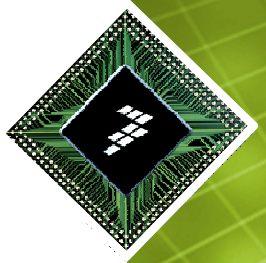
FRDM-KL25Z

Programming and Debugging in CodeWarrior



Agenda

1. What is FRDM-KL25Z?
2. Creating a new Project
3. C/C++ Perspective
4. Debugging a Project
5. Debug Perspective
6. Settings
7. Flashing a Binary File
8. Running Example

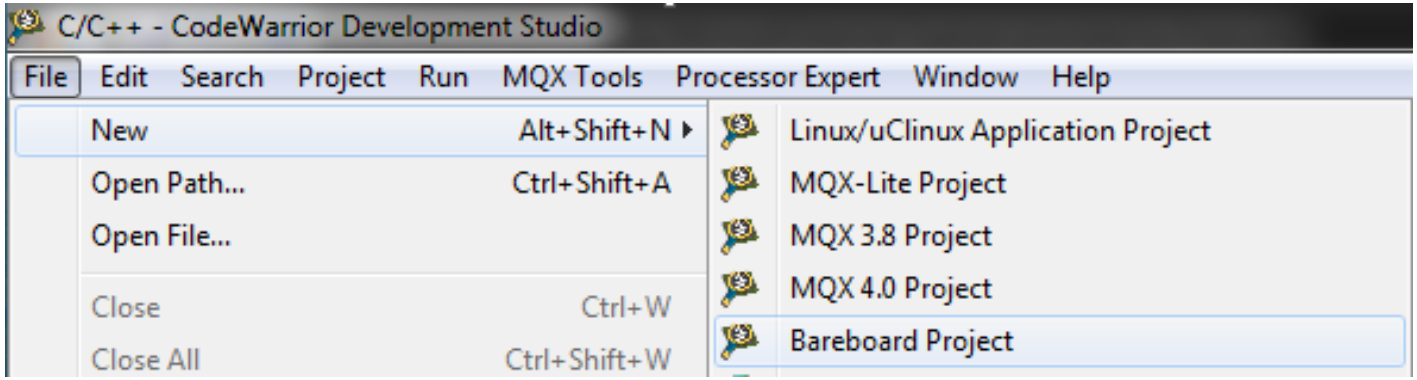
What is FRDM-KL25Z?

- The FRDM-KL25Z is an ultra-low-cost development platform enabled by Kinetis L Series KL1x and KL2x MCUs families built on ARM® Cortex™-M0+ processor.

- Documentation can be found at:
  - www.freescale.com/FRDM-KL25Z
  - FRDMKL25ZUM: FRDM-KL25Z User's Manual

