Start A New Project with Keil Microcontroller Development Kit Version 5 and Freescale FRDM-KL25Z

This tutorial is intended for starting a new project to develop software with Freescale FRDM-KL25Z board using Keil Microcontroller Development Kit for ARM (MDK-ARM) version 5.11a.

Starting version 5 the device support software components are separated from the MDK software and called Device Family Packs. As release v5.11a, many ARM microcontroller Device Family Packs are available. Software development with Device Family Packs is slightly different from software development in MDK v4.

Create a New Project in Keil MDK v5.11a

1. Launch Keil uVision IDE by double clicking on the icon ![Keil uVision5 icon]. If the icon was not created by the installation, open the folder where Keil MDK was installed (by default C:\Keil_v5). Under the folder UV4 you should find UV4.exe. Right click and hold on UV4.exe then drag it to the desktop. Release the mouse button then select "Create shortcuts here". A short cut like the one above will be created.

2. From menu, select Project>New uVision Project…
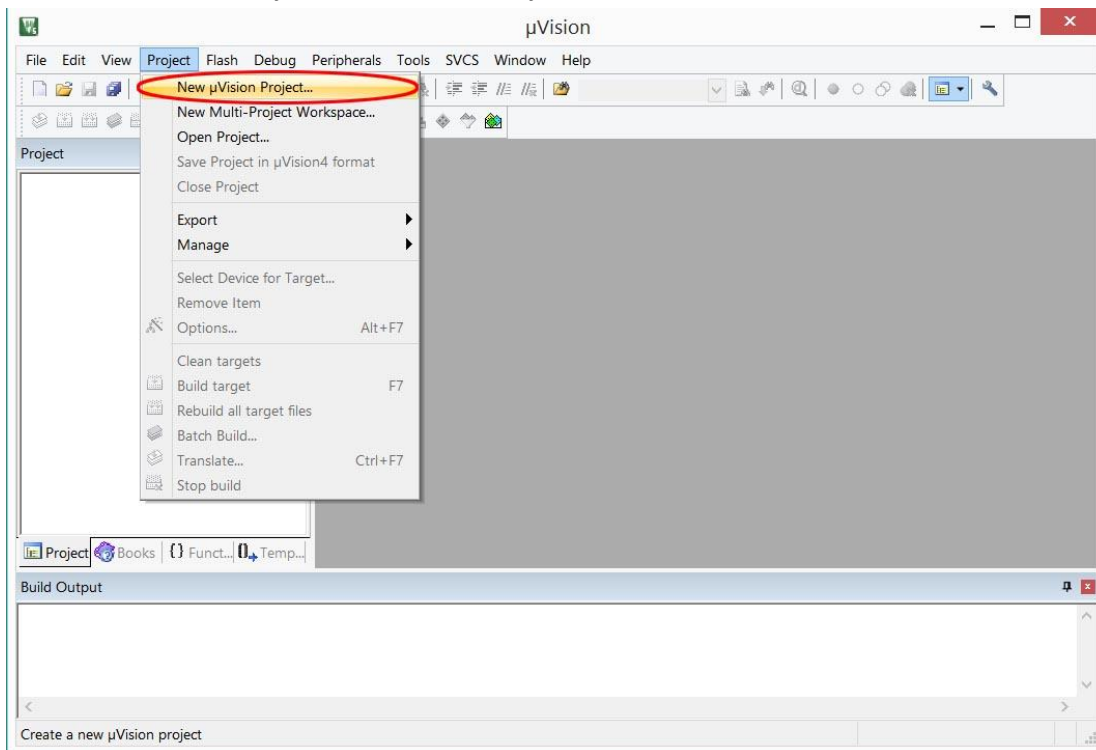
3. In the "Create New Project" dialog box, click "New folder". Browse to a folder location where you would like to create your project folder.
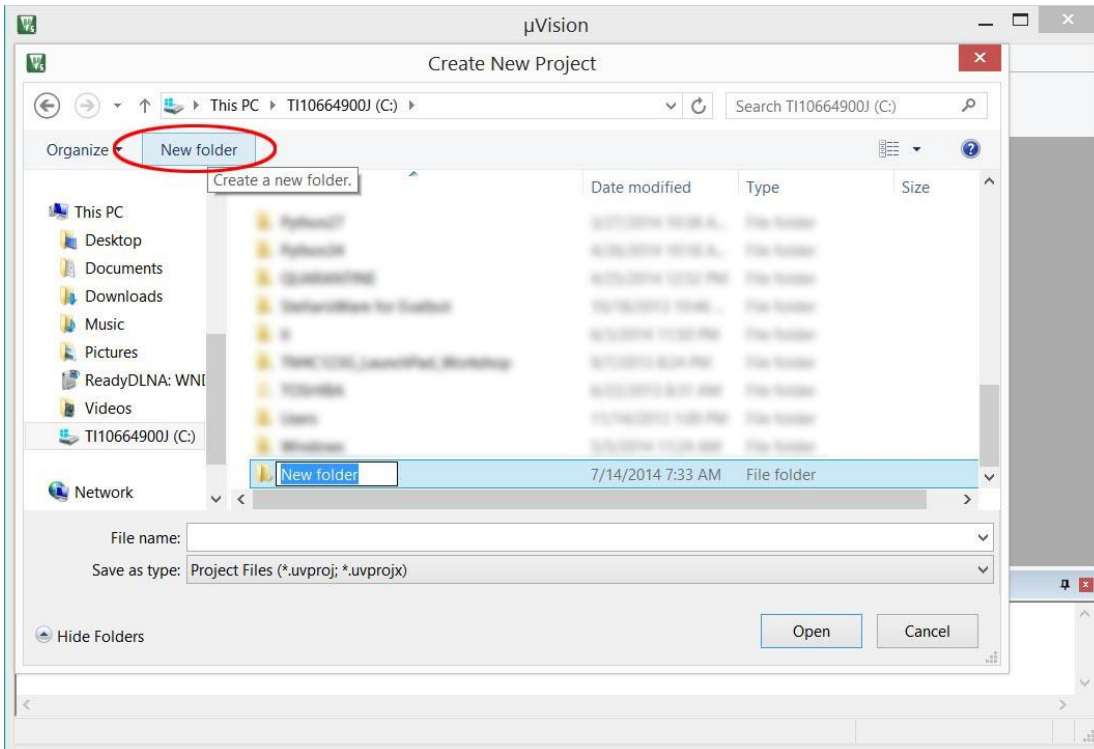

