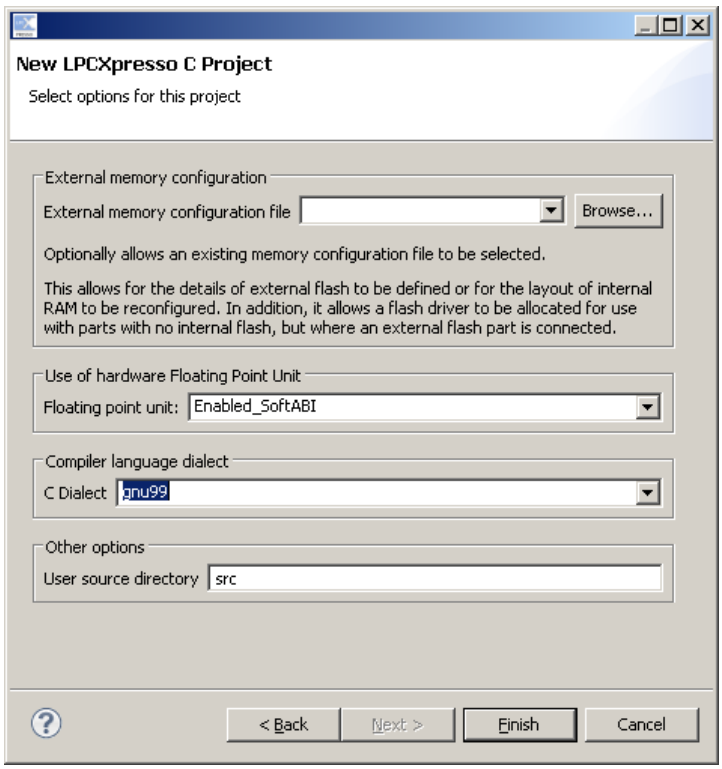
In Target Selection choose LPC4330 (for JellyBean) and click **Next**



For CMSIS DSP or Library just choose None for both and click **Next**



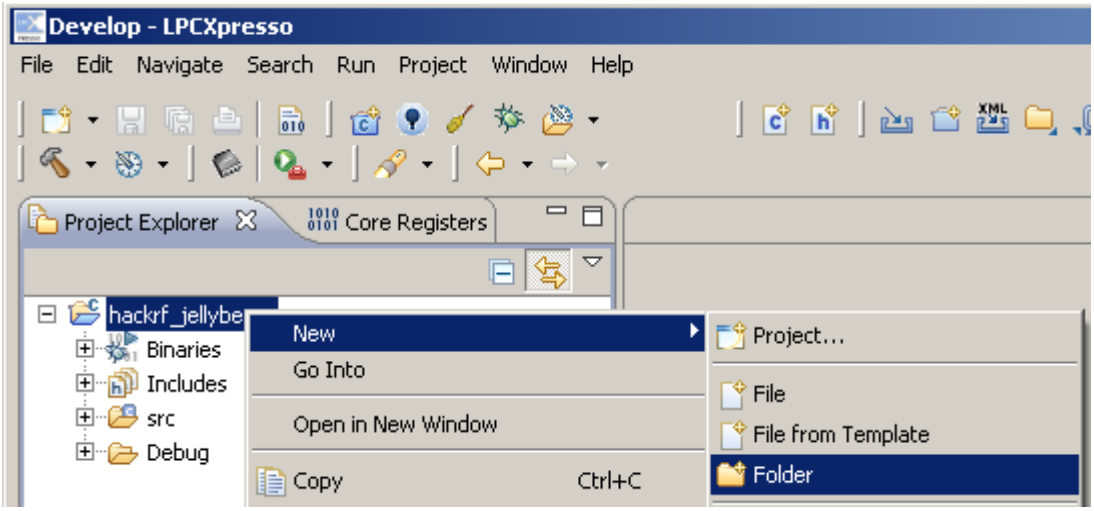
Just change C Dialect to gnu99 or what you prefer and click **Finish**



A basic project is now created.