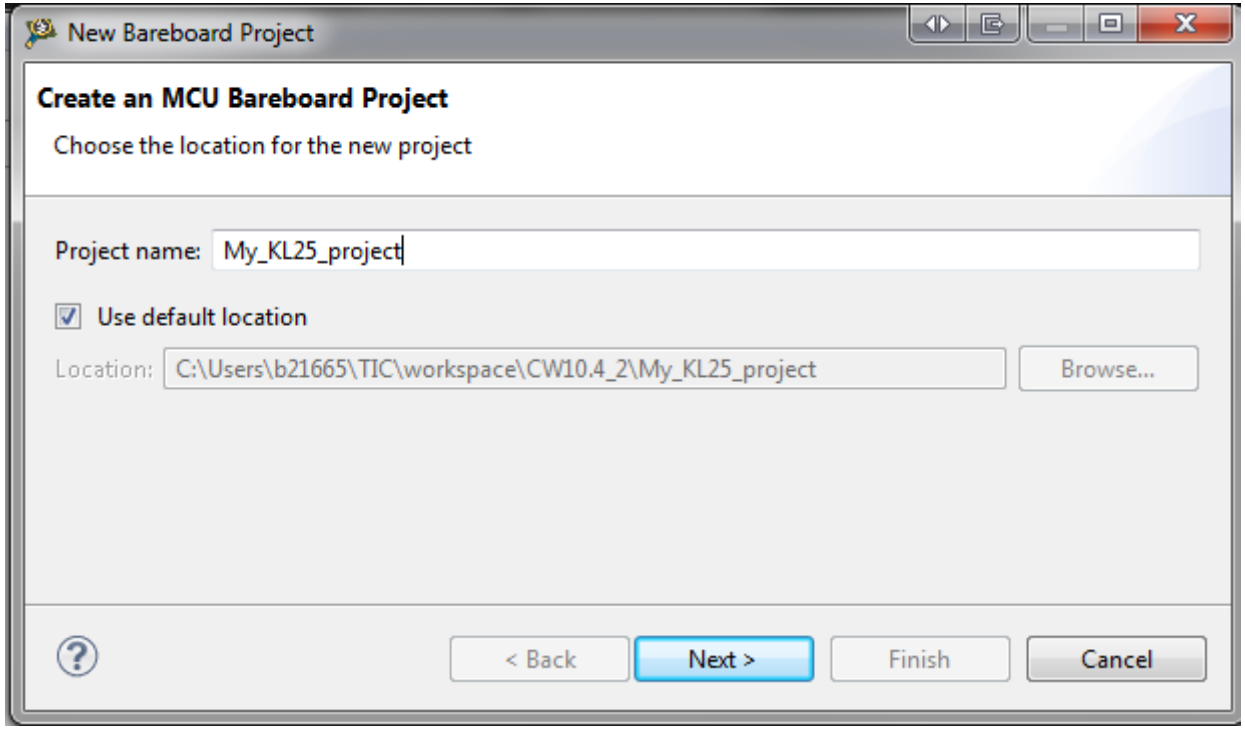
Creating a New Project

- Open CodeWarrior 10.3 or above
- Select menu File > New > Bareboard Project



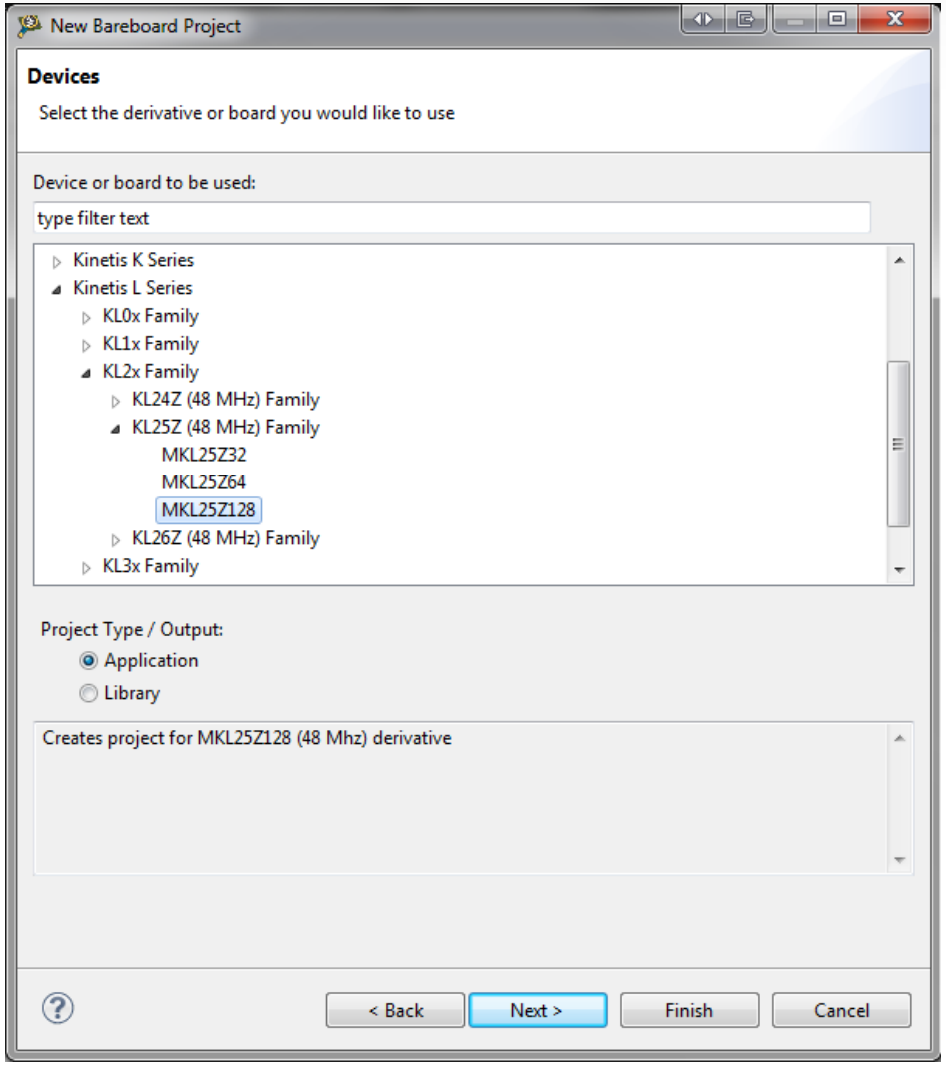
Creating a New Project

- In the next window write a name for your project and click ‘Next >’ button

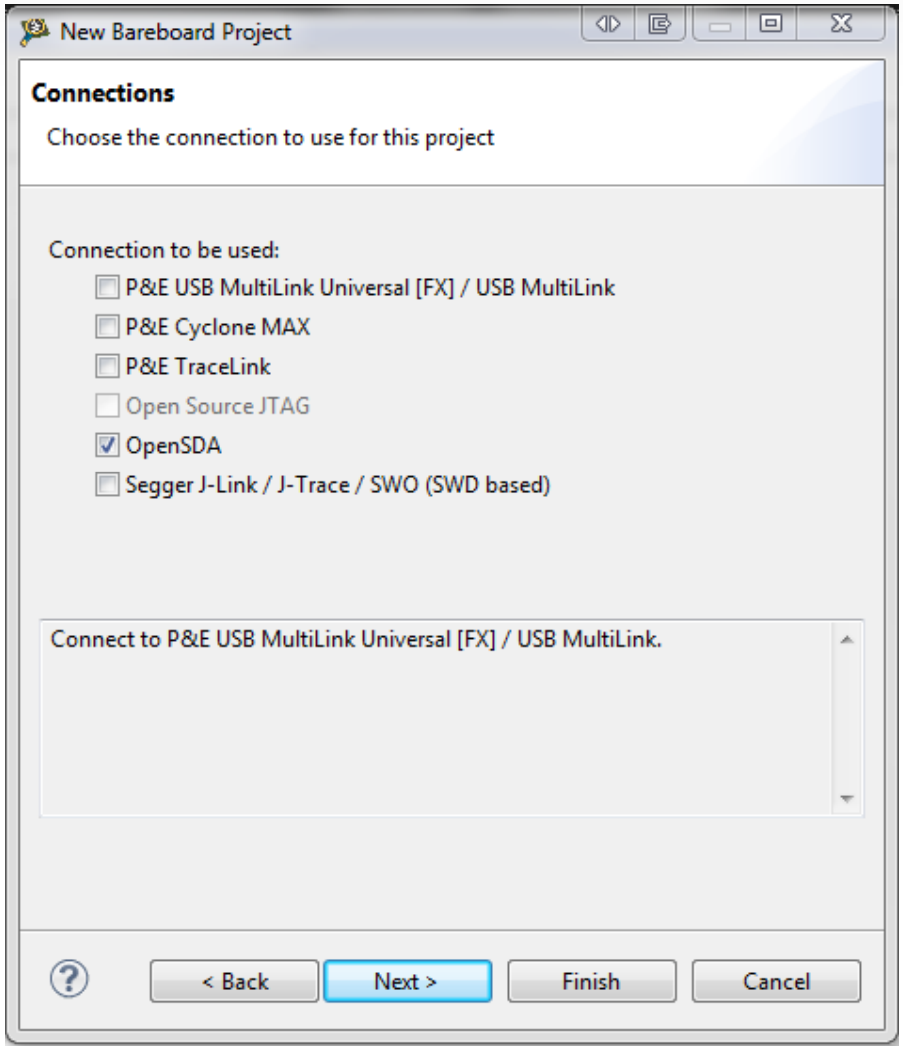


