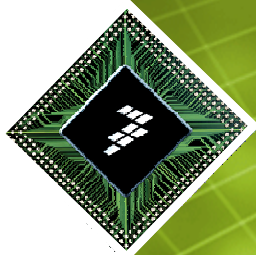


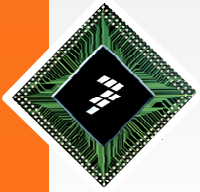


FRDM-KL25Z

Programming and Debugging in CodeWarrior

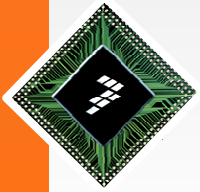


Freescale, the Freescale logo, Altivec, C-5, CodeTEST, CodeWarrior, ColdFire, C-Ware, the Energy Efficient Solutions logo, mobileGT, PowerQUICC, QorIQ, StarCore and Symphony are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. BeeKit, BeeStack, ColdFire+, CoreNet, Flexis, Kinetis, MXC, Platform in a Package, Processor Expert, QorIQ, Qonverge, Qorivva, QUICC Engine, SMARTMOS, TurboLink, VortiQa and Xtrinsic are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © 2011 Freescale Semiconductor, Inc.



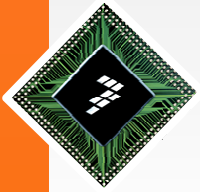
Agenda

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



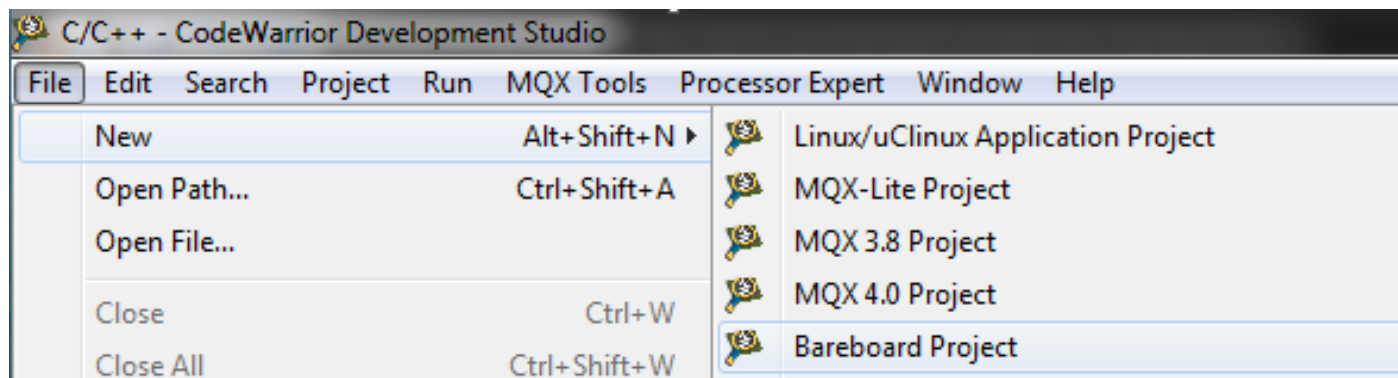
What is FRDM-KL25Z?

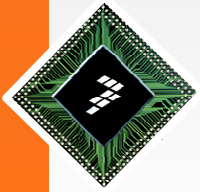
- The FRDM-KL25Z is an ultra-low-cost development platform enabled by Kinetis L Series KL1x and KL2x MCUs families built on ARM® Cortex™-M0+ processor.
- Documentation can be found at:
 - www.freescale.com/FRDM-KL25Z
 - FRDMKL25ZUM: FRDM-KL25Z User's Manual