You can import existing source project using link:

With mouse right click on project, choose New -> Folder

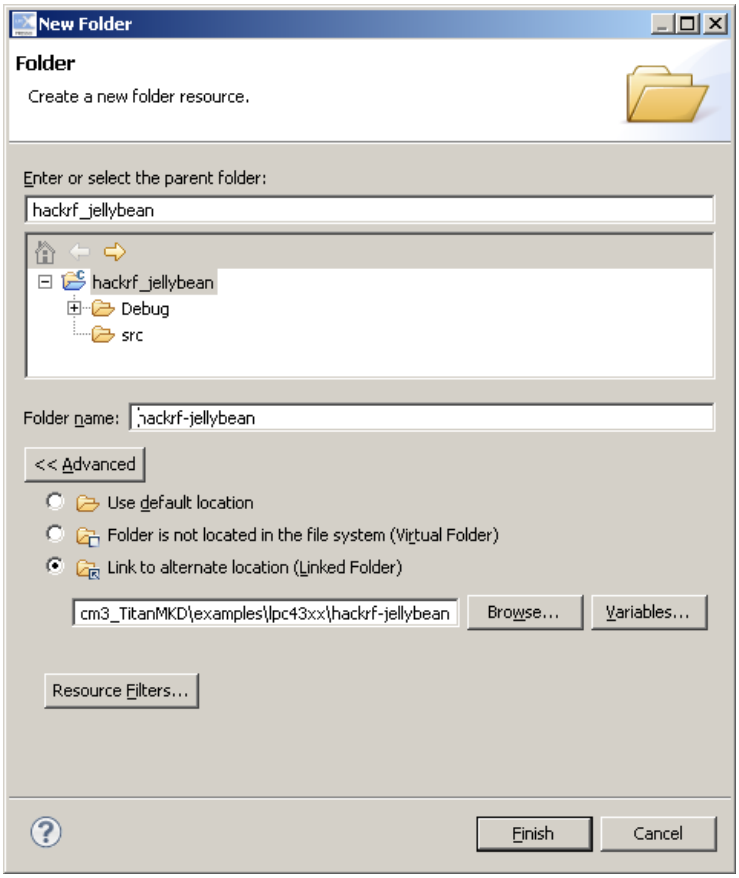


In Folder name: Enter the full path in the example hackrf-jellybean

Then click on Advanced and choose "Link to alternate location (Linked Folder)

And choose the full path in the example F:\SVN\libopencm3_TitanMKD\examples\lpc43xx\hackrf-jellybean

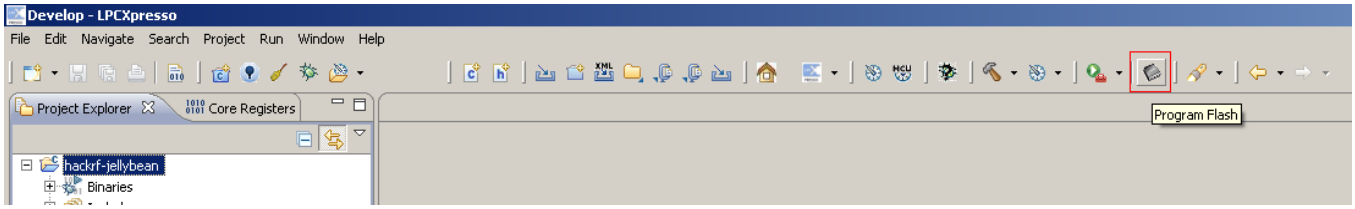
Then click on **Finish**.



Now you have access to you external source from LPCXpresso IDE.