Creating a New Project

- Then select the MCU you will use. In this case MKL25Z128 is used.



Creating a New Project

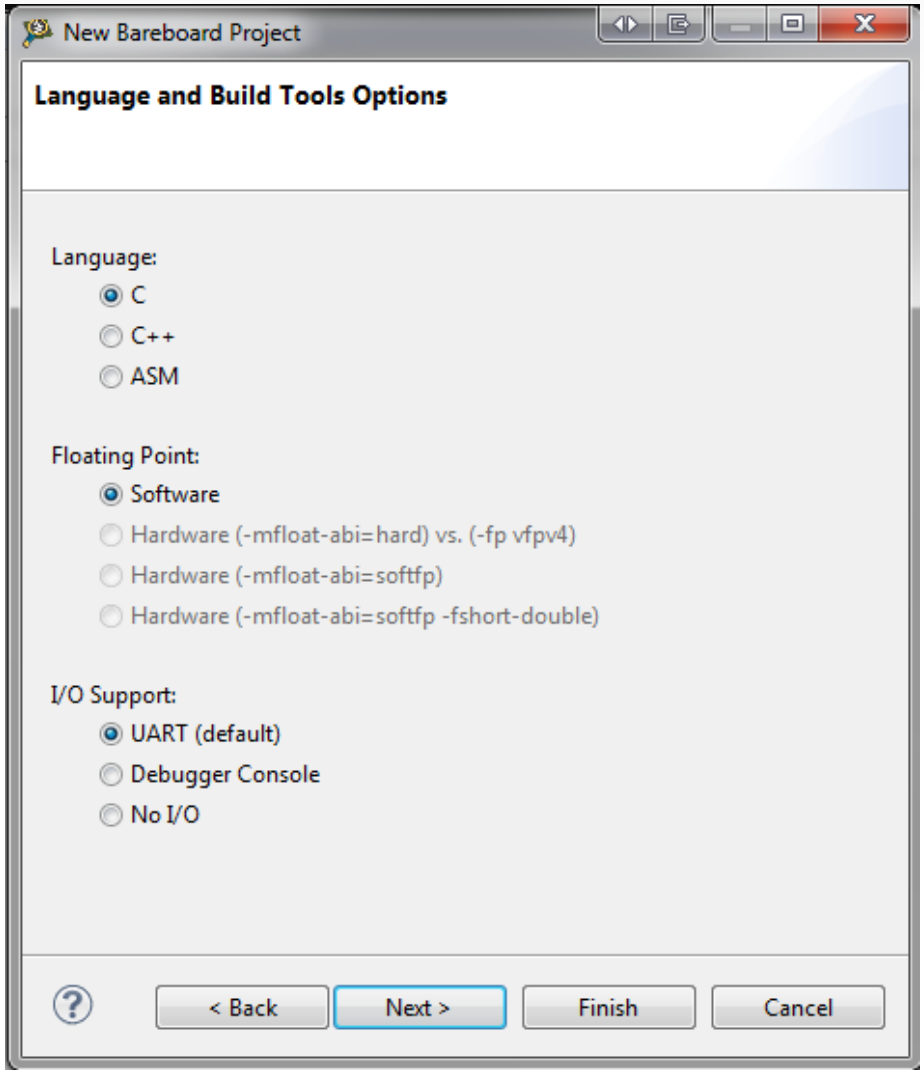


- Select the connection type that you are using, you can select as many connections as you want. In this lab only 'OpenSDA' is used.