4. Type in the folder name, for example "Blinky511a" then click the Open button. This will create a folder named Blinky511a to hold all the files for the new project. It will also take you inside that folder.
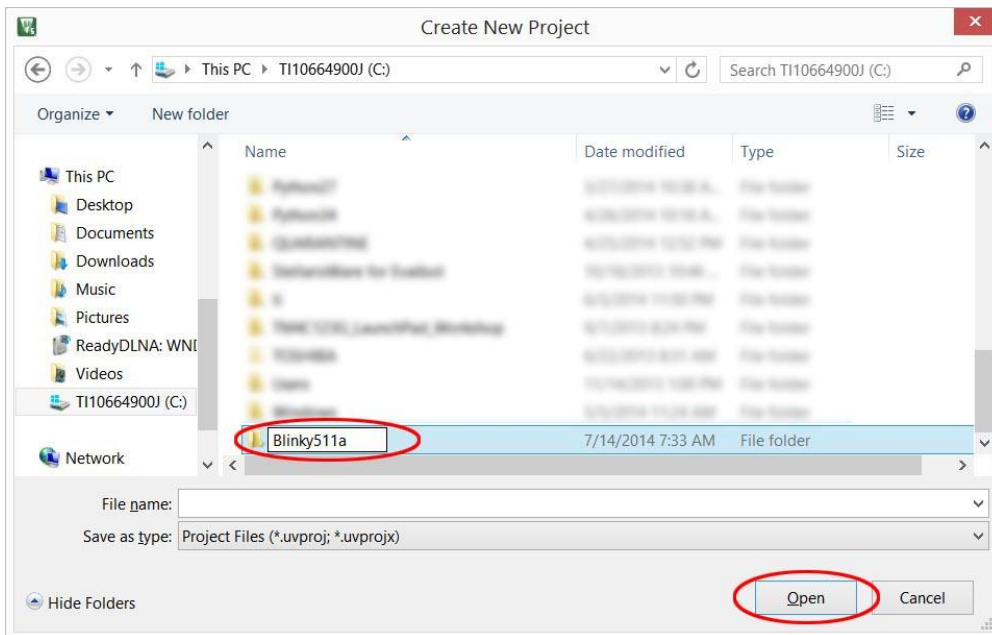
5. While in the project folder, type "Blinky" in the File name field and click "Save" button. This creates a project with the project named Blinky. For the new MDK v5 project, the project file has the file extension of ".uvprojx" (the MDK v4 and MDK v5 with legacy package creates project file with extension of ".uvproj".
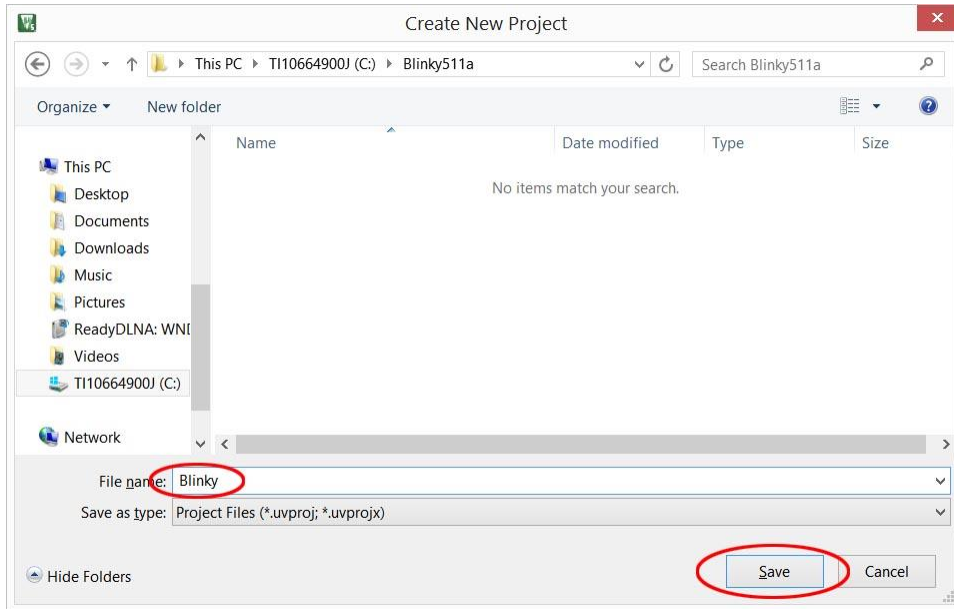