Flashing ".bin" or ".elf" in SPIFI flash memory

With mouse click on project hackrf-jellybean previously created and click on the "Chip/Component" Icon



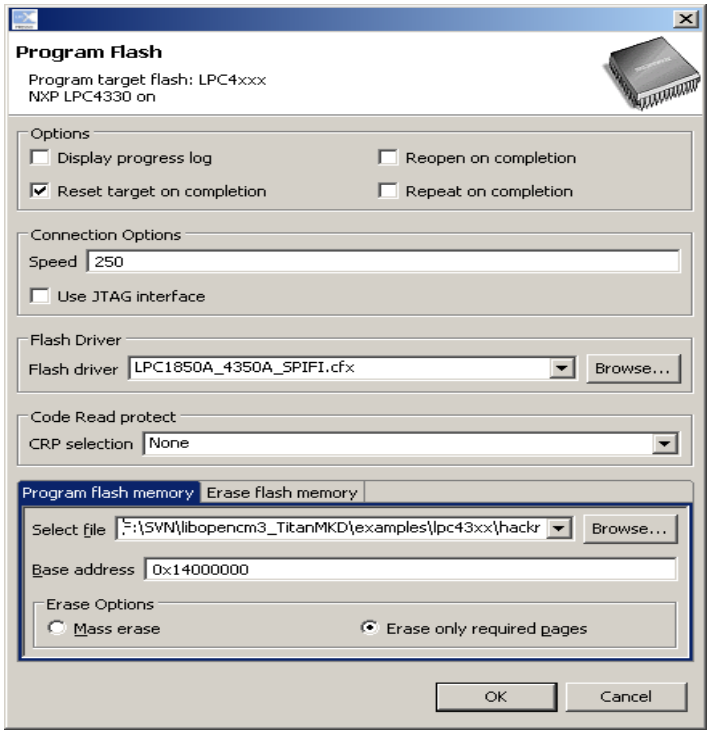
Configure Flash with following parameters:

1) Choose Flash driver file (using Flash driver Browse) in Default Installation path \LPCXpresso_4.2.3_292\lpcxpresso\bin\Flash\LPC1850A_4350A_SPIFI.cfx

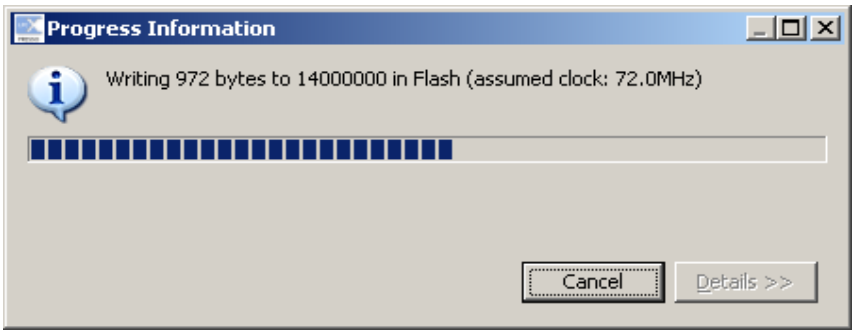
2) Select the file to write in flash in our example F:\SVN\libopencm3_TitanMKD\examples\lpc43xx\hackrf-jellybean\miniblink\miniblink.bin

3) Choose base address: 0x14000000

Click OK to start.