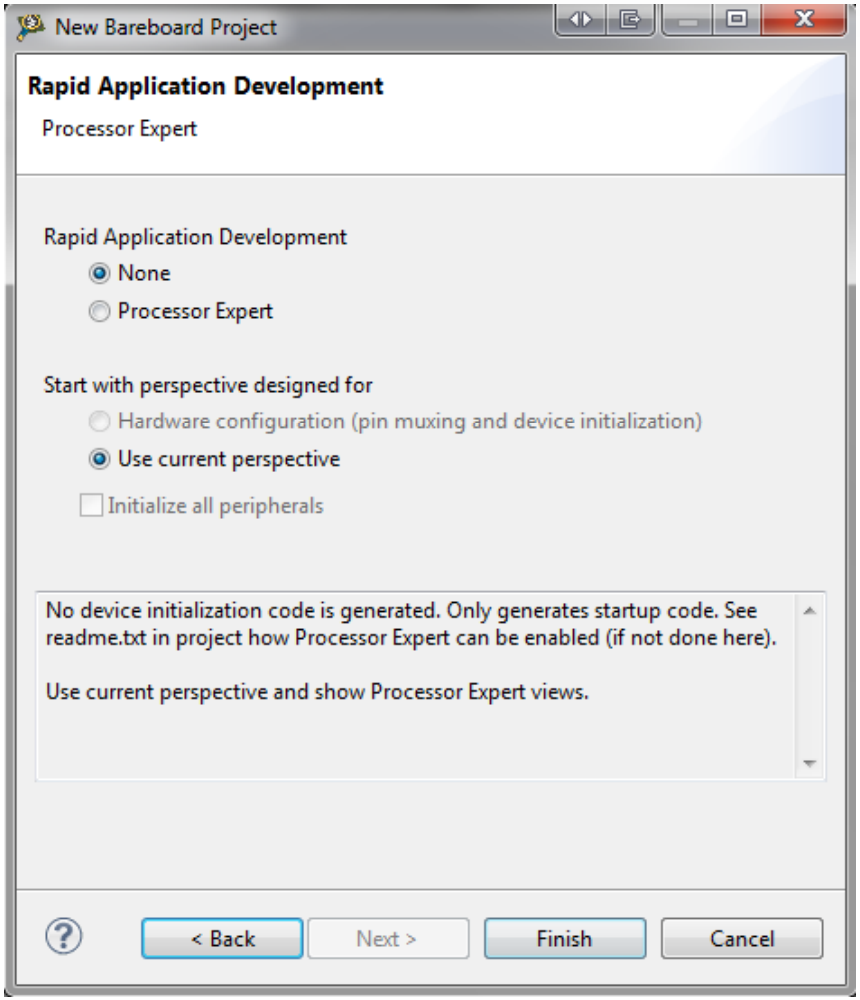
Creating a New Project

- In the next window you can set:

- Language
  - Select C, C++ or ASM according to your requirements. For this lab select C language.
- Floating Point support
  - KL25 does not feature Hardware Floating Point unit. Software is selected automatically.
- I/O Support
  - Codewarrior provides 3 different library options for I/O Support. Select UART for this lab.
  - You can change your library set later in menu Project Properties > C/C++ Build > Settings > Librarian