6. The Project Wizard will prompt you to select the device type for the project target. You may drill down the device family tree to find "MKL25Z128xxx4". If you installed a large number of supported devices, it may be a large task to find the device in the family trees. In that case, type in part of the device name in the "Search" field and the matching devices will appear in the window below.
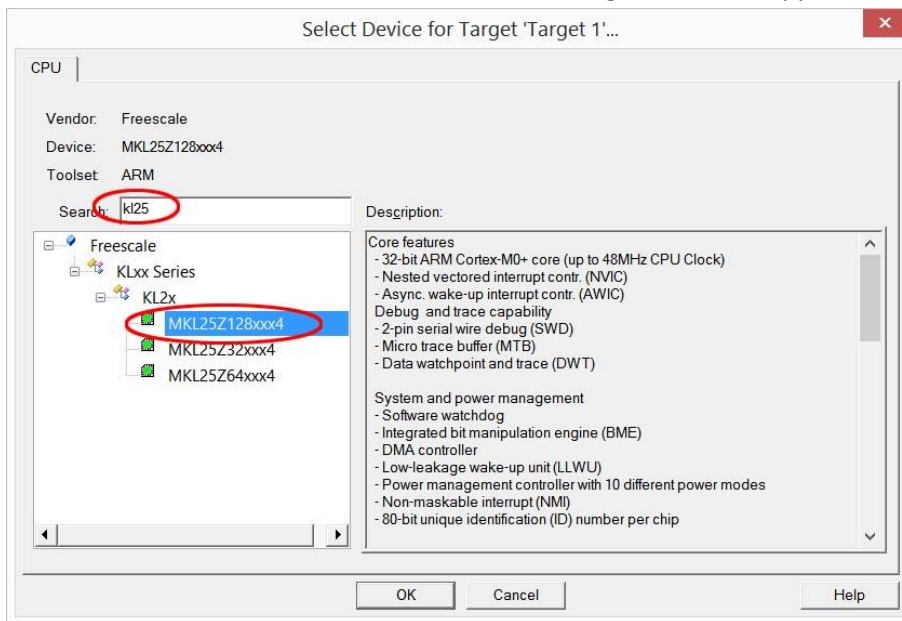
7. Next, the Project Wizard will help you manage the run-time environment. Expand the Software Component selections and check CMSIS>CORE and Device>Startup then click OK button.
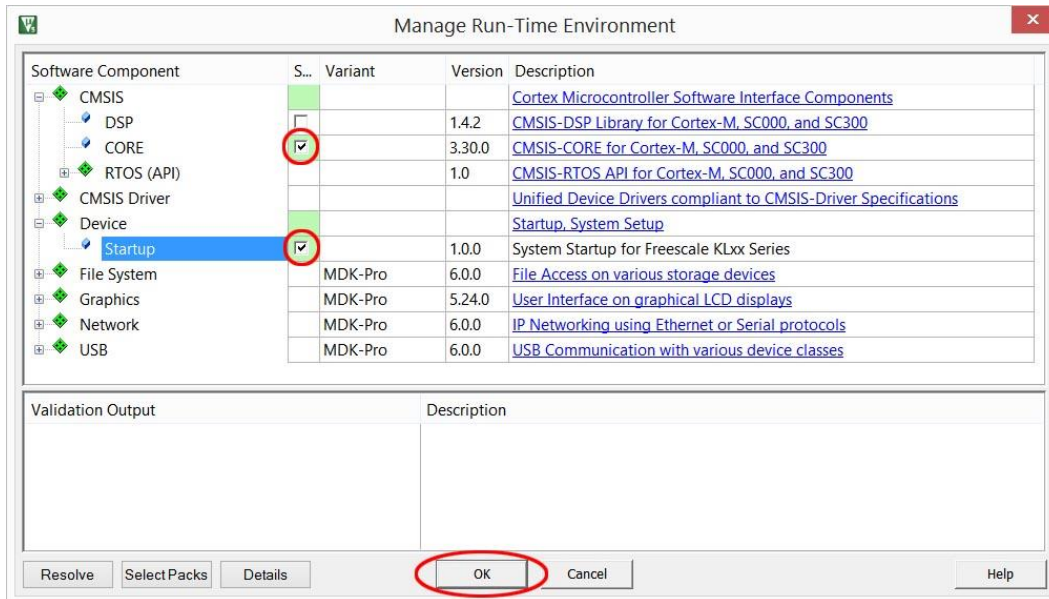
8. You should see a target created in the Project pane. If you click the + signs to open the target, you would see the CMSIS and startup codes were created in the project. Click "New" button for a new text file in the editing window.
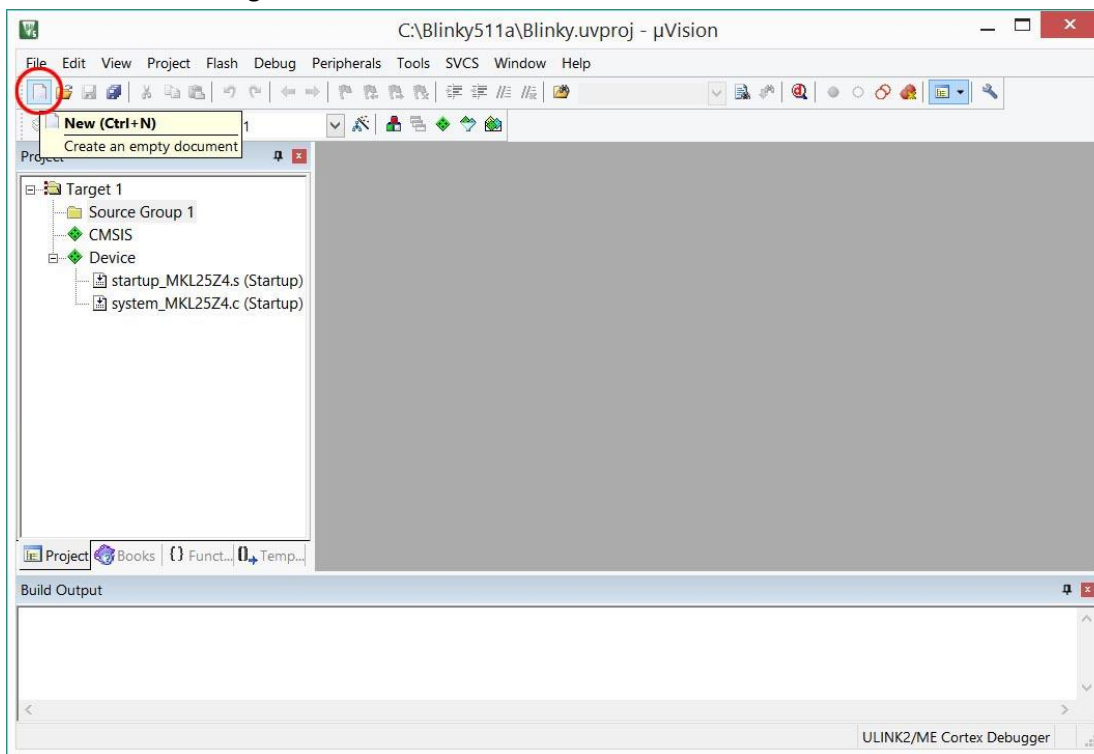
9. Copy the code below into the new text file window. This is a simple program to blink the red LED on the FRDM-KL25Z board.

```c
// This program blinks the red LED on the FRDM-KL25 board.
// The red LED is connected to PORTB pin 18 and is low active.

#include <MKL25Z4.H>

void delay(int n) {
    int i;
    for(i = 0; i < n; i++) ;
}

int main (void) {
    SIM->SCGC5 |= 0x0400;       // enable clock to Port B
    PORTB->PCR[18] = 0x0100;    // make pin PTB18 as GPIO

    FPTB->PDOR = 0x40000;       // switch Red/Green LED off
    FPTB->PDDR = 0x40000;       // enable PTB18/19 as Output

    while(1) {
        FPTB->PCOR = 0x40000;   // make the pin output low
        delay(5000000);
        FPTB->PSOR = 0x40000;   // make the pin output high
        delay(5000000);
    }
}
```
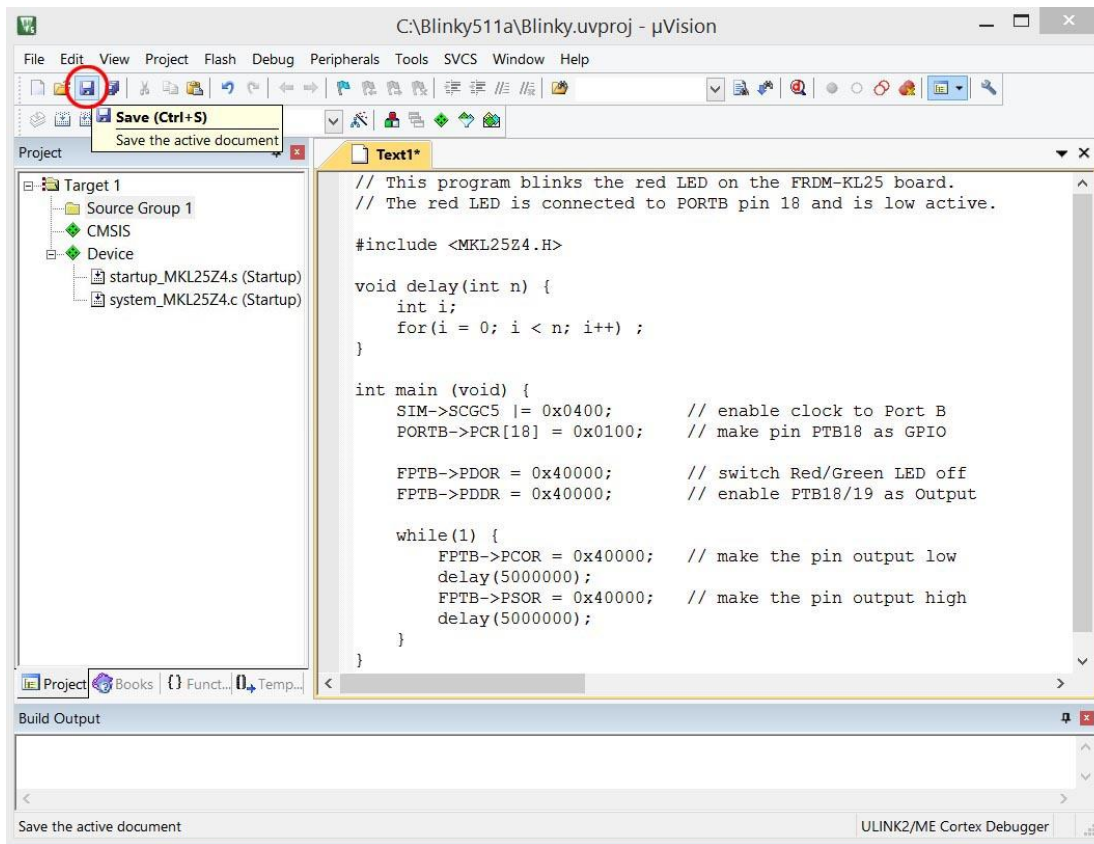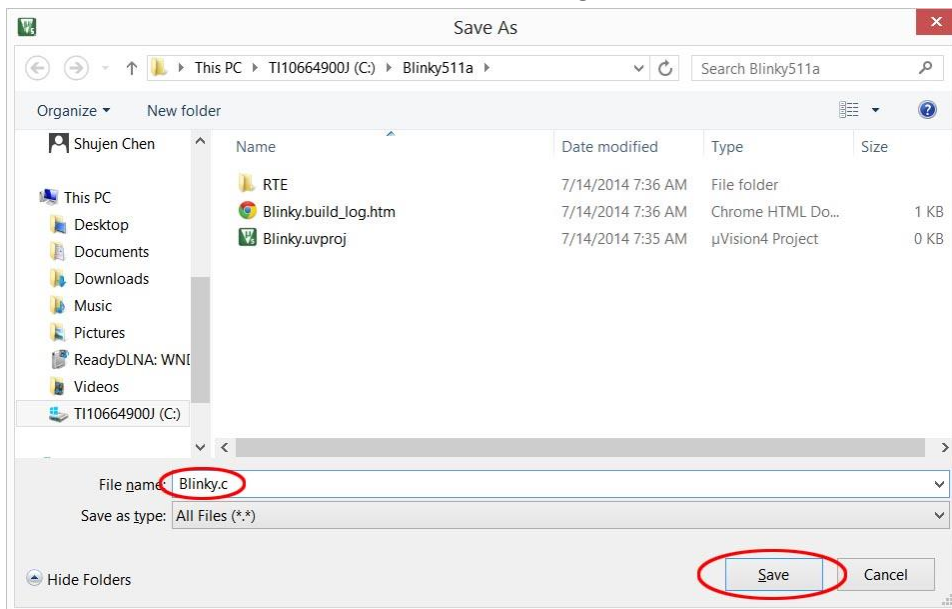
10. Click "Save" button to save the file with the code. By default the file will be saved in the project folder.