Progress shall be displayed and all disappear when it is finished



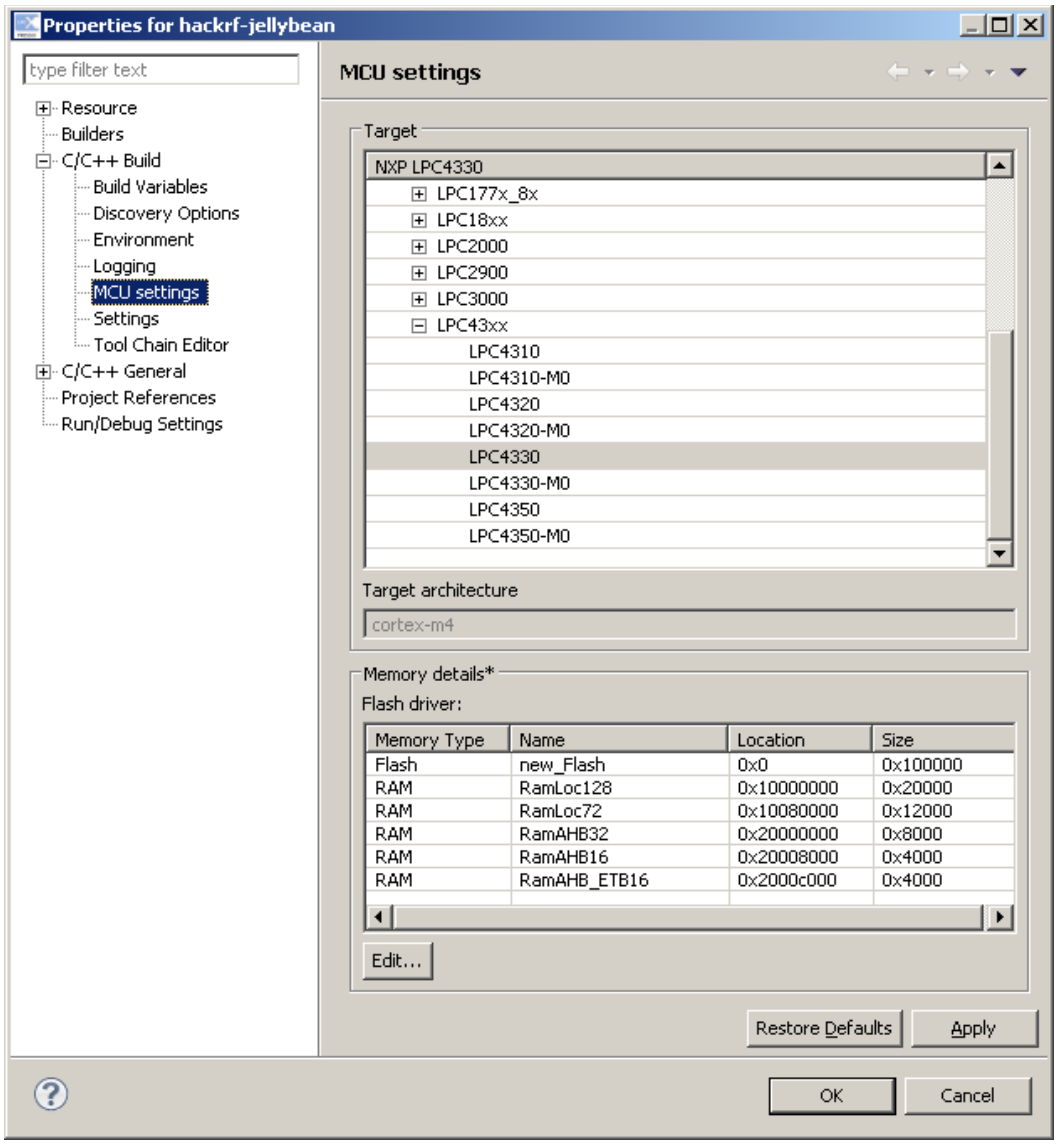
Nota: Flashing with JTAG work fine too and is a bit faster than default SWD mode.

Debugger configuration

With mouse Right click on project (or Alt Enter), choose Properties

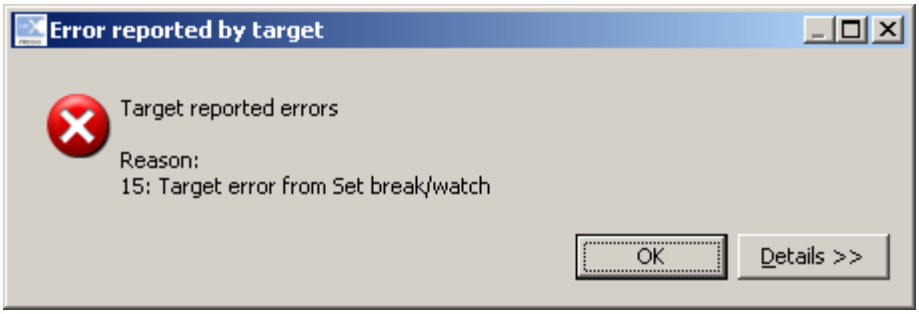
Choose MCU settings and Edit to add Flash memory at location 0x0 with size 0x100000 (1MB)