Creating a New Project

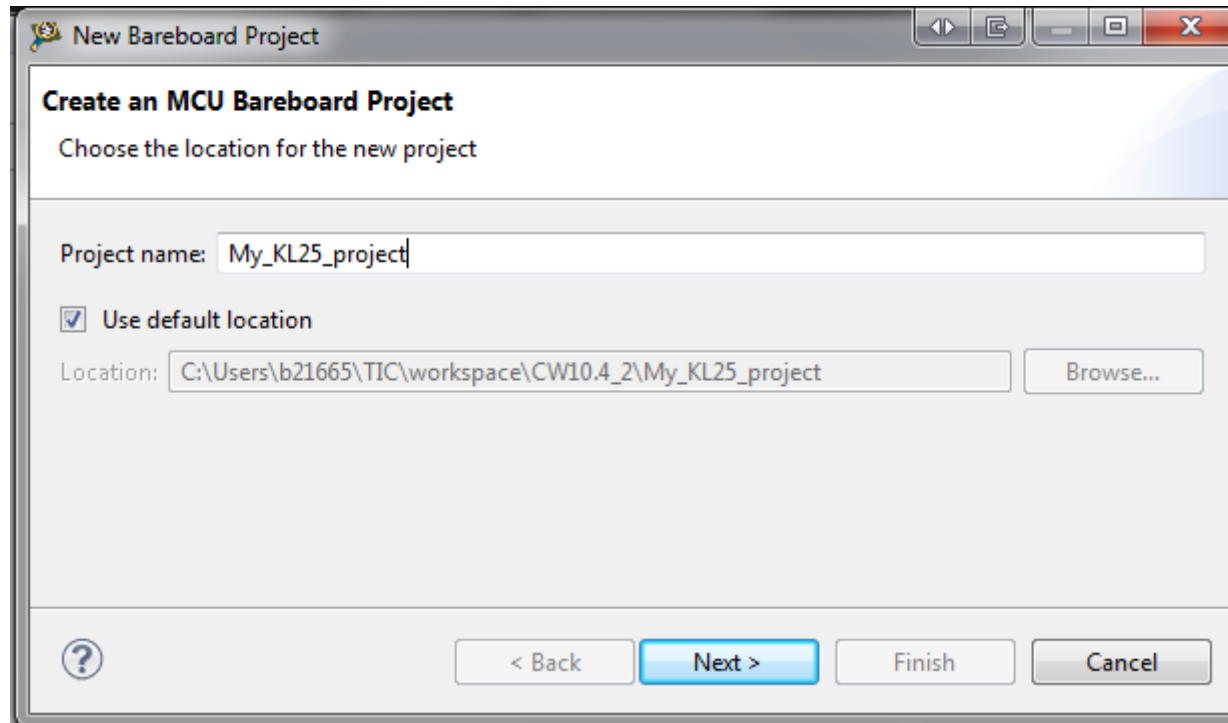
- Open CodeWarrior 10.3 or above
- Select menu File > New > Bareboard Project

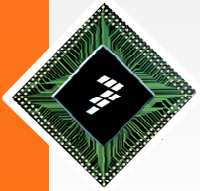




Creating a New Project

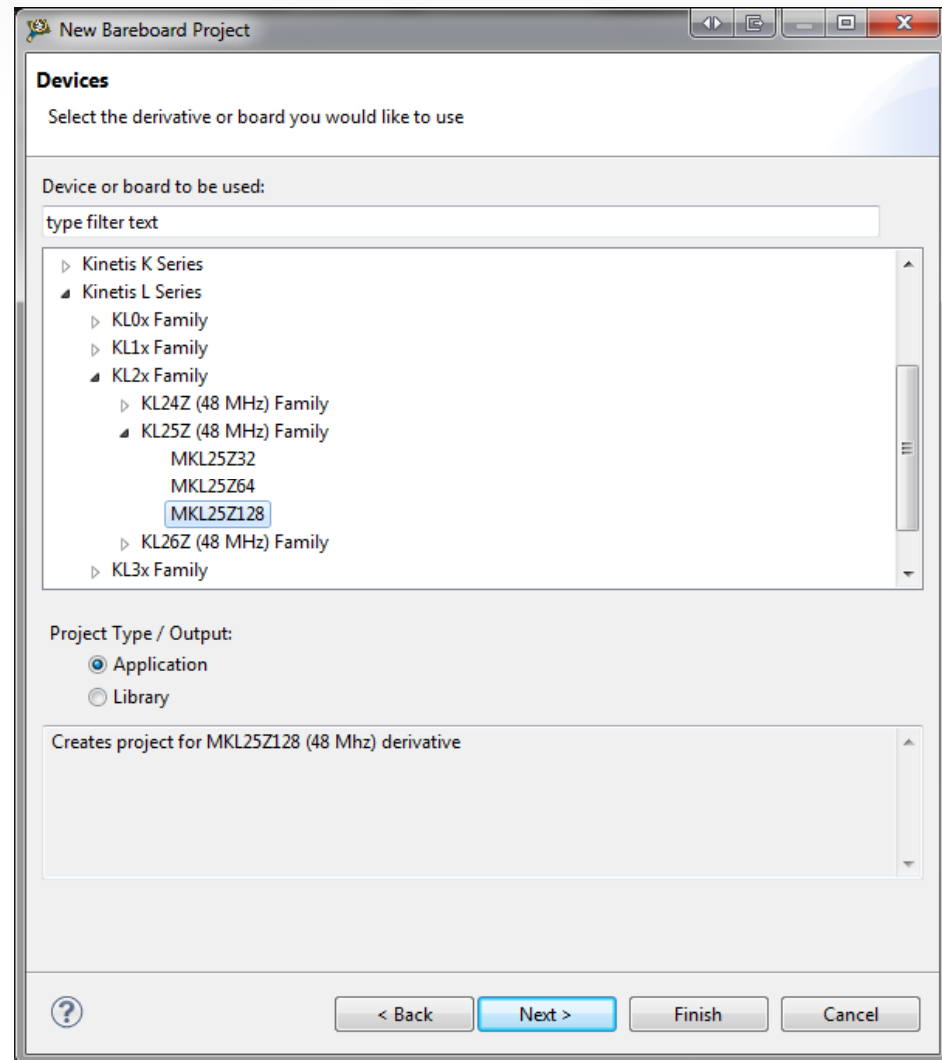
- In the next window write a name for your project and click 'Next >' button