11. Give the file name "Blinky.c" and click "Save" button. The editor is context sensitive. Once you make it a C source file, the file content will change the color.
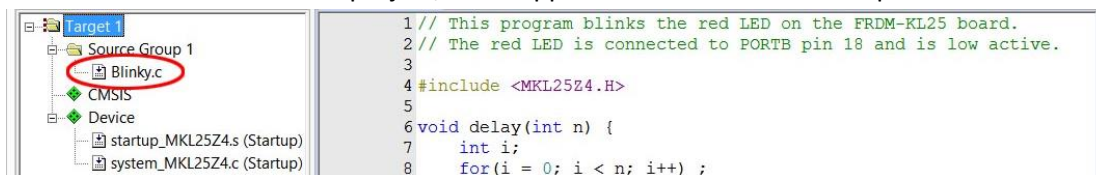
12. Once the source file is saved, it needs to be associated with the project. Right click on "Source Group 1" and select "Add Existing Files to Group 'Source Group 1'…"
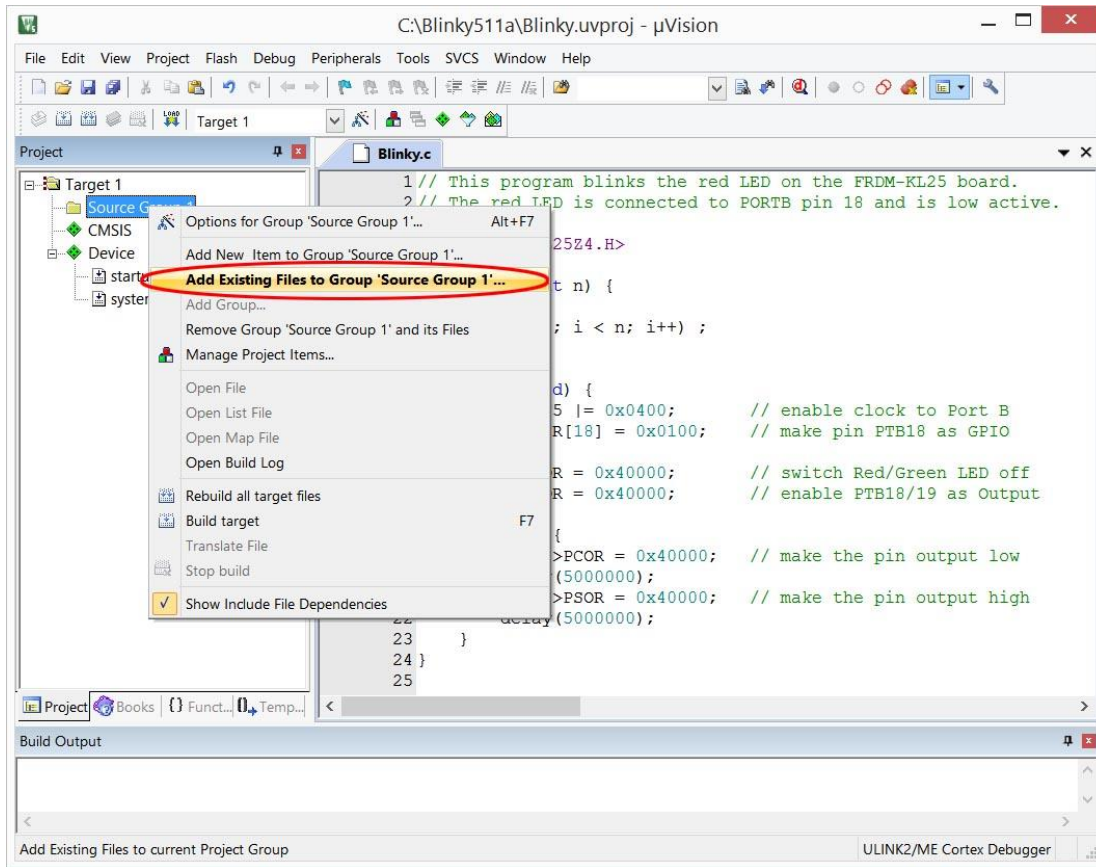
13. Locate the file Blinky.c, click select it then click the "Add" button. Click "Close" button to proceed.
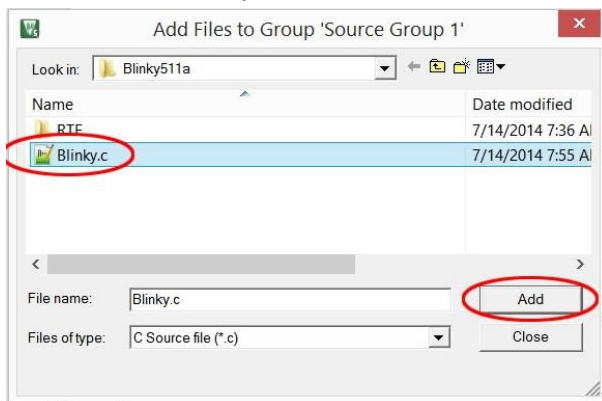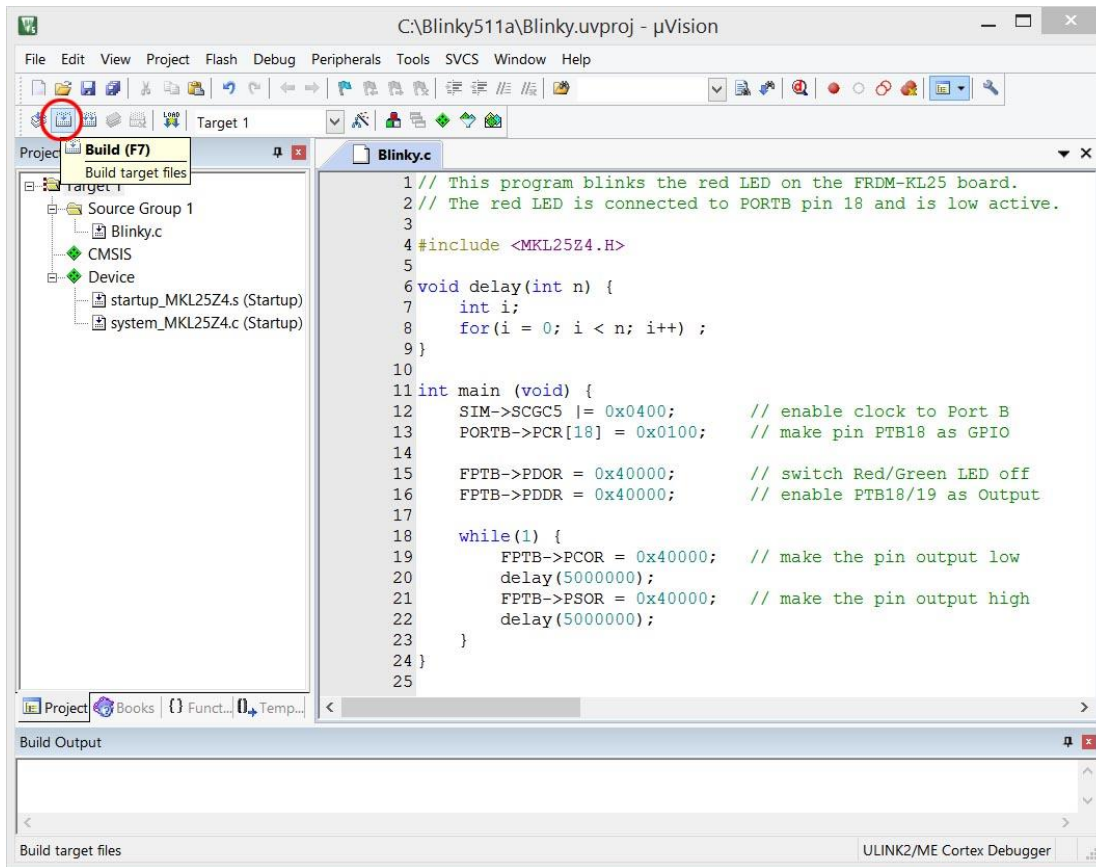