Click on Apply and OK



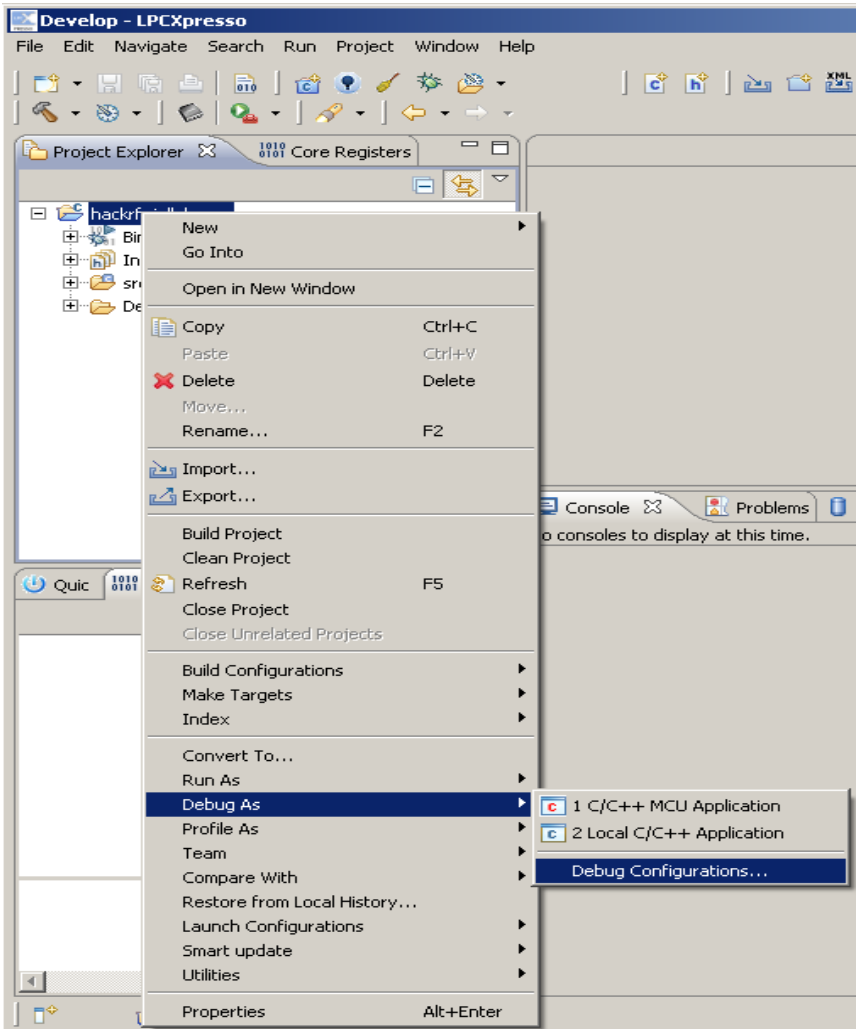
This step is mandatory to debug code executing from SPIFI flash, else debugger will always fail to set breakpoint to debug code.

