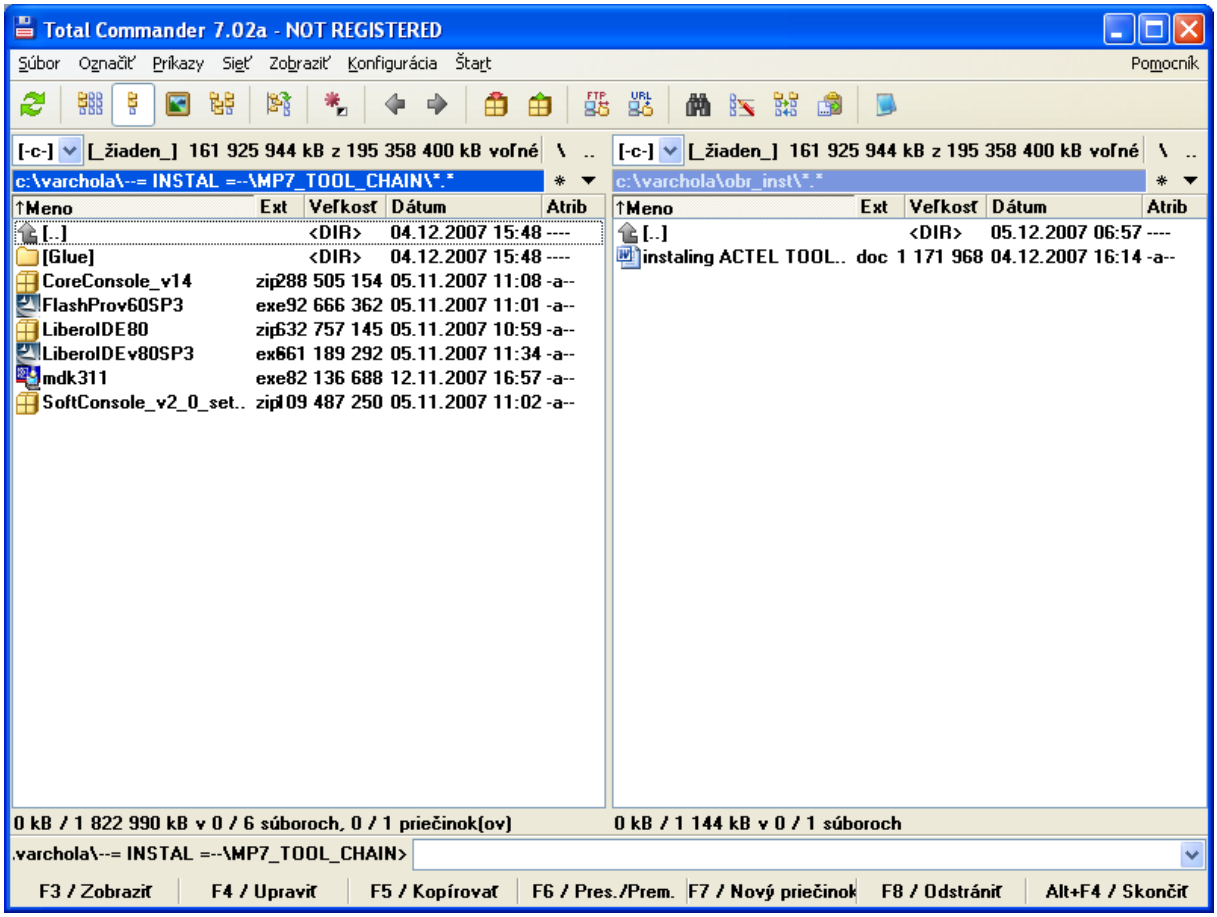
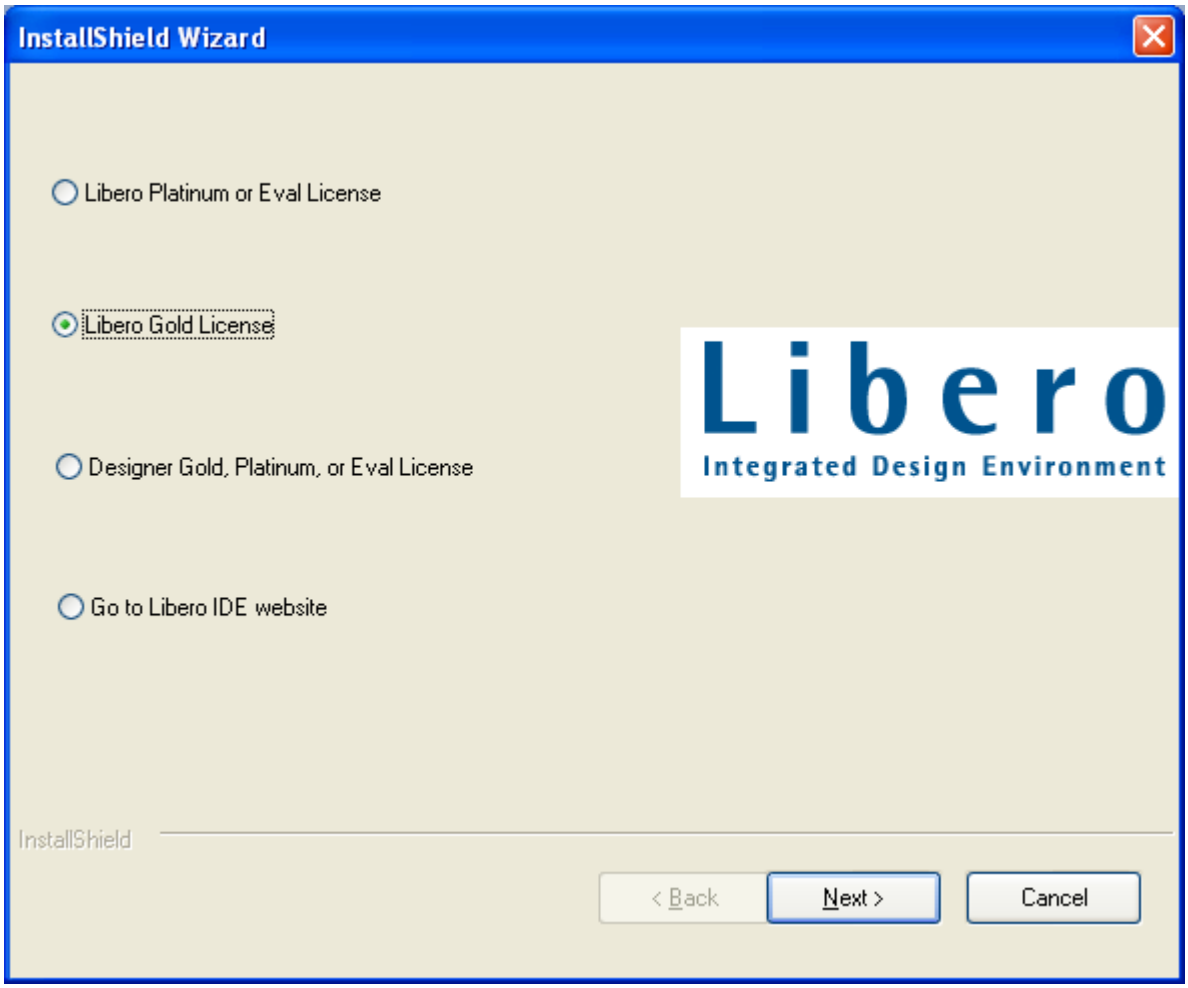
Necessary files to install Tool-chain for ACTEL MP7



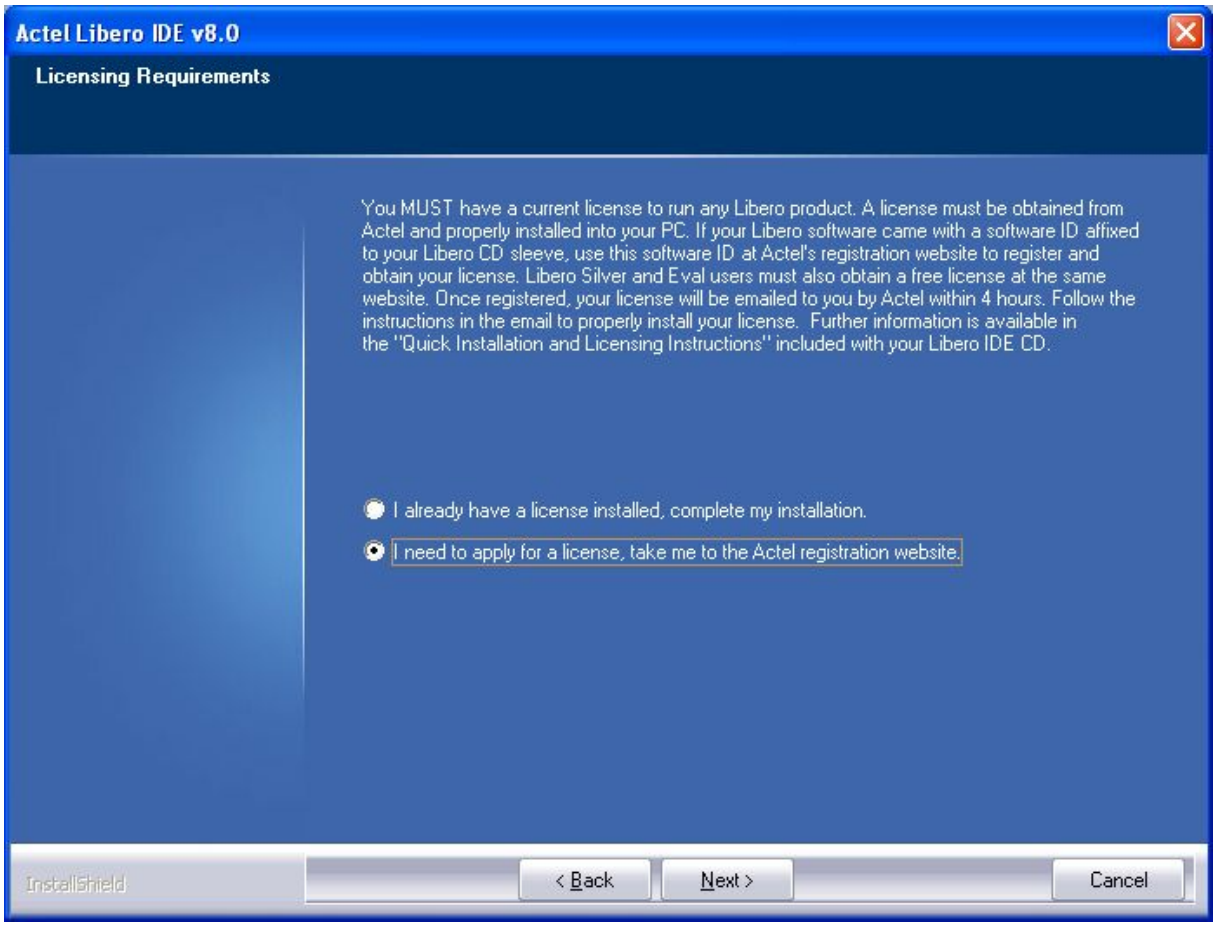
Start by installing Libero IDE
Unzip LiberoIDE80.zip and run the exe file

Install Libero Gold License

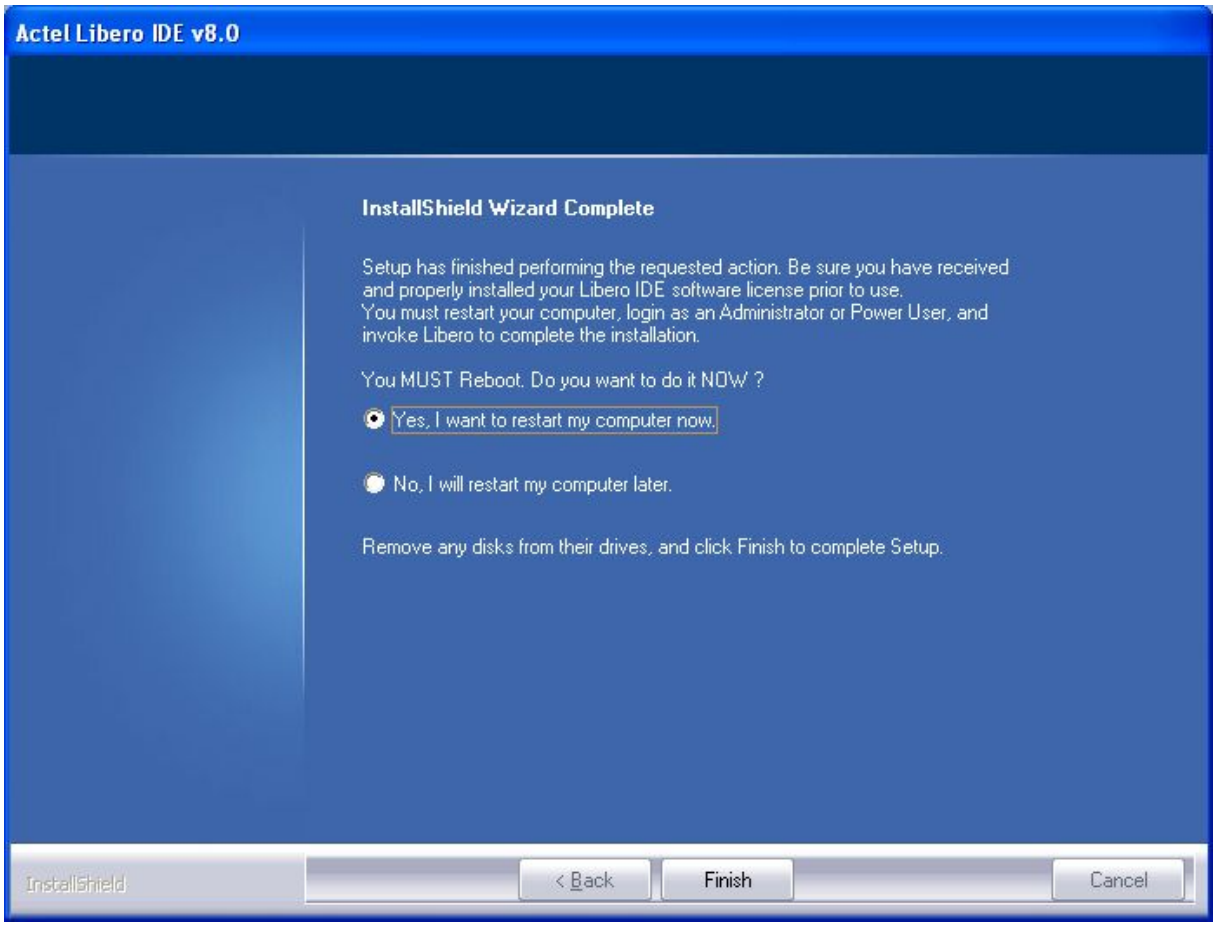


Click NEXT for several times and wait to complete the installation of LIBERO.

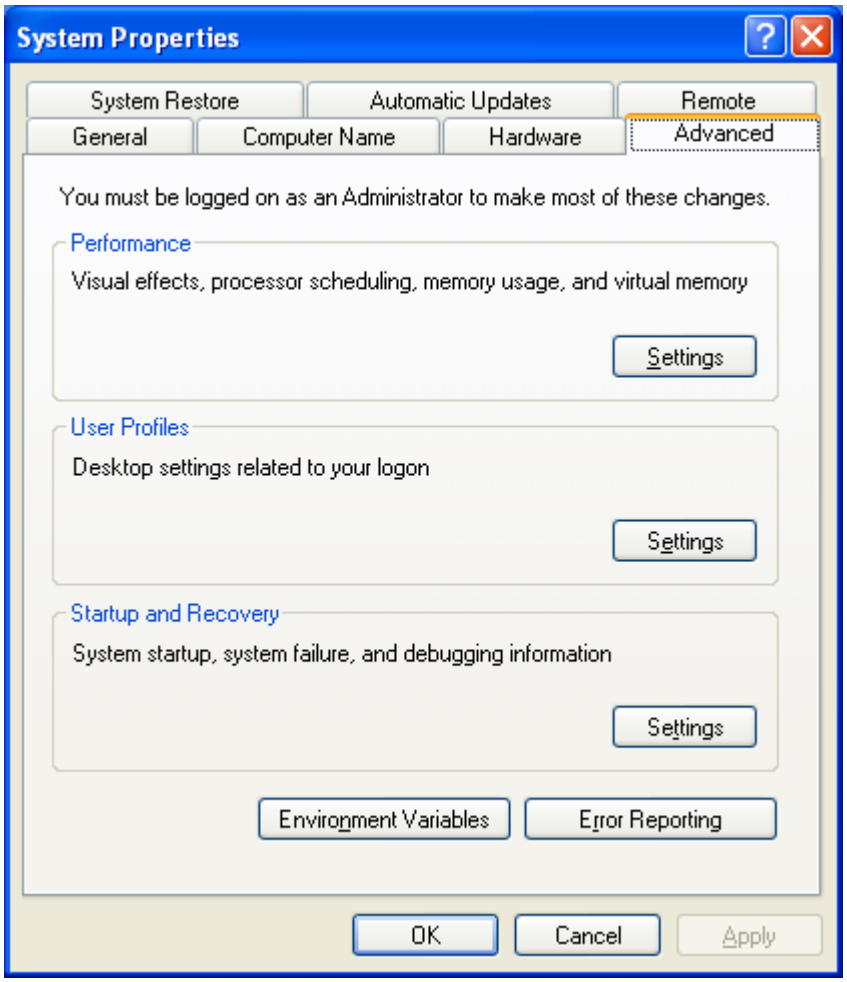
You were been asked to choose way of installing the license at the end of installation.
You have to visit actel website before the installation to request license.dat file for LIBERO IDE. This file could be downloaded from actel website or from your mailbox after requesting the LIBERO GOLD LICENCE (1 year license for using up to 1.000.000 gates). Copy this file into C:\flexlm\licence.dat. In following window choose:



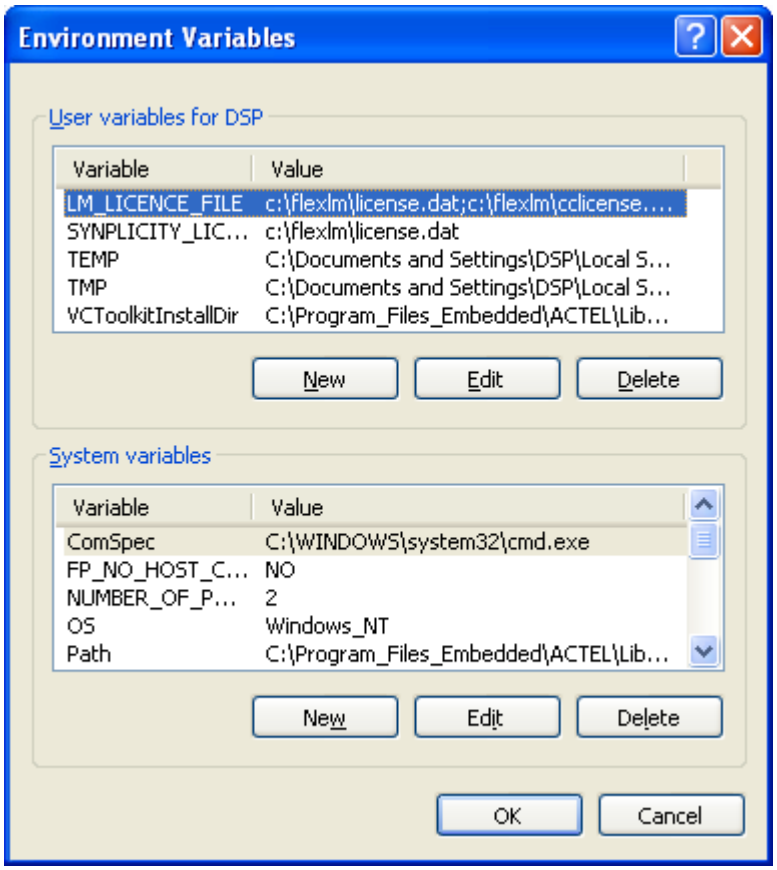
And reboot your computer.



Open the System Properties Window by right-click on My Computer. Select the Advanced tab and click on the Environment Variables:

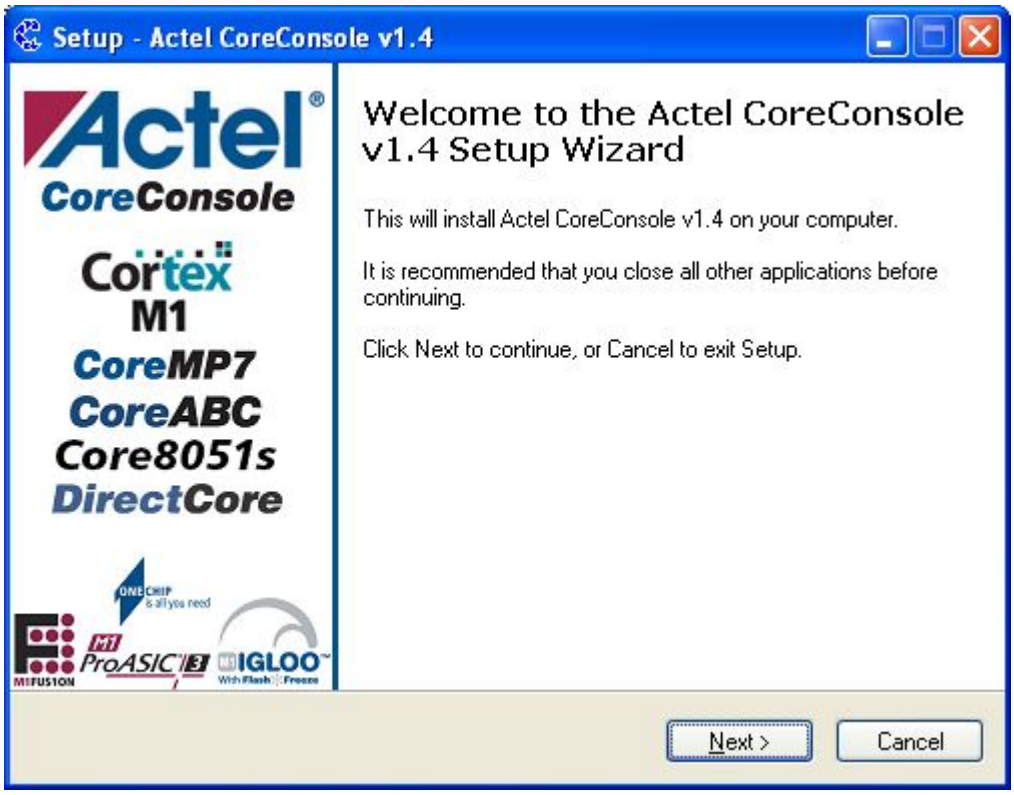


Create new variables: LM_LICENCE_FILE and SYNPLICITY_LICENSE_FILE with C:\flexlm\License.dat file path. File License.dat is also necessary to copy into folder: Libero/Desinger/adm/



Install the SP3 service pack for Libero 8.0 by executing LiberoIDEv80SP3.exe file. Click NEXT for several times.

Continue with installing the CoreConsole: CoreConsole_v14.exe



Click NEXT for several times to complete the installation

You will be asked about licences again:

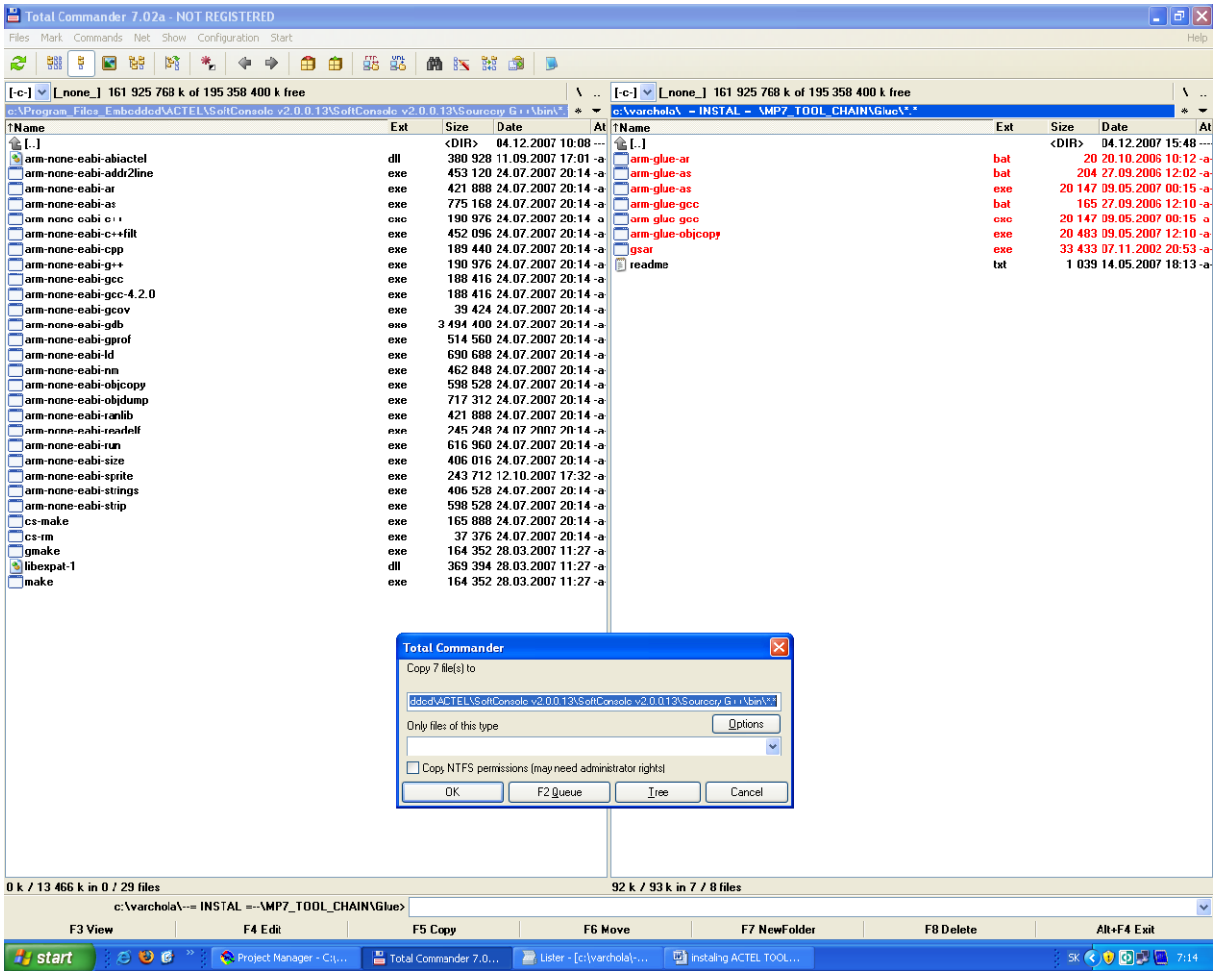
So, if you have not a license, visit ACTEL's website and request for it.

Received license rename to cclicense.dat and copy into C:\flexlm\
And into LM_LICENCE_FILE path in Envirionment variables.

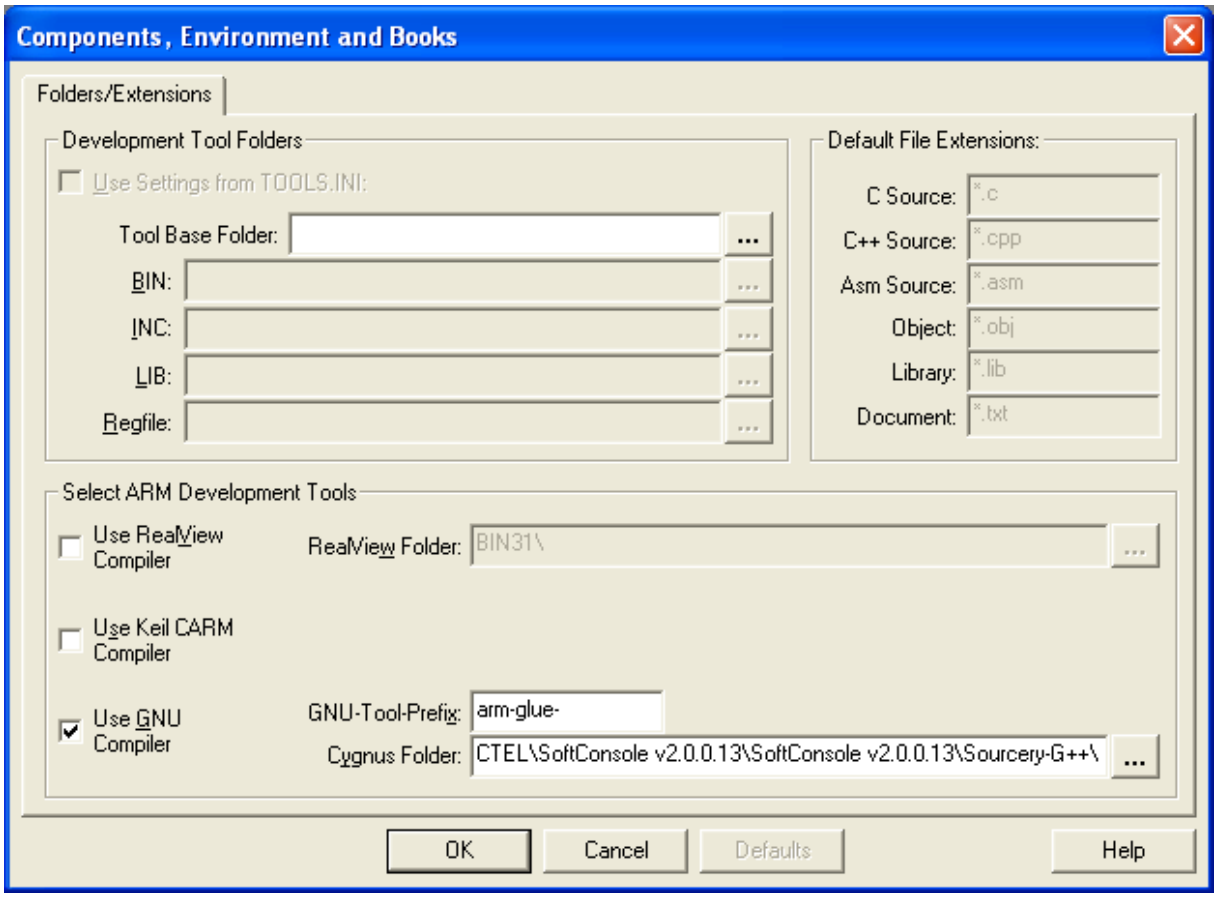
Last stage for MP7 designing is the SofConsole. Run its installation by executing:
SoftConsole_v2_0_setup.exe

The alternate option of ARM software development toolkit is KEIL uVision3 IDE with RealView compiler (in demo-version up to 16kB of code) or free GNU GCC by CodeSourcery. CodeSourcery compiler is used in softconsole too, so we will make interconnection between KEIL uVision IDE and GNU GCC compiler. At first run installation of uVision3 by executing: mdk311.exe. Click several times NEXT for success installation of KEIL.