14. Once the file is added to the project, it will appear in under "Source Group 1".
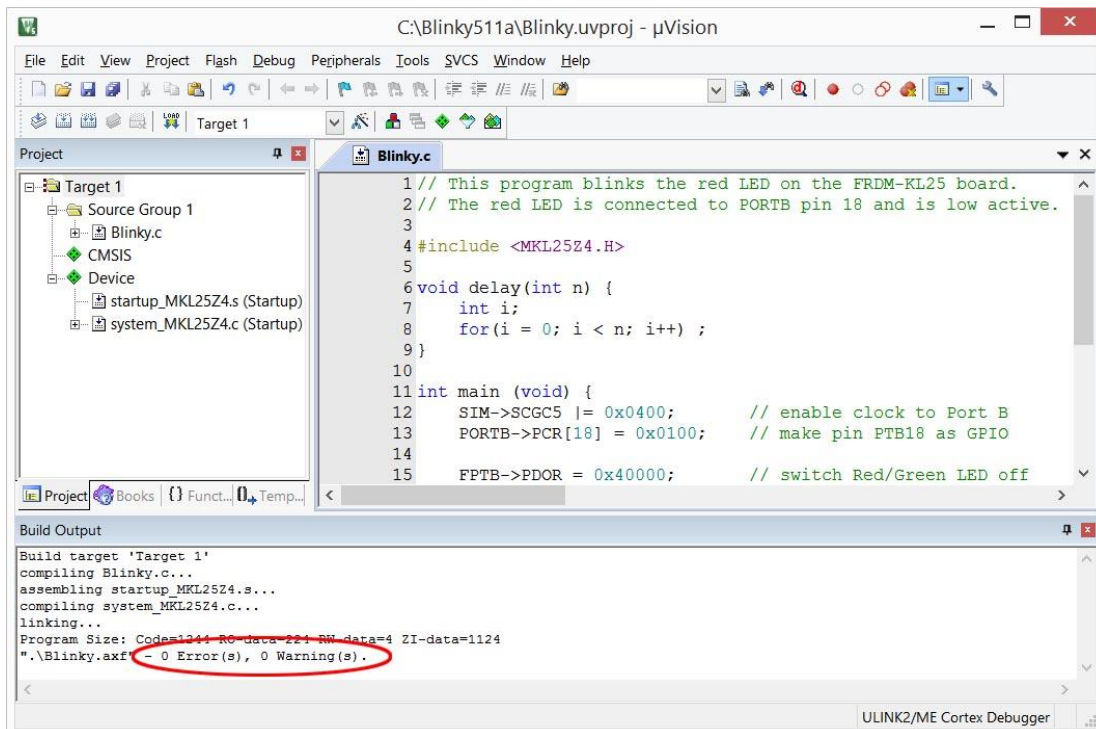
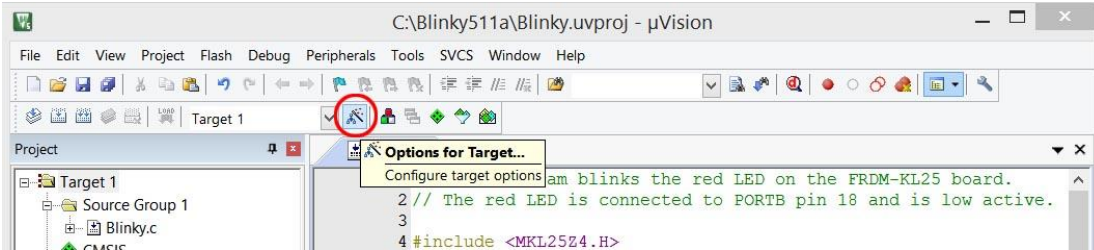15. Click "Build" button to build the project.

16. You should get a clean build with this project.

17. Once we have a project built, we need to configure the debug interface so that we may download the program to the target and test it. Make sure the FRDM-KL25Z board is connected to the computer with the USB cable.
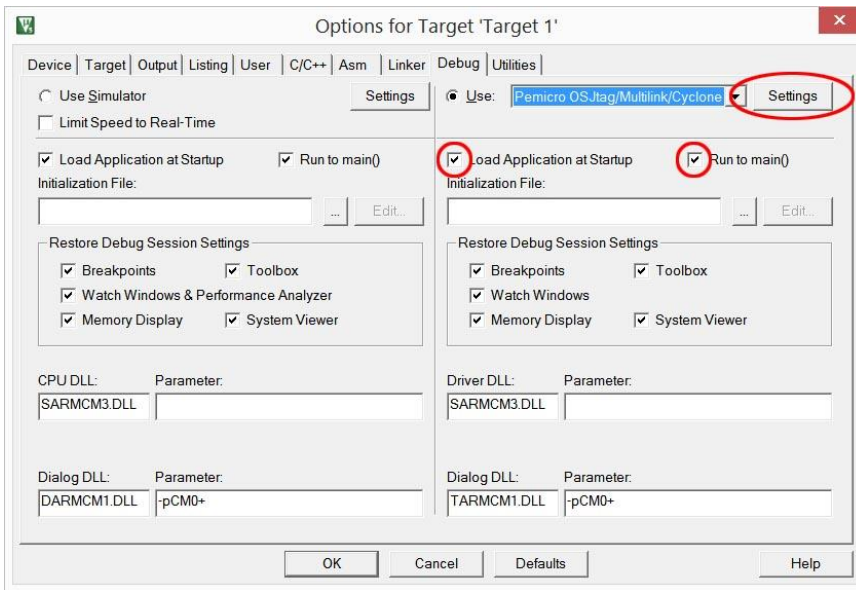
18. Click "Options for Target…" button.