With following error:

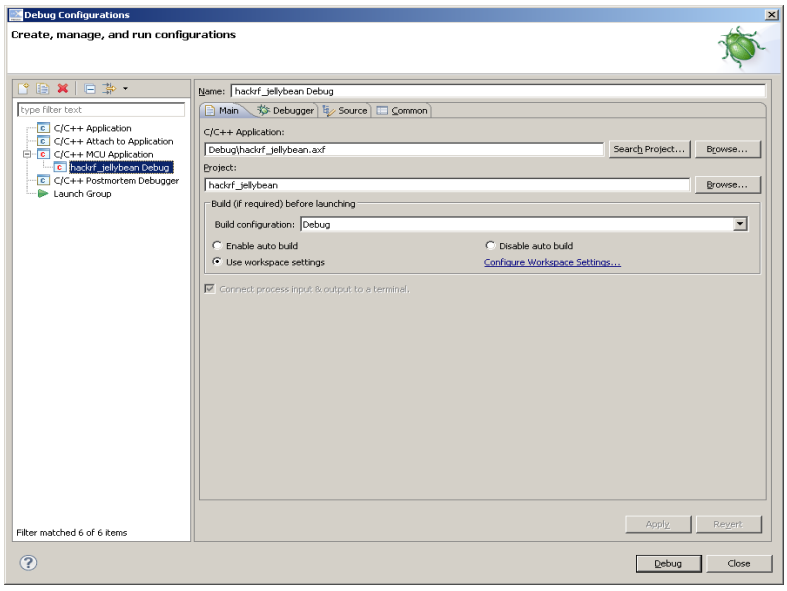


Now configure the debugger

With mouse Right click on project, choose Debug As -> Debug configurations...



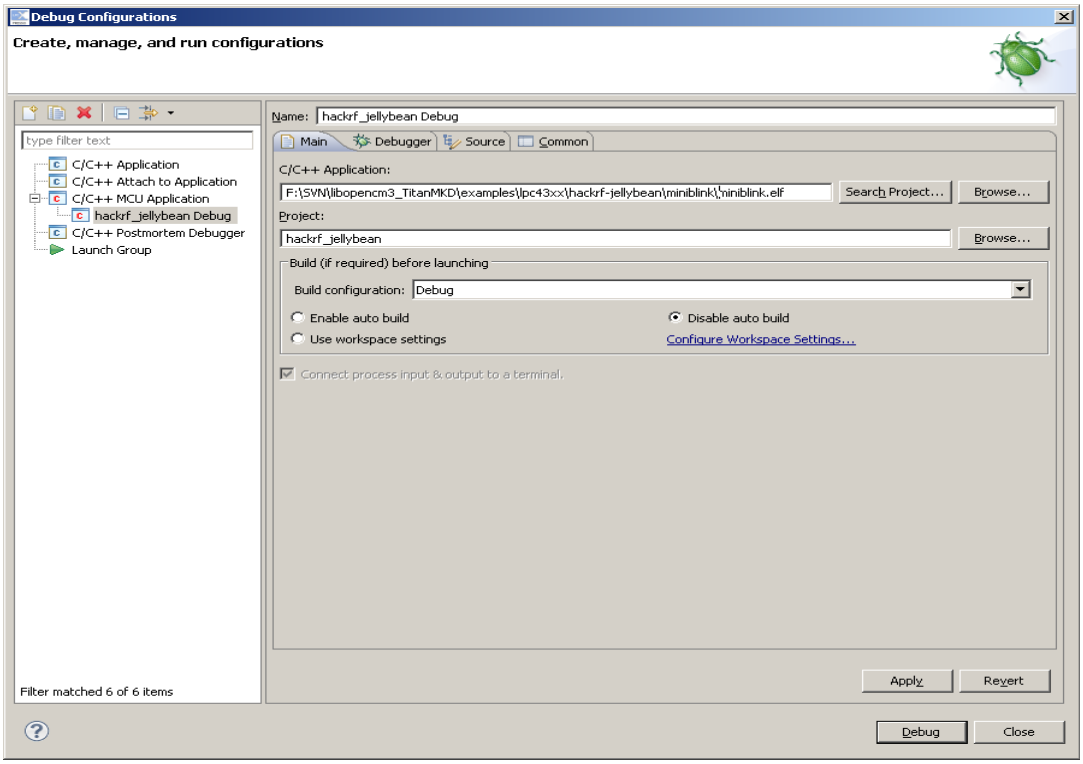
Double click on C/C++ MCU Application to automatically create hack_rfjellybean Debug Entry



Now change de C/C++ Application to miniblink.elf using full path

In the example F:\SVN\libopencm3_TitanMKD\examples\lpc43xx\hackrf-jellybean\miniblink.elf

Disable the auto build too.