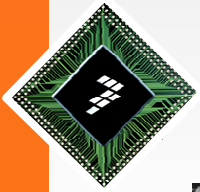




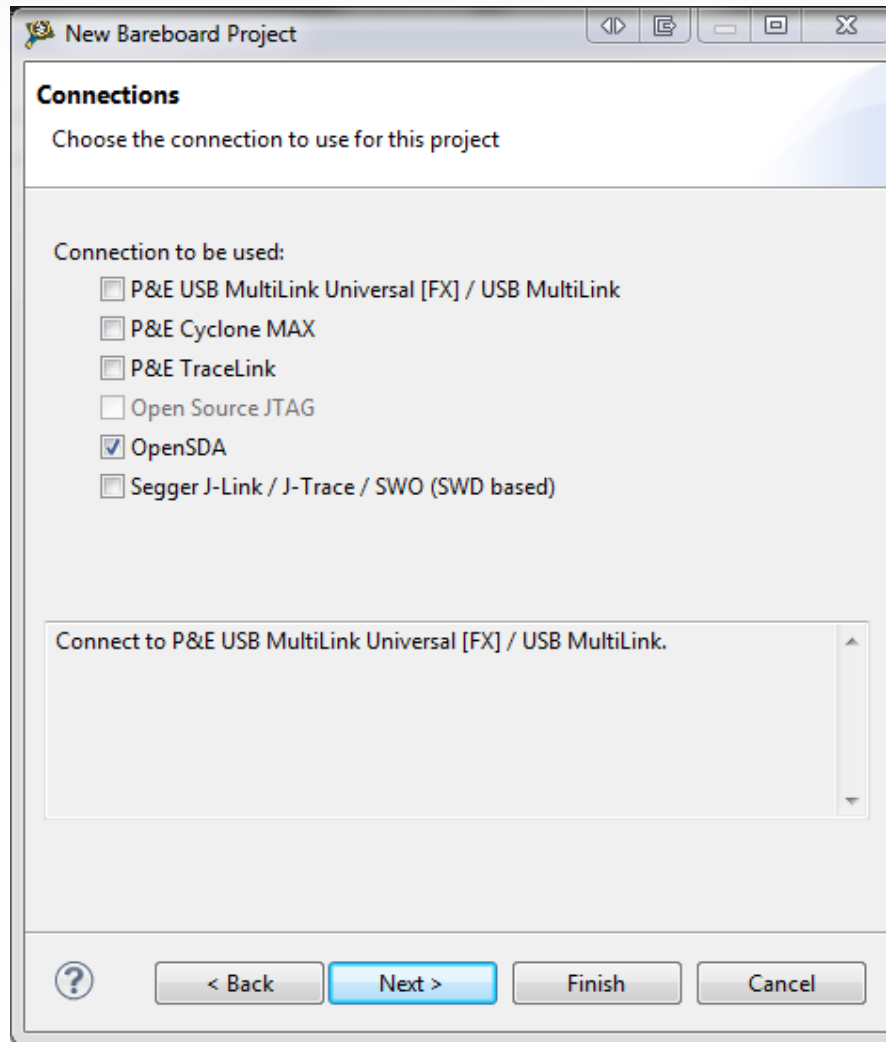
Creating a New Project

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

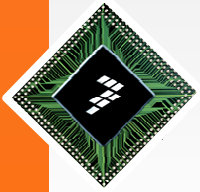




Creating a New Project

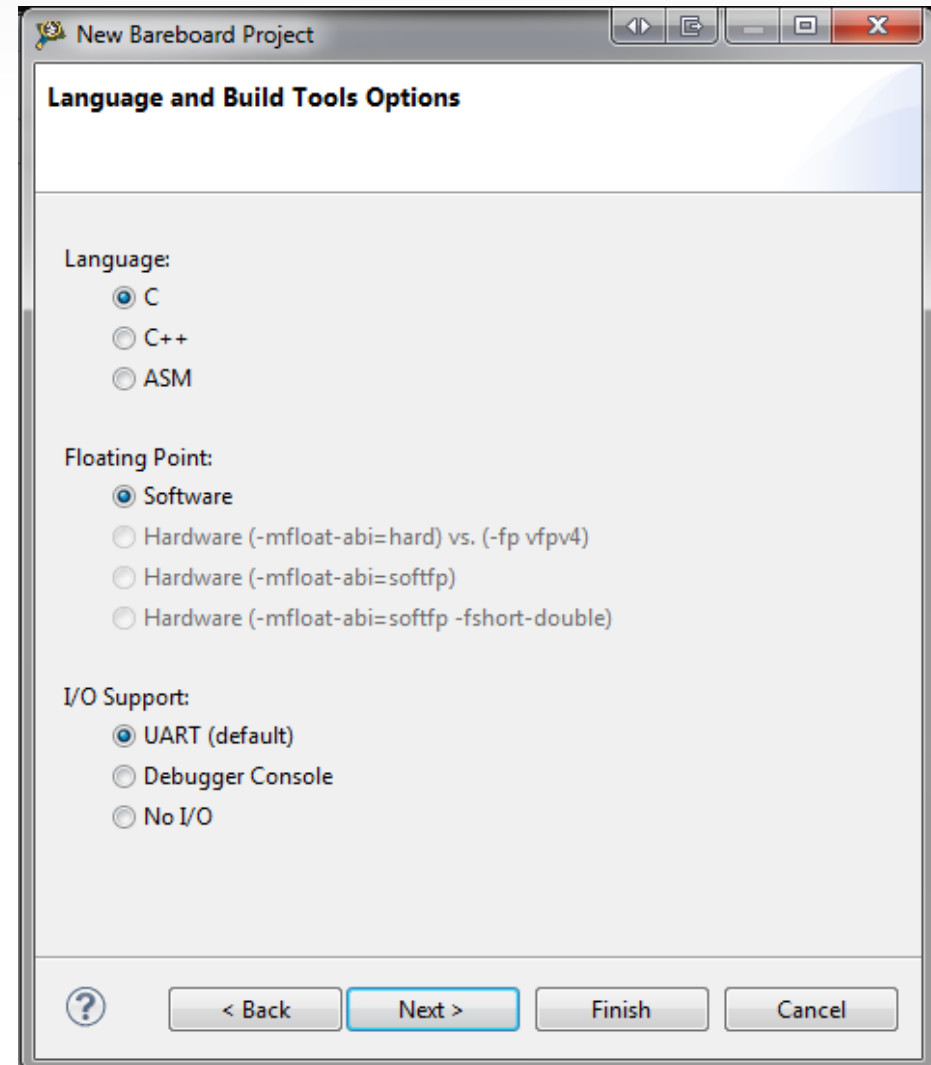


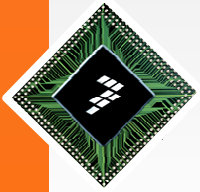
- Select the connection type that you are using, you can select as many connections as you want. In this lab only 'OpenSDA' is used.