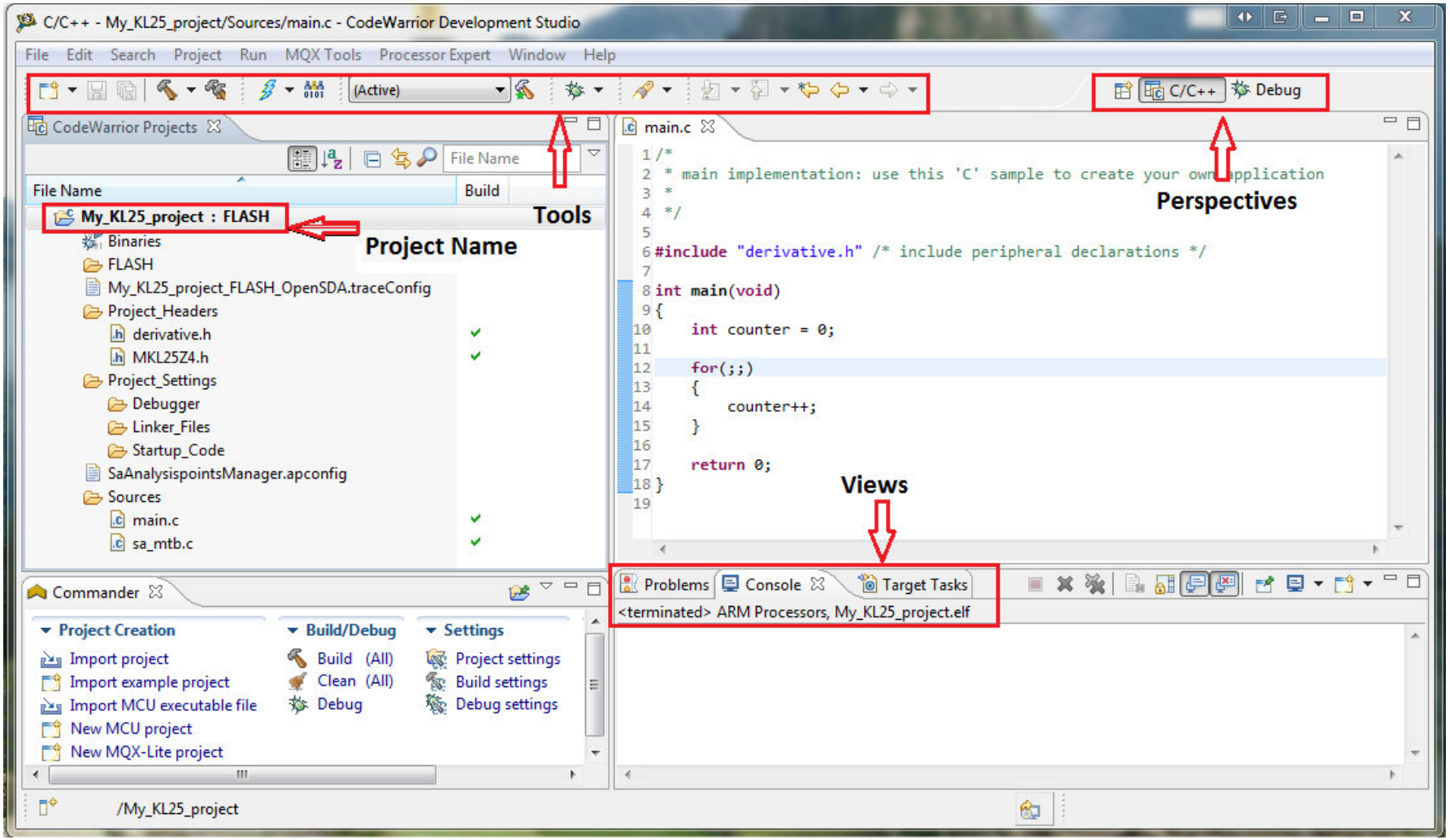


Creating a New Project



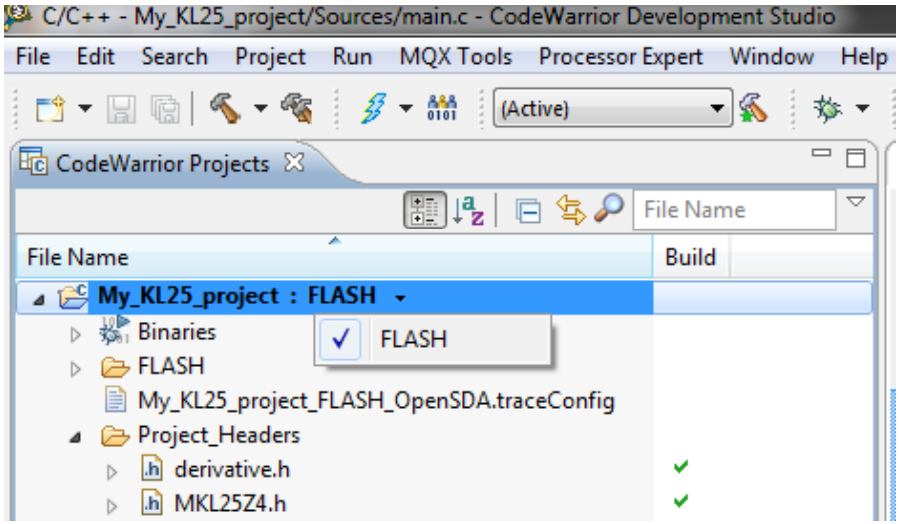
- Rapid Application Development features are not covered in this lab, select 'None' and click 'Finish' Button.