19. Select Debug tab. Click the pull-down menu on the right and select "Pemicro OSJtag/Multilink/Cyclone". Click the radio button before "Use:".
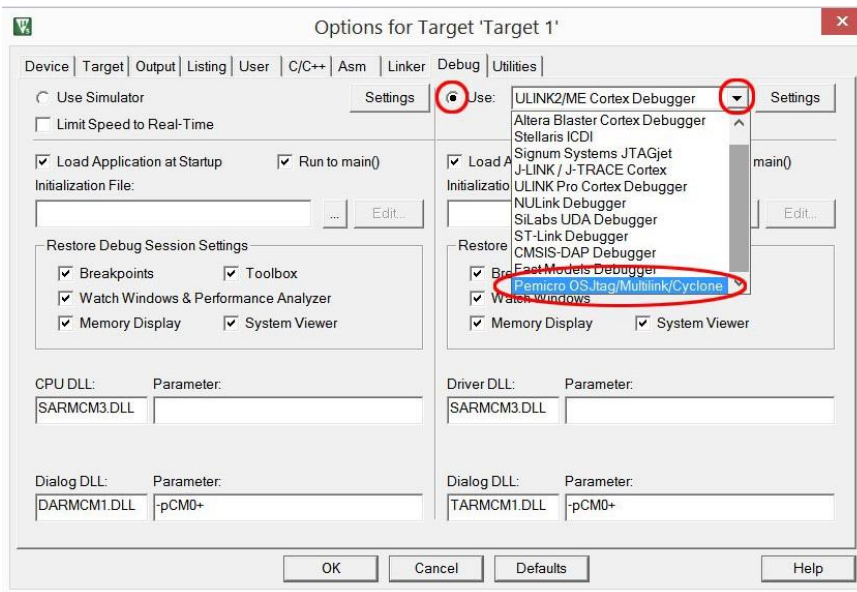
20. Check "Load Application at Startup" and "Run to main()". Click the "Settings" button.


21. In the next P&E Connection Manager dialog box, pull down the Interface: and select "OpenSDA Embedded Tower Debug - USB Port".
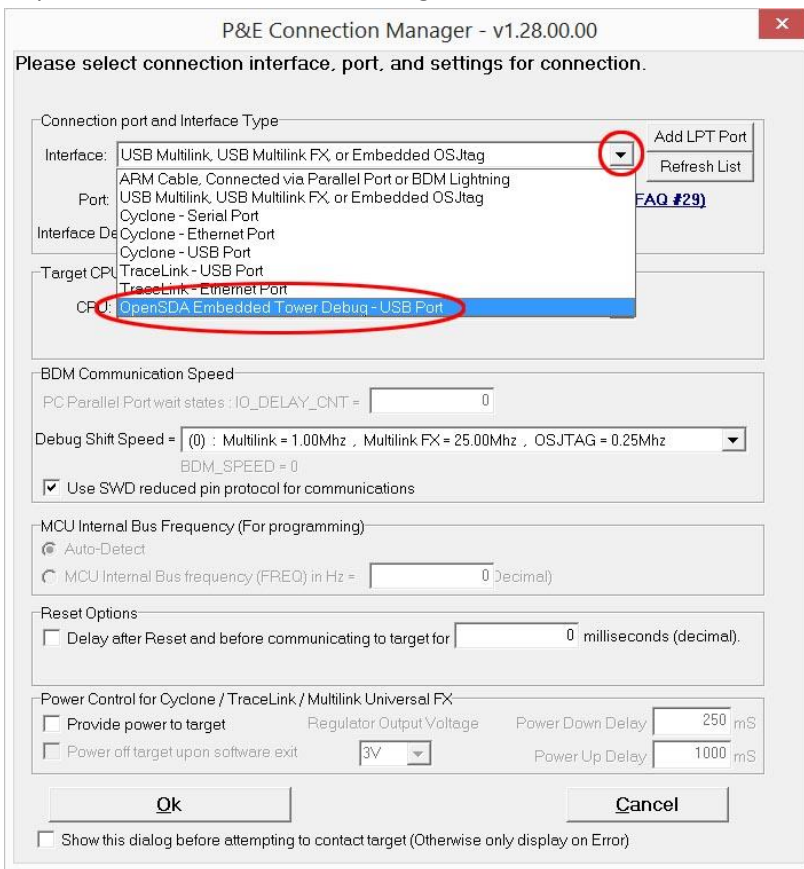
22. If the board is properly connected to the computer, the Port: information should recognize the USB connection. If the Port: field does not display a USB connection, you might want to verify the P&E driver installation.
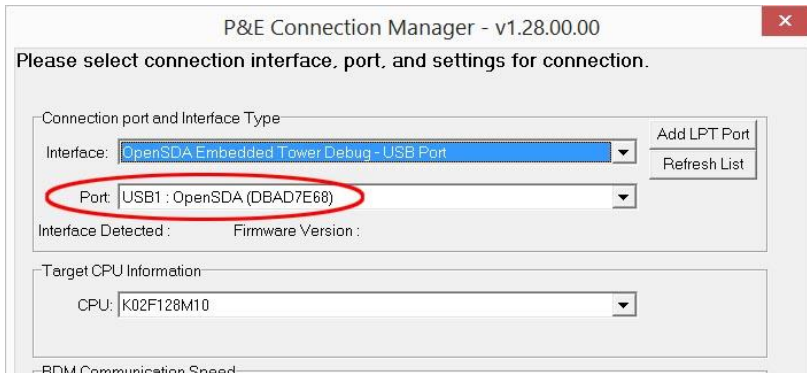