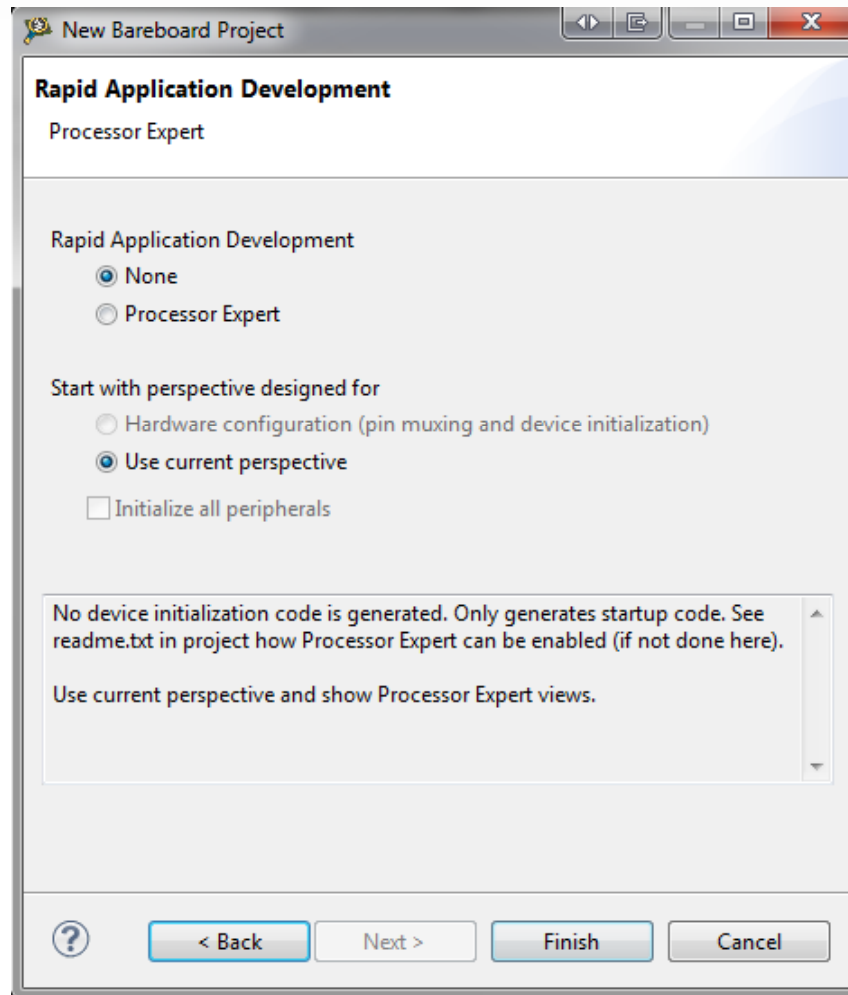
Creating a New Project

- 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

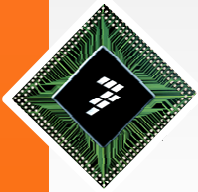




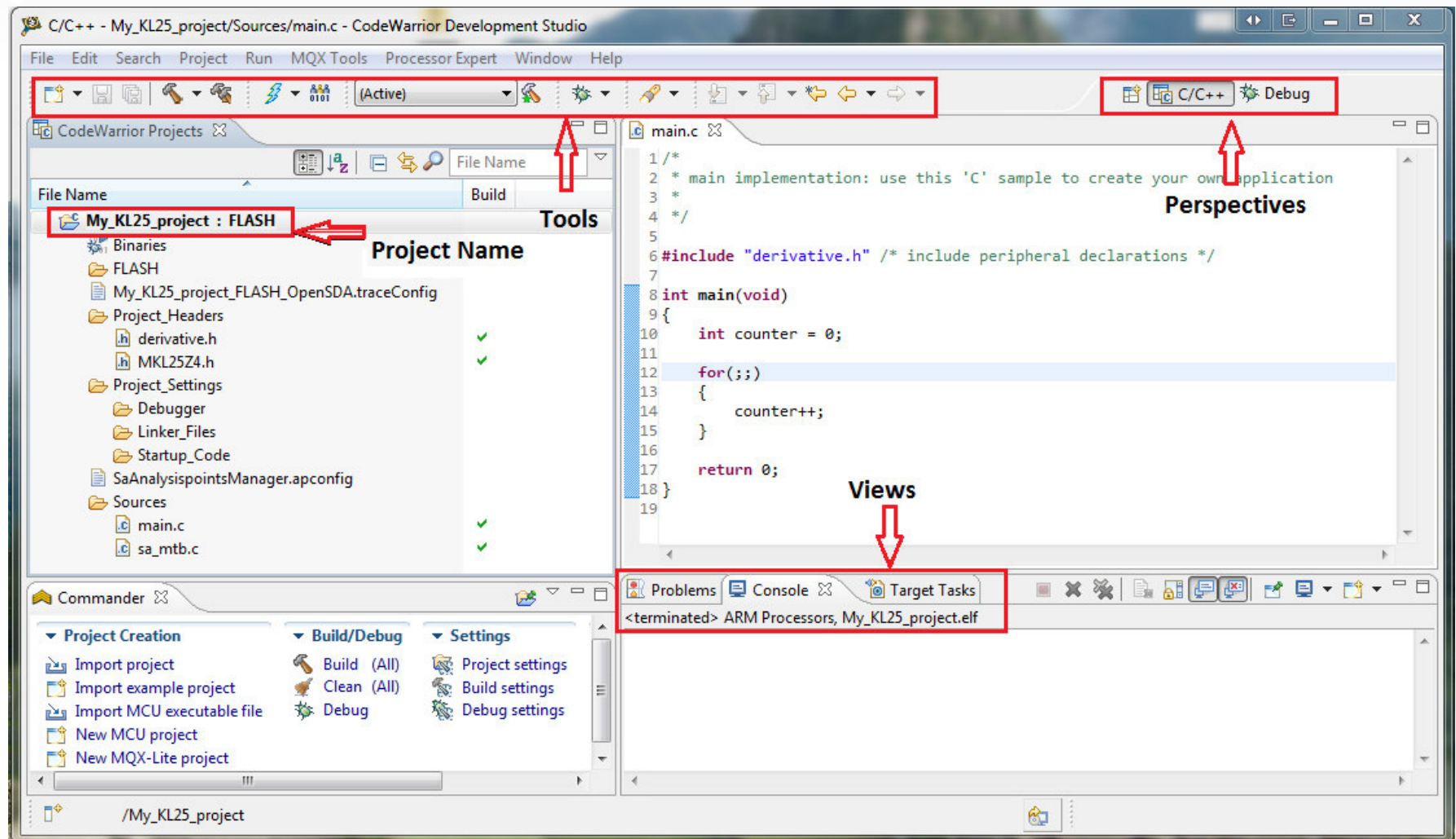
Creating a New Project

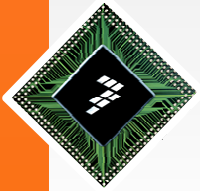


- Rapid Application Development features are not covered in this lab, select 'None' and click 'Finish' Button.



