Concordia University
Engineering and Computer Science

User Guide document for FRDM-KL25Z Evaluation board, Keil-MDK5 software development environment and mbed development platform

Course Instructor:
Dr.Otmane Ait Mohamed
Prepared by:
MohammedHossein Askari Hemmat
COEN 6711
Concordia University

Winter 2014

Note: Before attaching the FRDM-KL25Z board to your computer, please read the P&E OpenSDA Firmware section on this document.

Contents

1 Introduction

This document guides you through the installation process of the following softwares and drivers on your PC:

- KEIL MDK5
- P&E OpenSDA Firmware

Also at the end of this document, you can find a thorough description of the Blinky project. Starting from building a simple project and adding the necessary libraries/drivers to the project to compiling the source code and debugging the project while it is executing on the FRDM-KL25Z platform.
You can find more information about the FRDM-KL25Z platform on the Freescal website. Also, you can find many useful information regarding this platform on mbed and element14 websites.

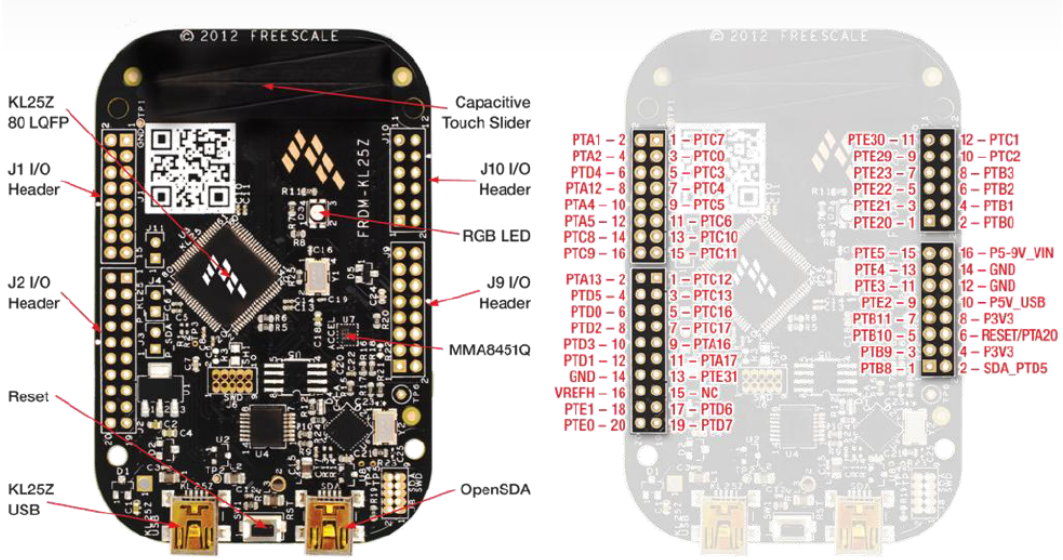


Figure 1: FRDM-KL25Z board pin description

2 Step1: Keil MDK5 Installation process

Keil MDK5 is a complete software development environment for almost all of the ARM based microcontrollers. MDK-ARM V5 is the latest version which were released in 2013. The installation process is pretty straight forward:
First you need to download the latest version of MDK-ARM V5 from KEIL website. Before downloading, you will be asked to fill out the installation form.