23. In the Target CPU Information, pull down the CPU selection and select KL25Z128M4. If you don't see choices for CPU, the P&E Driver update was not completed properly. Go back and try to reinstall the update and make sure it is installed at the same folder Keil MDK-ARM was installed.
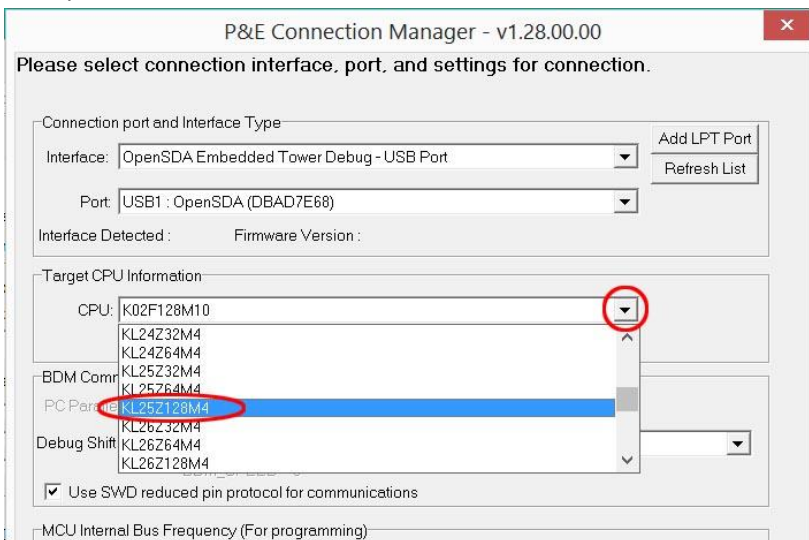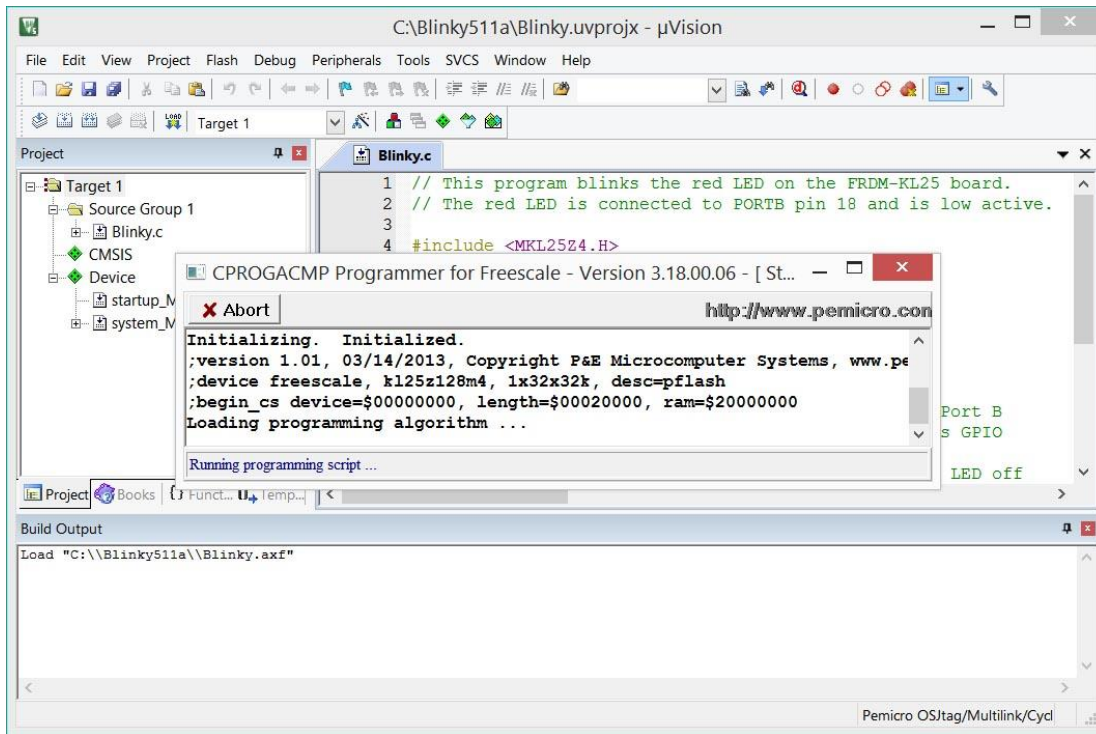
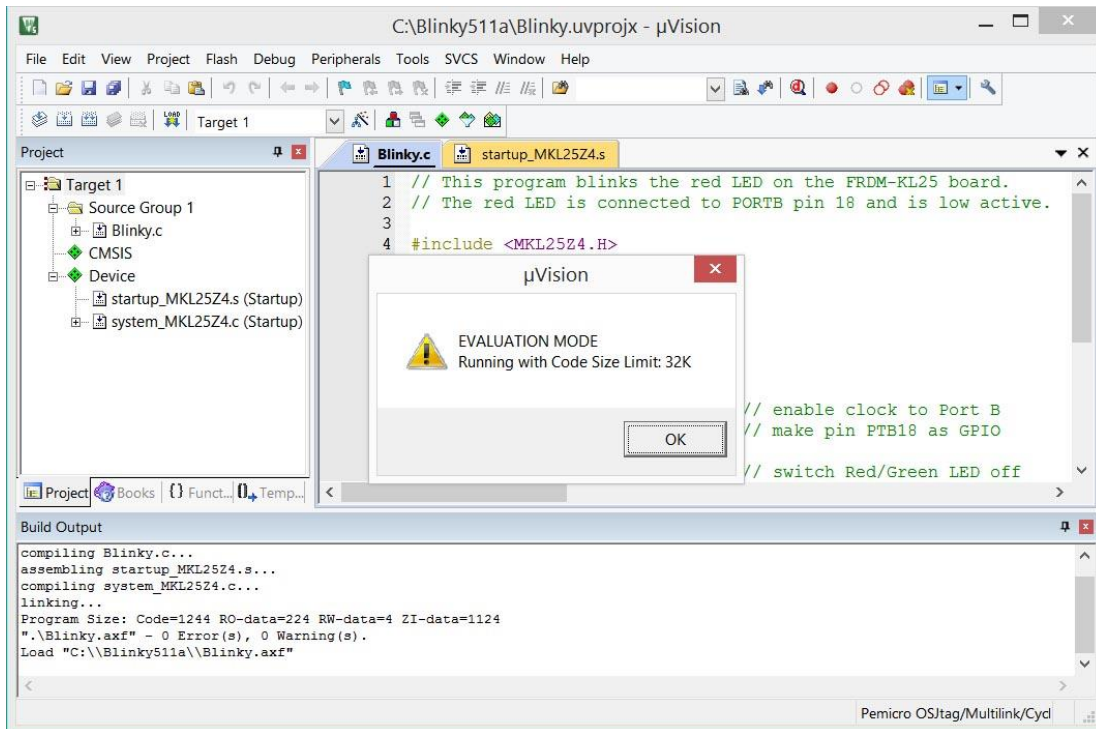24. Make sure "Use SWD" is checked then click the OK button.
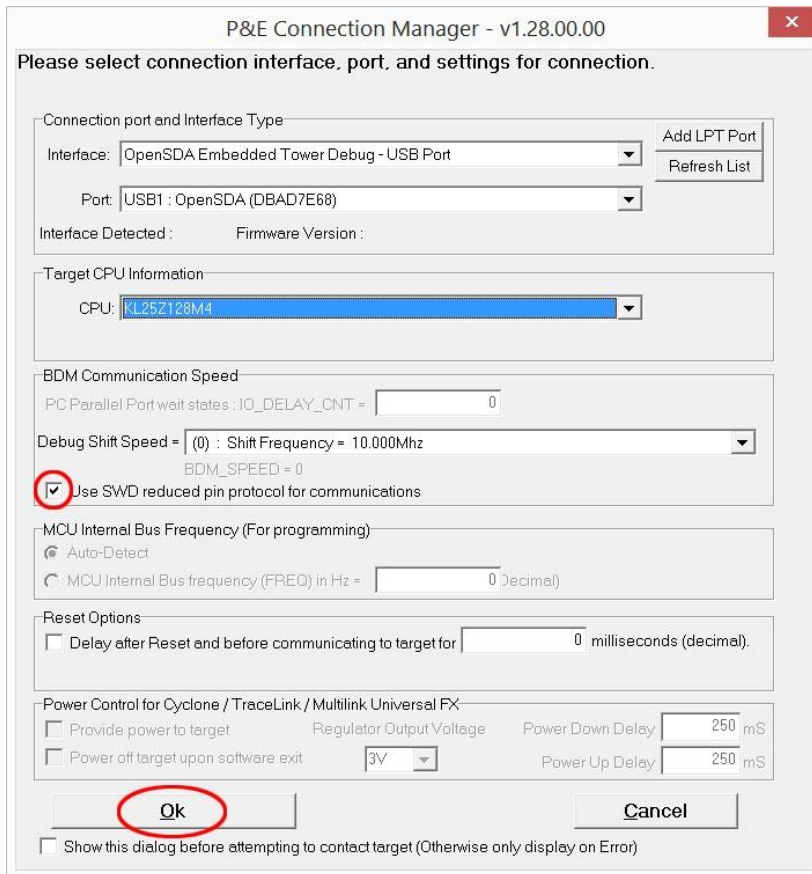
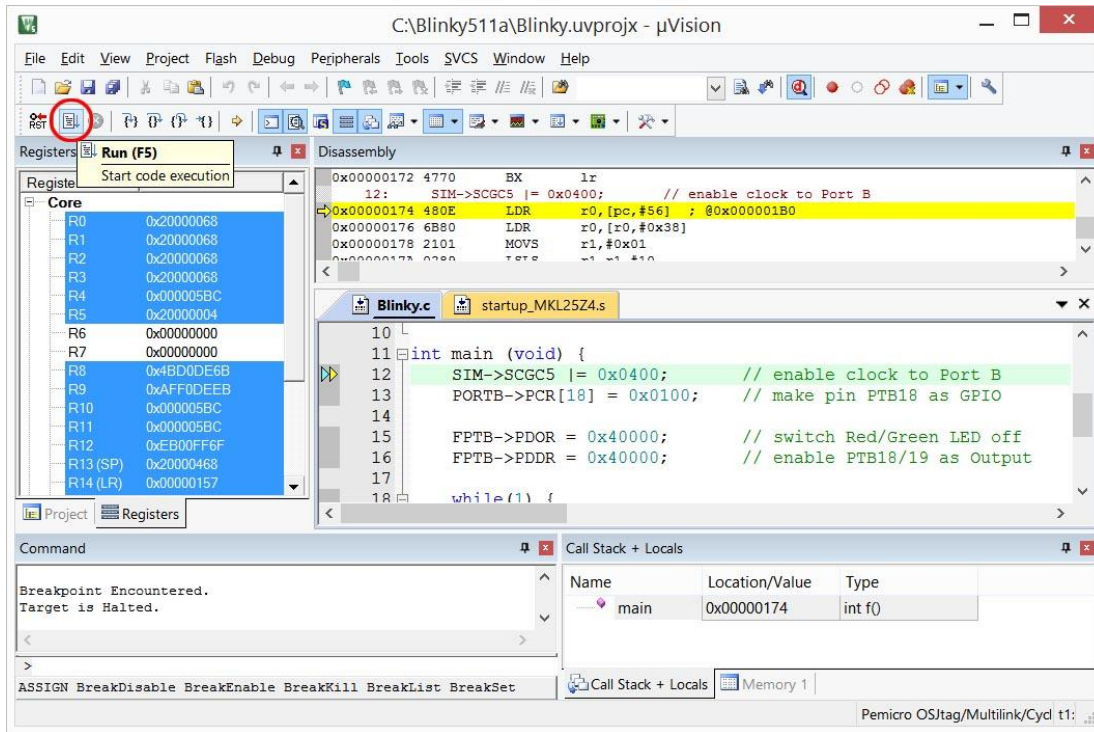25. Click on the "Start/Stop Debug Session" button.
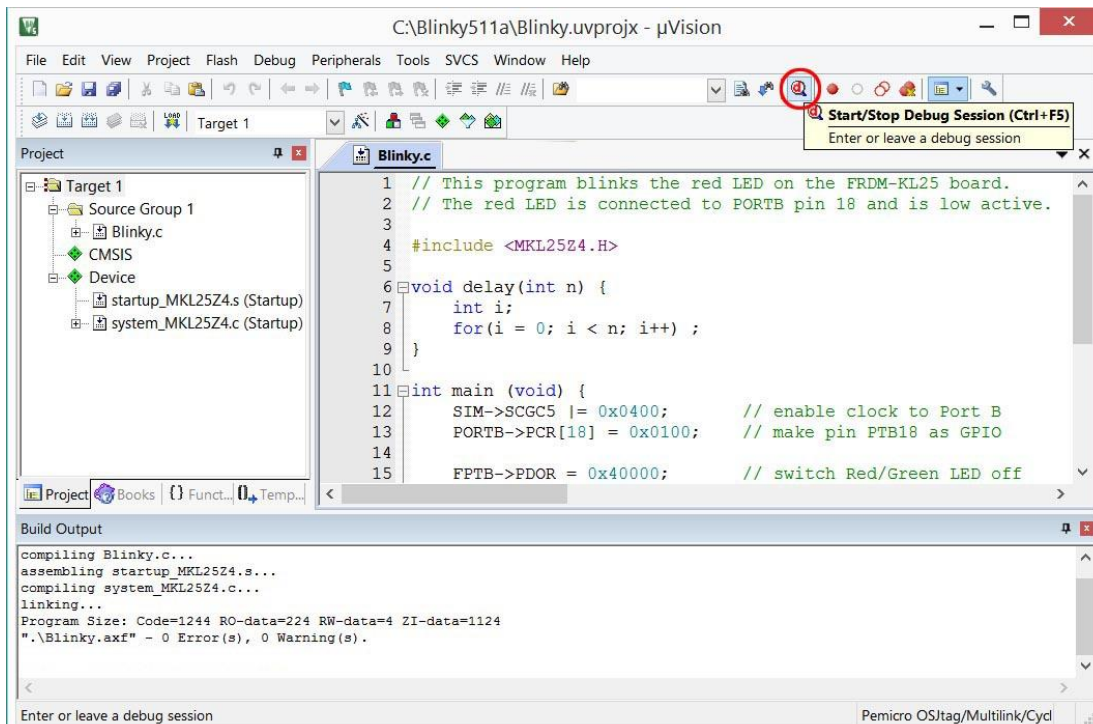
26. The programmer message window will pop up and the programming of the target device starts.