C/C++ Perspective



Debugging a Project

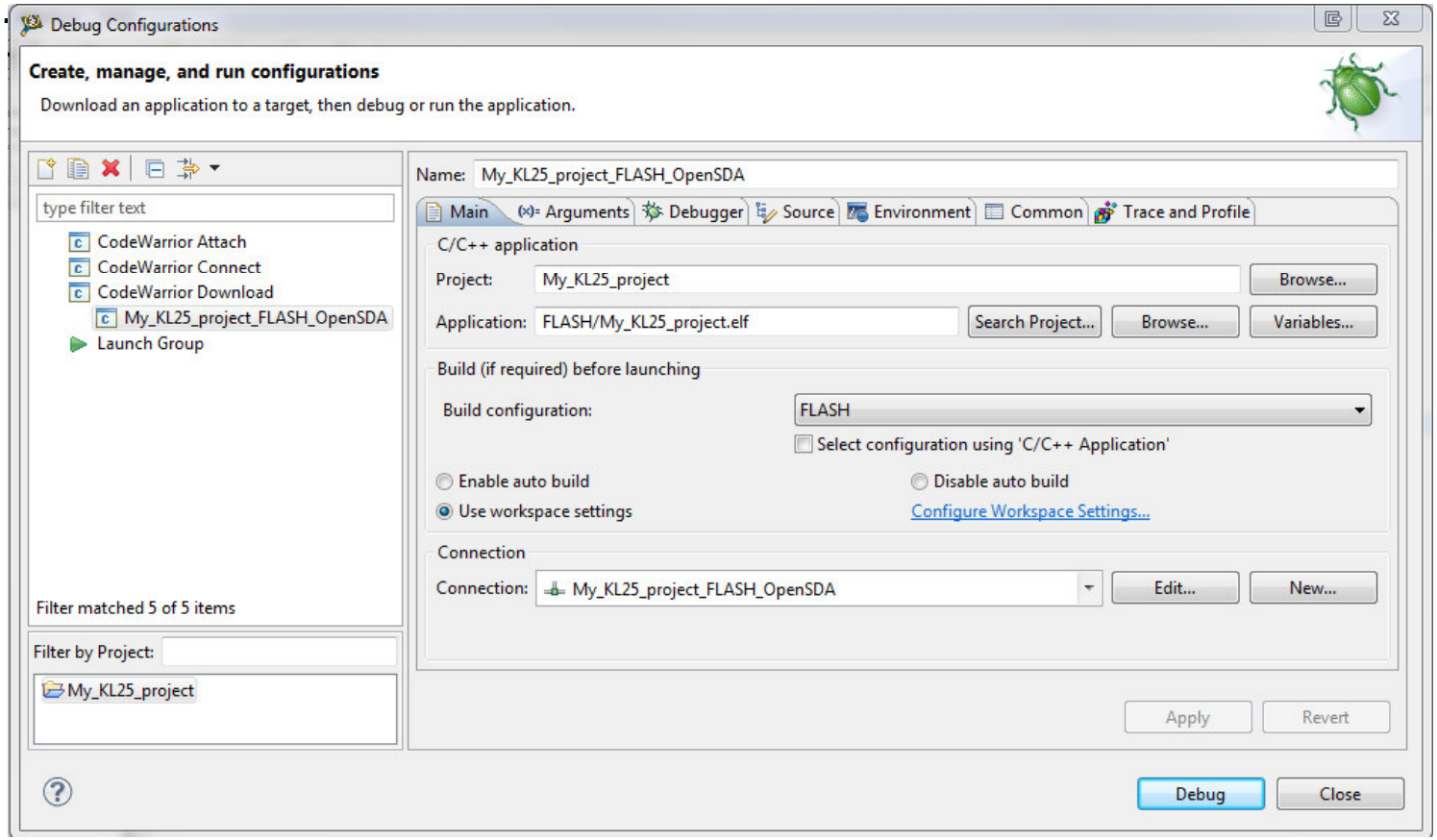
- Once your project is created you must choose a 'Build Configuration'. KL25 project only provide FLASH configuration, but other Freescale devices provide FLASH and RAM targets.