Now we prepare interconnection between KEIL and CodeSourcery GNU GCC from the SoftConsole installation by following instructions in GLUE/readme.txt
At first copy all files instead readme.txt from GLUE folder into BIN folder of Sourcery G++:



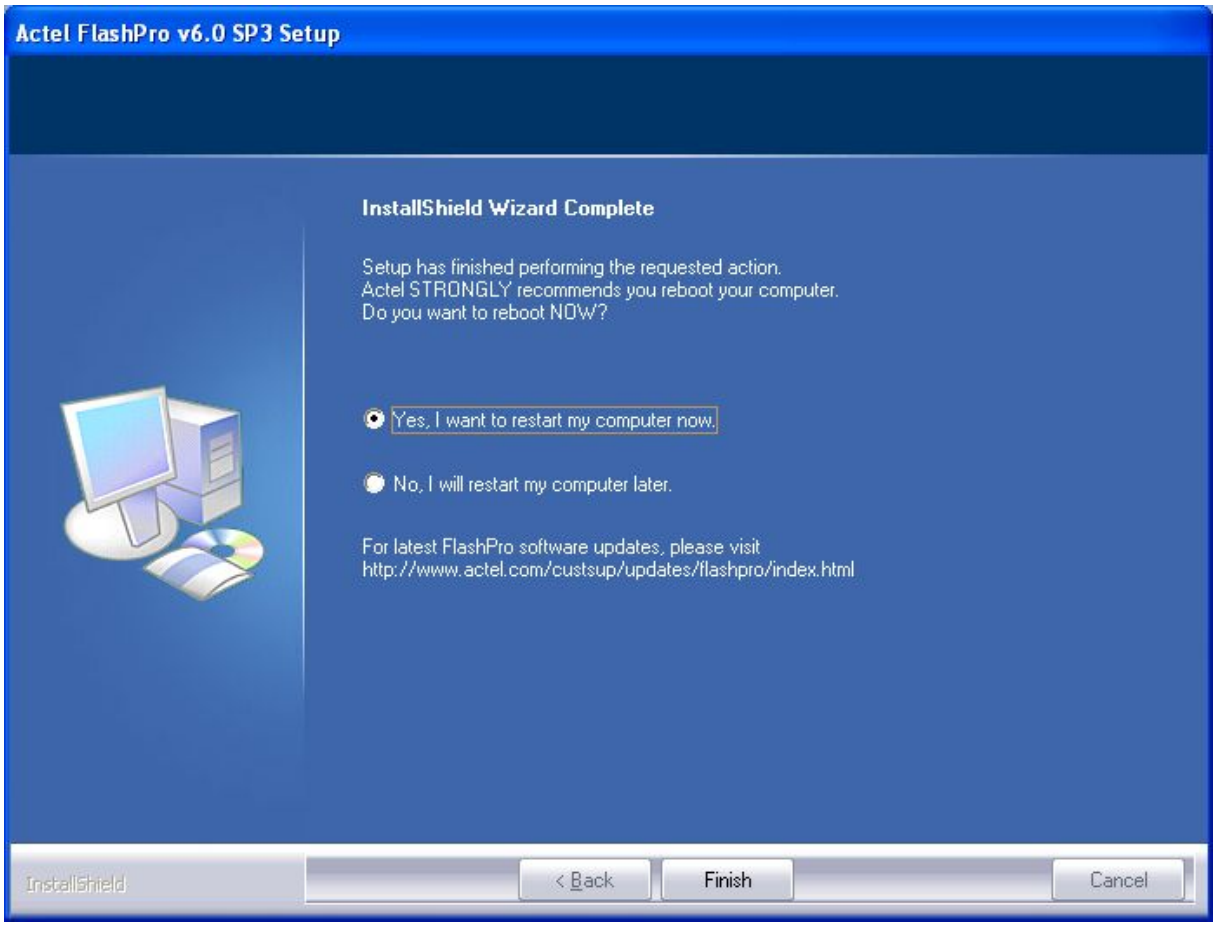
At the second, run KEIL. Close opened project. From toolbar menu choose Project -> Manage -> Components, Environments, Books…
In Folders/Extensions tab check Use GNU Compiler
In GNU-Tool-Prefix insert arm-glue-
And browse location of GNUARM folder of CodeSourecry GNU GCC in Cygnus Folder.
Click OK.



Close the Keil.

INSTALING DRIVERS FOR FLASHPRO3

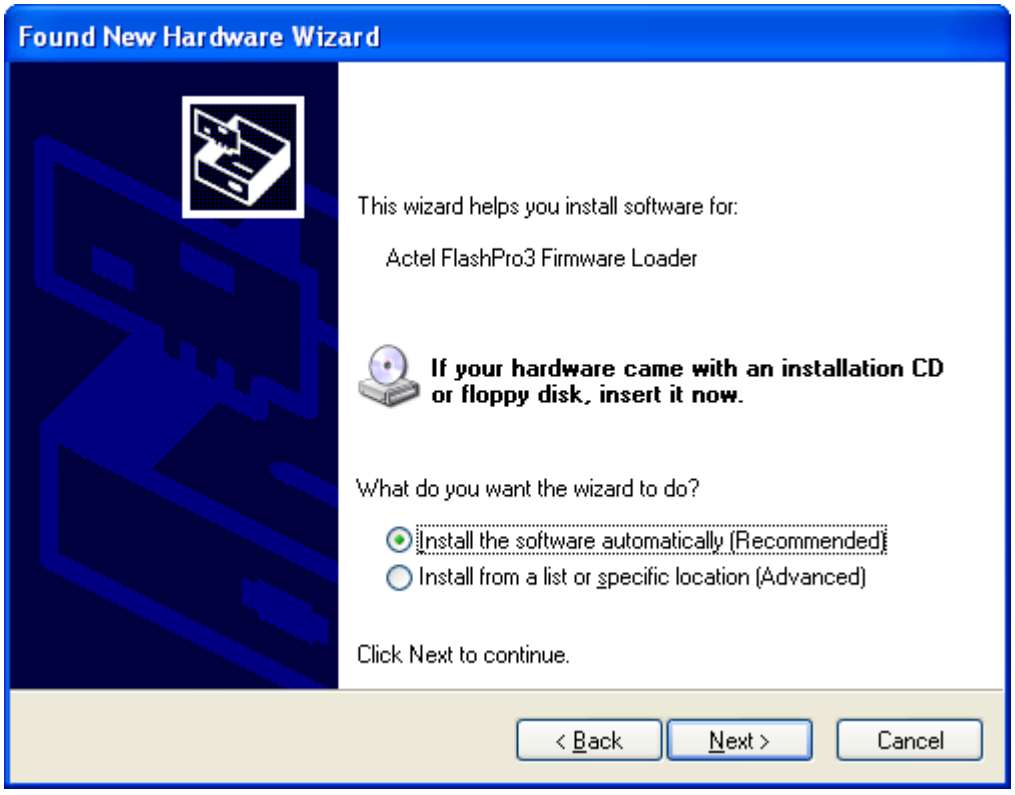
Execute FlashProv60SP3.exe. Click NEXT and OK for several times. You will prompted to restart the PC.

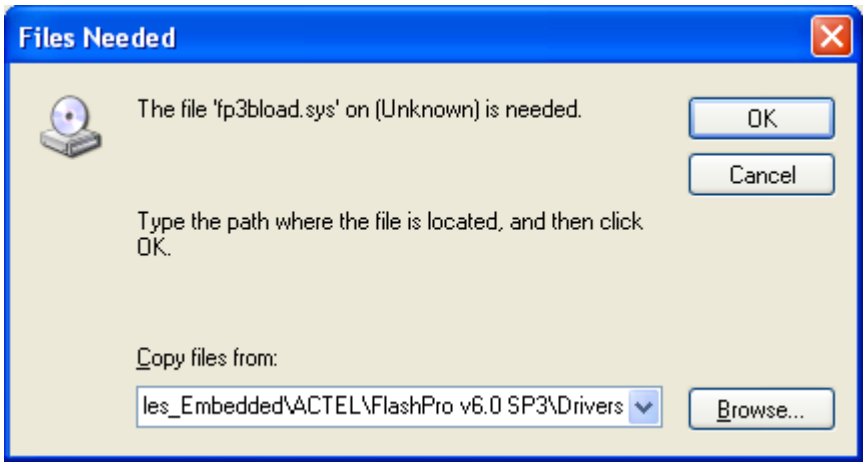


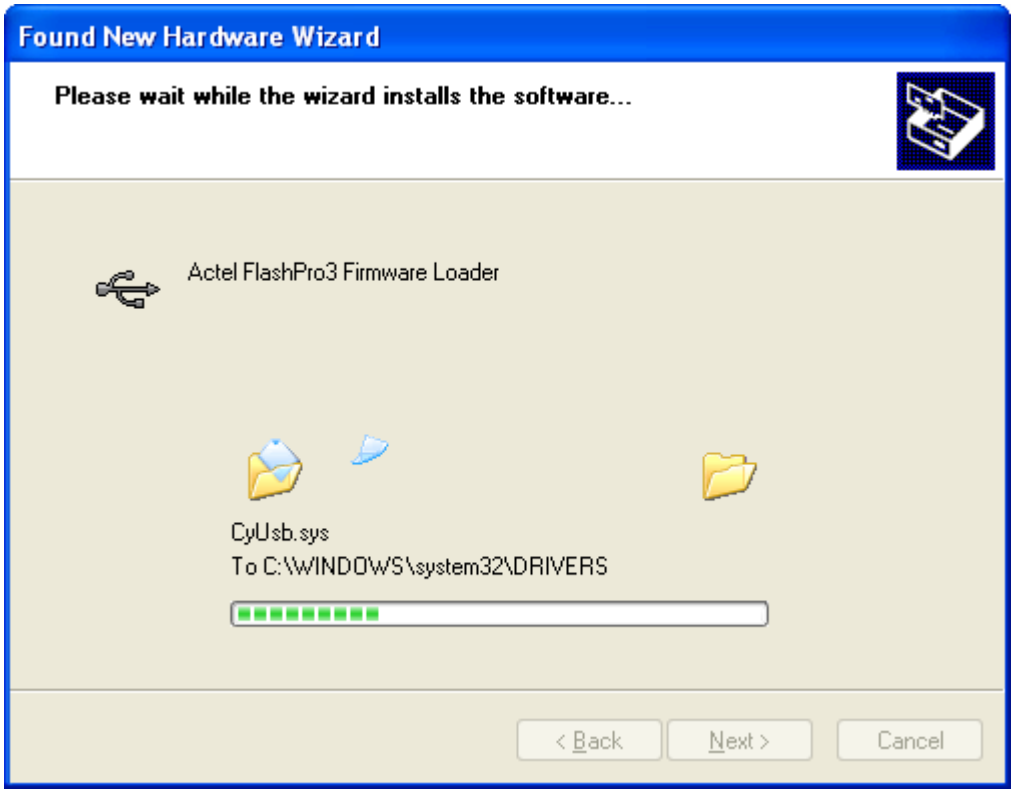
Restart it.

After restart connect your FlashPro3 into USB.


Found New Hardware Wizard
Welcome to the Found New Hardware Wizard
Windows will search for current and updated software by looking on your computer, on the hardware installation CD, or on the Windows Update Web site (with your permission).
Read our privacy policy
Can Windows connect to Windows Update to search for software?
Yes, this time only
Yes, now and every time I connect a device
No, not this time
Click Next to continue.
< Back
Next >
Cancel

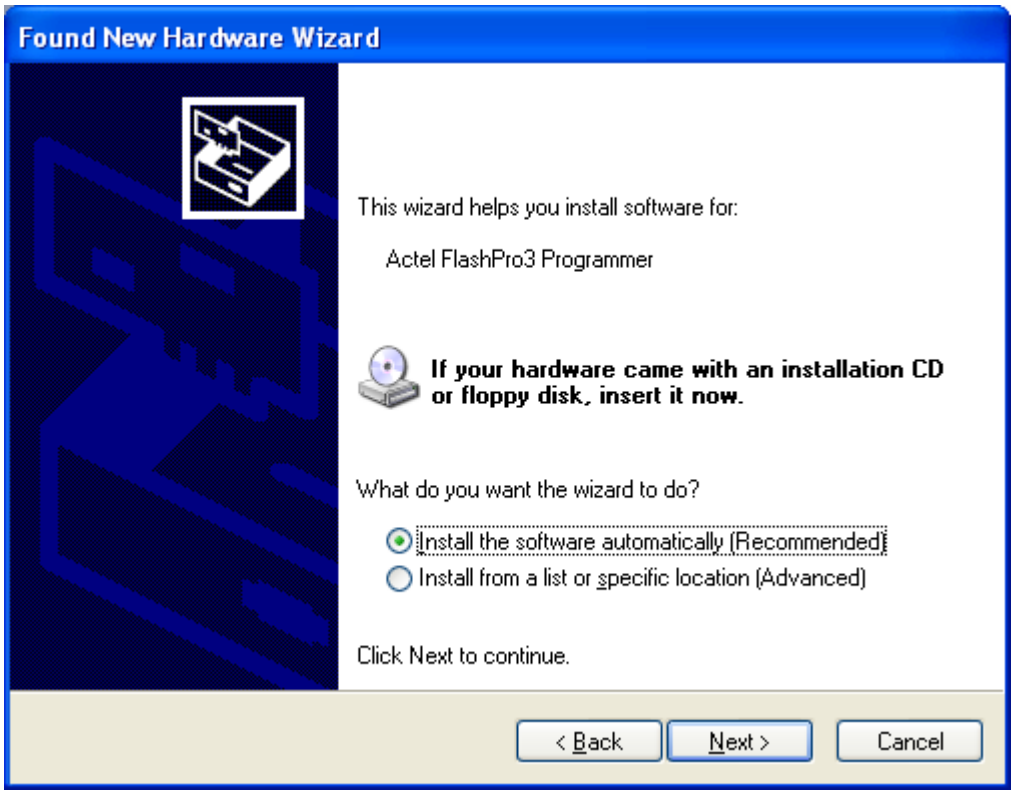

Found New Hardware Wizard
This wizard helps you install software for:
Actel FlashPro3 Firmware Loader
If your hardware came with an installation CD or floppy disk, insert it now.
What do you want the wizard to do?
Install the software automatically (Recommended)
Install from a list or specific location (Advanced)
Click Next to continue.
< Back
Next >
Cancel


Files Needed
The file 'fp3bload.sys' on (Unknown) is needed.
OK
Cancel
Type the path where the file is located, and then click OK.
Copy files from:
les_Embedded\ACTEL\FlashPro v6.0 SP3\Drivers
Browse...


Found New Hardware Wizard
Please wait while the wizard installs the software...
Actel FlashPro3 Firmware Loader
CyUsb.sys
To C:\WINDOWS\system32\DRIVERS
< Back
Next >
Cancel


Found New Hardware Wizard
Completing the Found New Hardware Wizard
The wizard has finished installing the software for:
Actel FlashPro3 Firmware Loader
Click Finish to close the wizard.
< Back
Finish
Cancel


Found New Hardware Wizard
Welcome to the Found New Hardware Wizard
Windows will search for current and updated software by looking on your computer, on the hardware installation CD, or on the Windows Update Web site (with your permission).
Read our privacy policy
Can Windows connect to Windows Update to search for software?
Yes, this time only
Yes, now and every time I connect a device
No, not this time
Click Next to continue.
< Back
Next >
Cancel

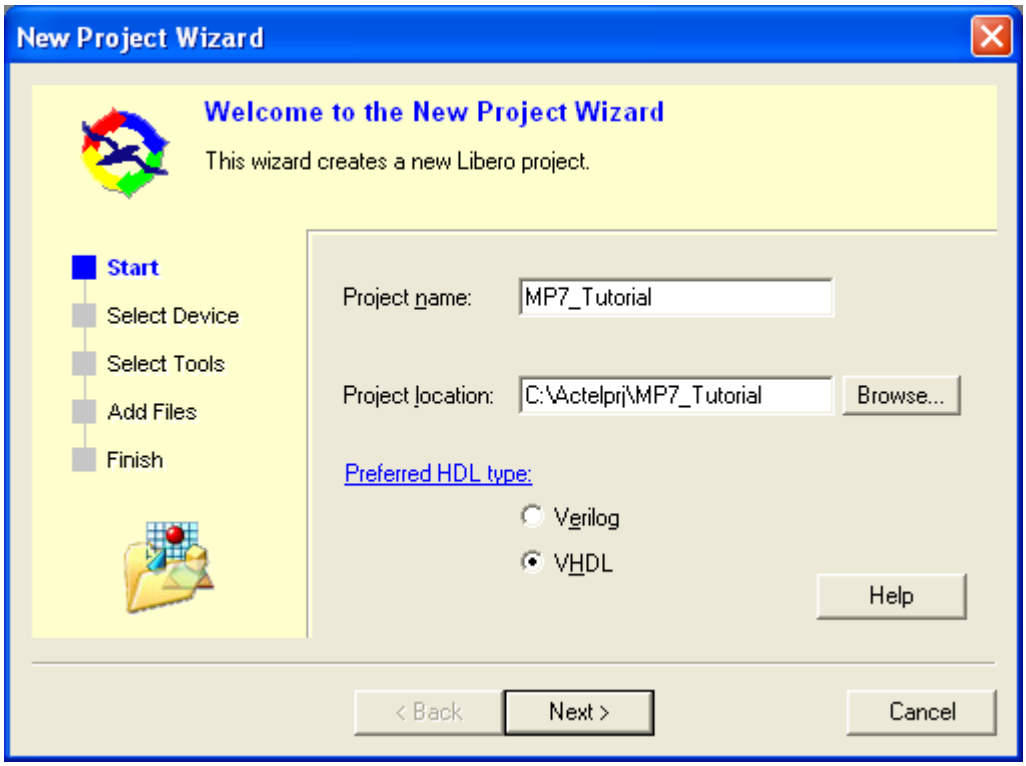


Now is your FlashPro3 ready to use.

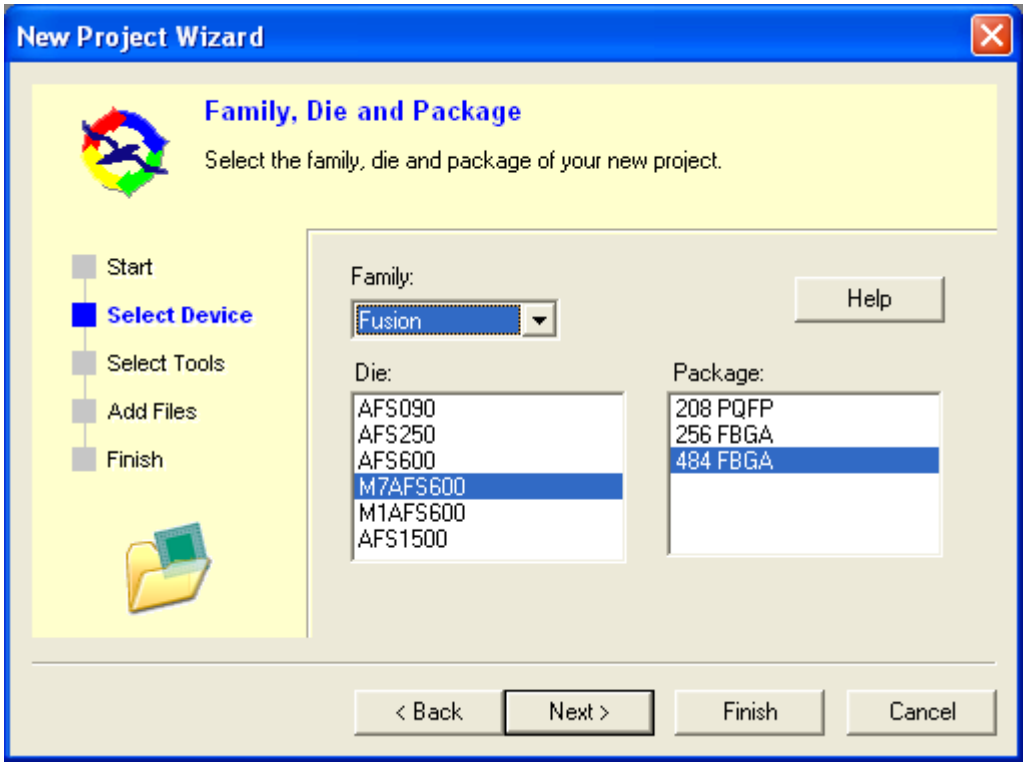
CREATING THE NEW PROJECT WITH CORE MP7

Run Libero IDE: start -> program files -> Actel Libero IDE v8.0 -> Project manager

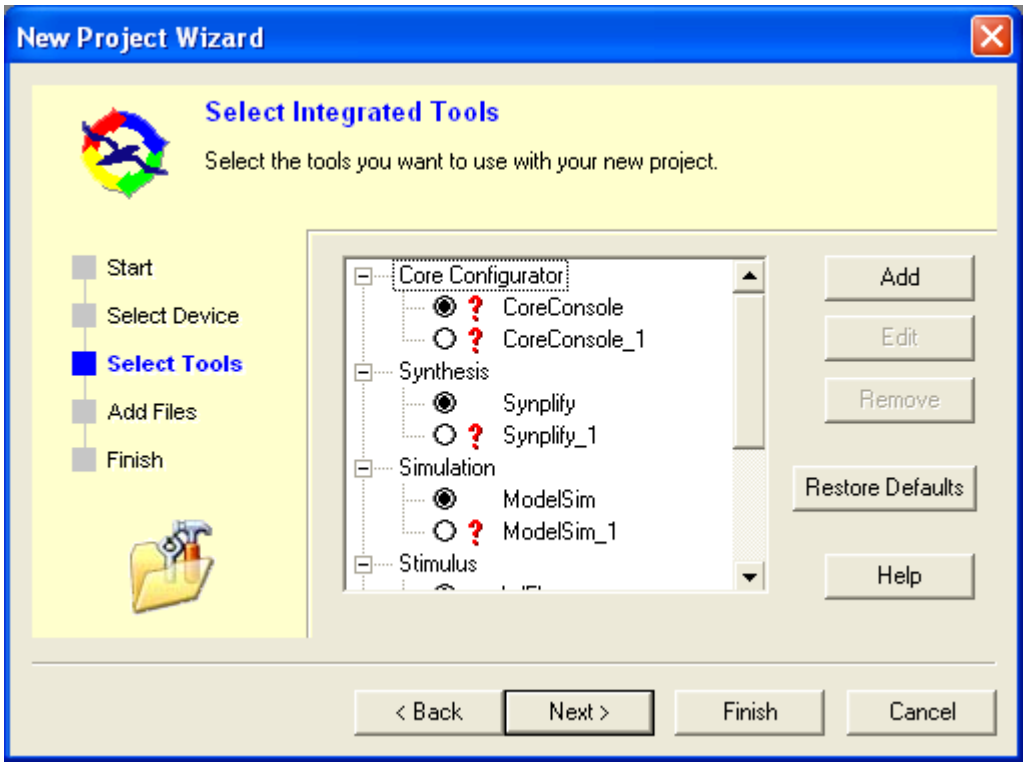
Create new project -> Project -> New Project



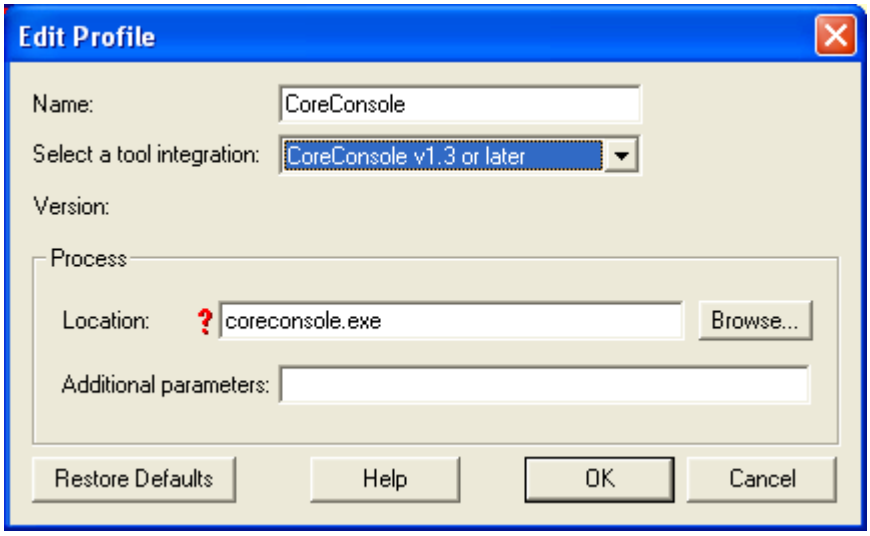
Click NEXT



Make same selections and click NEXT:

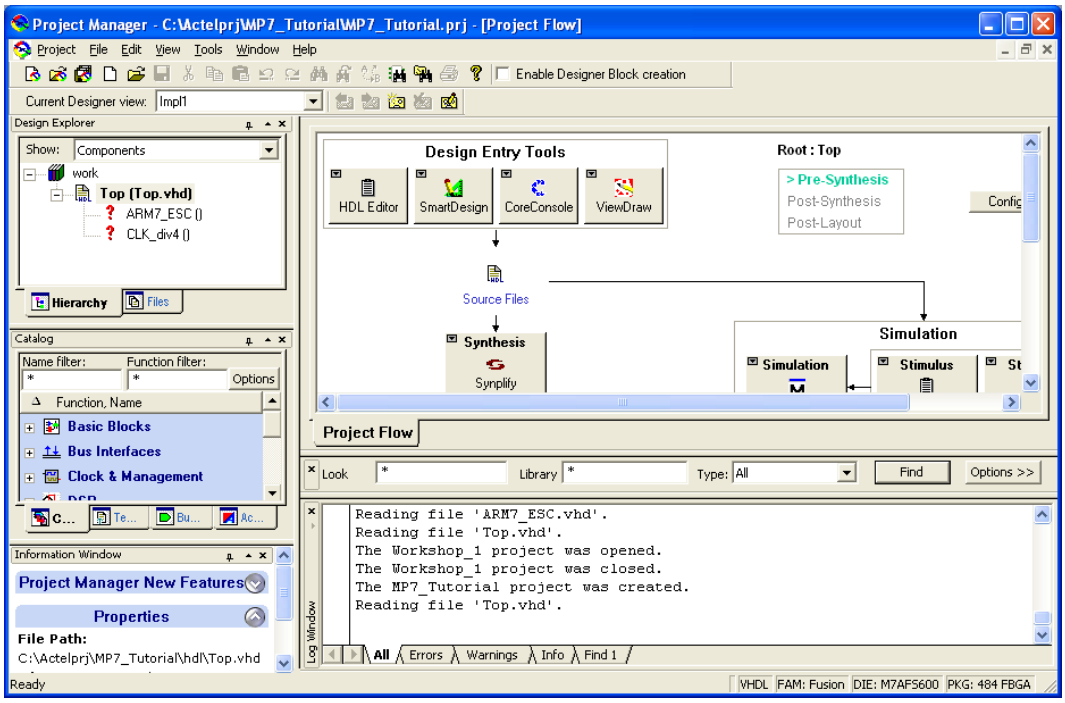


Select Tools, but, there is a question mark by Core Console after a new installation. So, Double click onto CoreConsole:

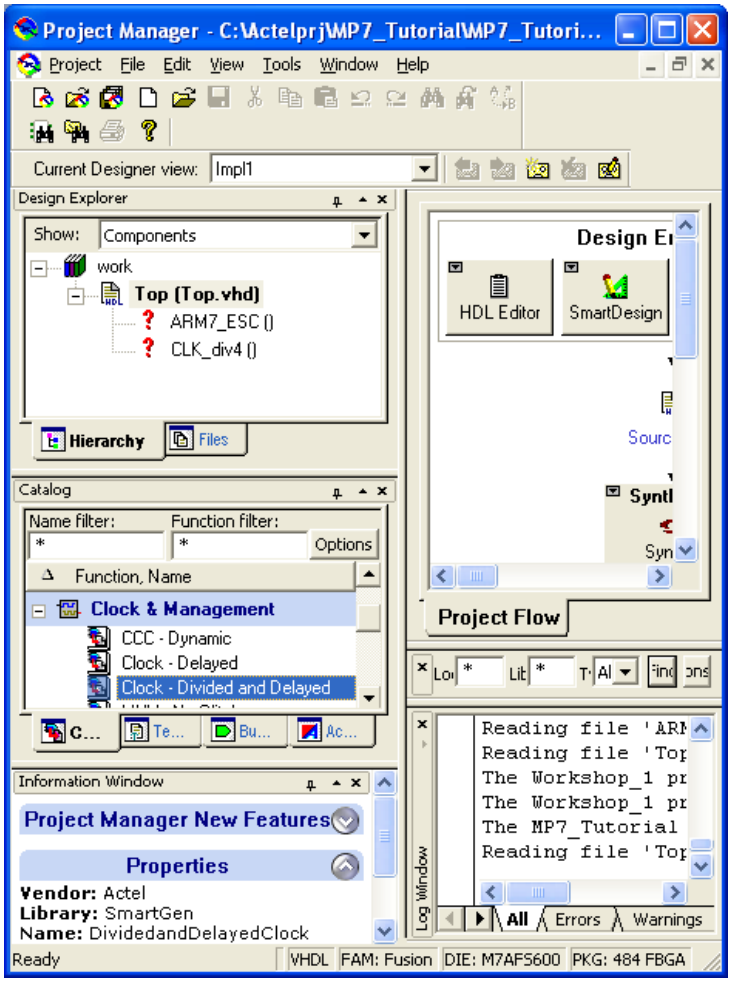


And browse location of coreconsole.exe: ...\CoreConsole_v1.4\bin\CoreConsole.exe, click OK. In New Project Wizard click NEXT and NEXT once again and FINISH.