27. If you are using the evaluation version of the Keil uVision, a warning of the code size limitation will pop up after the target device is programmed.

28. The uVision IDE should change into debug perspective. Click "Run" button and the program should start running and the red LED should blink.

29. To stop the debug session and return to the build project perspective, click on the "Start/Stop Debug Session" button.
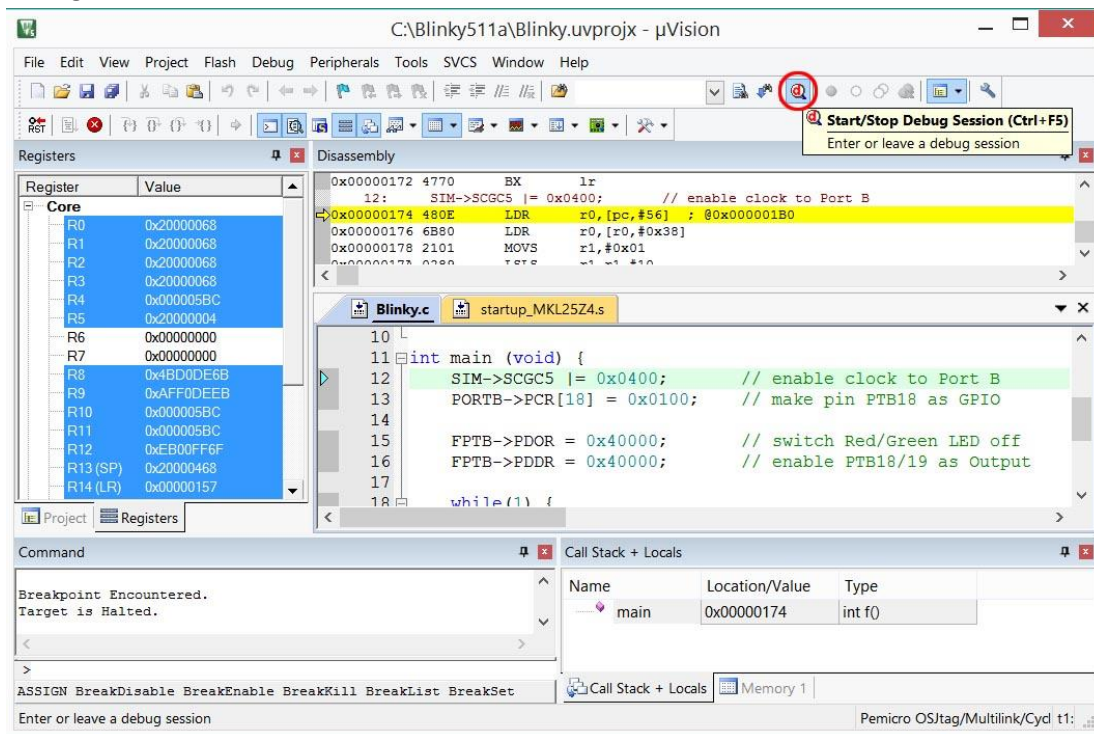
30. Congratulations! You have successfully finished the first programming project for FRDM-KL25Z board using Keil MDK v5 uVision IDE!