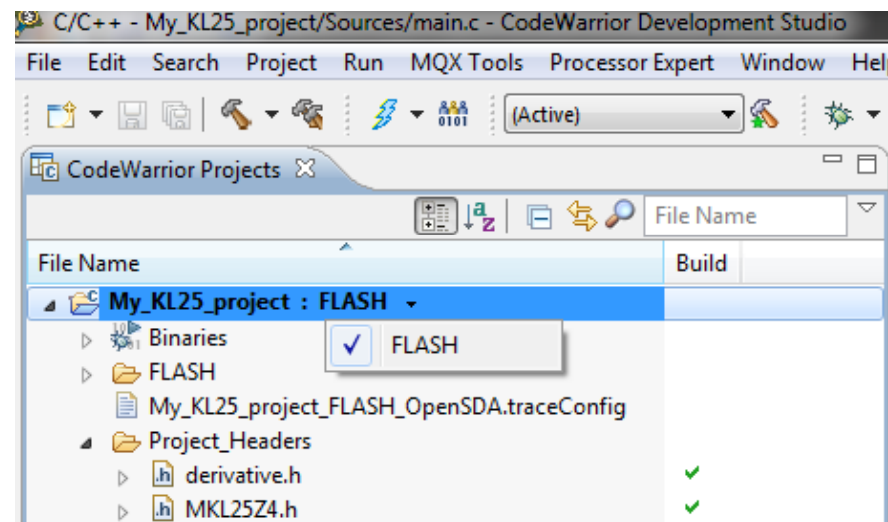
C/C++ Perspective




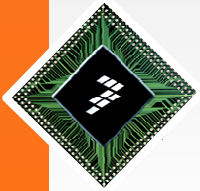


Debugging a Project

- 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.