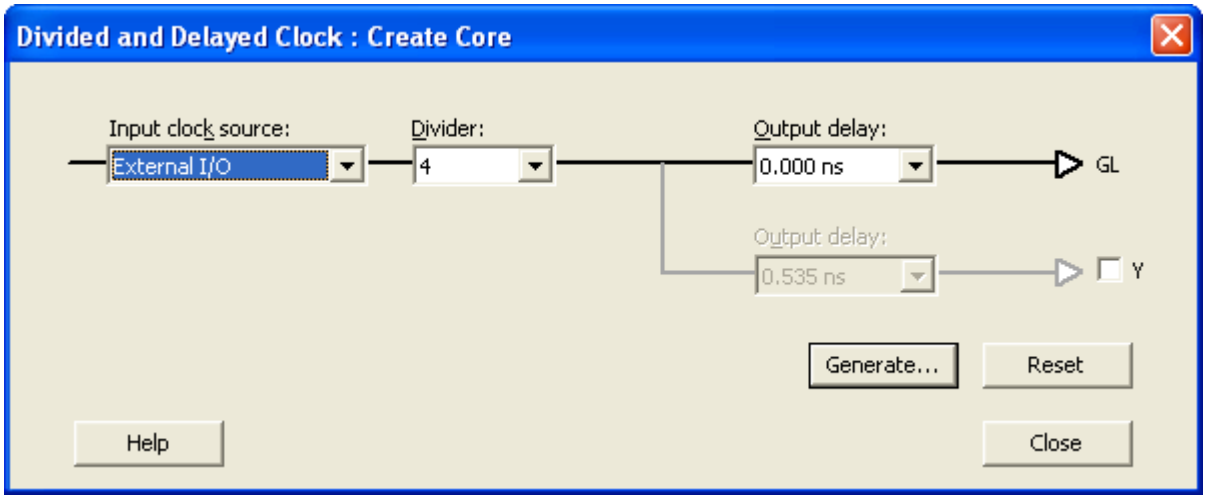
Import Top.vhd, that is vhdl file of top entity, where are two entities: MP7 processor and prescaler. Global clock of FUSION is 50MHz, whilst max frequency for MP7 is only 20MHz. We make 12.5MHz clock for MP7 by dividing the global clock by 4. So you can import Top.vhd by: File -> Import Files... Browse the Top.vhd



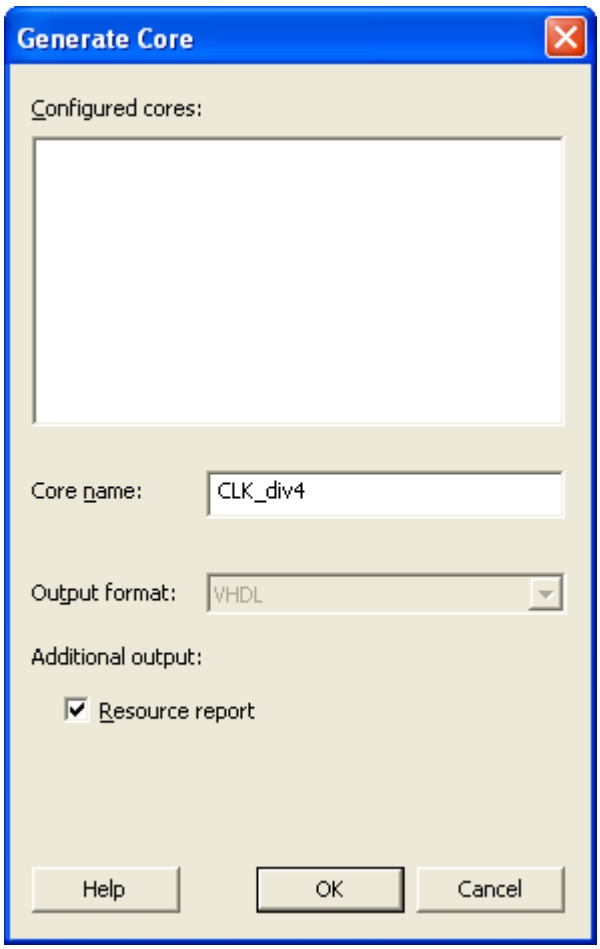
We create the Prescaler now: In left column of windows, select in CATALOG window Clock & Managemet -> Clock – Divided and Delayed



In opened Window make next selections:

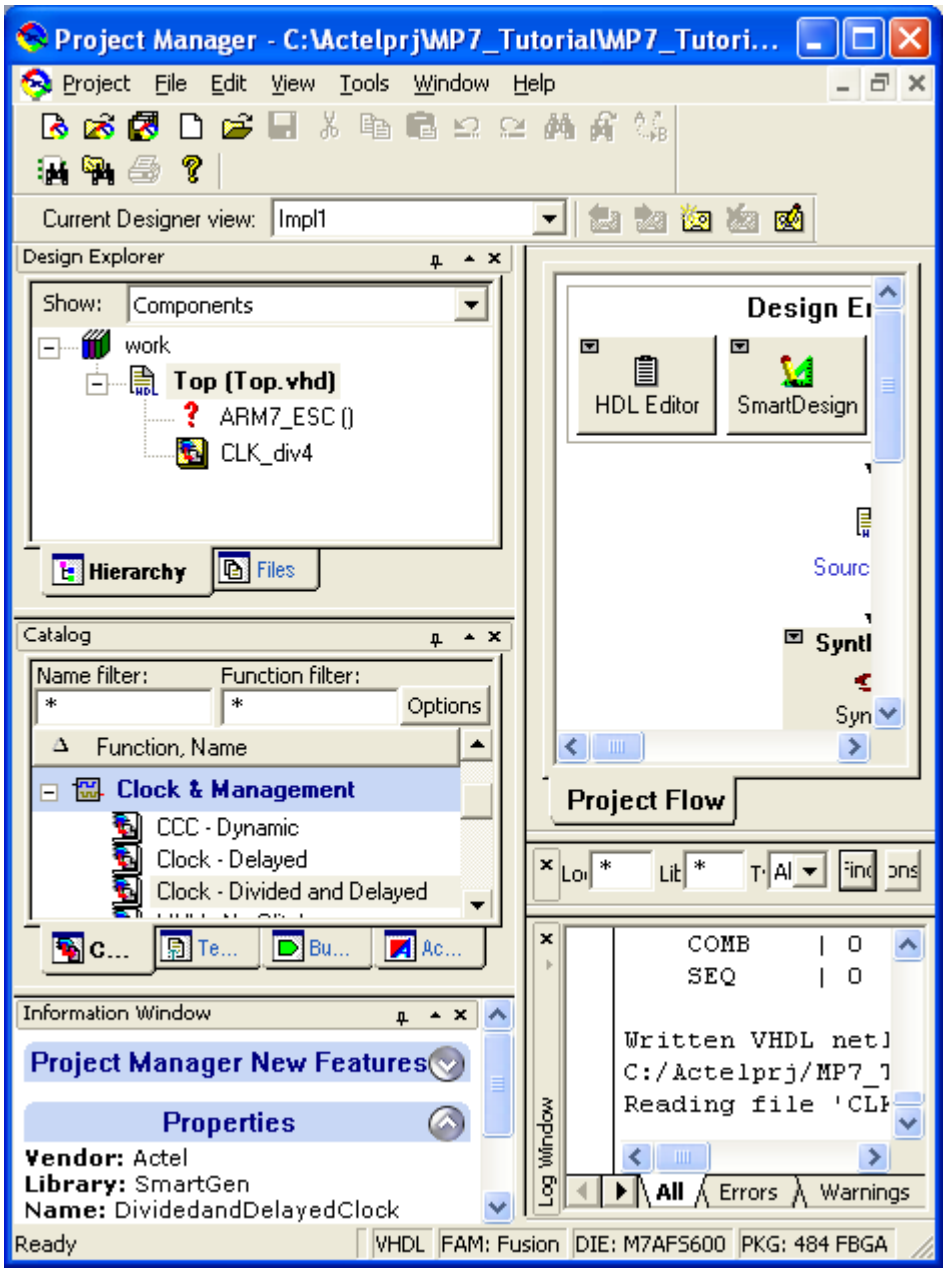


And click Generate button. Enter name of component CLK_div4.

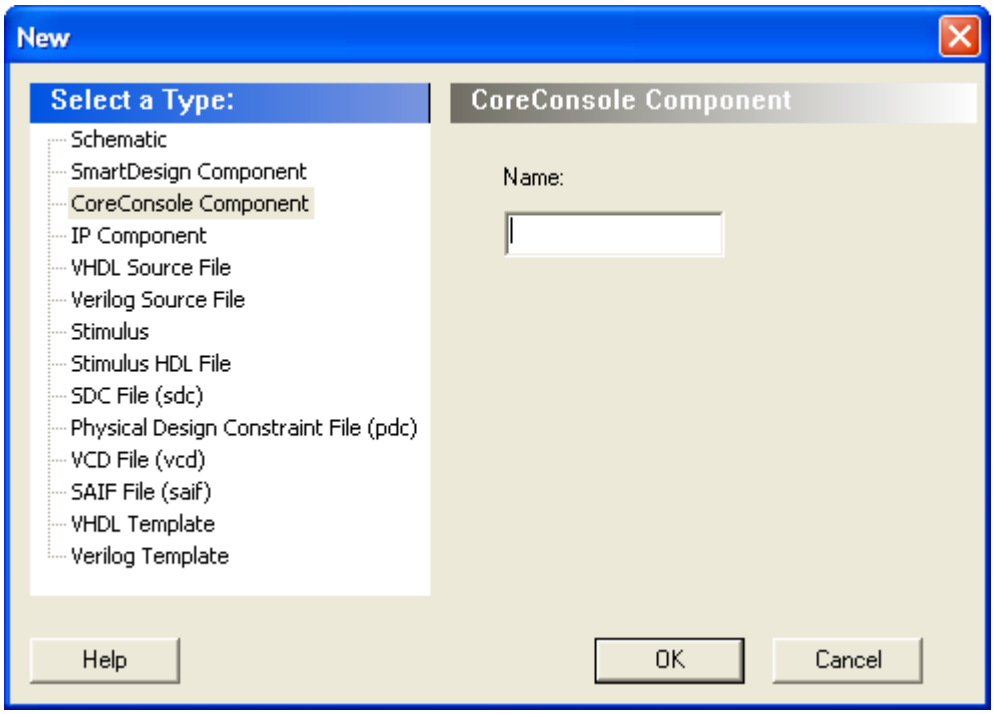


And click OK.

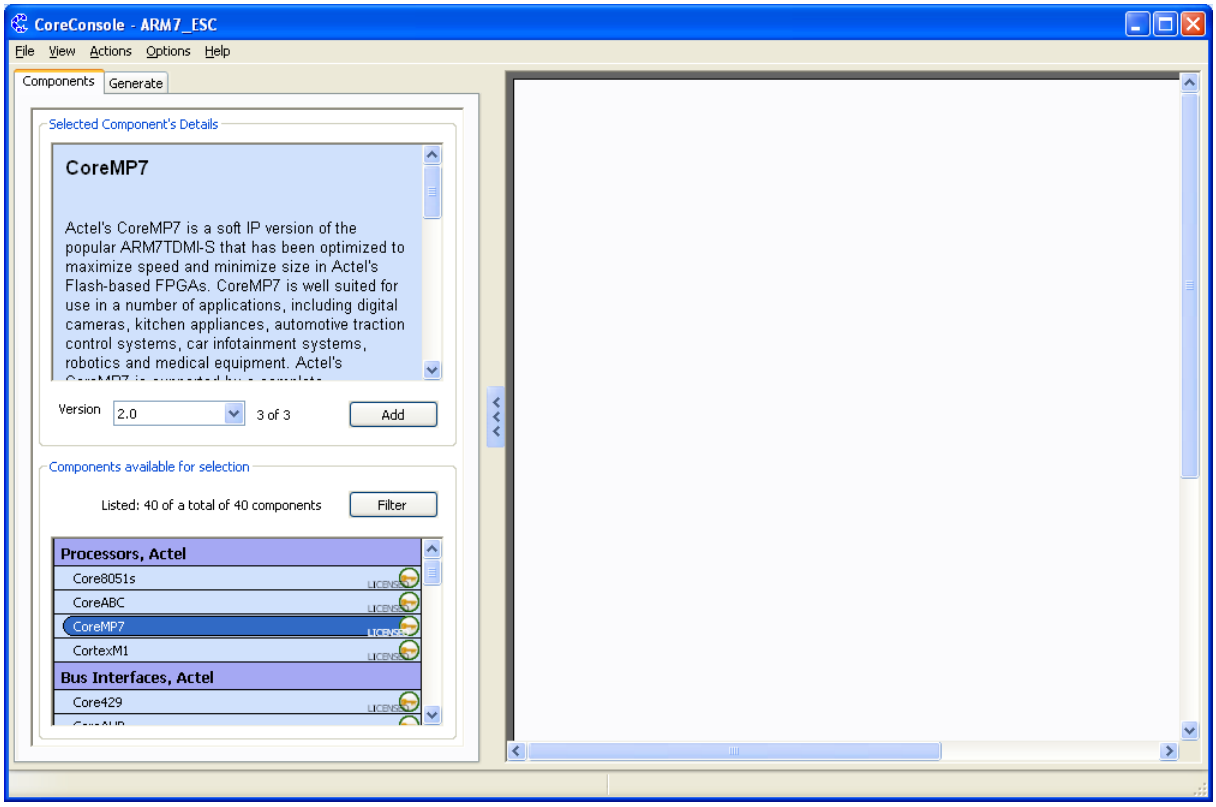
Be sure that a question mark infront of CLK_div4 in Design Explorer were changed into smartgen icon:



Continue by creating your MP7 core. In Design Flow window click on the Core Console in Design Entry Tools section. In opened window enter name of component: ARM7_ESC

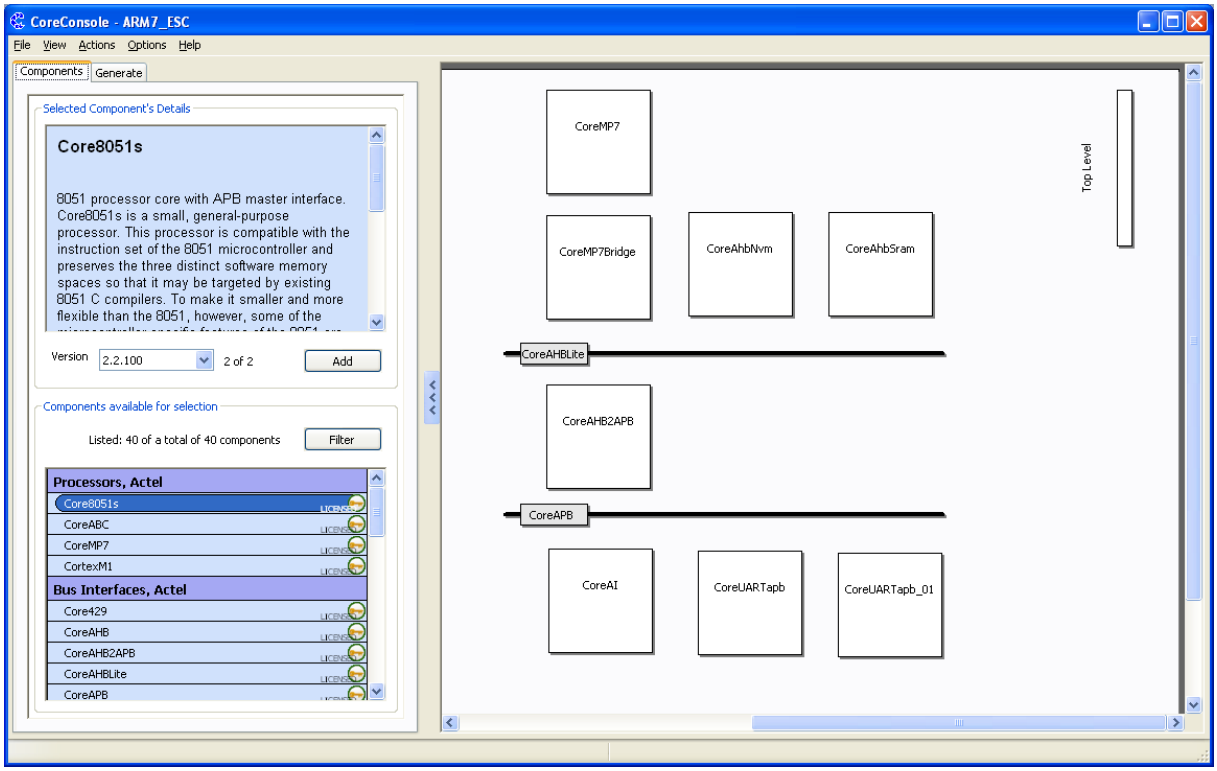


And click OK and the CoreConsole will be opened.



Choose the CoreMP7 from list Processors, Actel in the left side of your screen and click Add button. From Bus Interfaces, Actel list choose: CoreMP7Brigde, CoreAHB2APB and two buses: CoreAHBLite and CoreAPB. From Peripherals, Actel choose: CoreAhbNvm, CoreApbSram, CoreAI, CoreTimer and CoreUARTapb twice.

By Drag&Drop arange the components in next order:



It is necessaty to interconnect the peripherals, core and buses. I is very easy with Auto - Stitch function: Actions -> Auto Stitch...

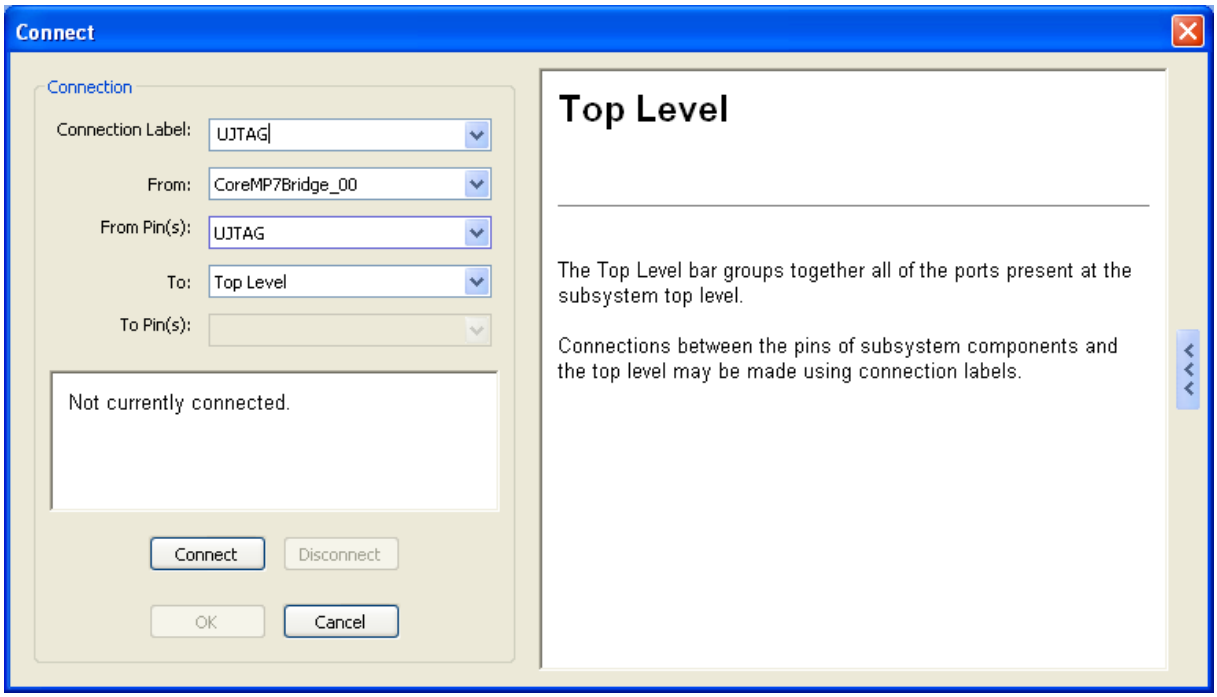
Set APB Masters & Slaves as folows:

And be sure that all boxes are checked. Click Stitch.

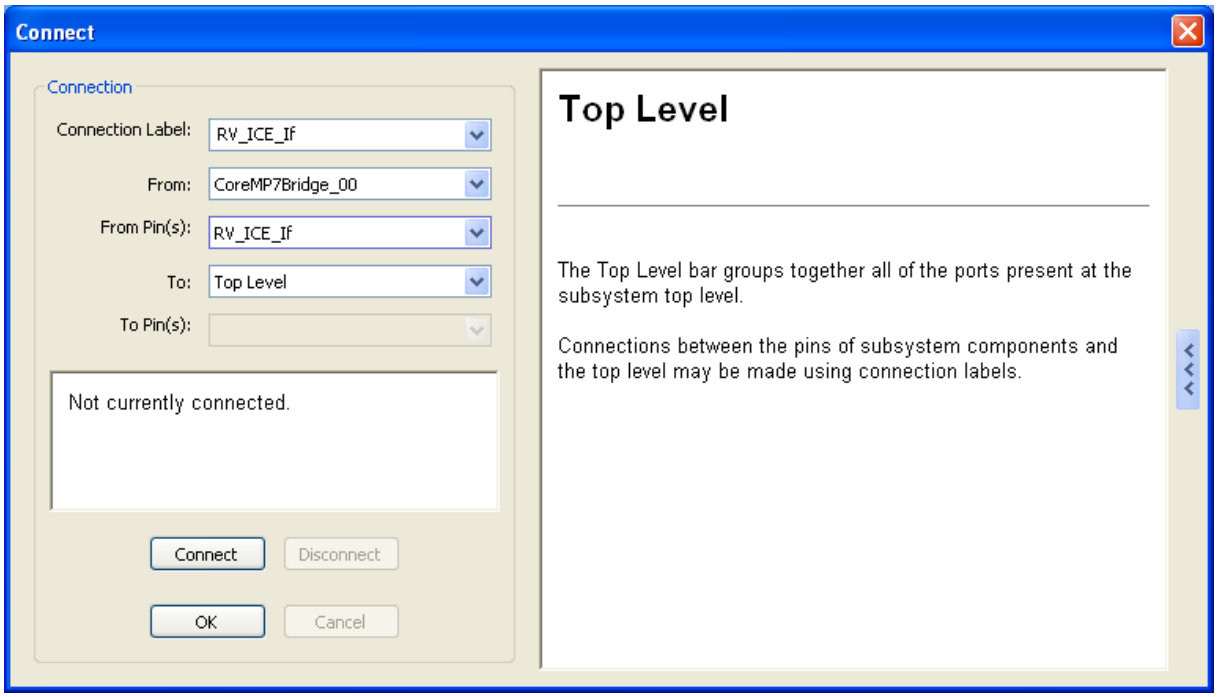
To connect other signals with top level entity apply following steps:

Move your mouse cursor over CoreMP7bridge, four icons will show. Click on the leftmost CONNECT icon.

Opened window fill in following way and click CONNECT:



And once again for RV_ICE_If:



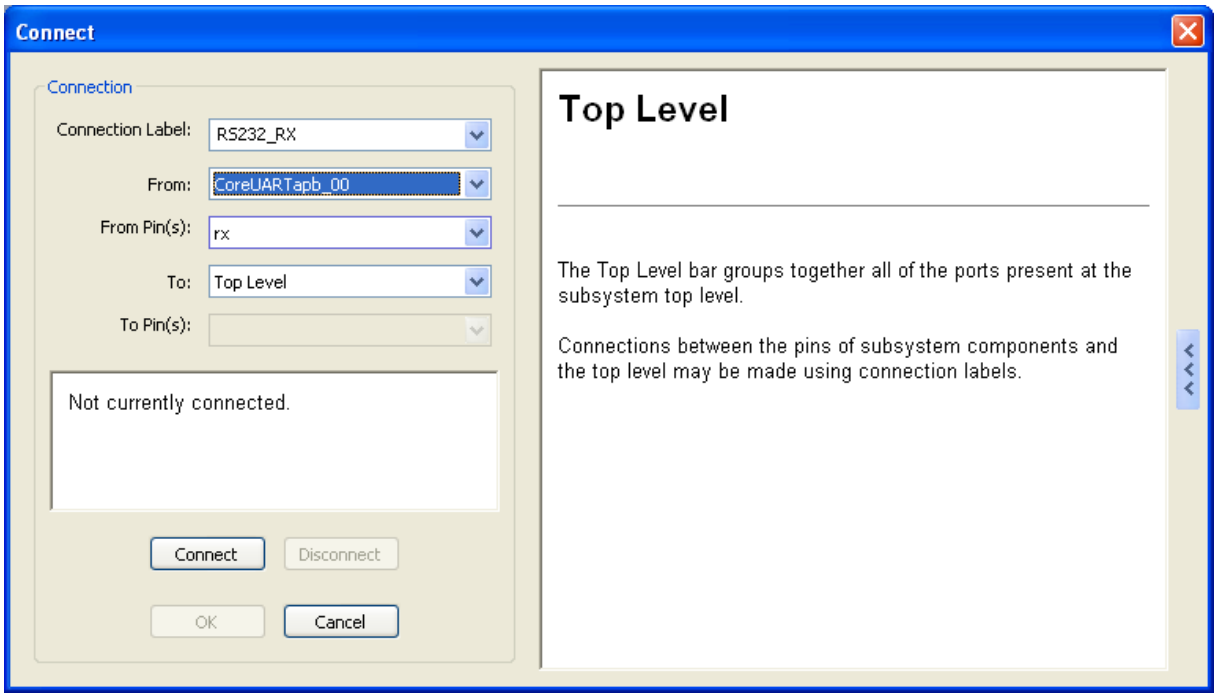
It was made UJTAG (debuging by FlashPro3 Interface) and RV_ICE_If interconection with top level entity by last two steps.

Move your mouse over both CoreUARTapb, click right button on the mouse and choose CHANGE VERSION. Choose 1.2 version instead 3.1.105 for both CoreUARTapb.

Apply an auto-stitch function again.

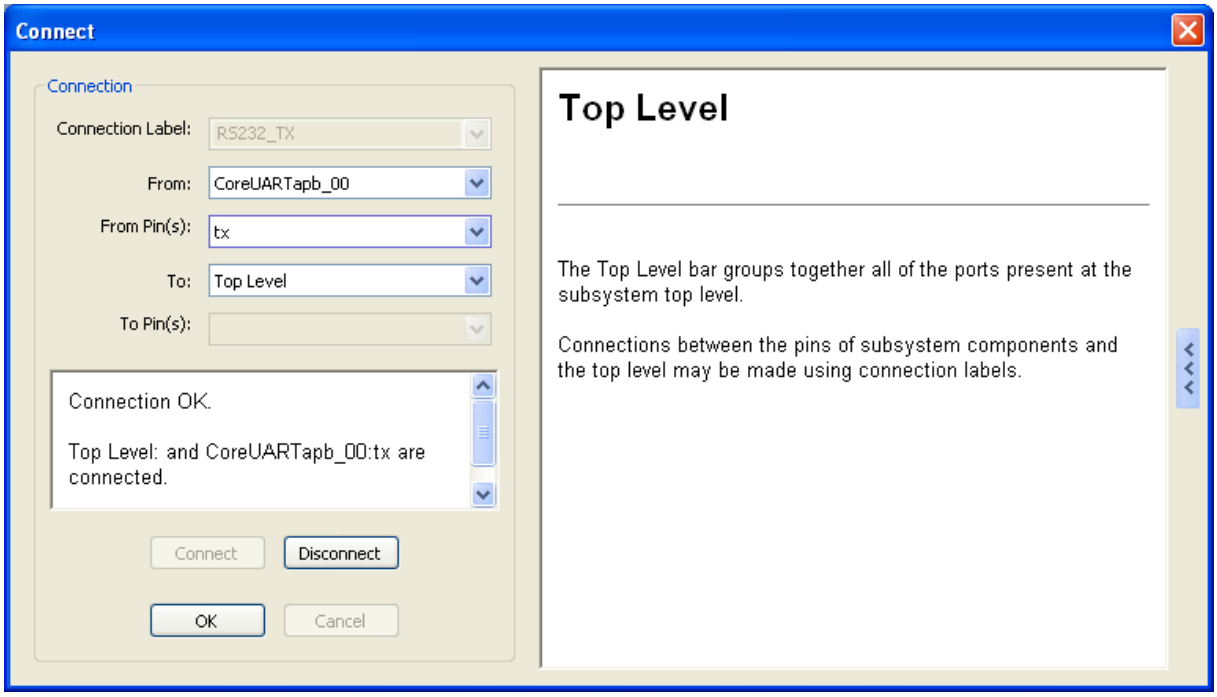
Move mouse button over CoreUARTapb which is configured as APBslave3 and click on the leftmost CONNECT buton.

Fill window as follows:



Click Connect button.

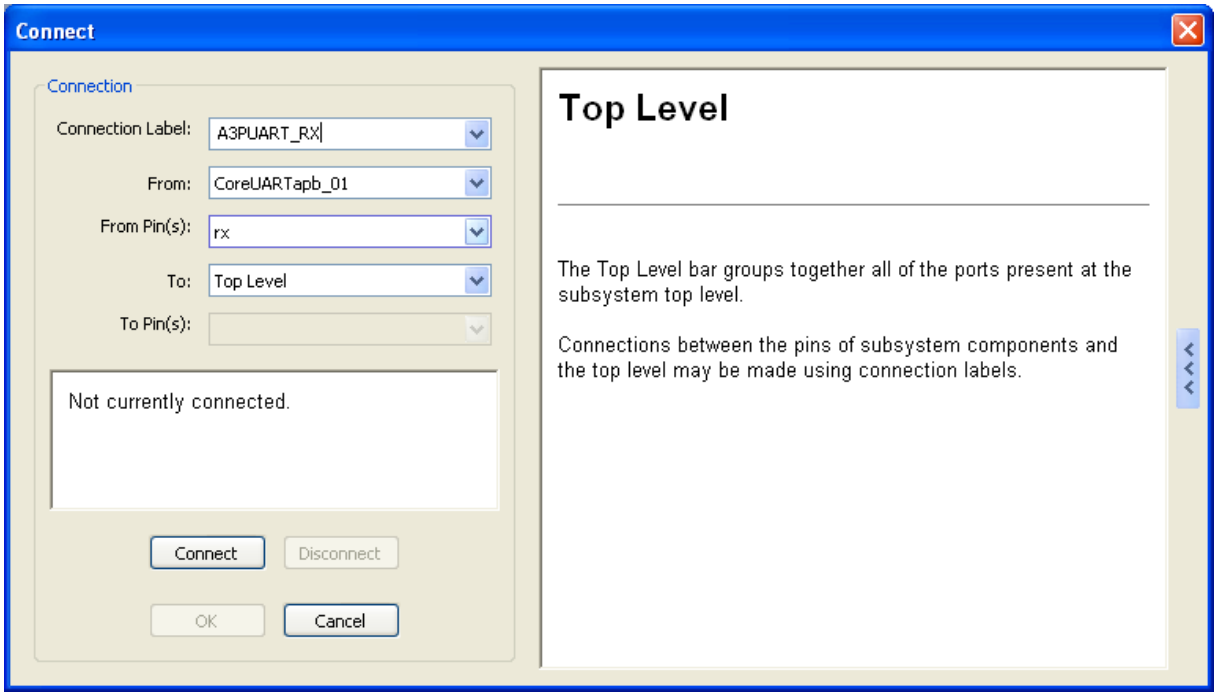
And for TX line:



Click OK.