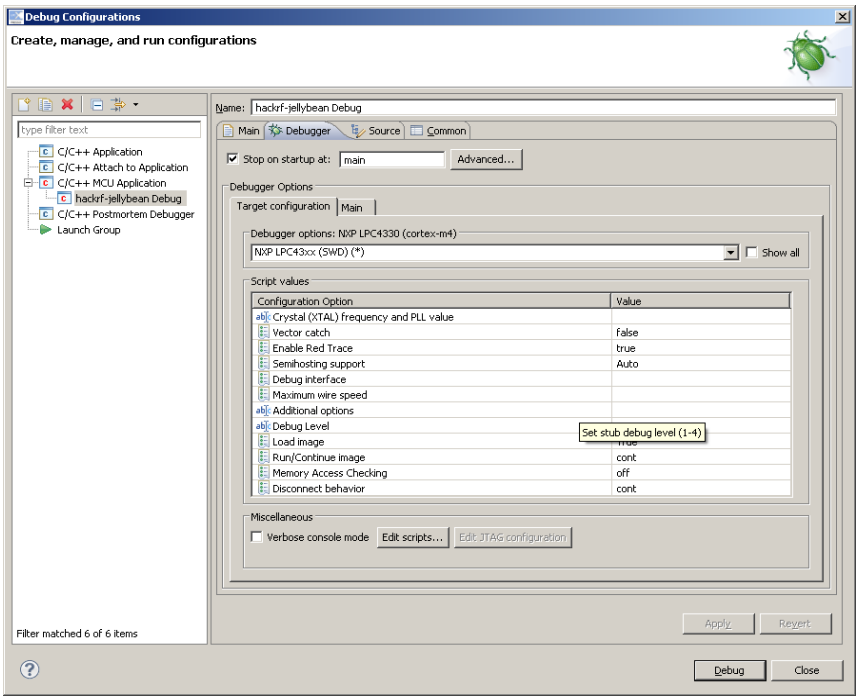


Configure the debugger:

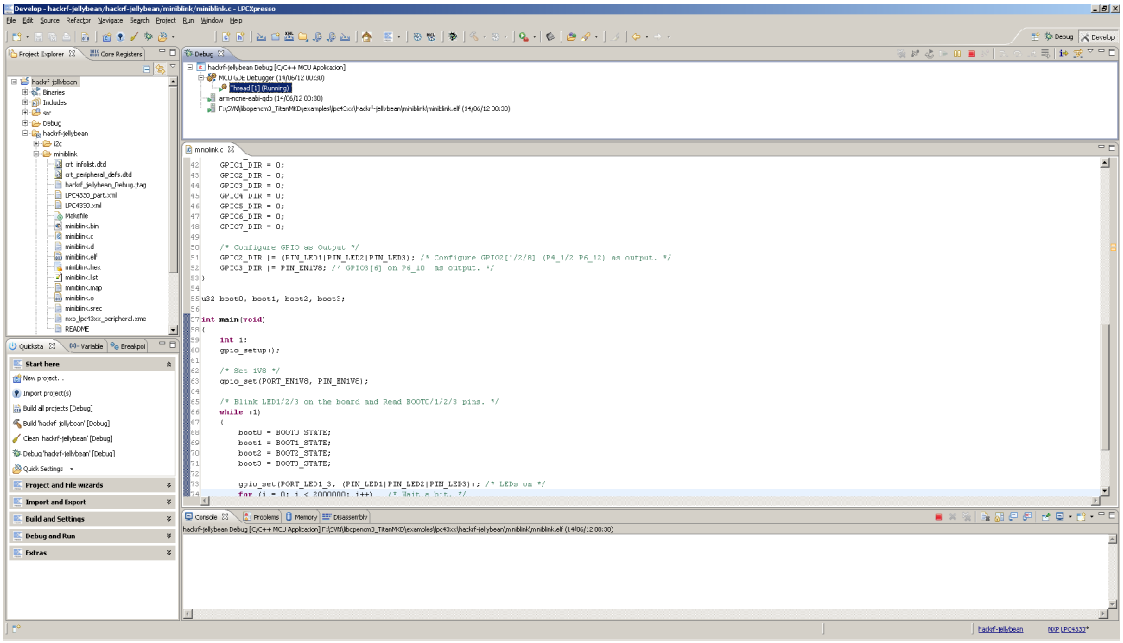
1) You can change "Stop on startup at:" by reset_handler or using an address like 0x00001234 if you want to stop execution before the default main code (it not always stop to debug it from start you need to flash it and then debug).