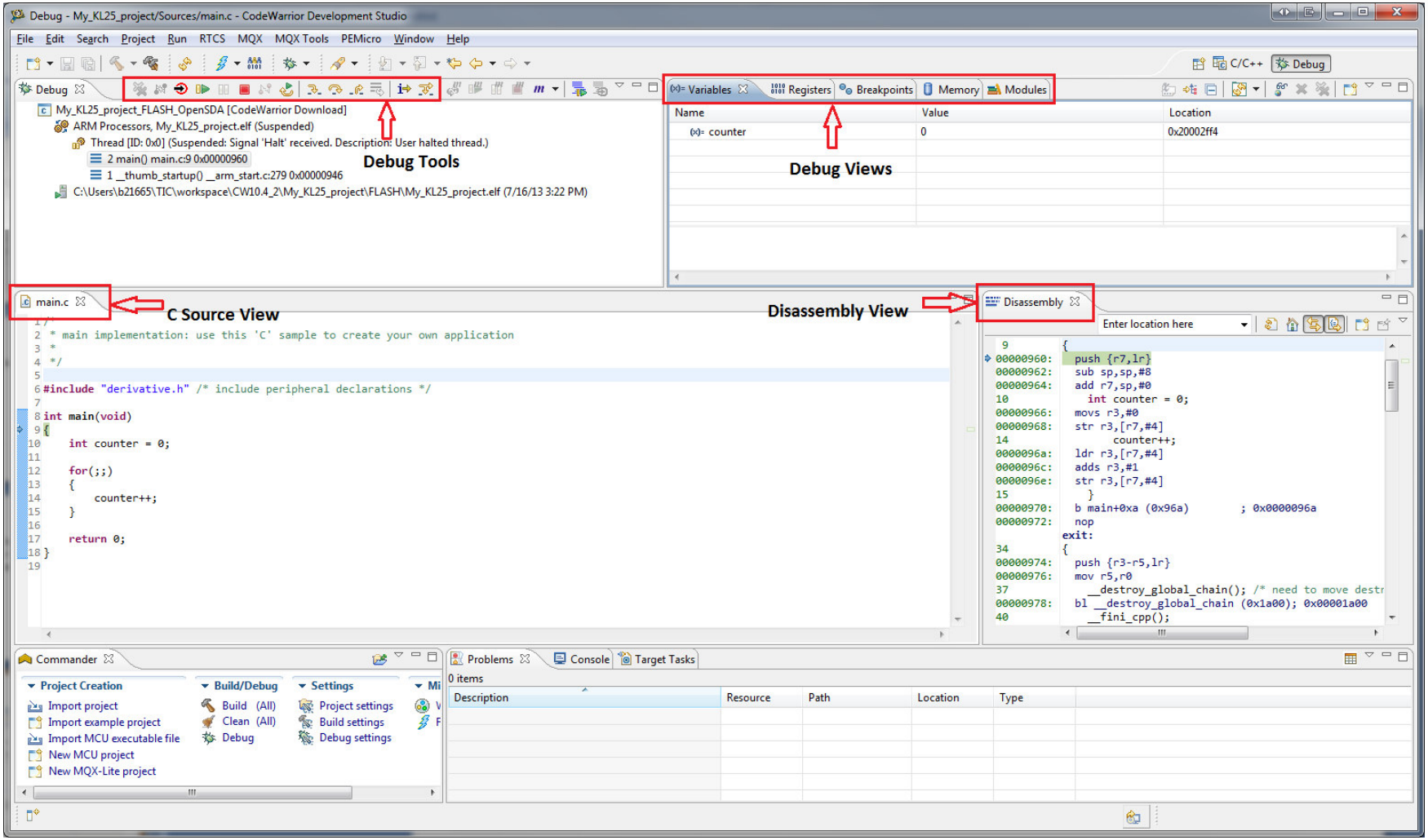
- To build the project, go to menu Project > Build Project, or alternately click the build button 

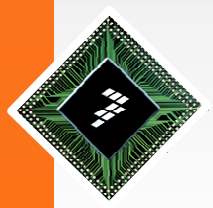
Debugging a Project

- To debug your project, go to menu Run > Debug Configurations… and select the Debug Configuration that matches with your Build Configuration and connection.
- Then click ‘Debug’



Debug Perspective



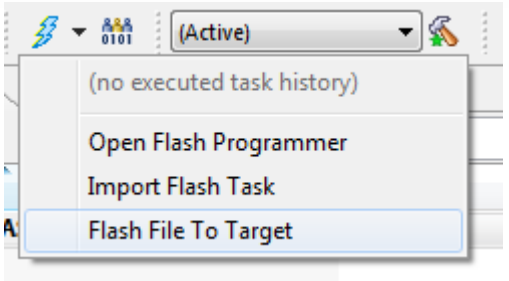


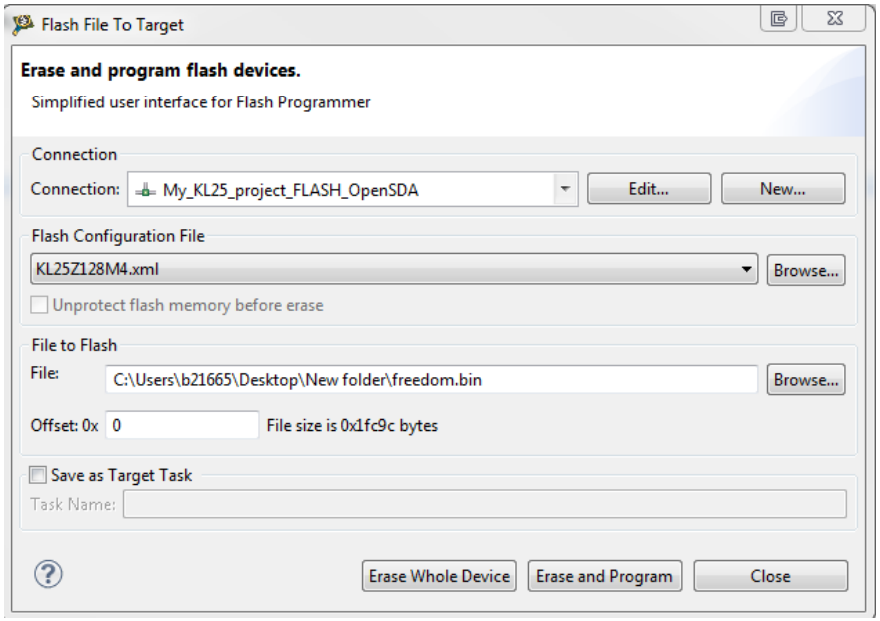
Settings

- To edit your project settings go to menu Project > Properties > C/C++ Build > Settings
- To open a new perspective go to menu Window > Open Perspective > Debug
- You can find all the views and open a new one in menu Window > Show View
- To edit CodeWarrior settings and prefernces go to menu Window > Preferences

Flashing a Binary File

- Click the drop down arrow besides the Flash programmer button to display the context menu and select ‘Flash File to Target’.