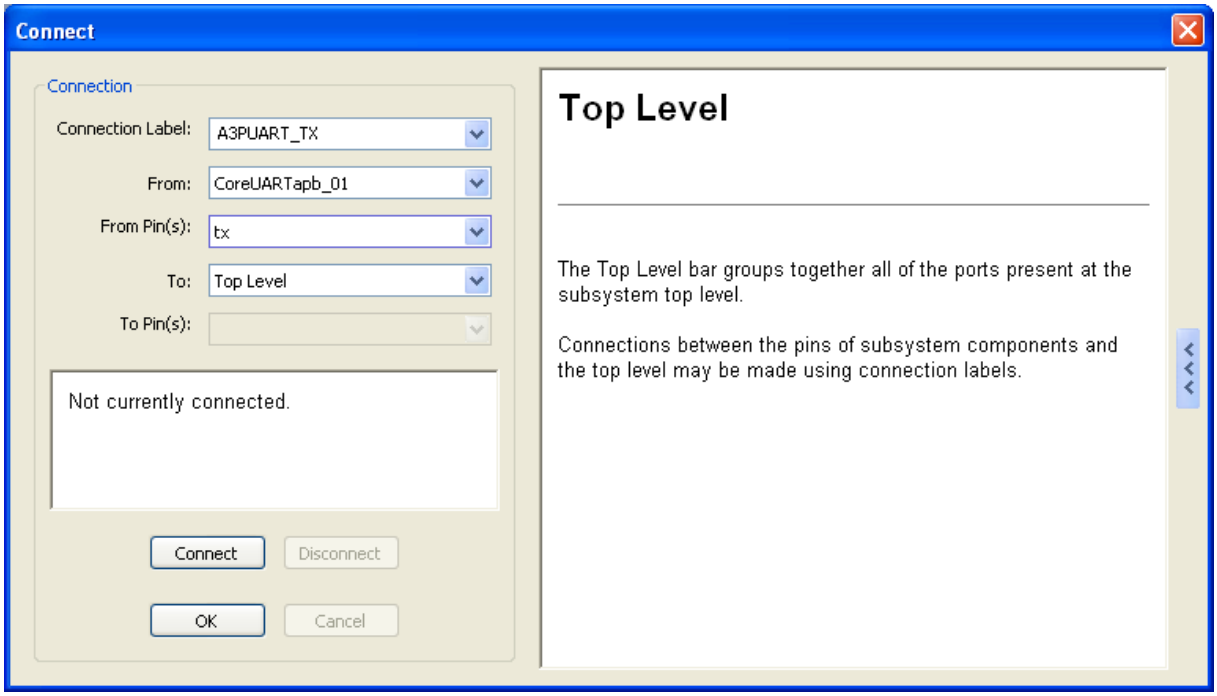
Move mouse button over CoreUARTapb which is configured as APBslave5 and click on the leftmost CONNECT buton.

Fill window as follows:



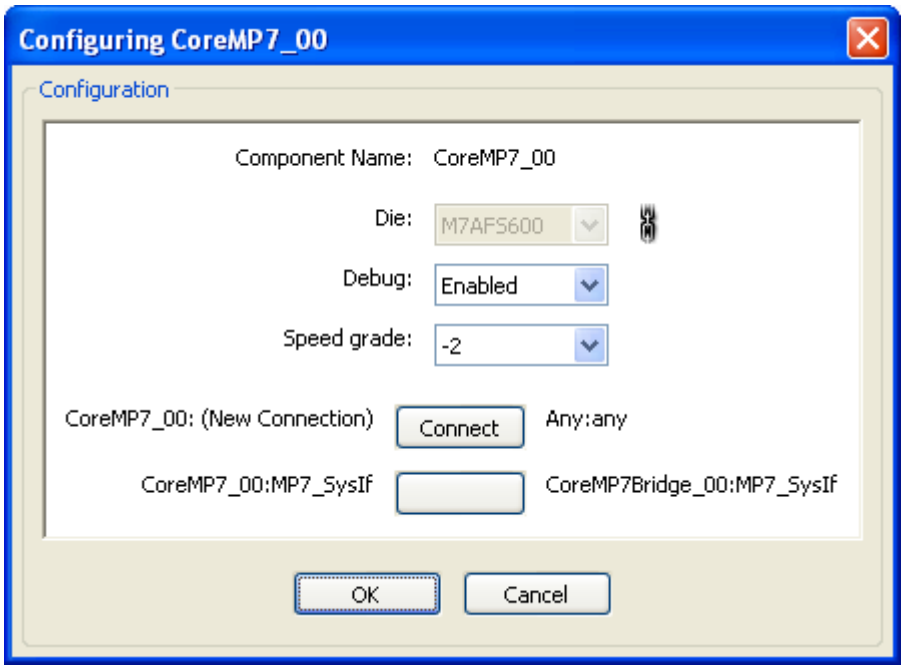
Click OK.

For TX line:

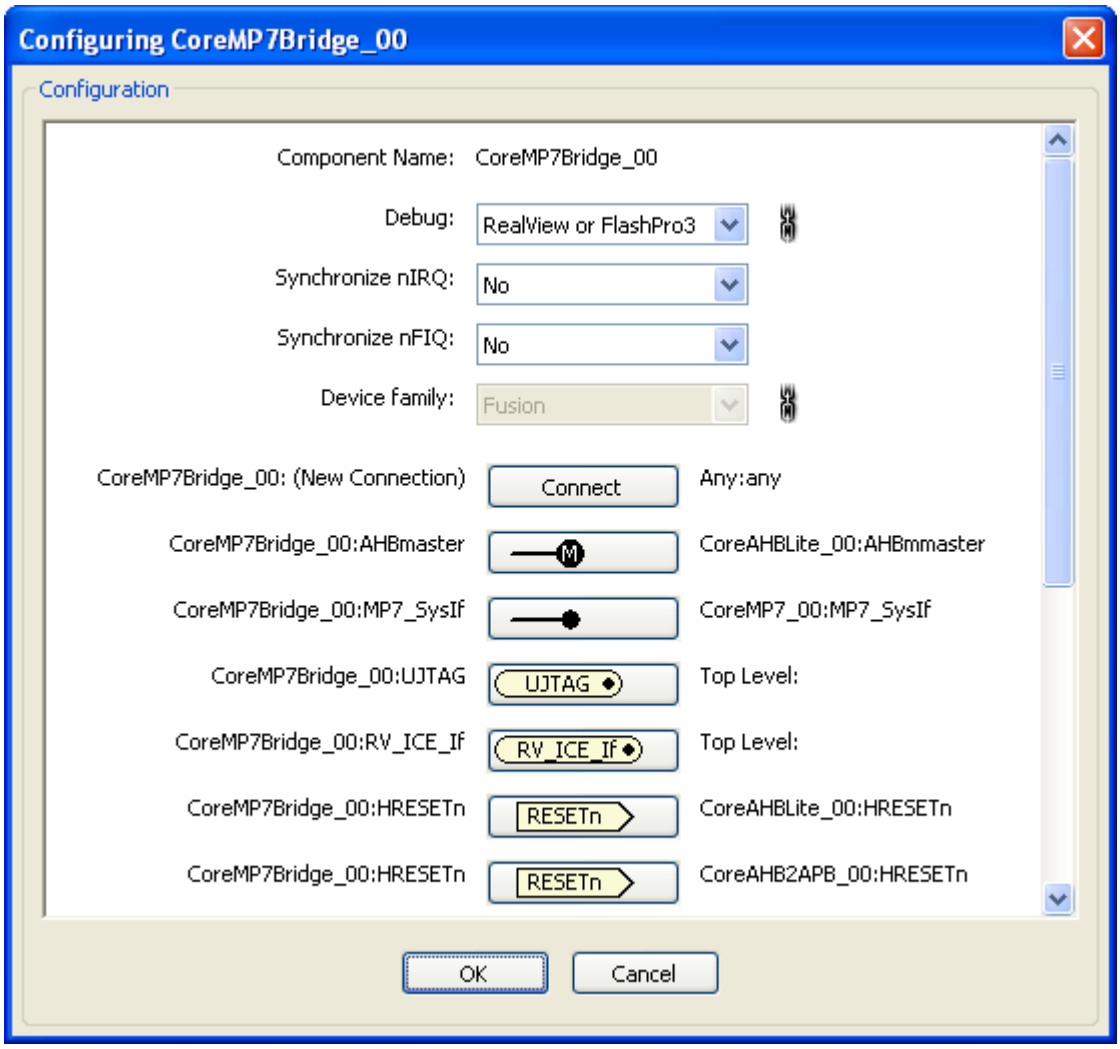


All interconnections are done for purpose of this tutorial.

Some of CoreXYZ bocks allow to configure its properties.
Move mouse cursor over CoreMP7 and click second icon – configure. Make sure of Debug is Enabled. Click OK.



Move mouse cursor over CoreMP7Bridge and click second icon – configure. Make sure of Debug is RealView or FlashPro3. Click OK



In the same way set Amount Of Internal Flash Required at 256kBytes in CoreAhbNvm. Amount of internal SRAM required set on 10kBytes in CoreAhbSram.

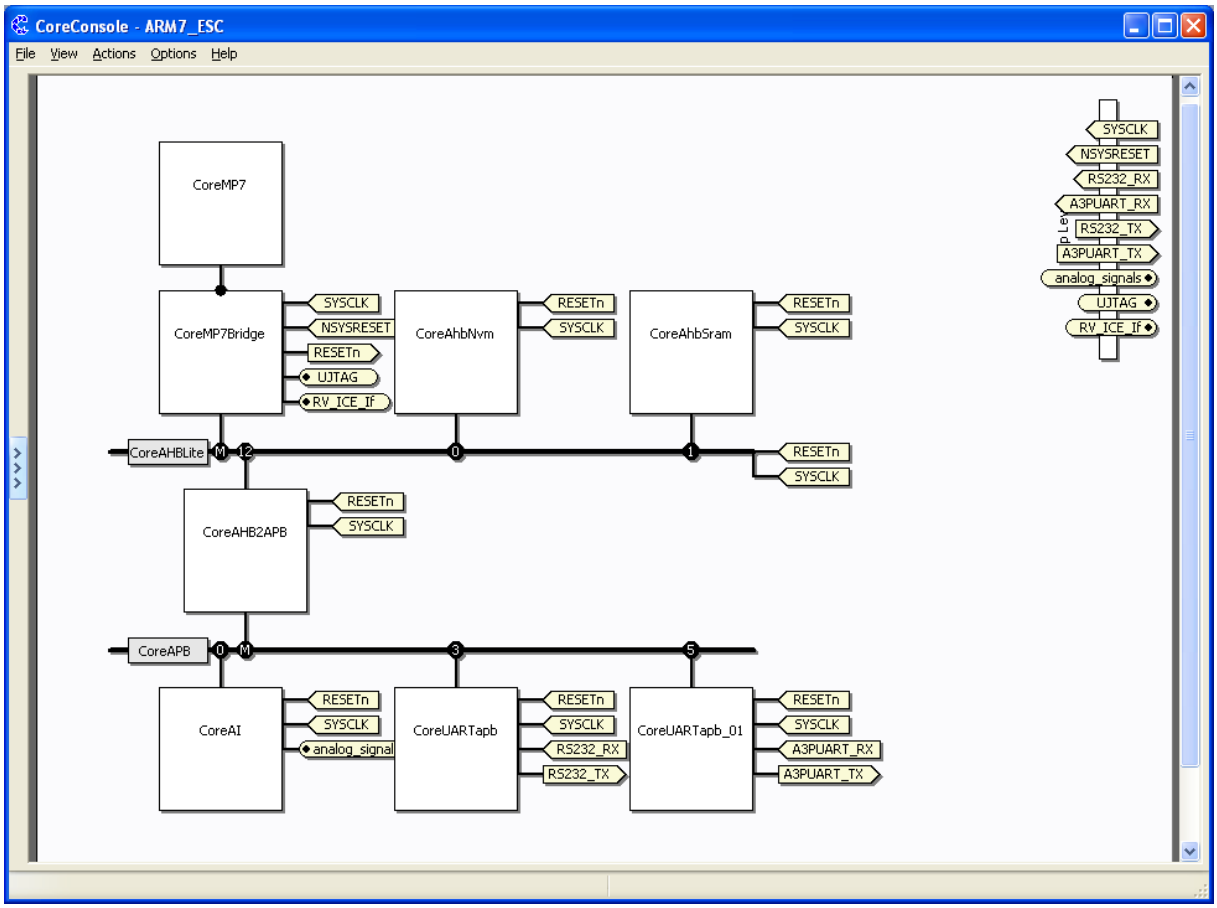
Click the **Configure** icon for the **CoreAI** component. Set the following in the Configuring CoreAI_00 dialog box. Accept the default settings for all other options and click **OK**:

- AT0 input: Temperature Monitor
- AT3 input: Temperature Monitor
- AT4 input: Temperature Monitor
- AC5 input: 0V to 8V analog input
- AT6 input: Temperature Monitor
- AG6 output: Software driven 25 mA sink
- AG7 output: Software driven 25 mA sink
- AT8 input: Temperature Monitor
- AG8 output: Software driven 25 mA sink
- ADC MODE control: Fixed constant
- ADC MODE fixed value: 12-bit
- TVC[7:0] pins control: Fixed constant
- TVC[7:0] fixed value: 0

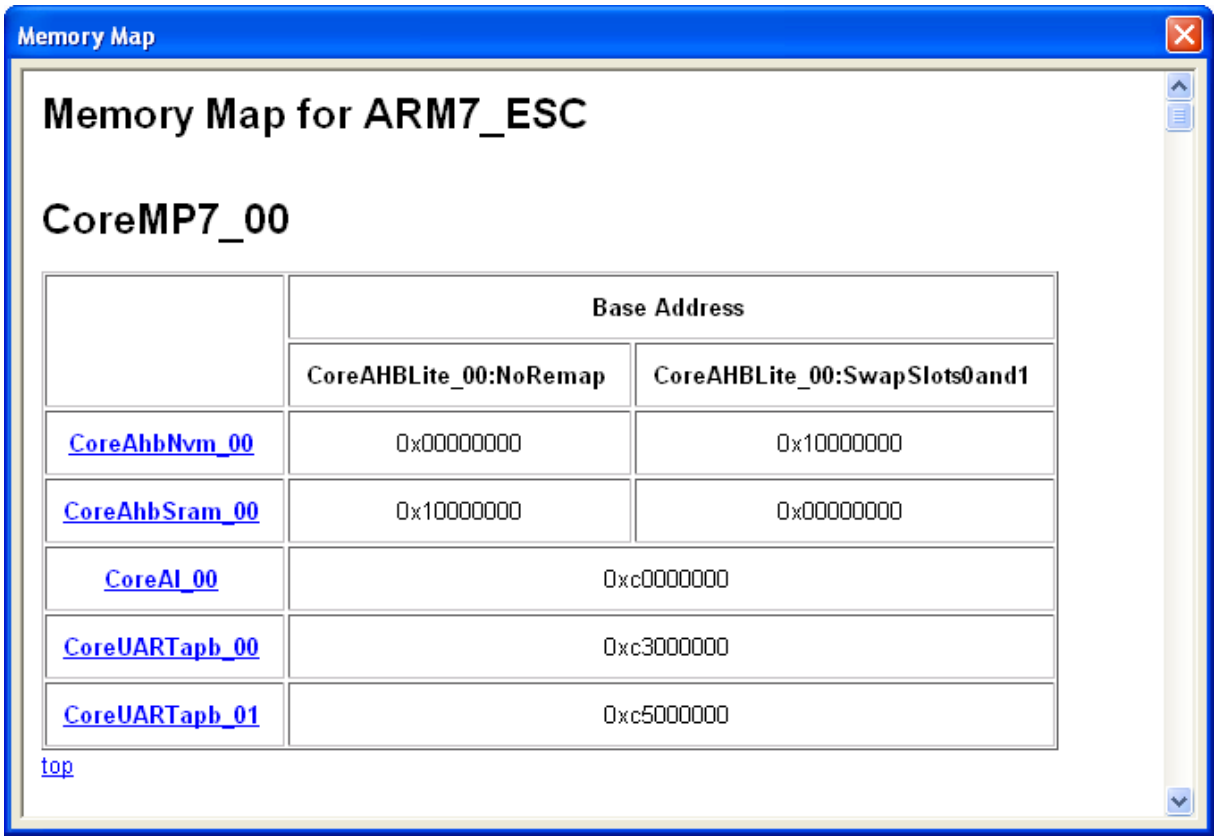
- STC[7:0] pins control: Fixed constant
- STC[7:0] constant: 23

Click Ok.

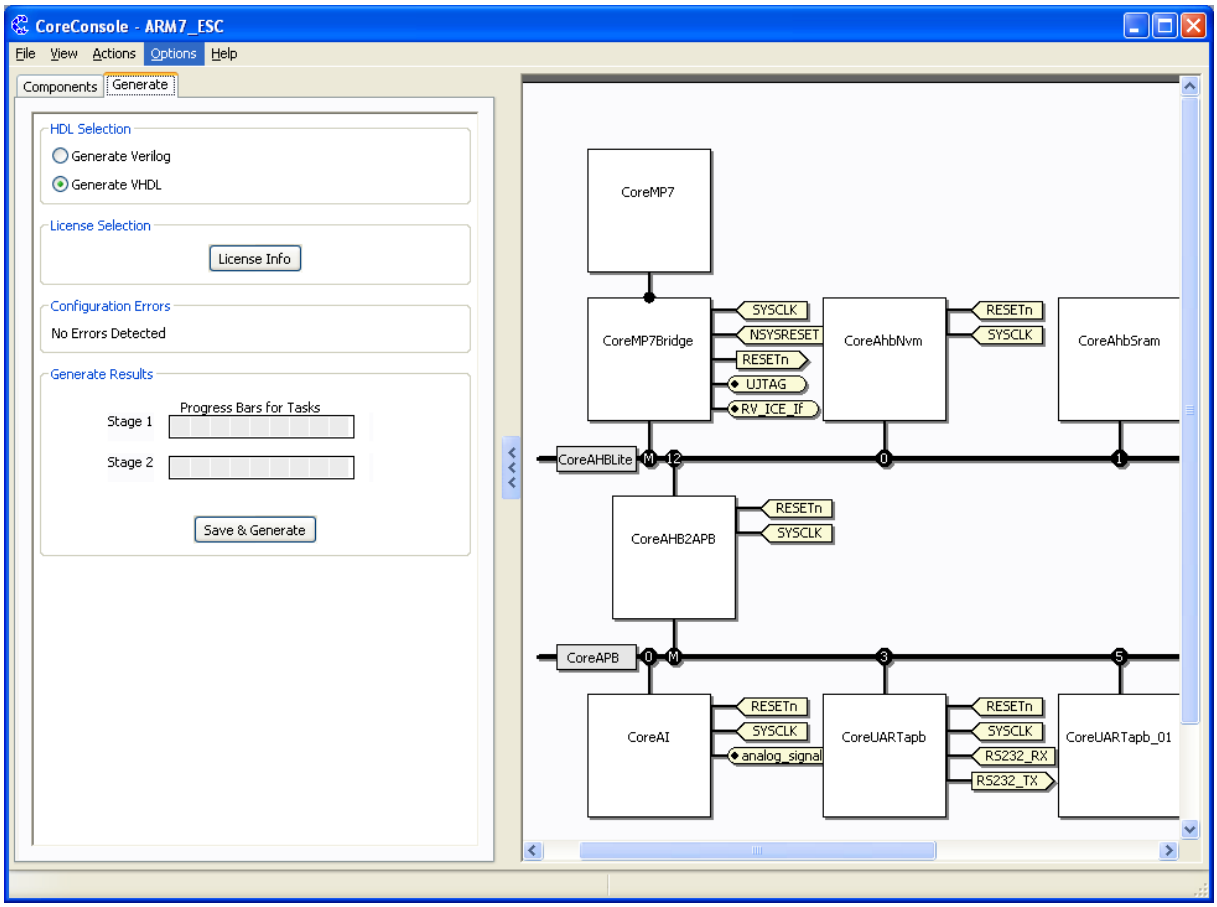
Your MP7 system could look like:



You can observe the memory map by: View -> Memory map.



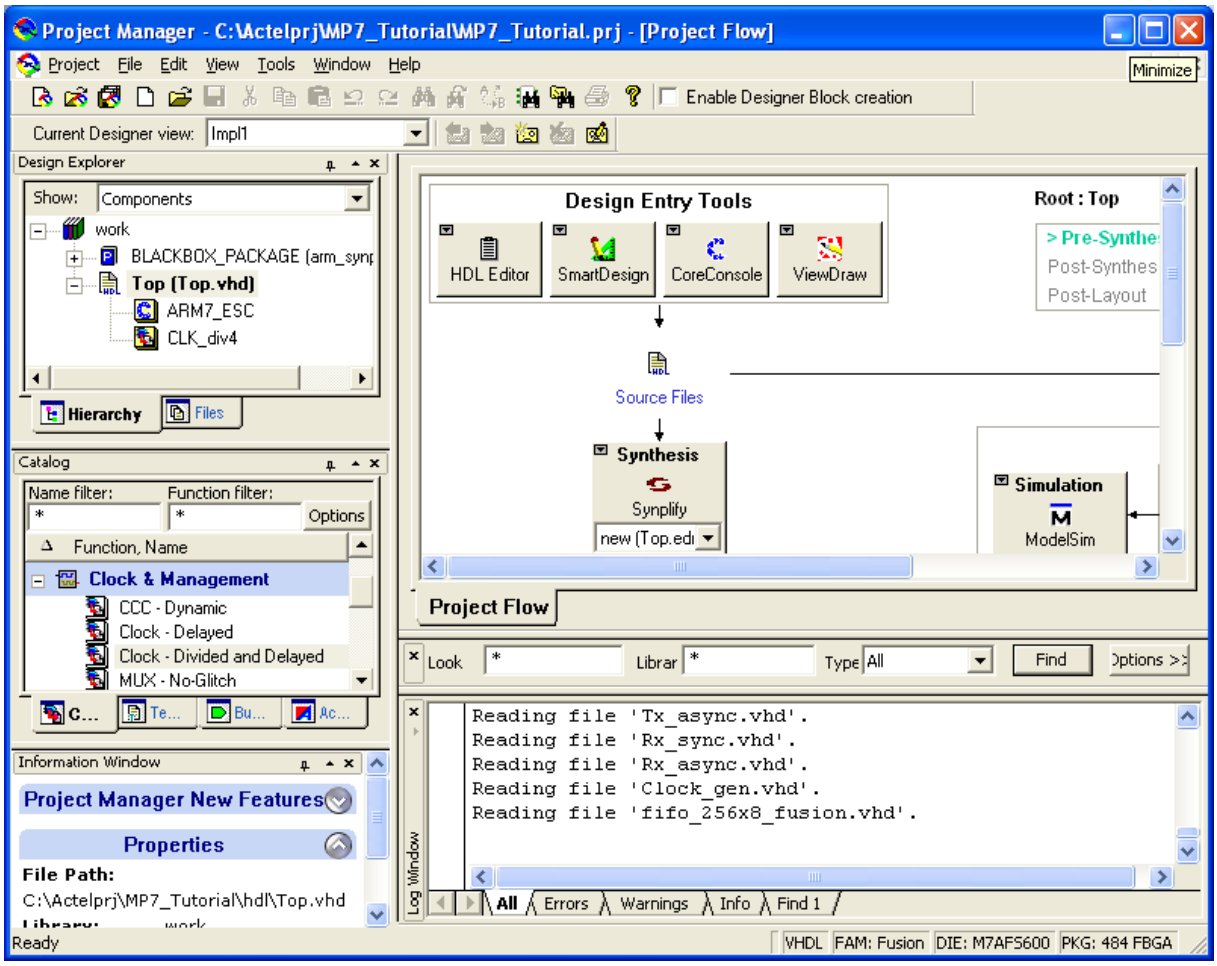
Select Generate Tab:



And click Save & Generate.

Now you can close the Core Console by File -> Exit

Note that question mark in Design Explorer changes into CoreConsole icon:



The sources are complete for physical design now, it is possible to synthetize, compile, place & route and download it into flash, but first, we prepare the firmware.

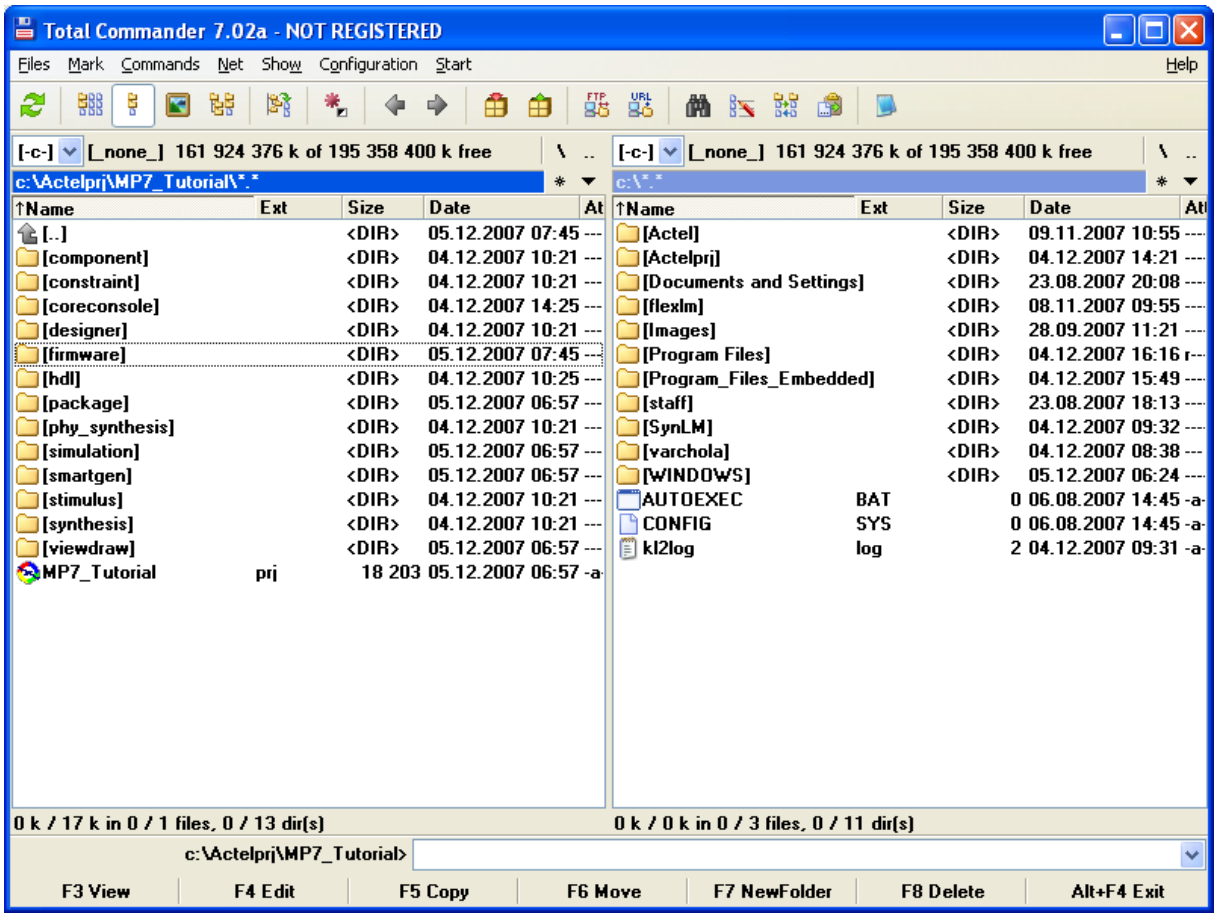
There is Two Ways to create MP7 firmware Intel HEX file

- ➔ In SoftConsole
- ➔ In Keil

We will use both ways, but, in this time (05. dec. 2007), we have under control generating the hex file in KEIL and dubuging in RAM in SoftConsole only.

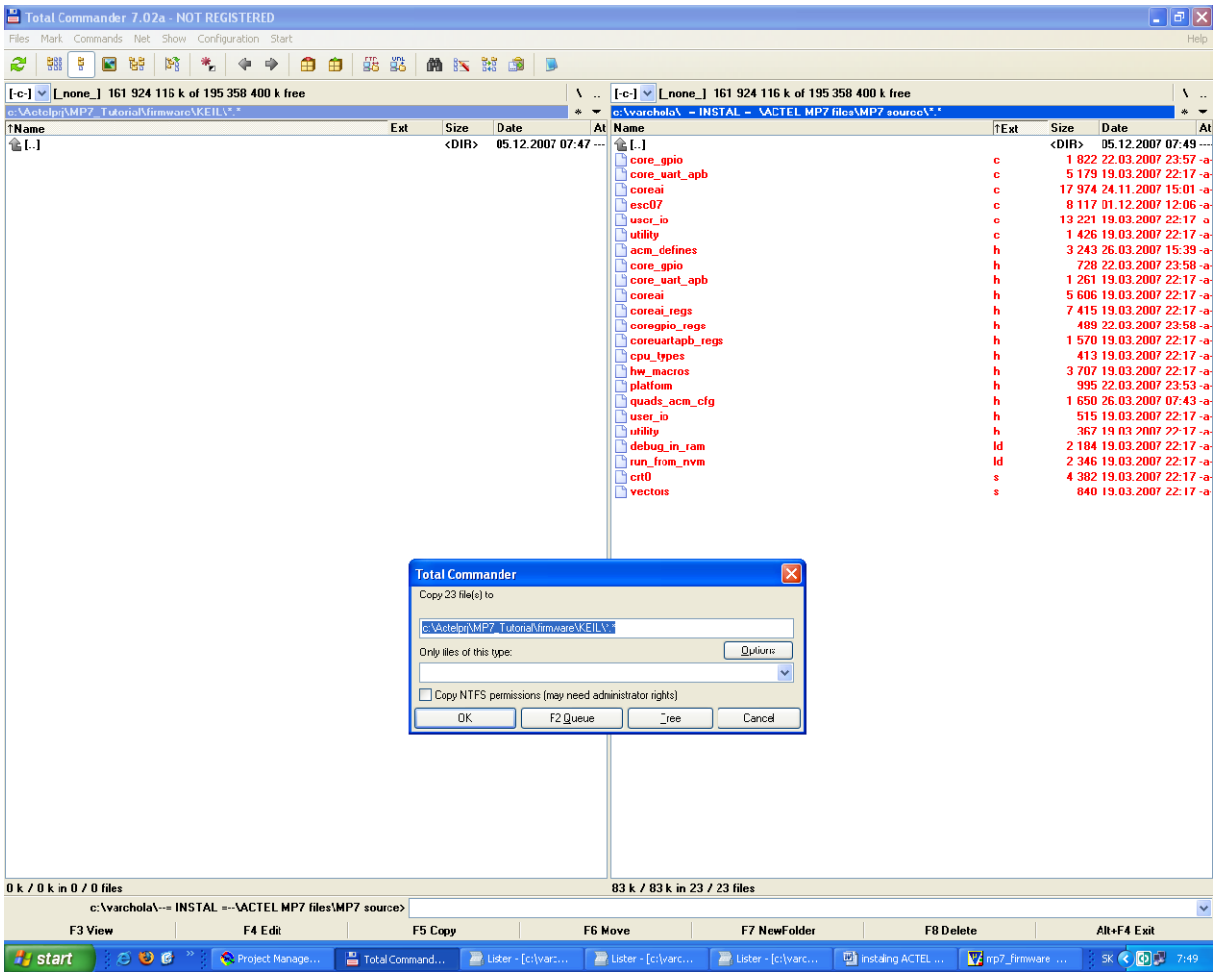
Run KEIL.