2) Change debugger options to use NXP LPC43xx (SWD) because JTAG never work for me.

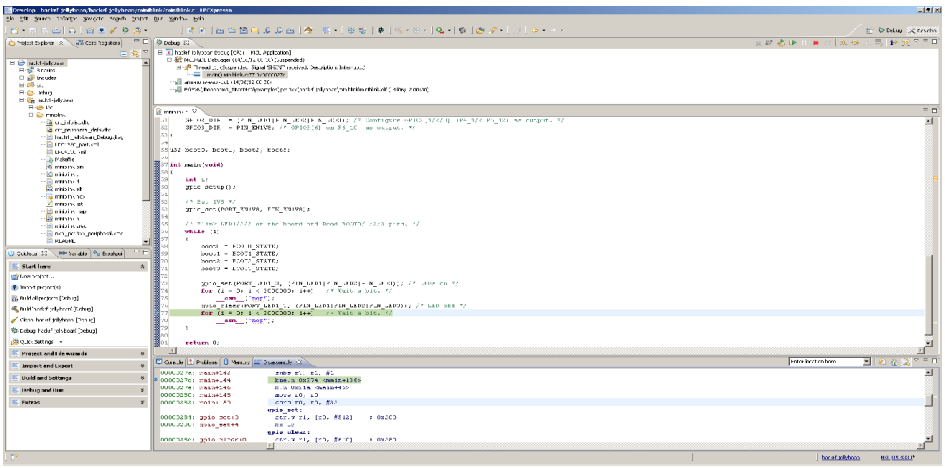
Now click on Apply and Debug.



The debugger is launched and code is running



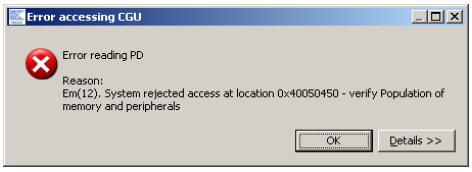
Just click on Pause "||" to stop it.

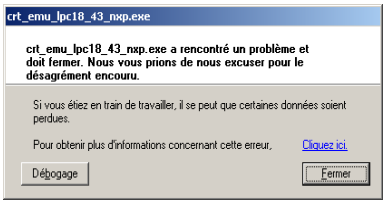


Now you can do step by step or check assembly code or core register (in debug view or by adding the window)

You can also view Peripherals with details (by default in debug view).

Warning View on CGU, CCU1 or CCU2 peripheral trig an error or even crash (for CCU1/2) crt_emu_lpc18_43_nxp.exe





After the crash of crt_emu_lpc18_43_nxp.exe you need to kill the debug session, power off the LPC43xx board, power on LPC43xx board and restart debug session, else all is crashed.