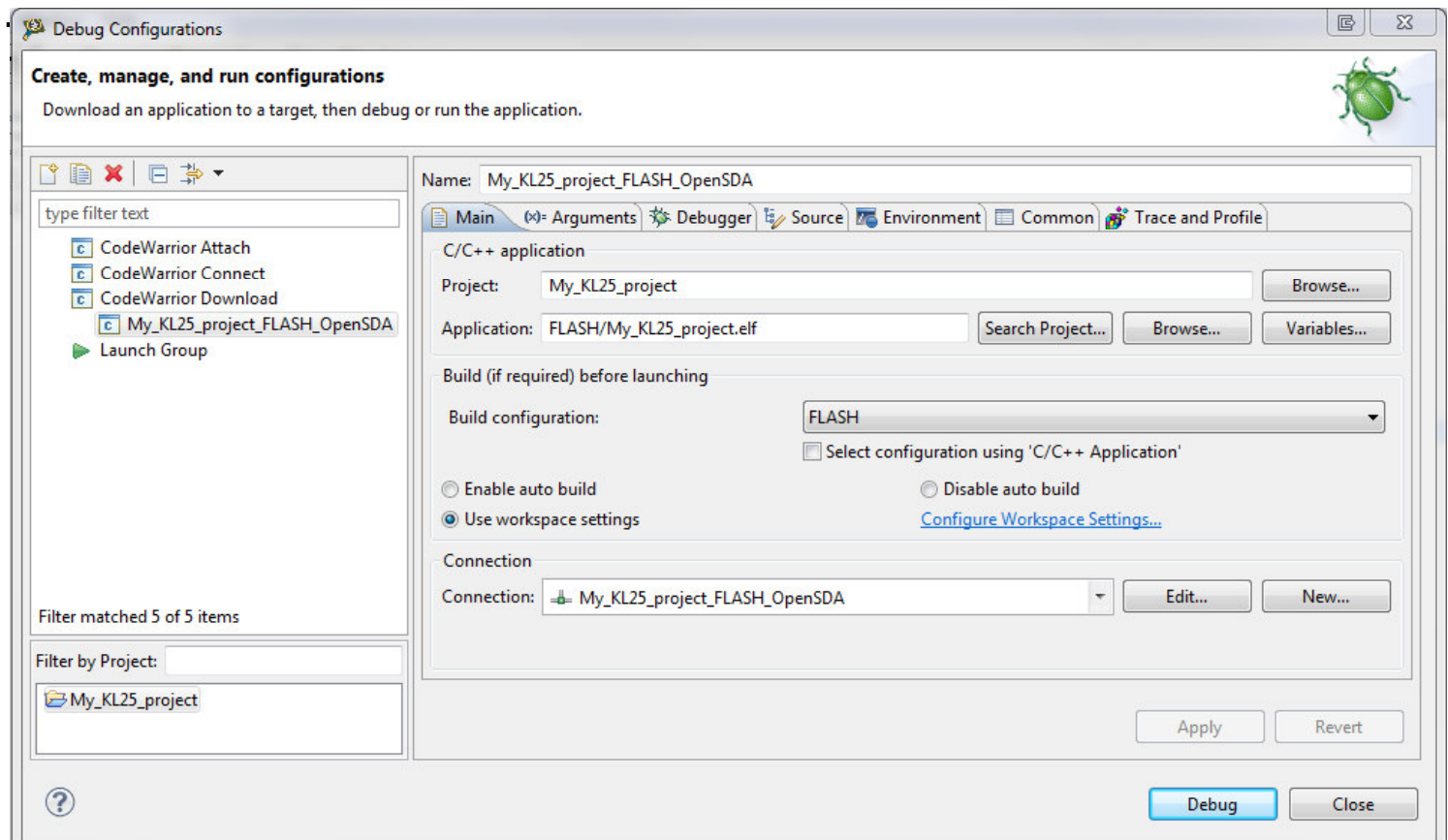


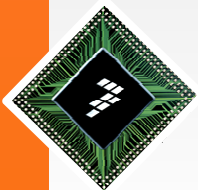
- To build the project, go to menu Project > Build Project, or alternately click the build button .



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

Debug - My_KL25_project/Sources/main.c - CodeWarrior Development Studio

File Edit Search Project Run RTCS MQX MQX Tools PEMicro Window Help

Debug Tools

Debug Views

C Source View

Disassembly View

Disassembly

Commander

Problems

Console

Target Tasks

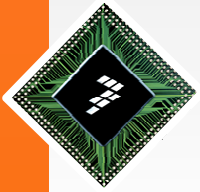
Name	Value	Location
counter	0	0x20002ff4

```
1 /*  
2 * main implementation: use this 'C' sample to create your own application  
3 *  
4 */  
5  
6 #include "derivative.h" /* include peripheral declarations */  
7  
8 int main(void)  
9 {  
10     int counter = 0;  
11  
12     for(;;)  
13     {  