Create firmware folder in your ACTEL project directory

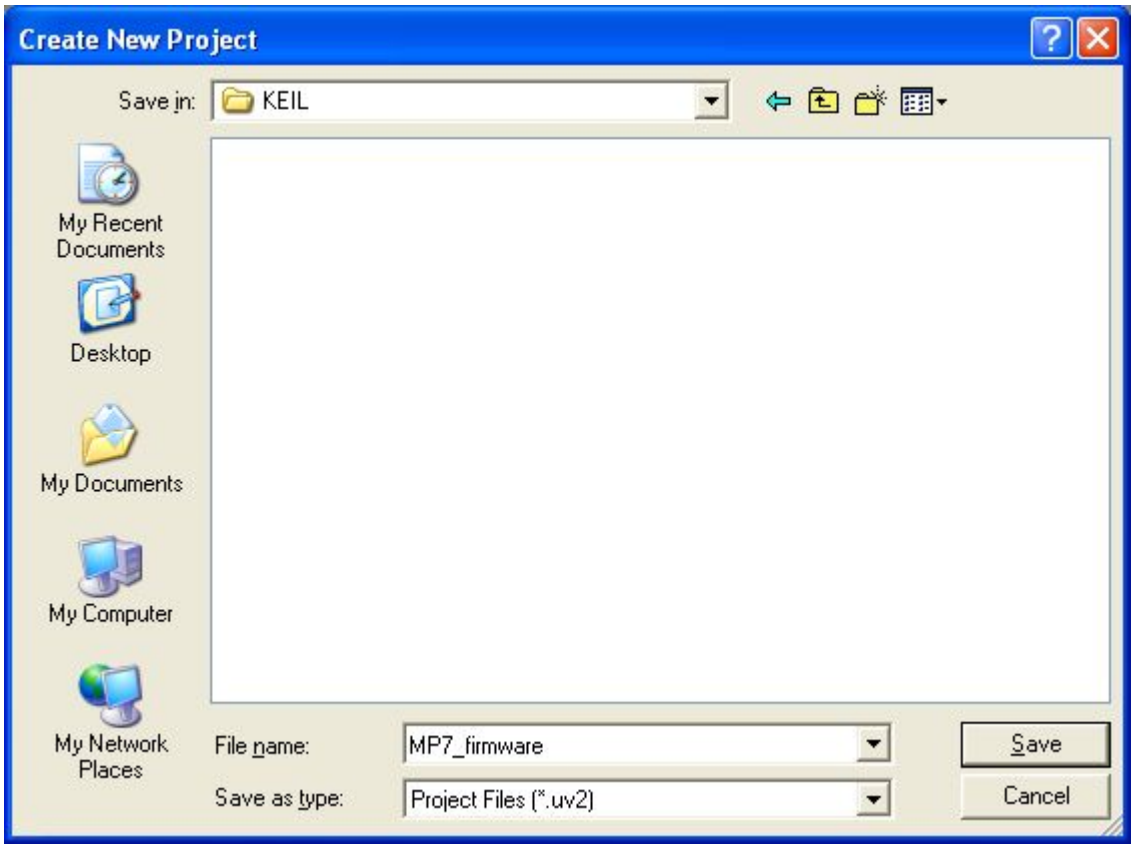


Open firmware directory and make subdirectories KEIL and SoftConsole in it.

Copy the source files (.c .h .ld .s) into KEIL subdirectory:

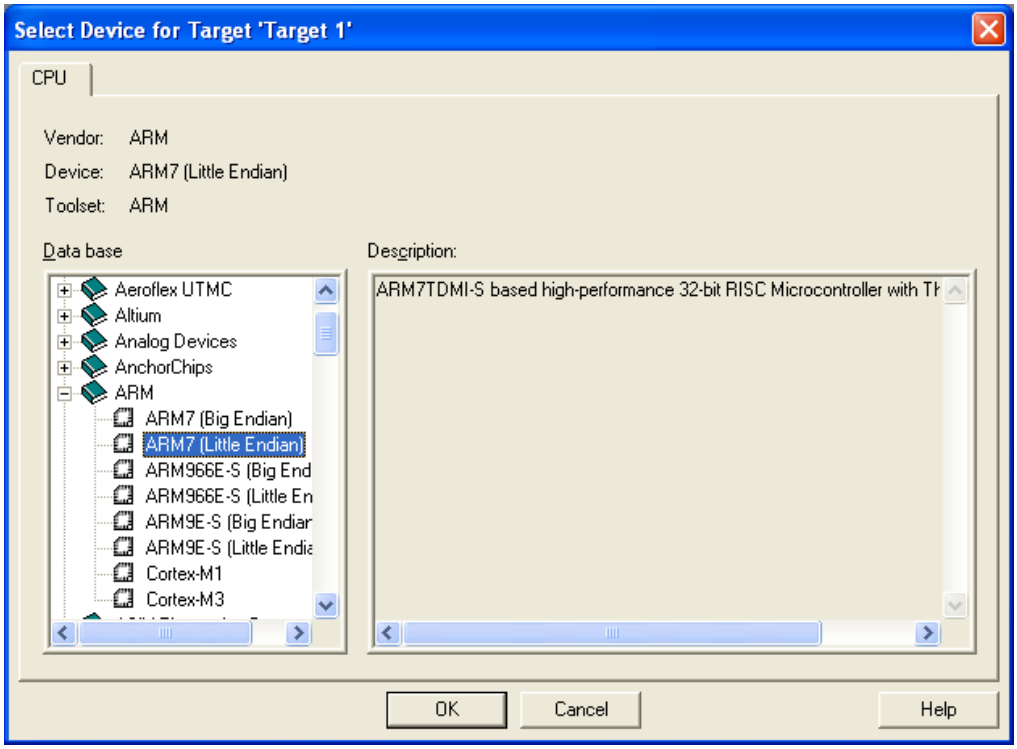


In Keil toolbar menu choose Project -> New uVision Project



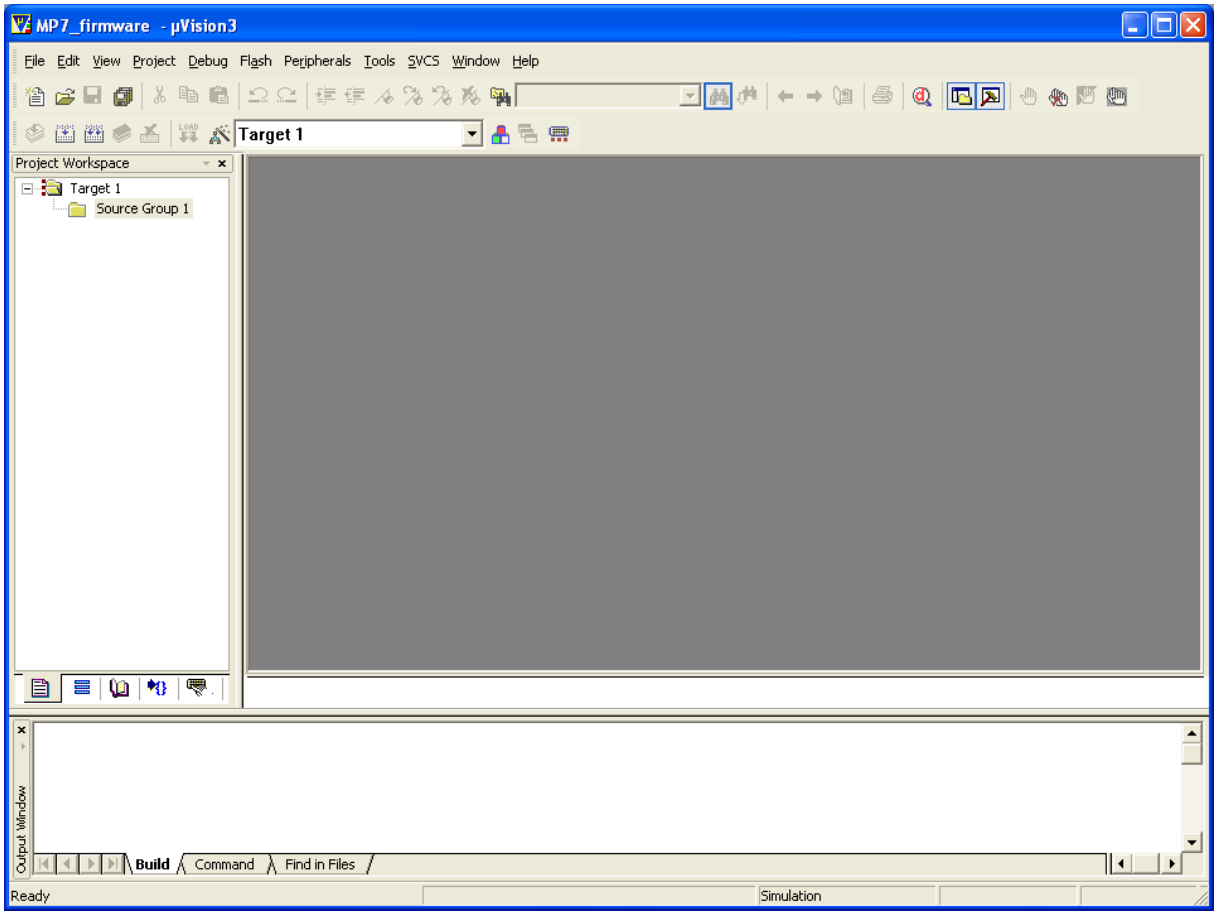
Enter name of new project (MP7_firmware) and save it in ...\your_actel_project\firmware\KEIL

Next you will be promt to choose device. In latest version of KEIL (05. dec. 2007) there is no option of MP7, so, you could to choose ARM -> ARM7 (Little Endian)

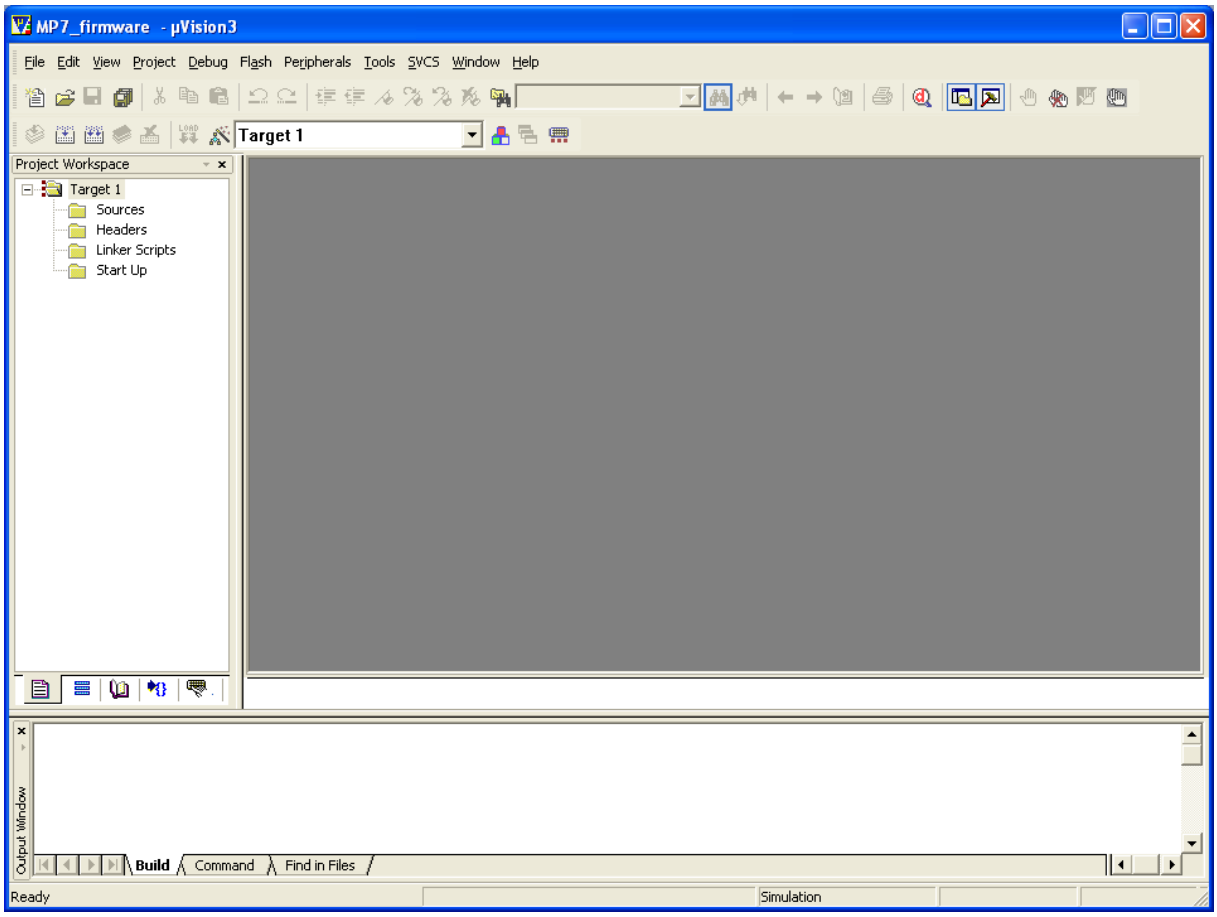


Click OK.

You will see:

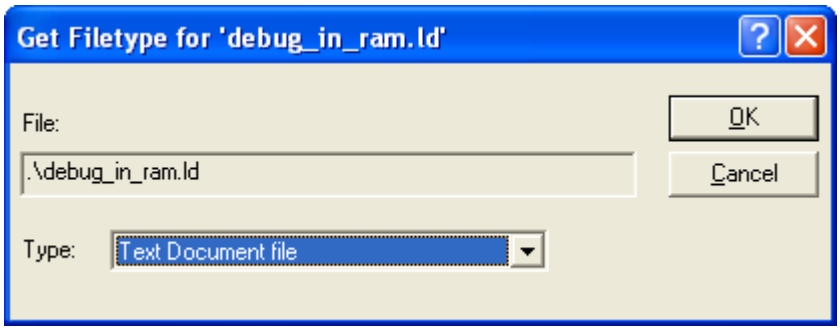


Now we prepare file structure in Project Workspace window.
Click twice slowly (not double-click) on Source Group1 to rename it. Enter new name: Sources. Right click on Target 1 and choose New Group and name this group Headers. Make further two groups „Linker Scripts“ and „Start Up“.

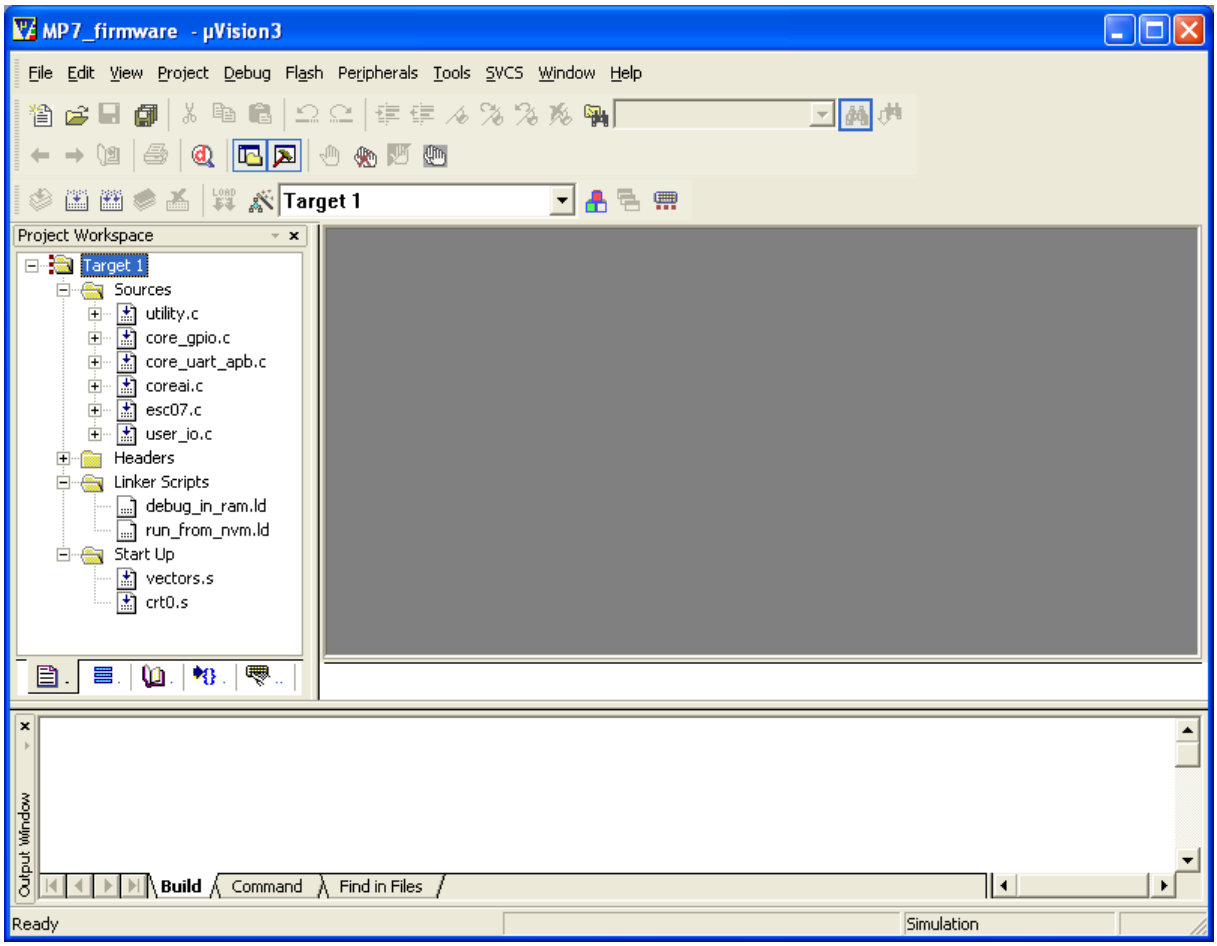


With double click on each you can add files. Add to:
Sources group all .c files
Headers group all .h files
Linker Scripts group all .ld files
Start Up group all .s files

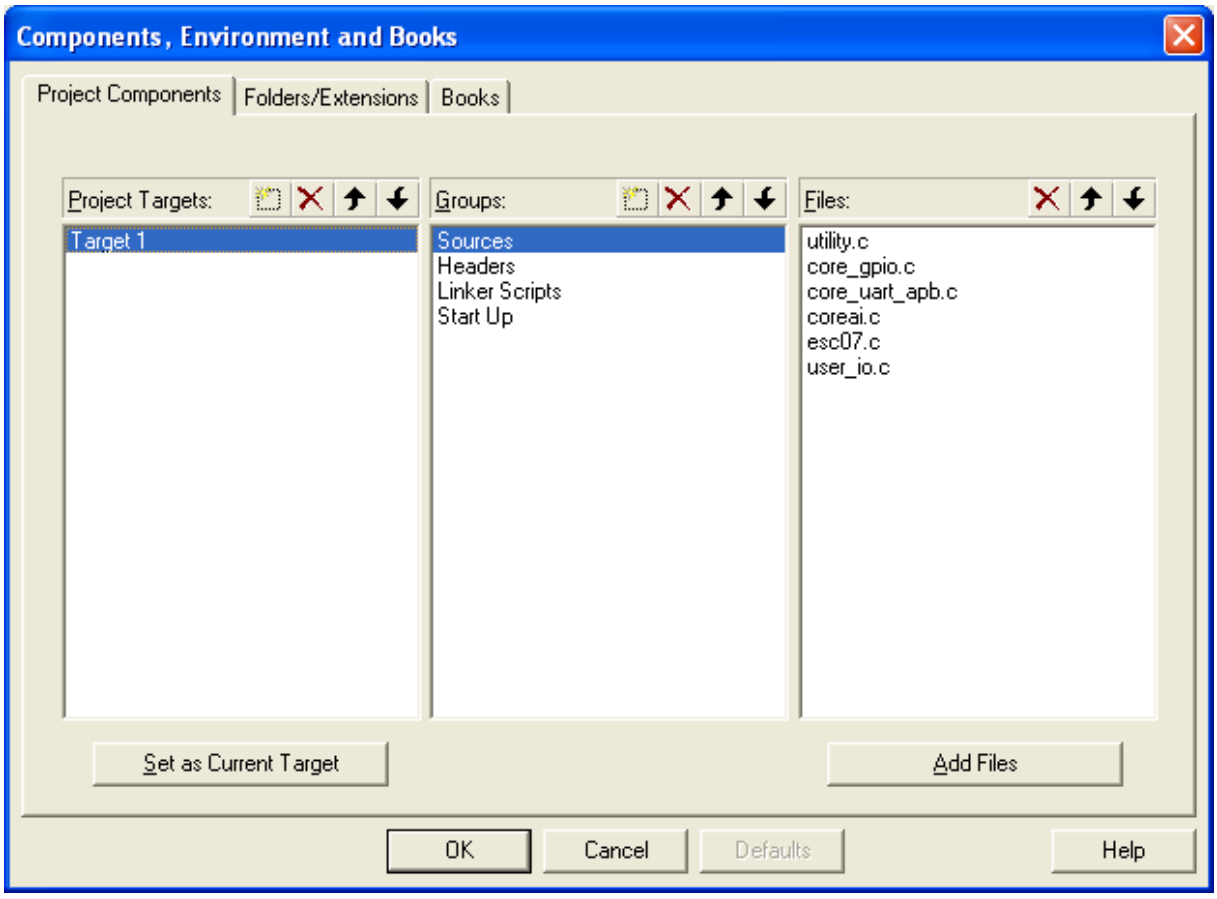
You will be promt about file type for .ld files, choose Text Documentation:



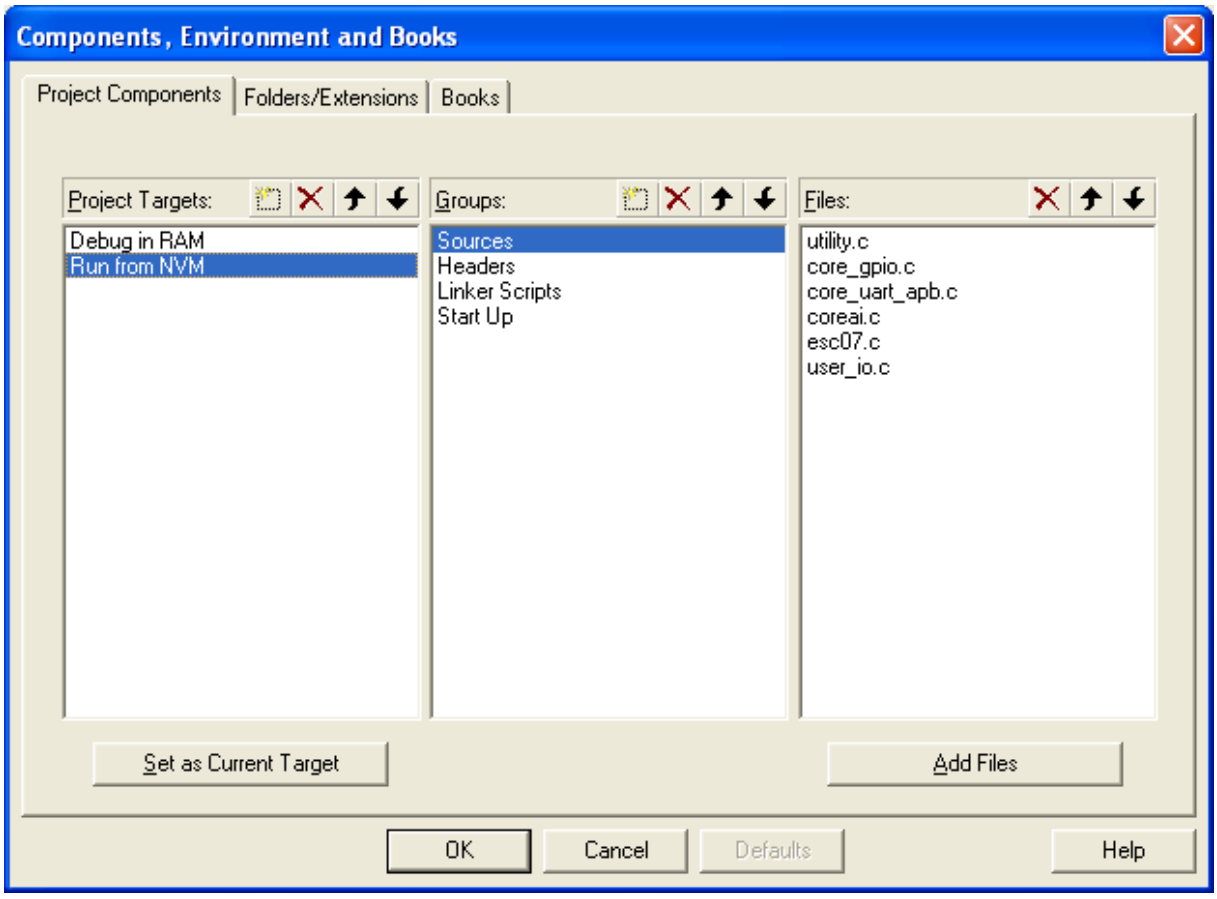
So now you could see:



Keil provides option to create more „Targets“ with various settings e.g. for Debuging in RAM or Run from FLASH settings. We create two targets instead Target 1: debug_in_ram and run_from_nvm according to names of linker scripts.
Choose from Toolbar menu: Project -> Manage -> Environments, Components, Books...
Select Project Components Tab
You will see:

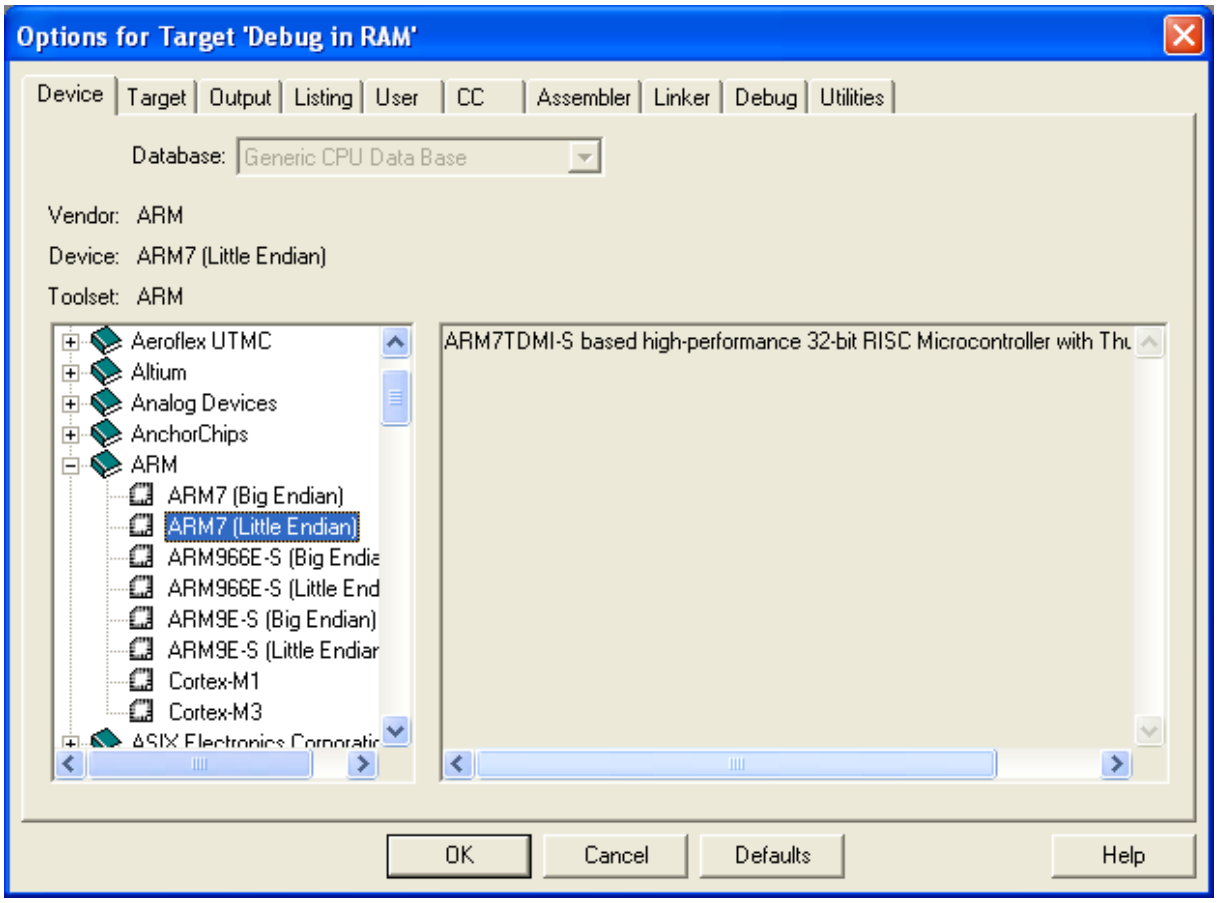


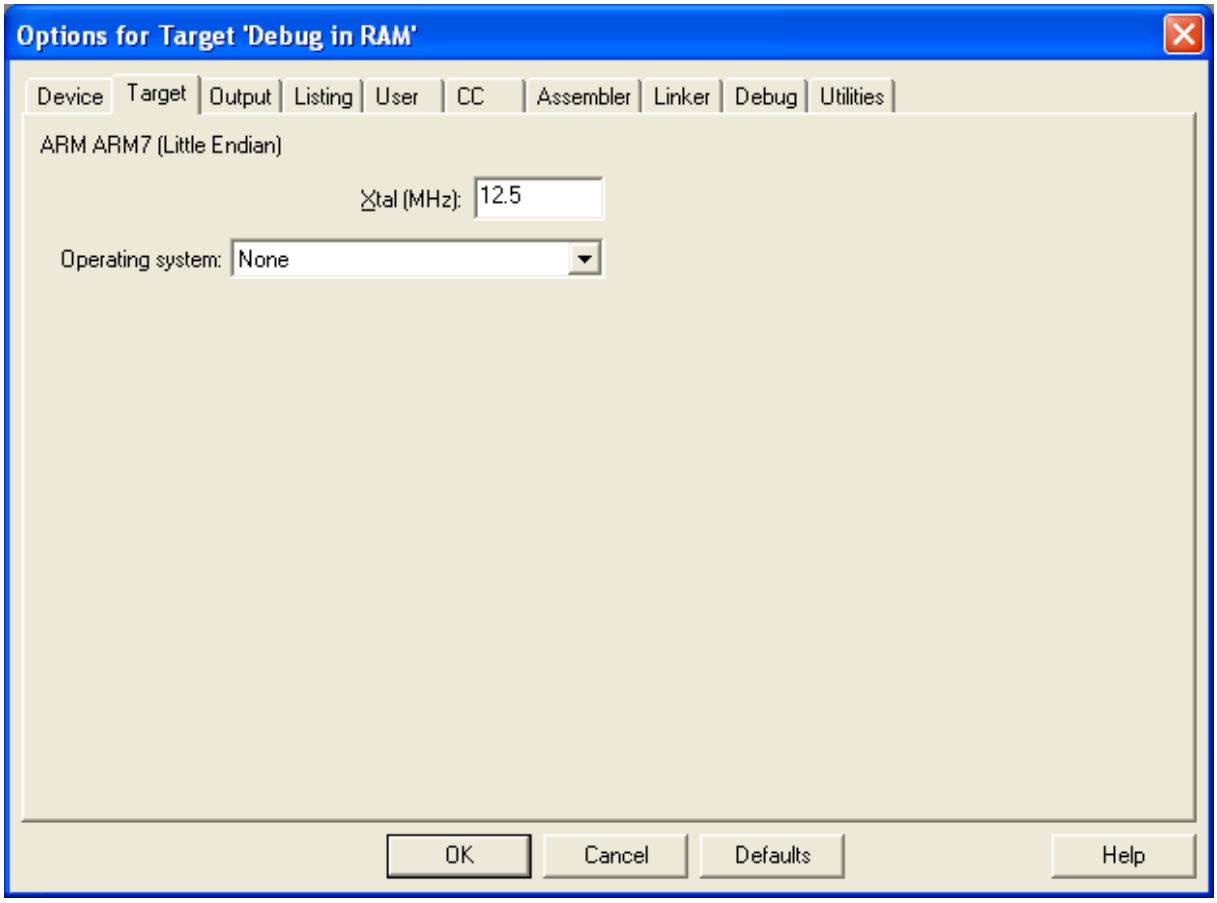
In Project Targets, double click on Target 1 to rename it into Debug in RAM. Then click onto leftmost icon: new(insert) to create new Target: Run from NVM.