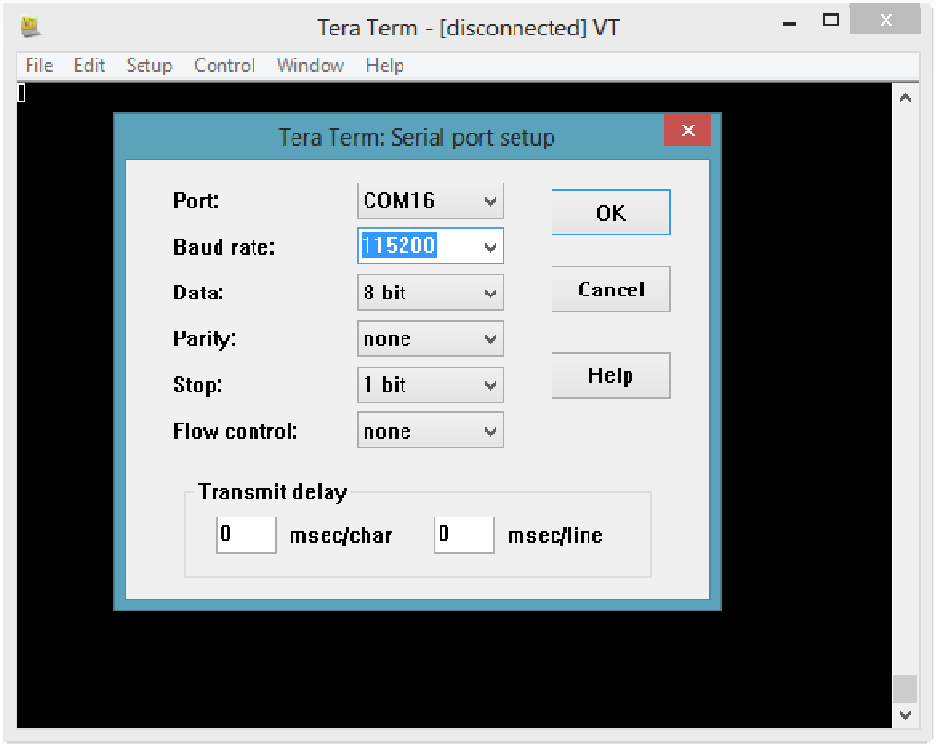




- Select your connection.
- In file to Flash search for freedom.bin
- Click ‘Erase and Program’.

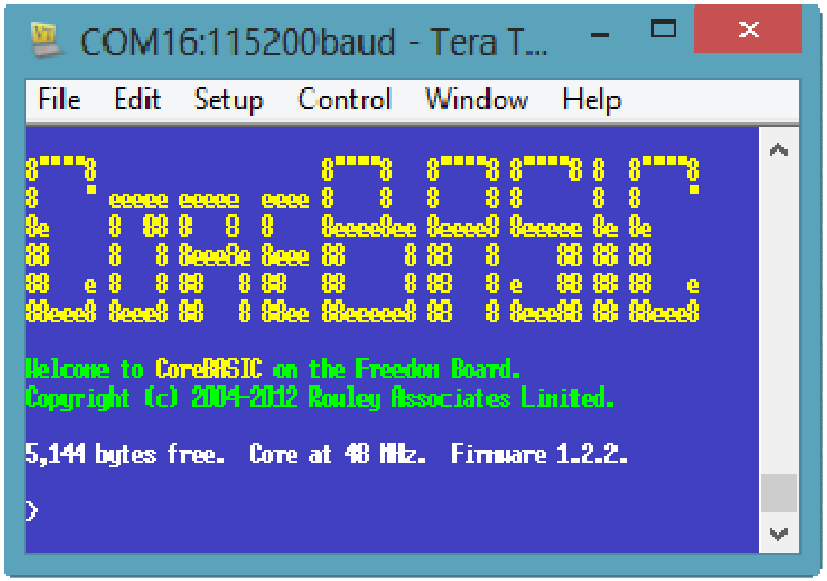
Running Example

- Open a terminal program. A good choice in my view is TeraTerm
- Connect to the board with 115200 baud



Run Example

- Push the onboard reset button. You will see next screen in terminal.



Running Example

- Below you can find some instructions

```
print "Hello World!“
core.led = %red
core.led = %blue
core.led = %green
core.led = %black
```

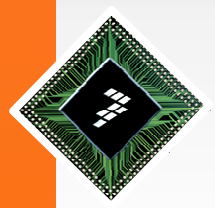
Other colors are accepted too (see below or the CoreBASIC manual).

Running Example

- A LED blink loop. You can copy and paste the following code:

```
10 for i = 1 to 5
20 print i
30 core.led = %magenta
40 pause 0.5
50 core.led = %black
60 pause 0.5
70 next I
```

- The command to run a program is: of course *run*



Running Example

- Please visit the link below to see the whole example post.
- http://mcuoneclipse.com/2013/01/18/back-to-basics-with-the-freedom-board/

freescale™


freescale™
semiconductor