14         counter++;  
15     }  
16  
17     return 0;  
18 }  
19
```

```
9  
00000960: push {r7,lr}  
00000962: sub sp,sp,#8  
00000964: add r7,sp,#0  
10     int counter = 0;  
00000966: movs r3,#0  
00000968: str r3,[r7,#4]  
14     counter++;  
0000096a: ldr r3,[r7,#4]  
0000096c: adds r3,#1  
0000096e: str r3,[r7,#4]  
15 }  
00000970: b main+0xa (0x96a) ; 0x0000096a  
00000972: nop  
00000974: exit:  
34 {  
00000974: push {r3-r5,lr}  
00000976: mov r5,r0  
37 __destroy_global_chain(); /* need to move destr  
00000978: bl __destroy_global_chain (0x1a00); 0x00001a00  
40     __fini_cpp();
```

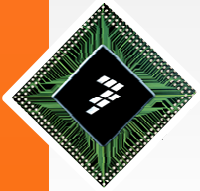
0 items

Description	Resource	Path	Location	Type
-------------	----------	------	----------	------




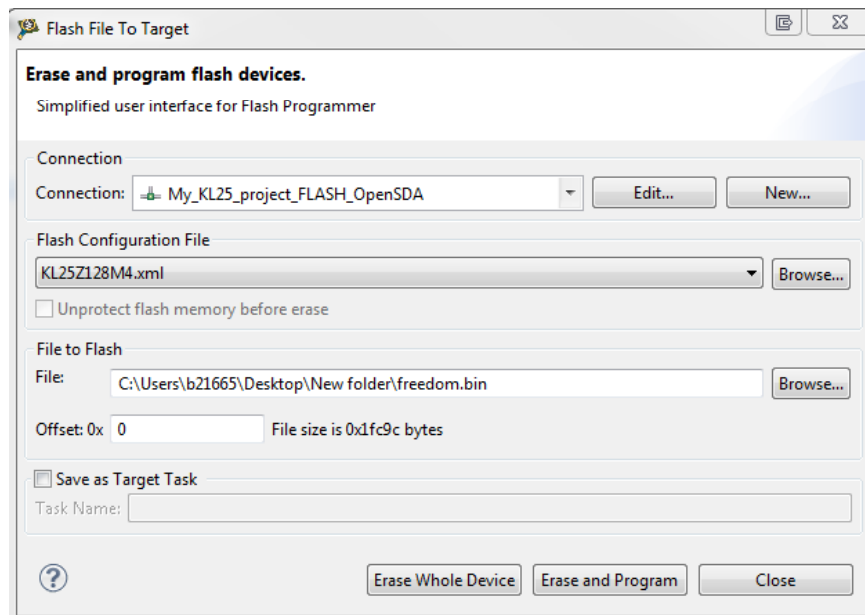
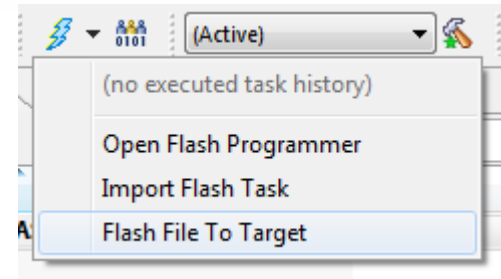
Settings