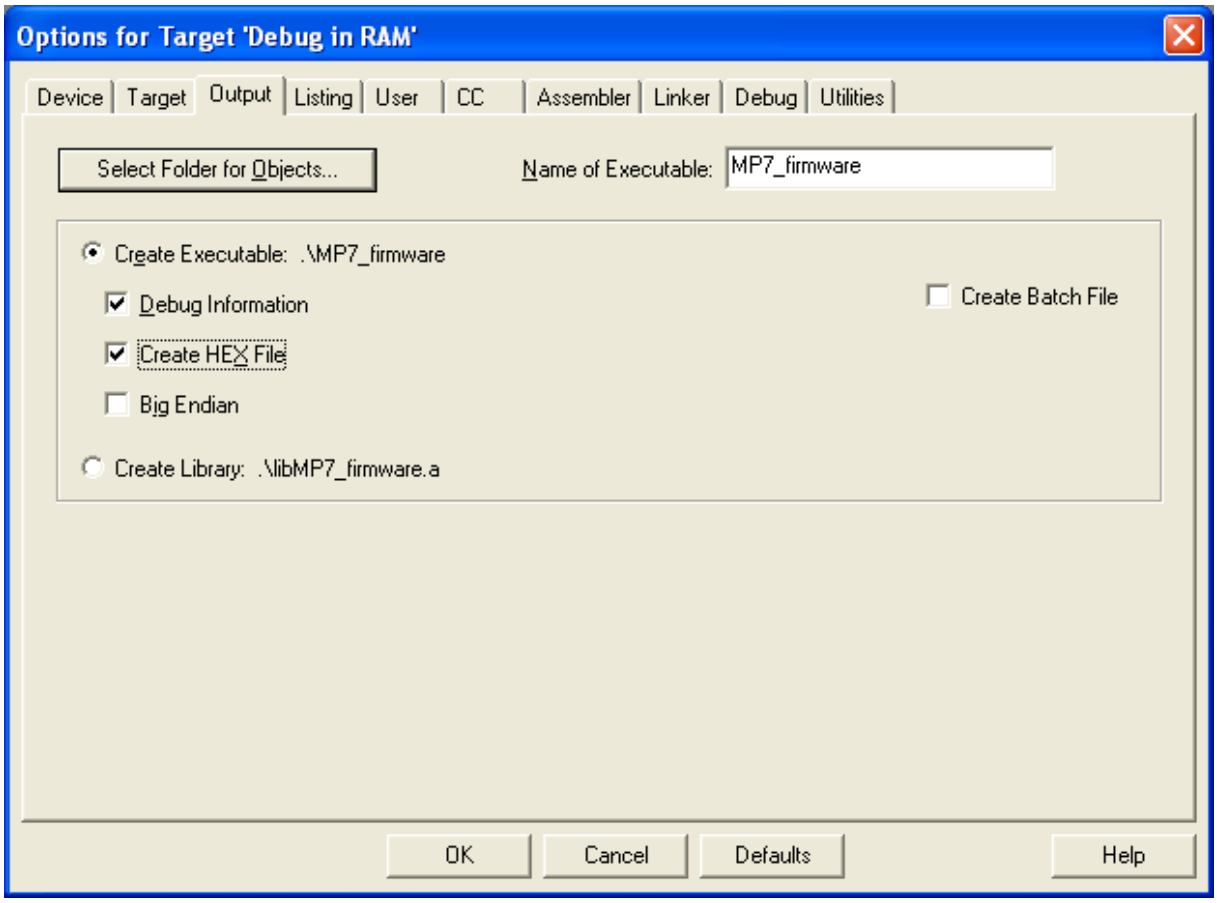


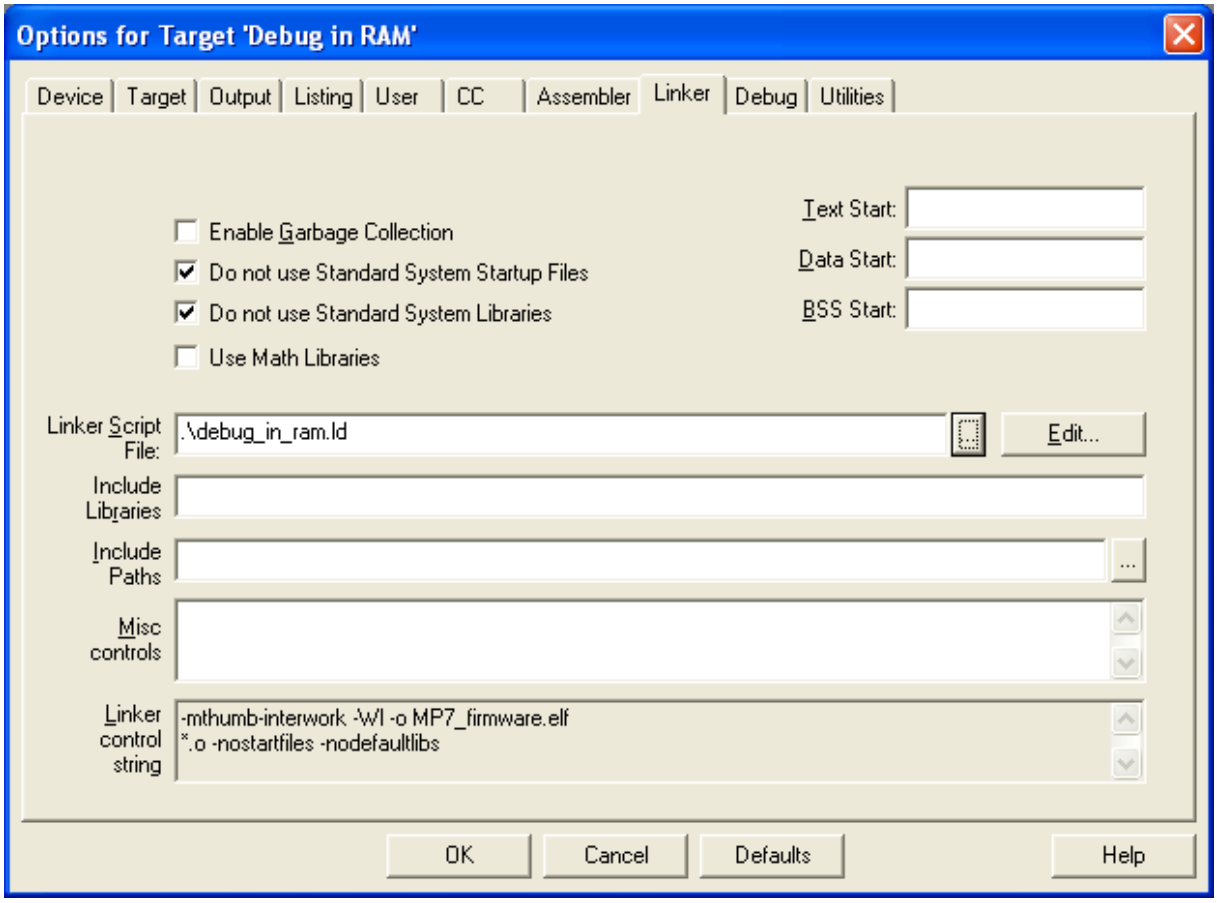
Click OK.

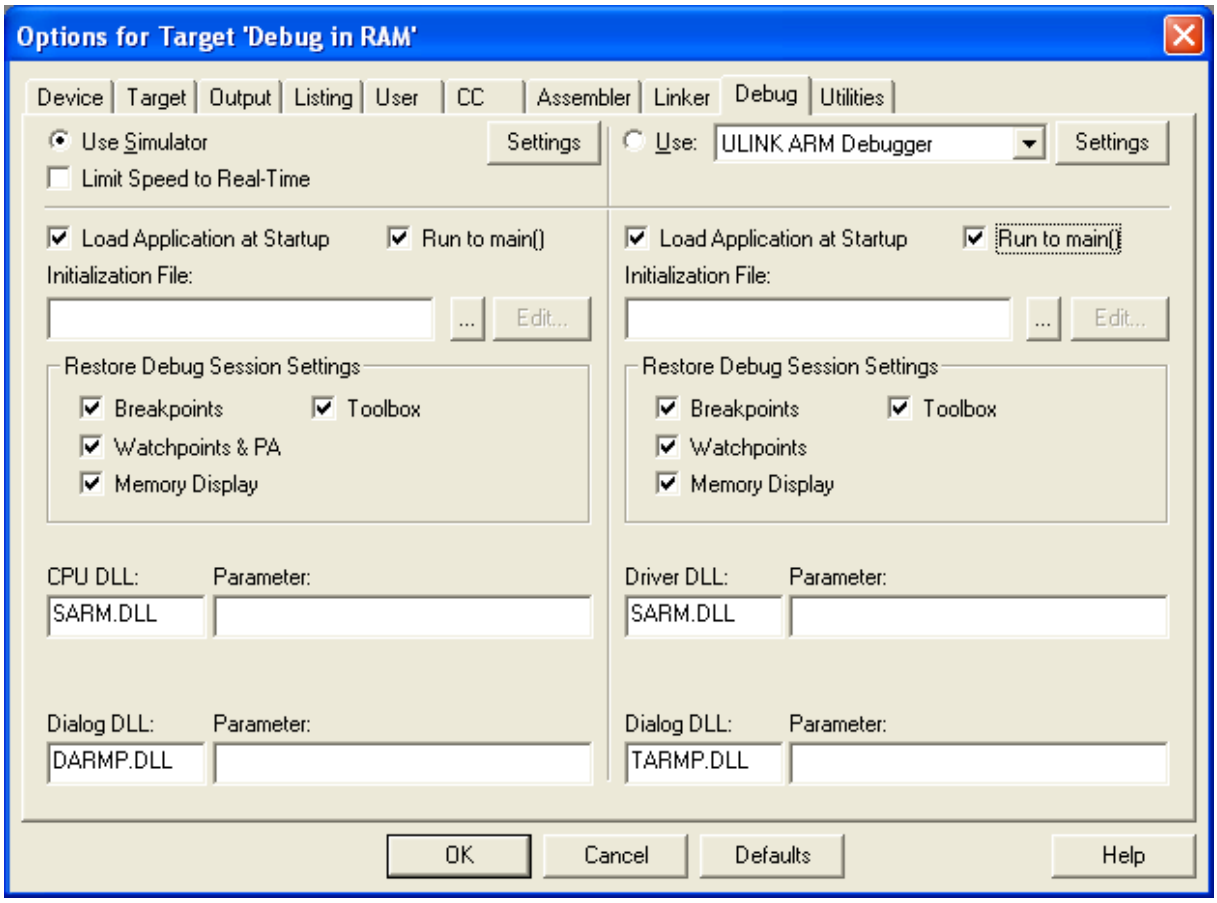
Now, set make the same settings for Debug in RAM:
Project -> Options for Target „Debug in RAM“


Options for Target 'Debug in RAM'
Device
Target
Output
Listing
User
CC
Assembler
Linker
Debug
Utilities
Database:
Generic CPU Data Base
Vendor: ARM
Device: ARM7 (Little Endian)
Toolset: ARM
Aeroflex UTMC
Altium
Analog Devices
AnchorChips
ARM
ARM7 (Big Endian)
ARM7 (Little Endian)
ARM966E-S (Big Endia
ARM966E-S (Little End
ARM9E-S (Big Endian)
ARM9E-S (Little Endiar
Cortex-M1
Cortex-M3
ARM7TDMI-S based high-performance 32-bit RISC Microcontroller with Thu
OK
Cancel
Defaults
Help

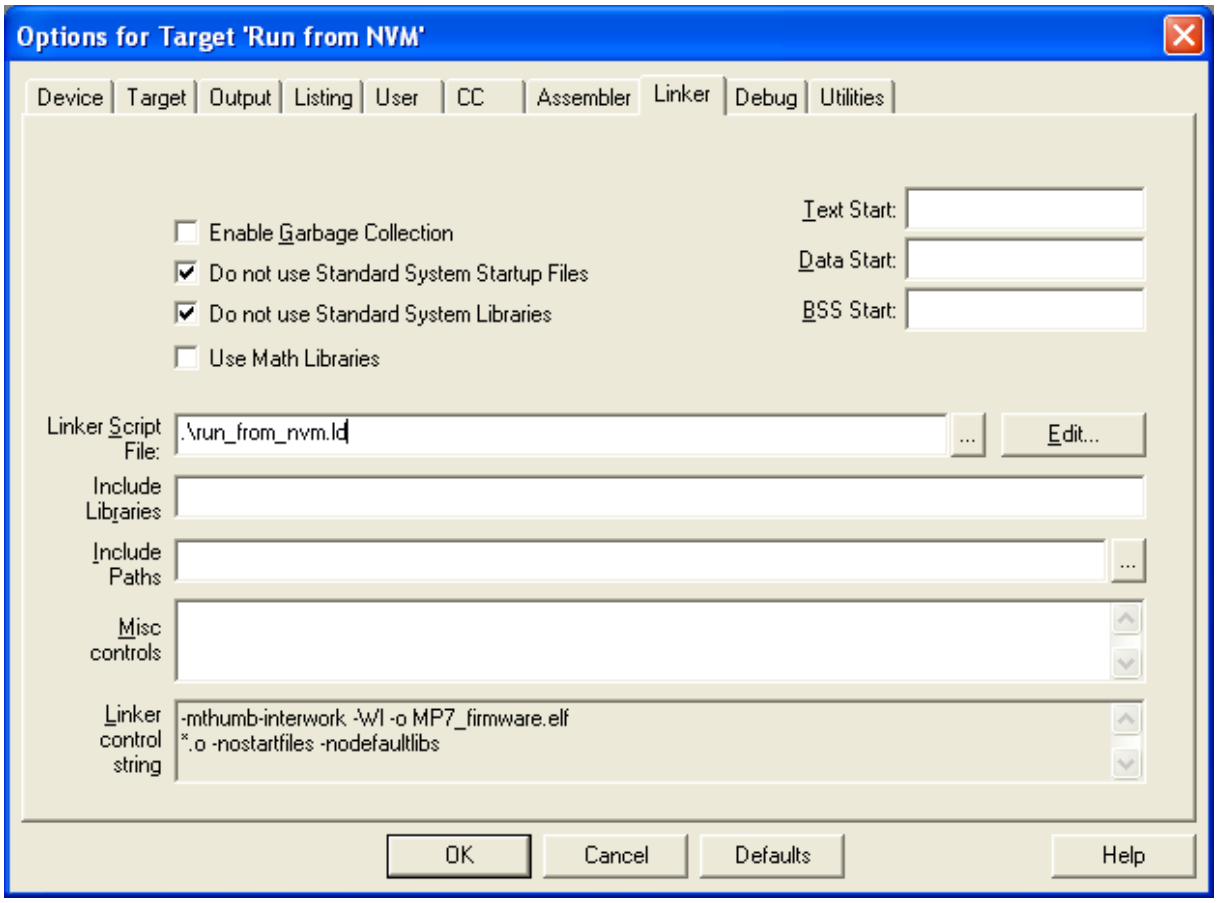

Options for Target 'Debug in RAM'
Device
Target
Output
Listing
User
CC
Assembler
Linker
Debug
Utilities
ARM ARM7 (Little Endian)
Xtal (MHz):
12.5
Operating system:
None
OK
Cancel
Defaults
Help


Options for Target 'Debug in RAM'
Device
Target
Output
Listing
User
CC
Assembler
Linker
Debug
Utilities
Select Folder for Objects...
Name of Executable:
MP7_firmware
Create Executable: .\MP7_firmware
Debug Information
Create HEX File
Big Endian
Create Batch File
Create Library: .\libMP7_firmware.a
OK
Cancel
Defaults
Help


Options for Target 'Debug in RAM'
Device
Target
Output
Listing
User
CC
Assembler
Linker
Debug
Utilities
Enable Garbage Collection
Do not use Standard System Startup Files
Do not use Standard System Libraries
Use Math Libraries
Text Start:
Data Start:
BSS Start:
Linker Script File:
.\debug_in_ram.ld
Edit...
Include Libraries
Include Paths
Misc controls
Linker control string
-mthumb-interwork -Wl -o MP7_firmware.elf
*.o -nostartfiles -nodefaultlibs
OK
Cancel
Defaults
Help



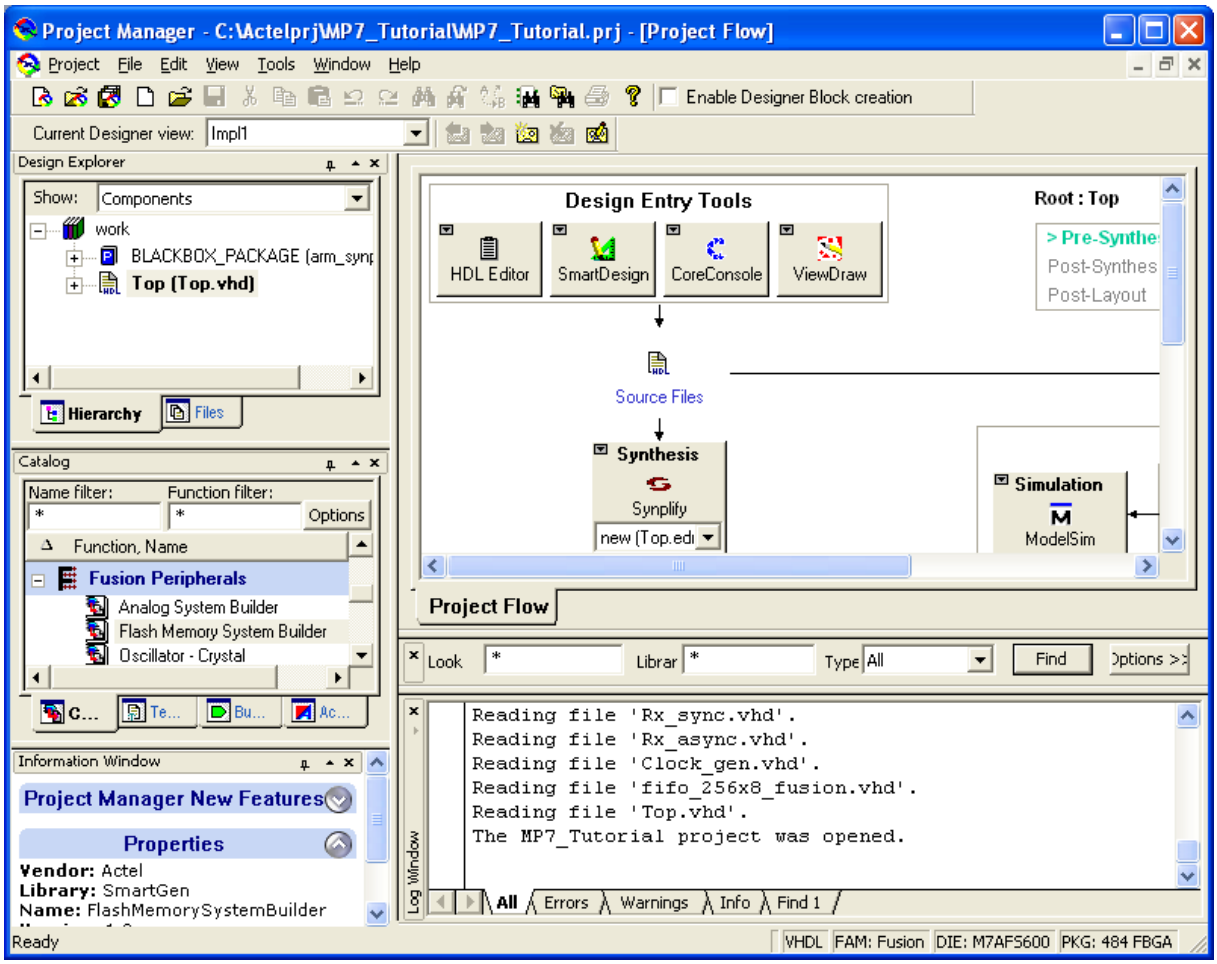
Make the same settings for Run from NVM, instead Linker, where browse the .\run_from_nvm.ld



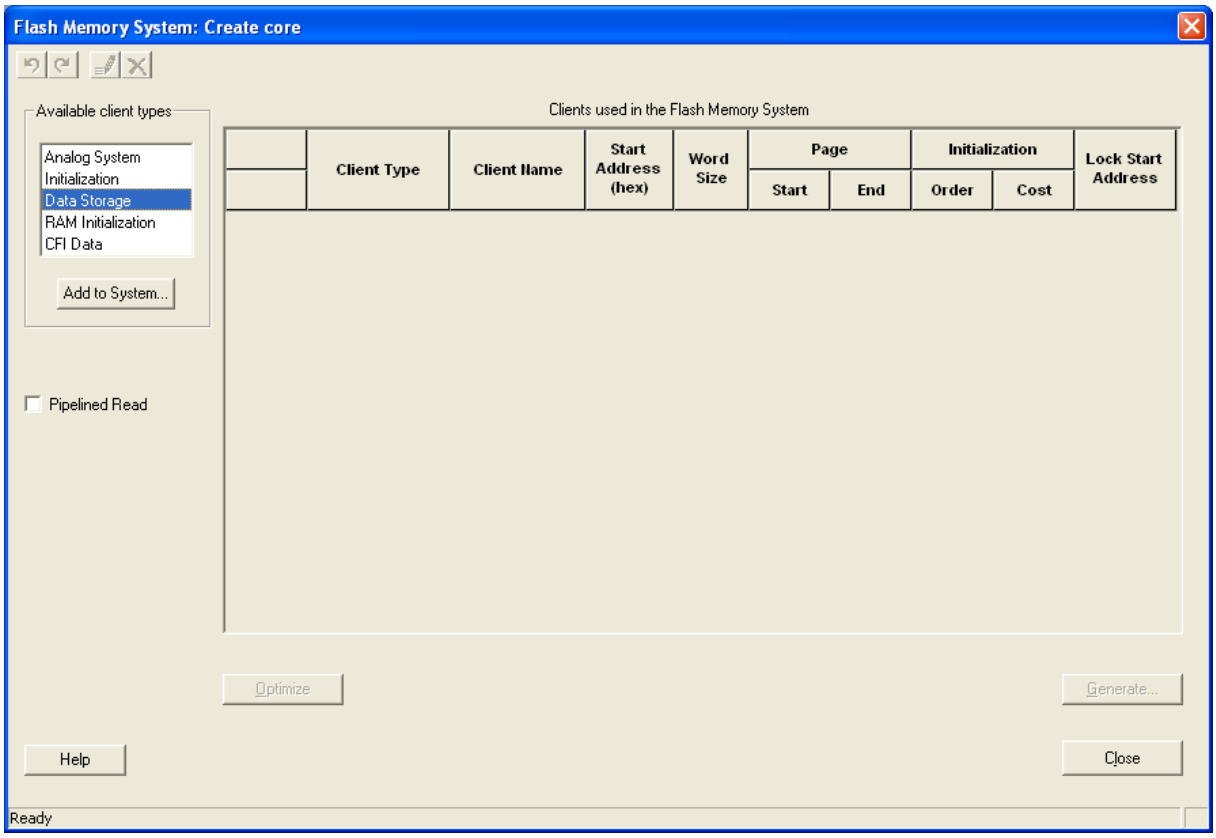
Build the project by:
Project -> Rebuild all target files...

Asigning the Intel HEX file into Libero project:

In LIBERO Catalog expand the Fusion peripherals and choose the Flash Memory System Builder.

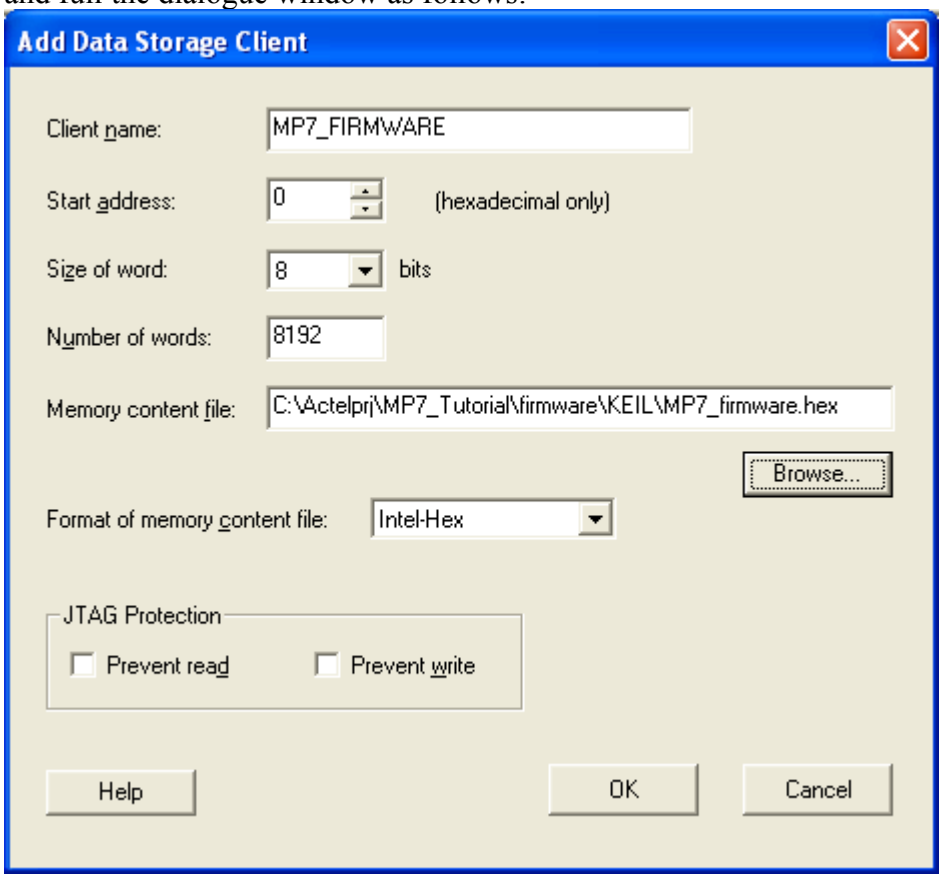

Project Manager - C:\Actelprj\MP7_Tutorial\MP7_Tutorial.prj - [Project Flow]
Project File Edit View Tools Window Help
Enable Designer Block creation
Current Designer view: Impl1
Design Explorer
Show: Components
work
BLACKBOX_PACKAGE (arm_synp
Top (Top.vhd)
Hierarchy
Files
Catalog
Name filter:
Function filter:
Options
Function, Name
Fusion Peripherals
Analog System Builder
Flash Memory System Builder
Oscillator - Crystal
Information Window
Project Manager New Features
Properties
Vendor: Actel
Library: SmartGen
Name: FlashMemorySystemBuilder
Design Entry Tools
HDL Editor
SmartDesign
CoreConsole
ViewDraw
Root : Top
> Pre-Synthe
Post-Synthes
Post-Layout
Source Files
Synthesis
Synplify
new (Top.edi
Simulation
ModelSim
Project Flow
Look
Librar
Type
All
Find
Options >>
Reading file 'Rx_sync.vhd'.
Reading file 'Rx_async.vhd'.
Reading file 'Clock_gen.vhd'.
Reading file 'fifo_256x8_fusion.vhd'.
Reading file 'Top.vhd'.
The MP7_Tutorial project was opened.
Log Window
All
Errors
Warnings
Info
Find 1
Ready
VHDL
FAM: Fusion
DIE: M7AFS600
PKG: 484 FBGA

You will see:



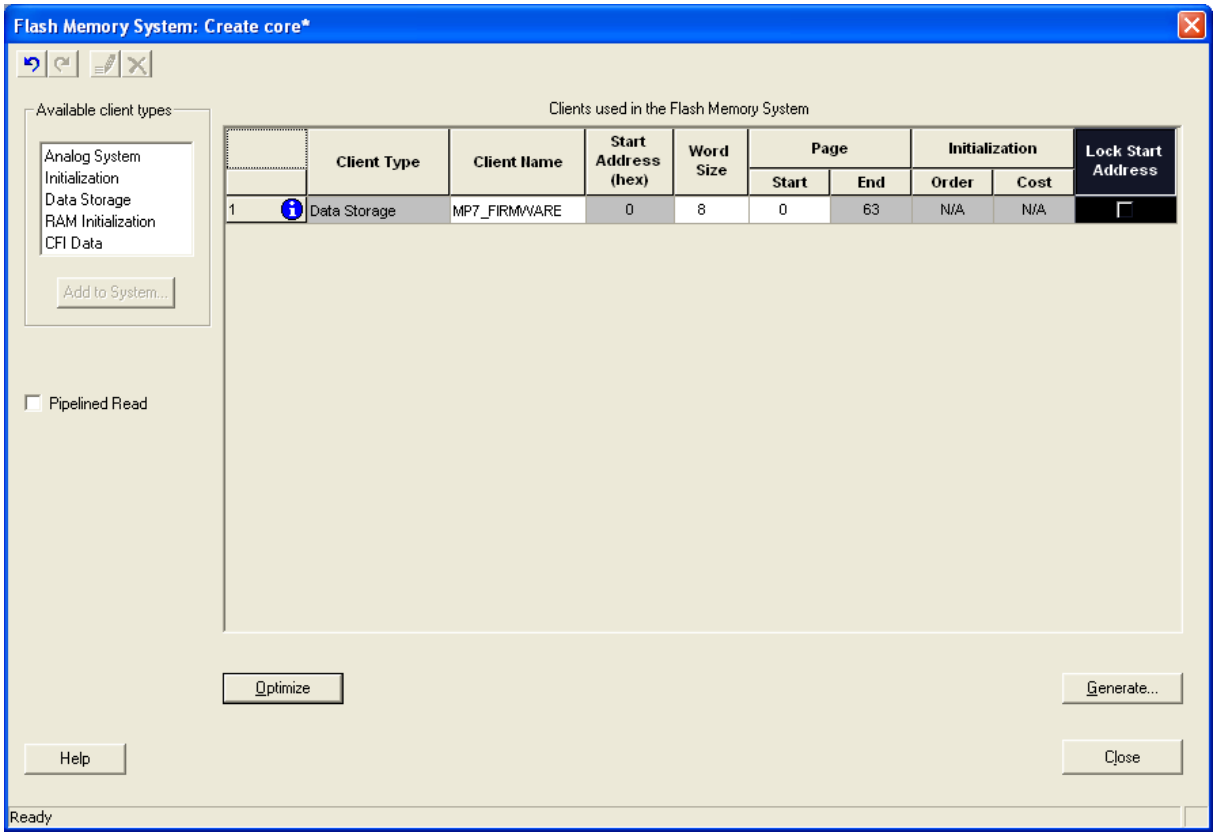
Click on the Data Storage

and full the dialogue window as follows:

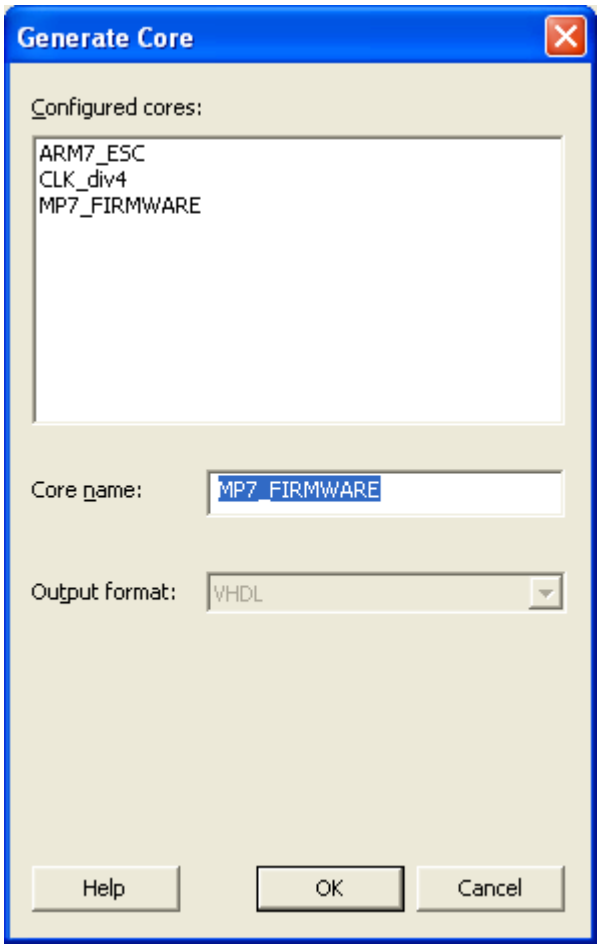


Into Memory content file browse the .hex file, which was generated by KEIL.

Click OK.



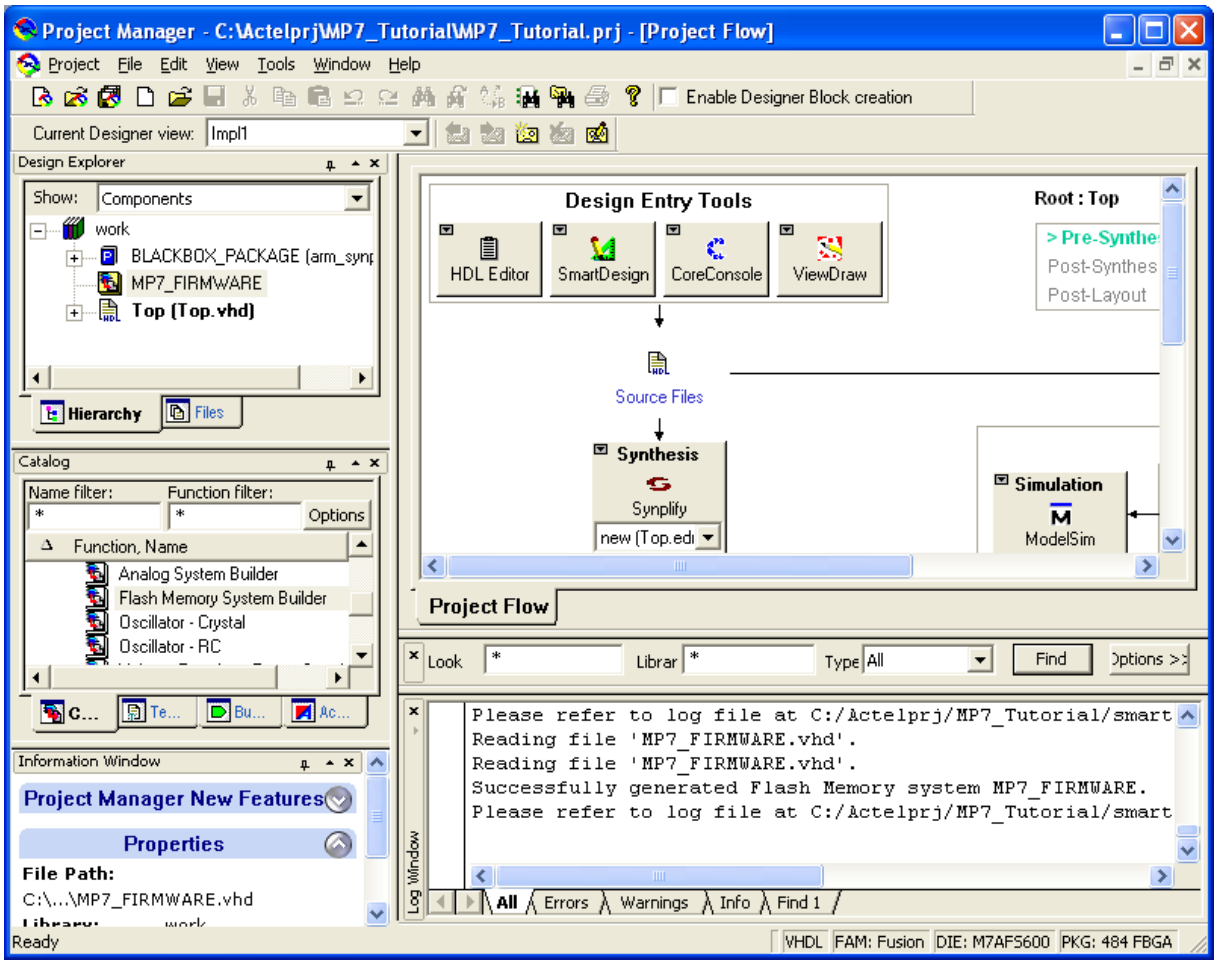
Click genrate.



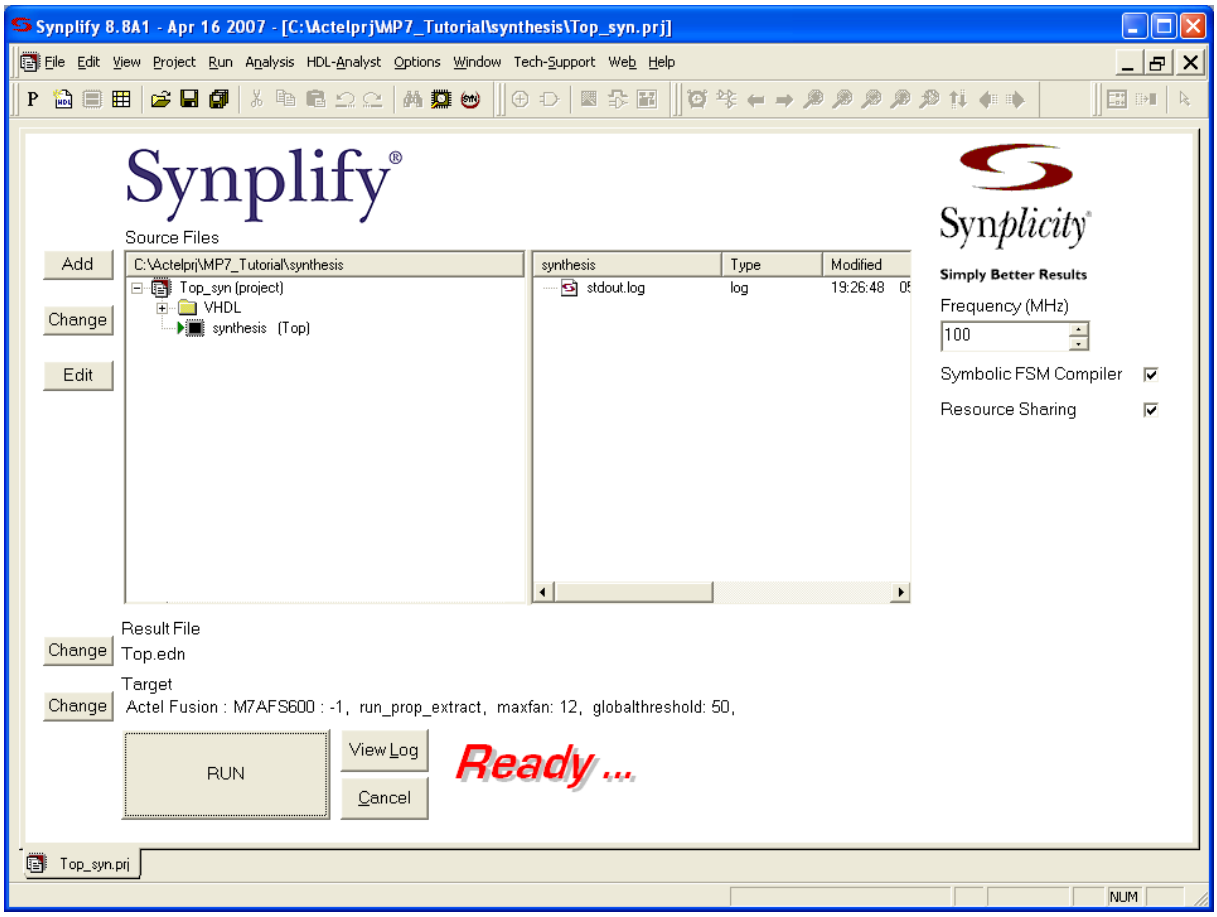
Write Core name and click OK

Click Close

Now you can see in LIBERO Design Explorer MP7_FIRMWARE component:



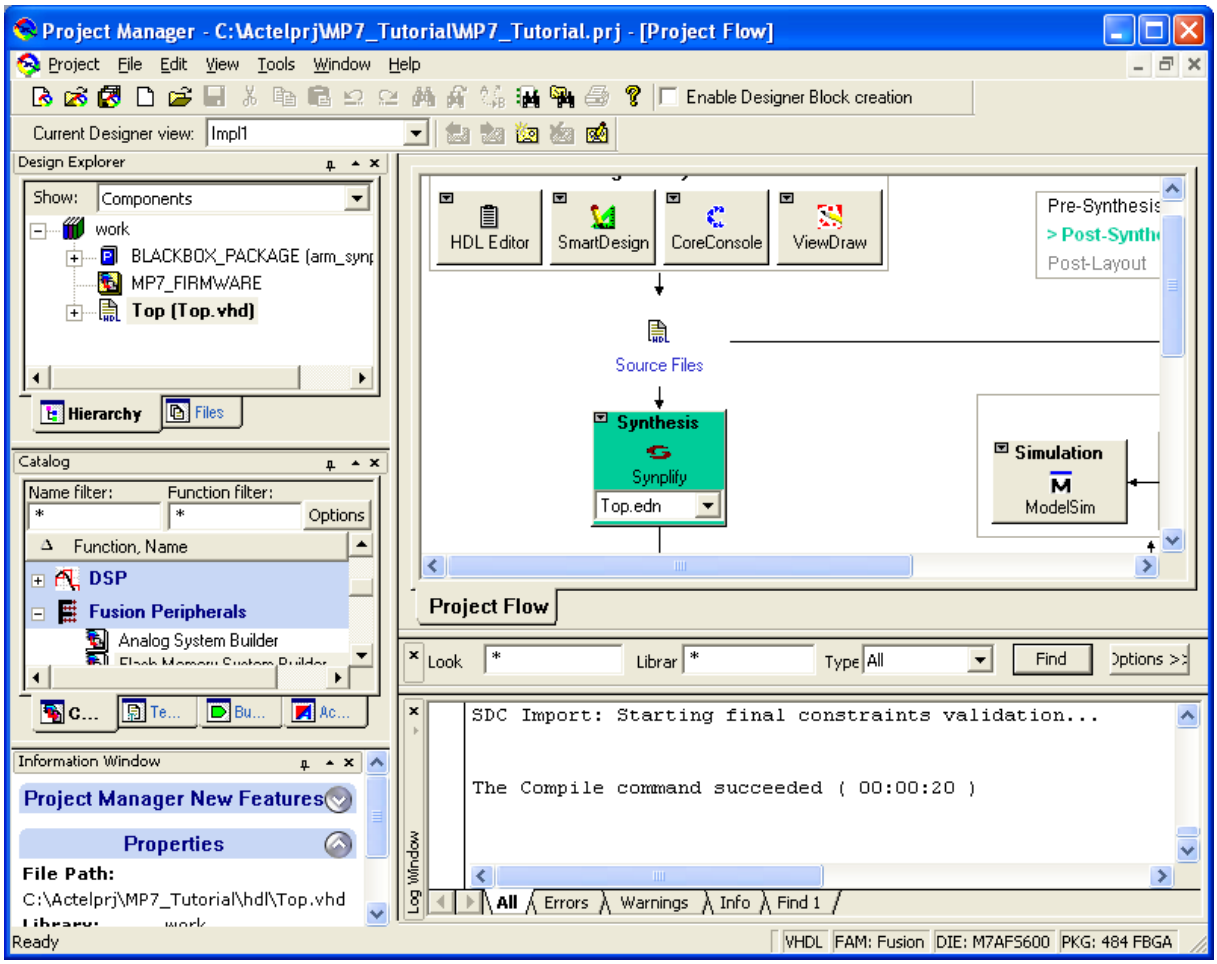
Click on the Synthesis button.
The Synplify starts:



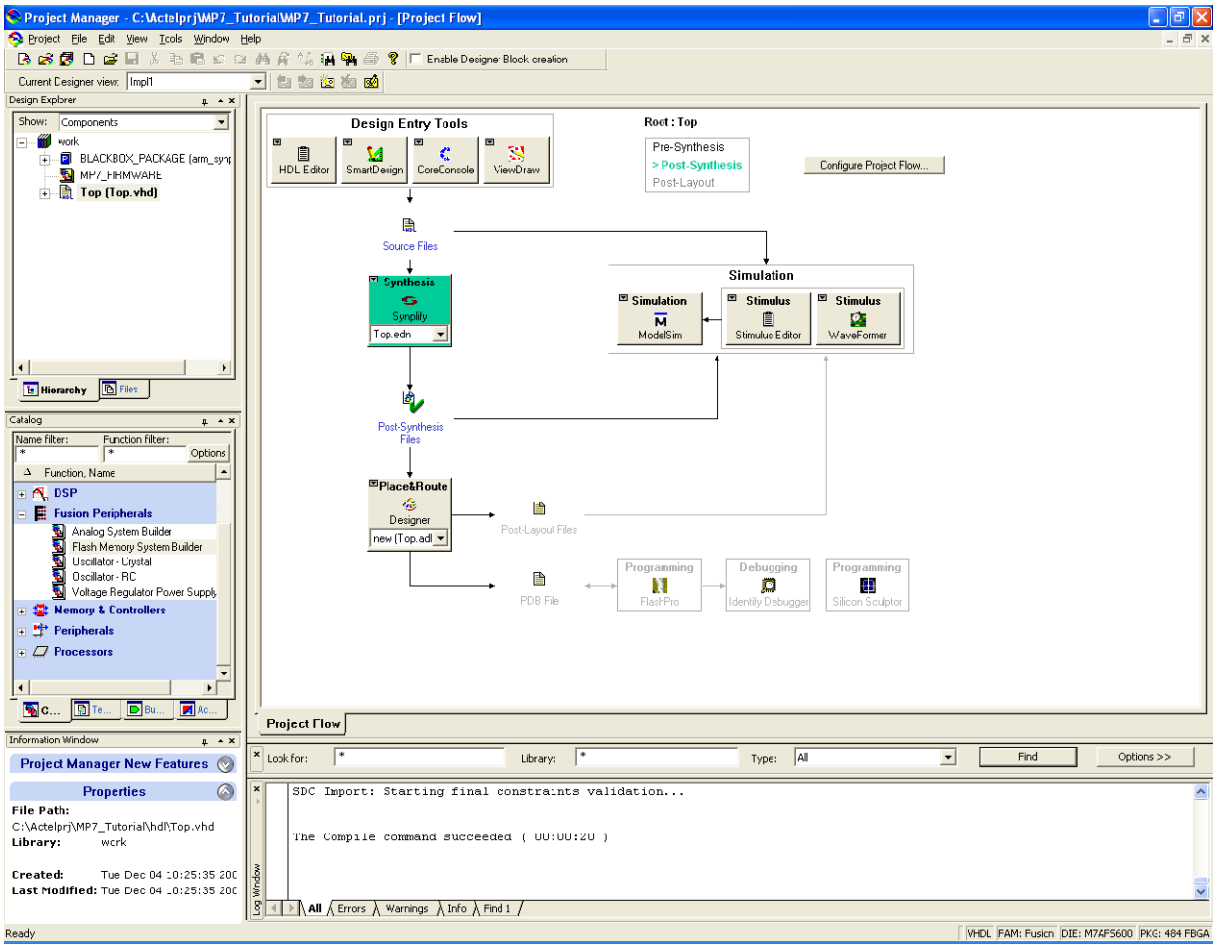
Click RUN button to synthetize your design

After synthetize the should not be any errors. Some warnings are OK...

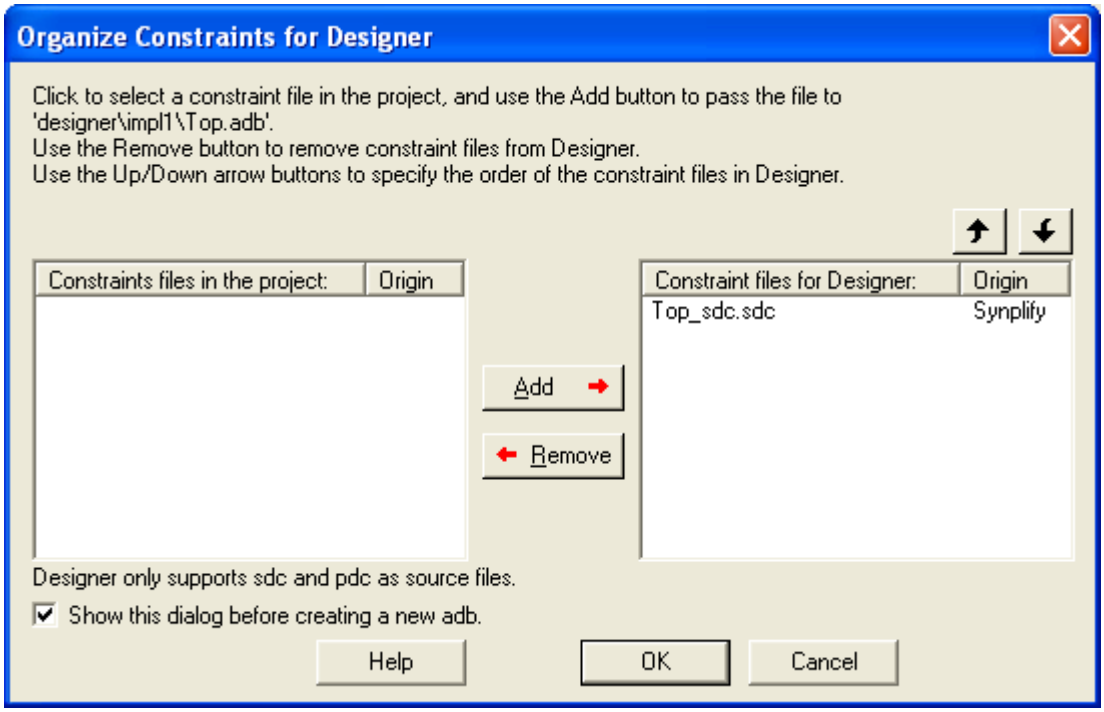
Synthetize icon in Libero should be green:



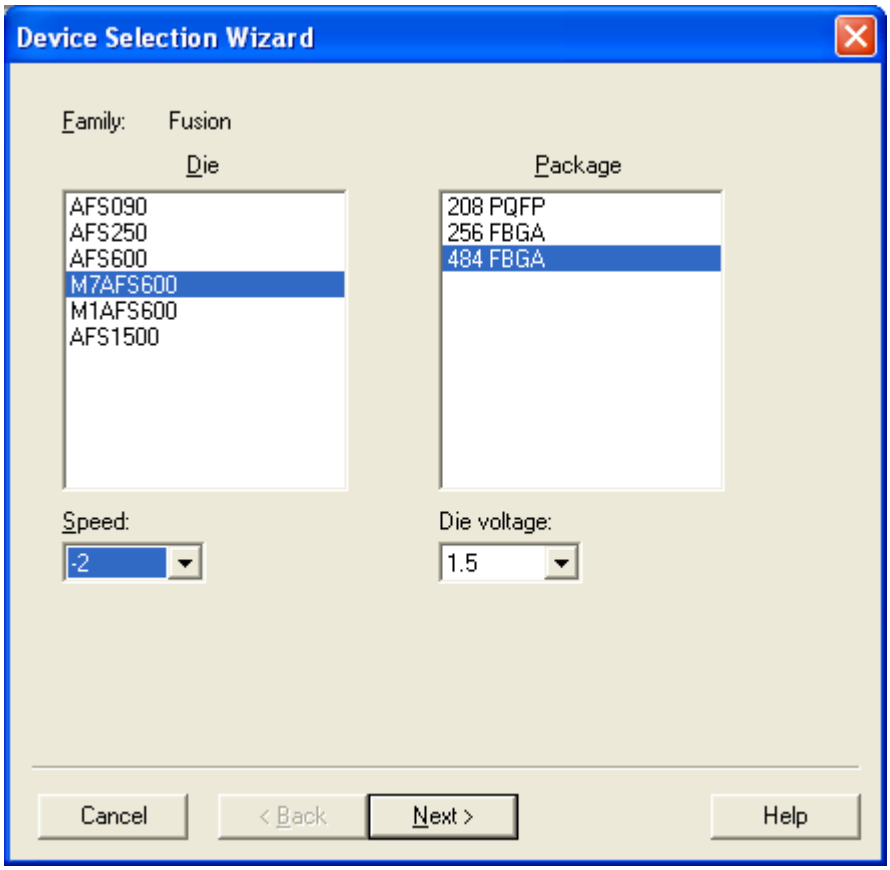
Now continue with Place and route. Click on this icon:

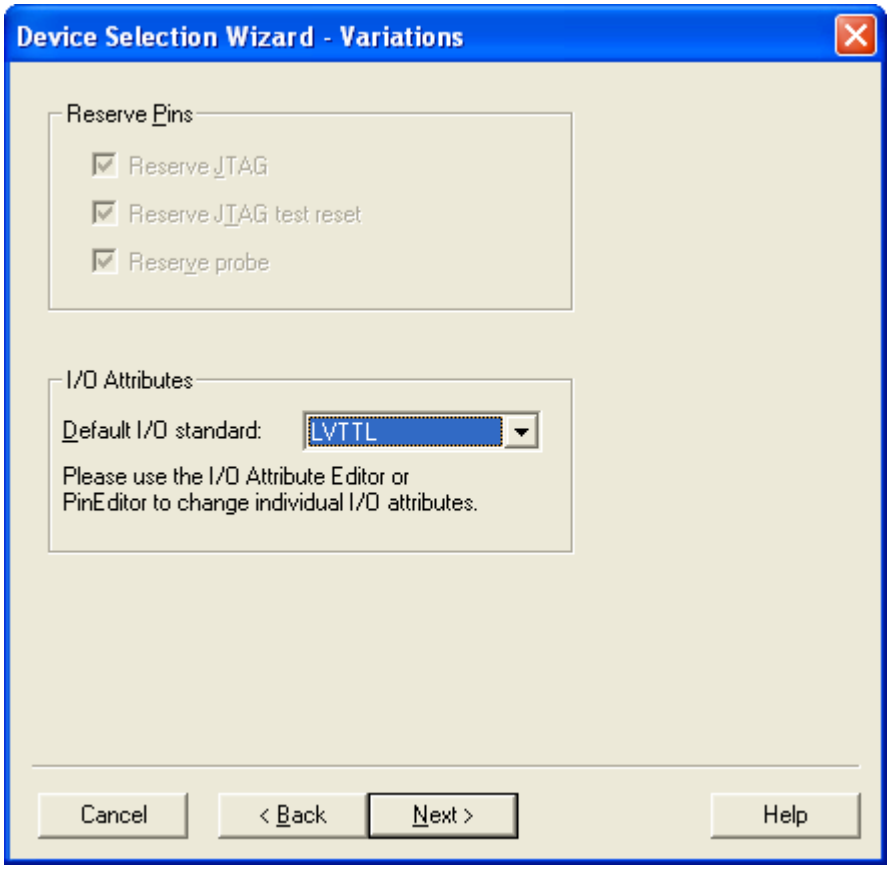


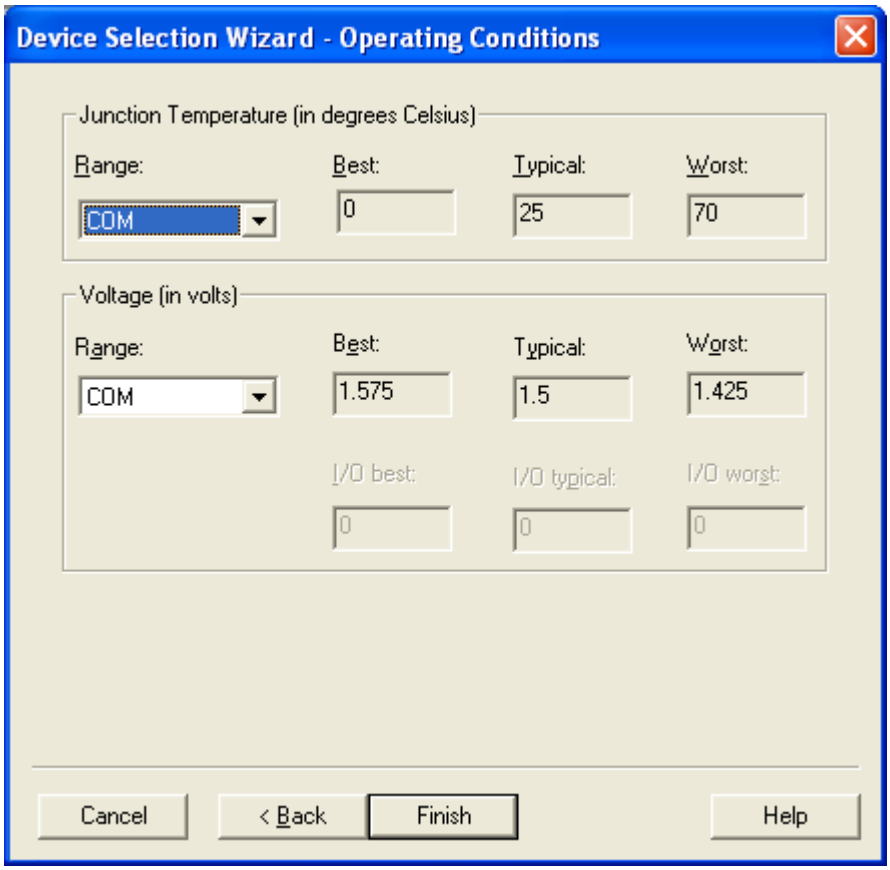
Make sure that you will see:



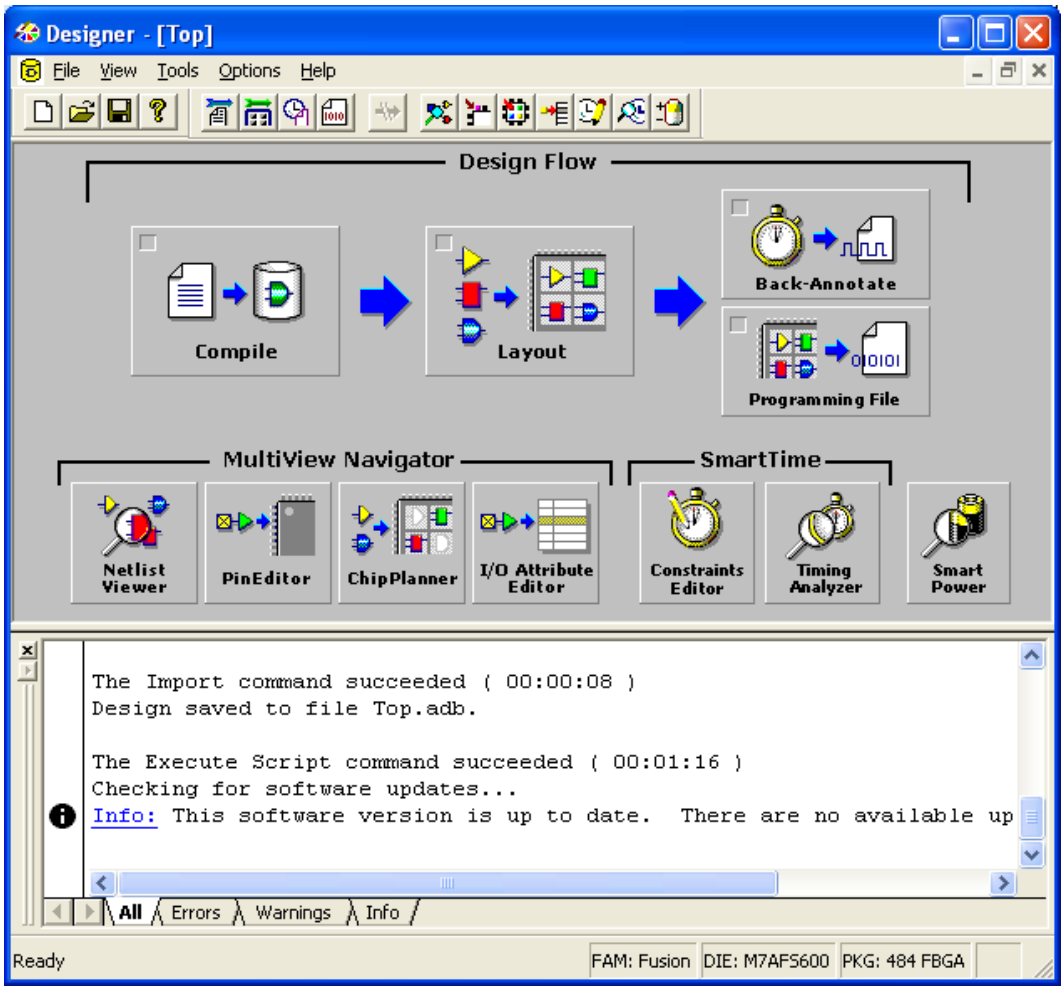
Desinger starts...
Proceed with these settings:


Device Selection Wizard
Family: Fusion
Die
AFS090
AFS250
AFS600
M7AFS600
M1AFS600
AFS1500
Package
208 PQFP
256 FBGA
484 FBGA
Speed:
-2
Die voltage:
1.5
Cancel
< Back
Next >
Help

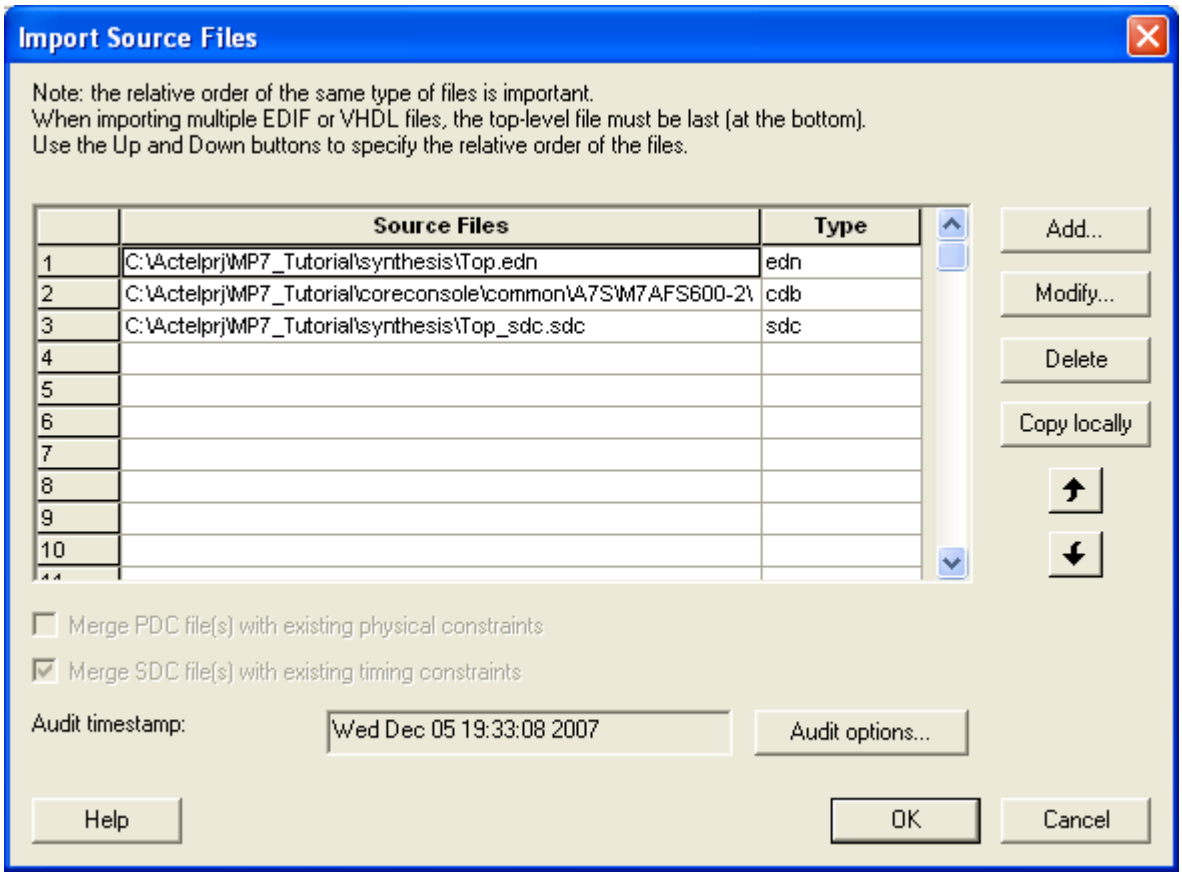

Device Selection Wizard - Variations
Reserve Pins
Reserve JTAG
Reserve JTAG test reset
Reserve probe
I/O Attributes
Default I/O standard:
LVTTL
Please use the I/O Attribute Editor or
PinEditor to change individual I/O attributes.
Cancel
< Back
Next >
Help


Device Selection Wizard - Operating Conditions
Junction Temperature (in degrees Celsius)
Range:
COM
Best:
0
Typical:
25
Worst:
70
Voltage (in volts)
Range:
COM
Best:
1.575
Typical:
1.5
Worst:
1.425
I/O best:
0
I/O typical:
0
I/O worst:
0
Cancel
< Back
Finish
Help

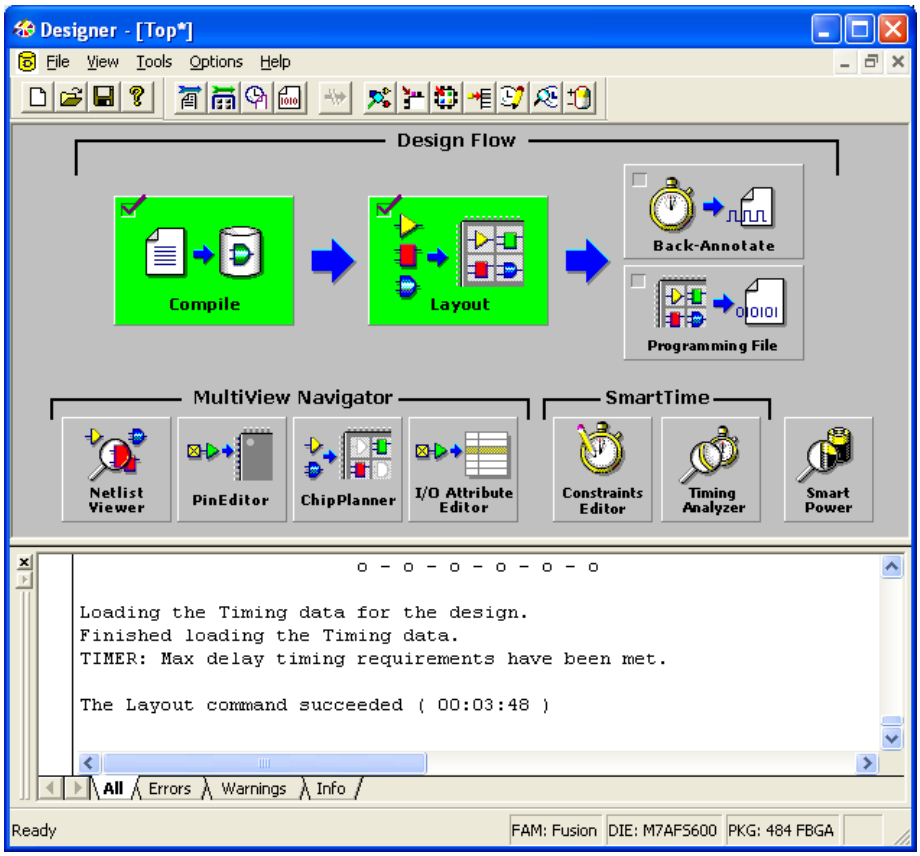
Desinger:



I tis necessary to import pin assigement file. Copy the Top.pdc file into Constraint folder. By File -> Import Source Files:

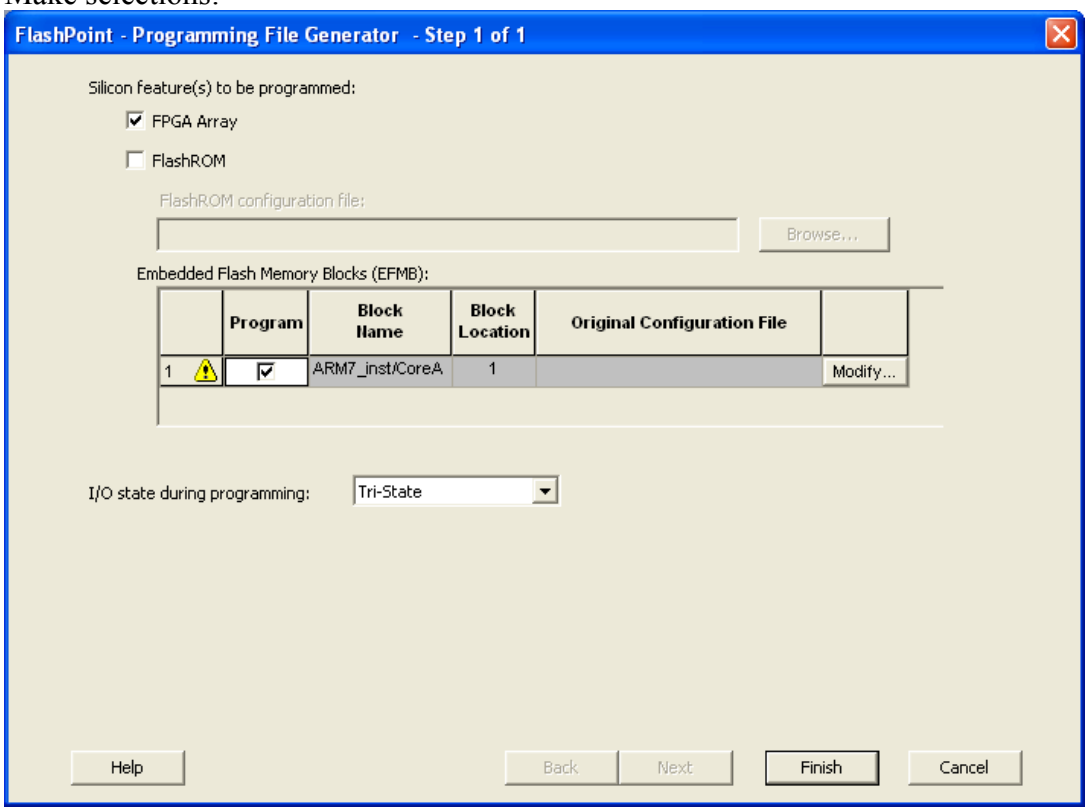


Click Add button and browse the Top.pdc file. Then two times the OK button. Hit Compile button. In dialogue window click OK. If everything is OK, Compile button colors green. Click on layout button and accept the default settings.



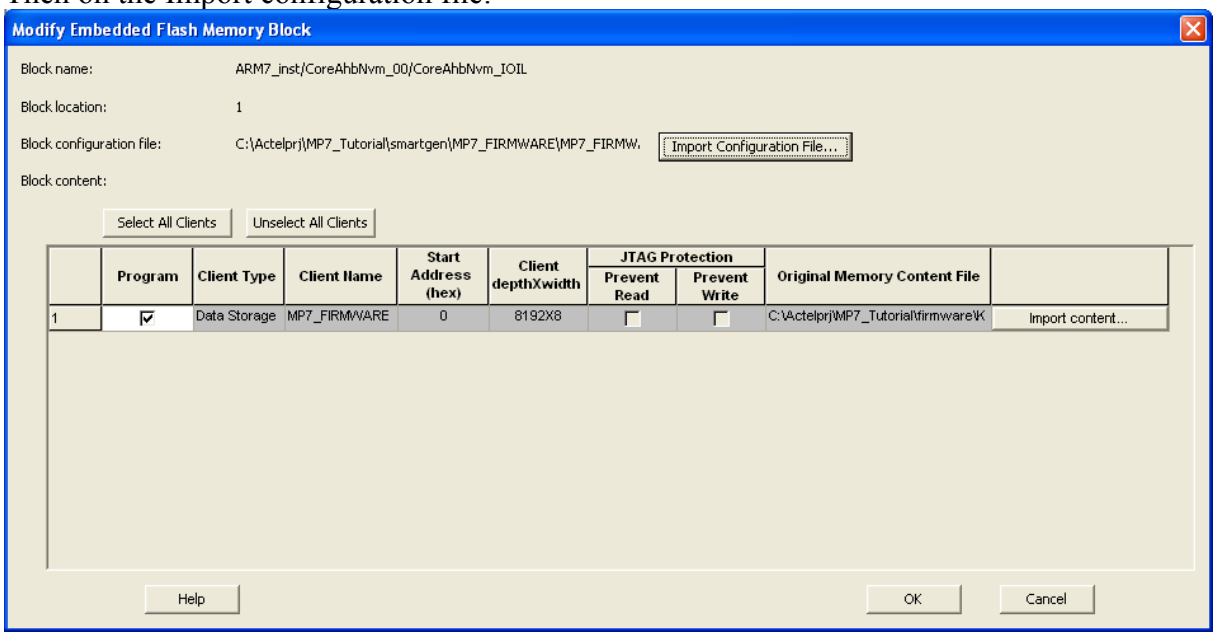
Create programming file with click on Programming File

Make selections:



And click on the modify button

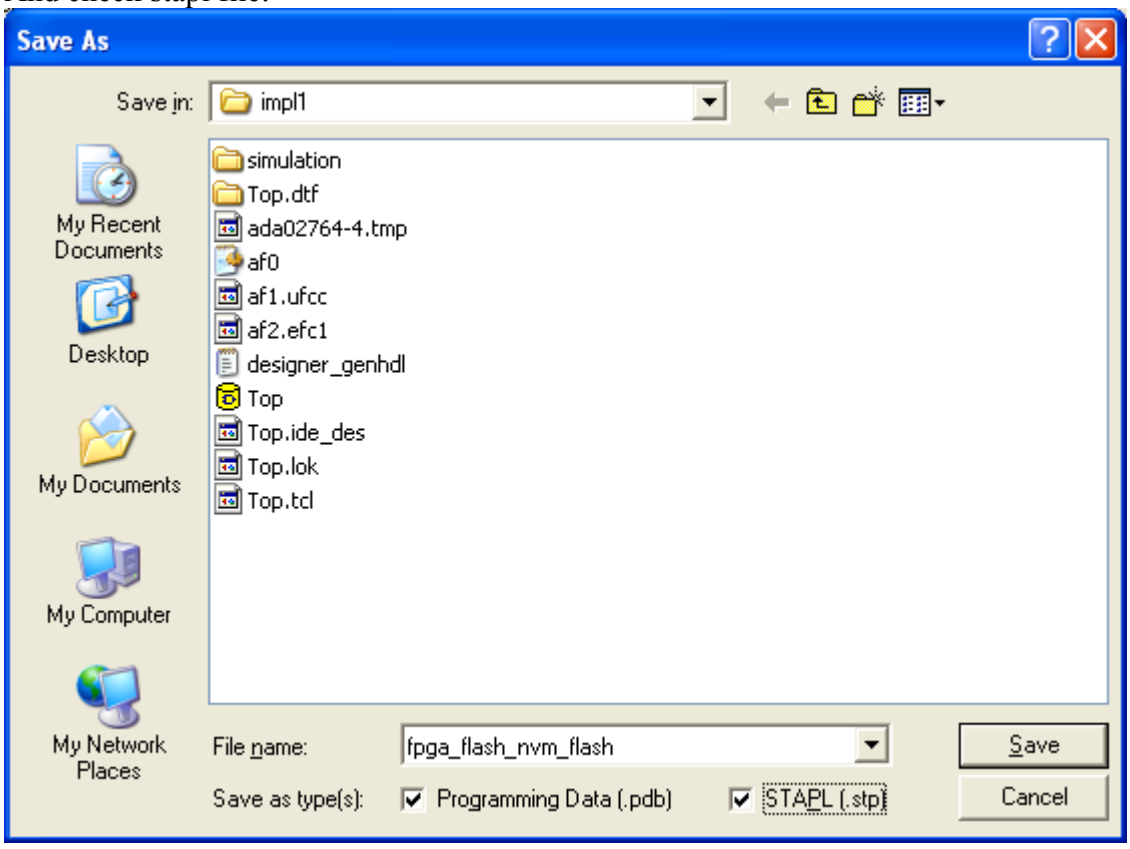
Then on the Import configuration file:



And browse the location ...\project_dir\smartgen\MP7_FIMWARE\MP7_FIRMWARE.efc

Click finish and name flash configuration file as fpga_flash_nvm_flash

And check stapl file:

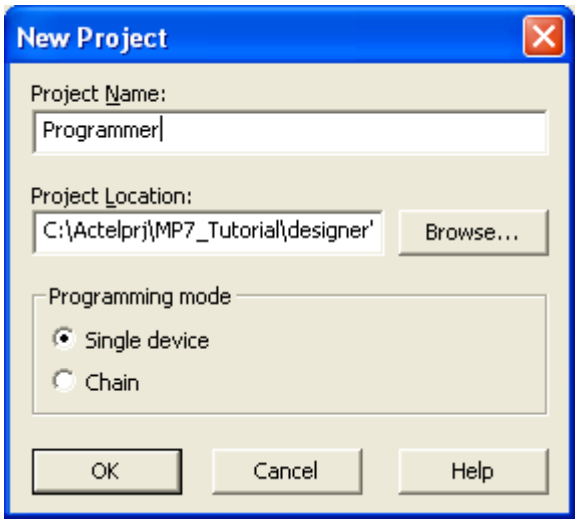


Click Save

In desinger Save projcet and exit it

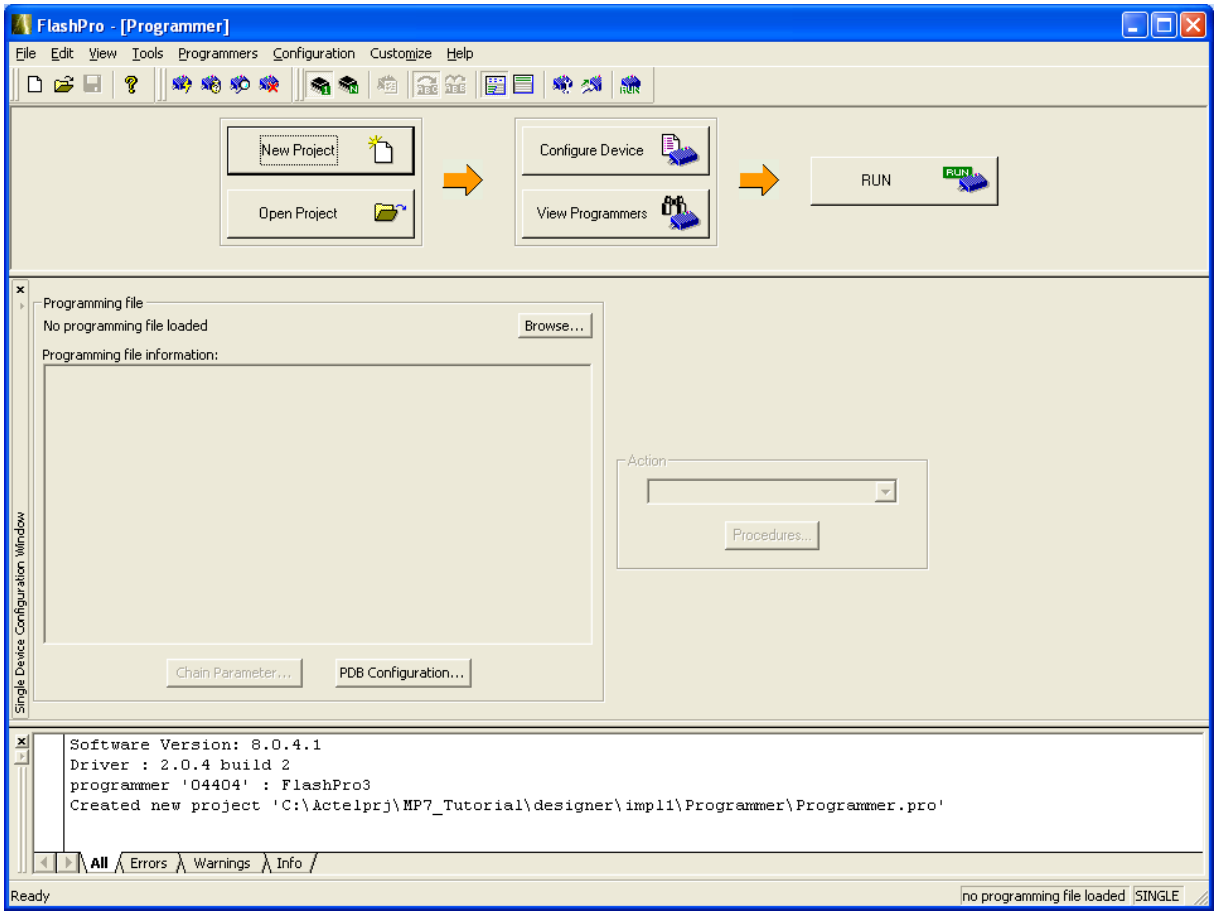
Now, you will download the configuration file into Fusion device. Run the programming with FlashPro

Create new project

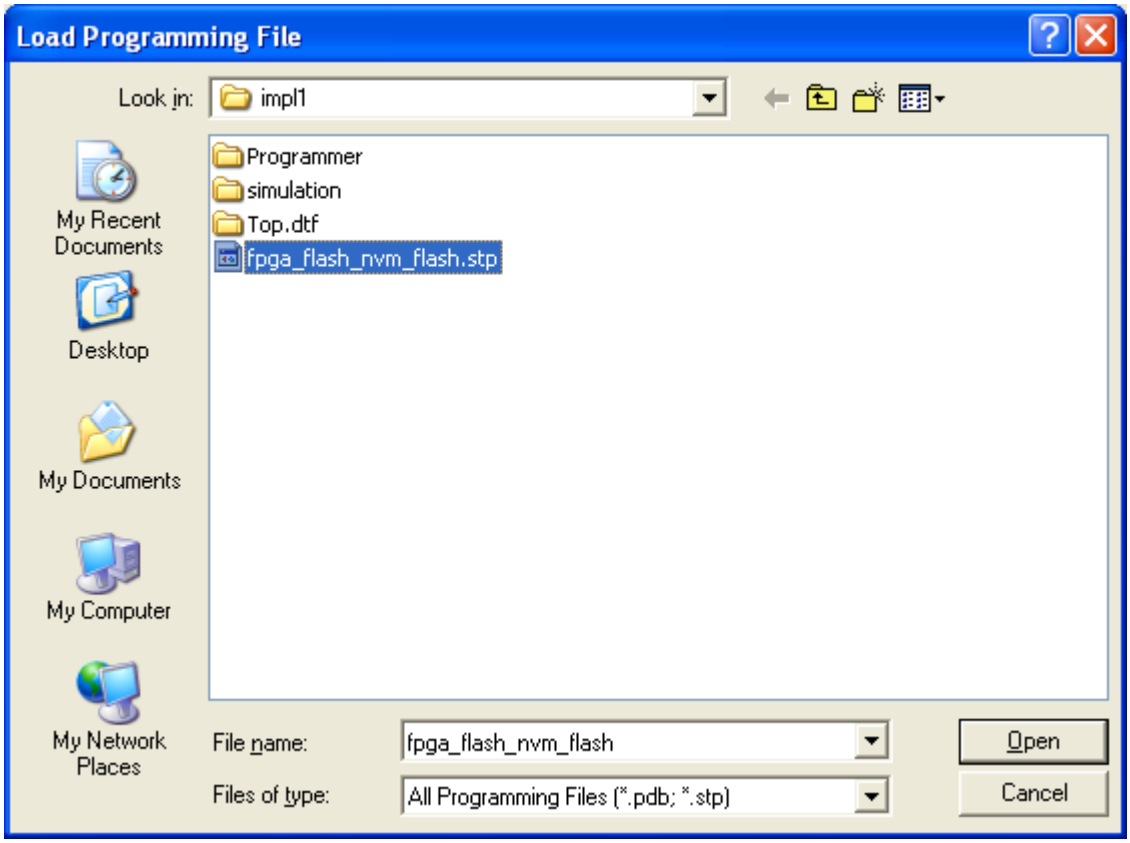


Click OK.

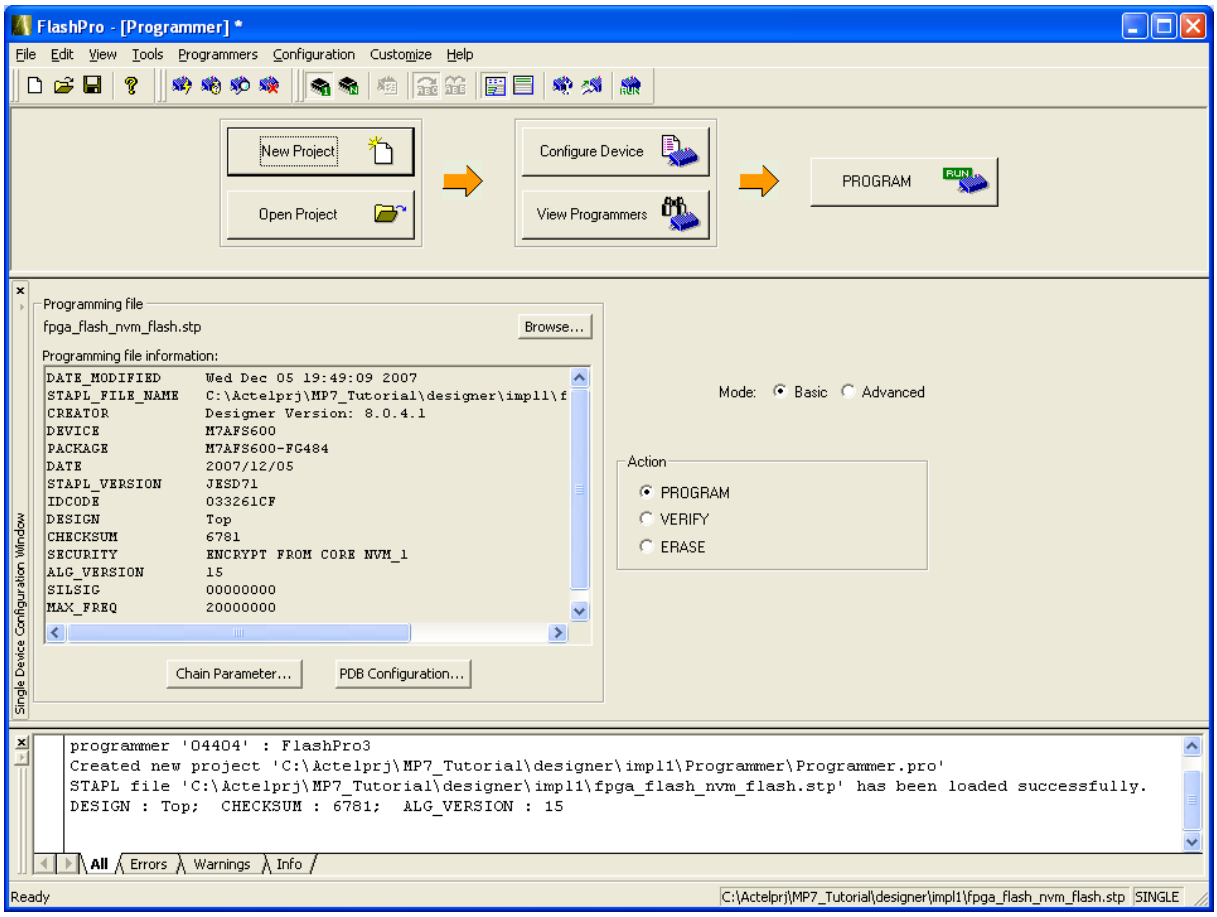
Click Configure device



And Browse programming file:



Click OK.



Click Program.