- 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 preferences go to menu Window > Preferences

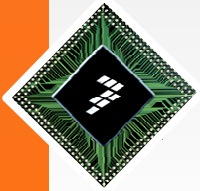


Flashing a Binary File

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

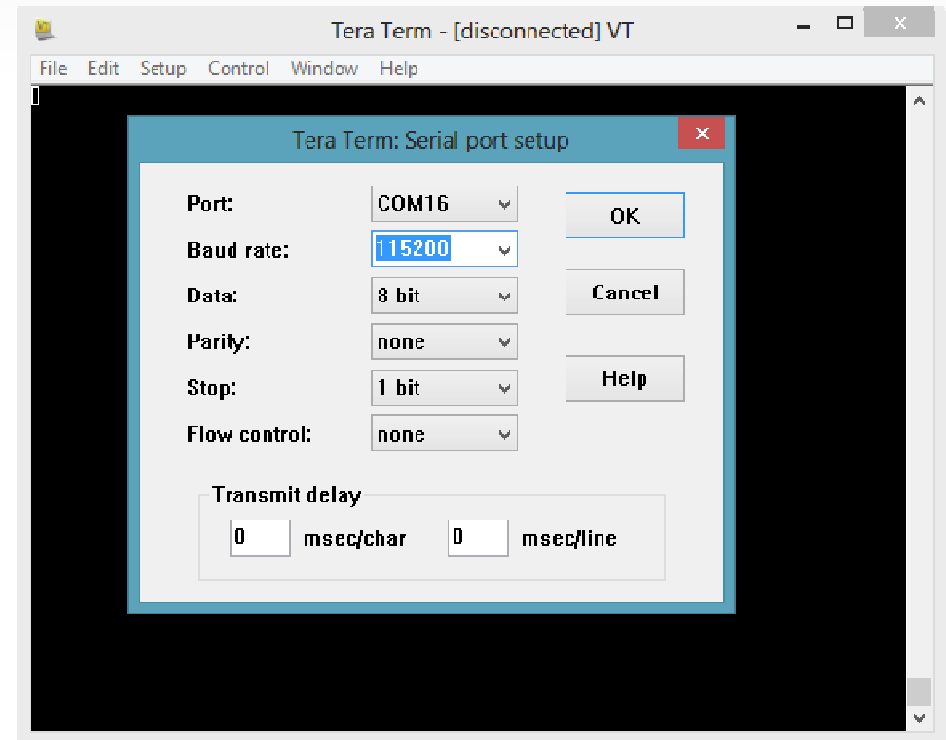


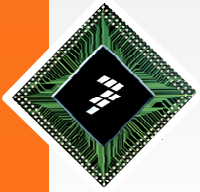
- Select your connection.
- In file to Flash search for freedom.bin
- Click 'Erase and Program'.



Running Example

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





Run Example

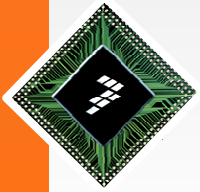
- Push the onboard reset button. You will see next screen in terminal.

```
COM16:115200baud - Tera T...
File Edit Setup Control Window Help

      _____
      |  eeeee eeeee eeee  |  |  |  |  |  |  |  |  |  |  | | | | | |
      |  e  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
      |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
      |  e  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
      |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
      |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Welcome to CoreBOOT on the Freedom Board.
Copyright (c) 2004-2012 Boulog Associates Limited.

5,144 bytes free. Core at 48 MHz. Firmware 1.2.2.
>
```



Running Example

- Below you can find some instructions

```
print "Hello World!"
```

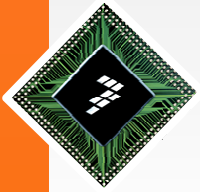
```
core.led = %red
```

```
core.led = %blue
```

```
core.led = %green
```

```
core.led = %black
```

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

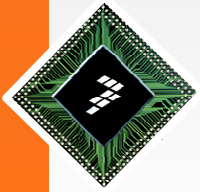


Running Example

- 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/>