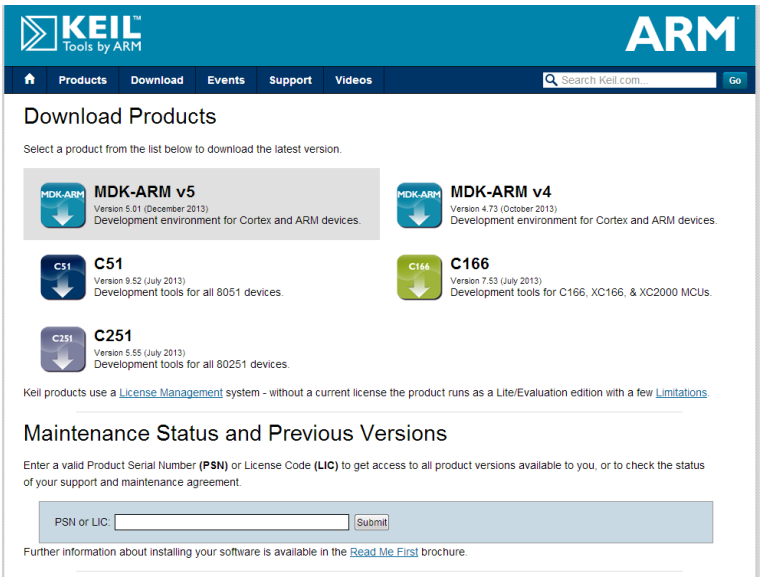


Figure 2: KEIL Product page

After you downloaded the file, you can run the mdk501.exe by double clicking on it. The program will install on your computer after several minutes. When you have successfully installed the program, the Pack Installer will pop-up where you can add the CMSIS library and some useful examples specially written for your specific platform to your program.

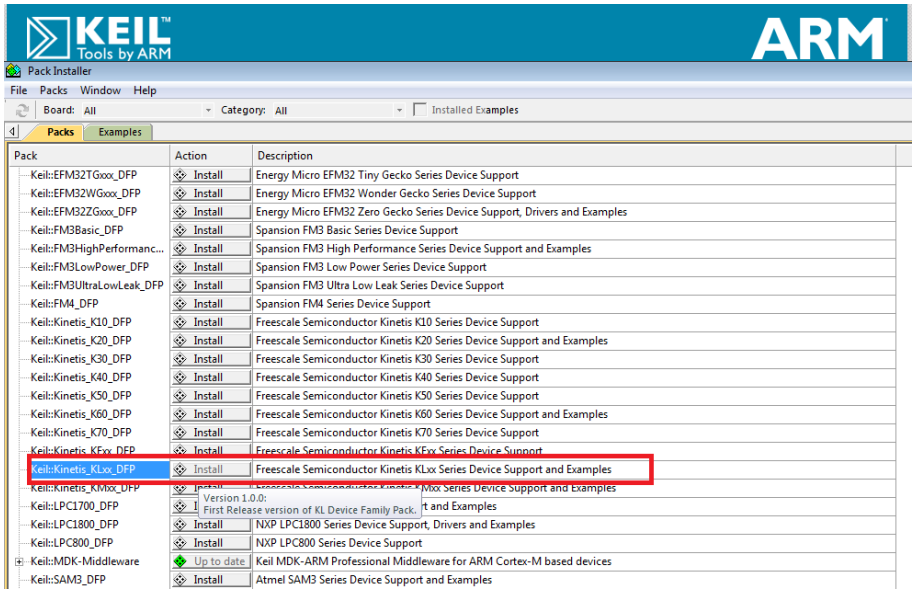


Figure 3: Pack Installer

After installing your desired libraries and Example package from Pack Installer window, you can then run the μVision program. You can find how to make a simple project on the μVision User's Guide

3 Step2: P&E OpenSDA Firmware

P&E OpenSDA Firmware provides the drivers for programming, debugging and communicating with FRDM-KL25Z platform. Without this Firmware you are not able to program your device. You can find the latest version of this software on the P&E micro website . You need to download and install the PEDrivers_install.exe under the Windows USB Drivers box.
The FRDM-KL25Z platform provides you an onboard debugging capability. On the other hand, OpenSDA provides you a UART communication channel with the onboard processor(KL25Z). So P&E OpenSDA Firmware will install not only the programming/debugging driver but also it will create OpenSDA-CDC serial port on your computer. This driver will add a virtual COM port on your computer when ever you attach the board to one of your USB ports.

4 Step3: Attaching FRDM-KL25Z board to PC

So far, we have installed the software development environment and P&E OpenSDA Firmware. At this point, we are all set to attach the board to the computer. Windows will automatically find and install all the proper driver for the board.

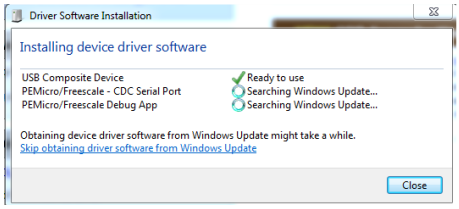


Figure 4: Driver software Installation

Since there is a pre-installed bootloader on the FRDM-KL25Z platform, after the first installation of the board, windows will detect FRDM-KL25Z as a storage device. So a new storage device will be added to your PC. You can drop your executable codes into this drive and OpenSDA will program the board. Since we are using the KEIL MDK5 as our programming environment, it will be more convenient to program the board using KEIL MDK5 as well. For this purpose, we need to change the pre-installed bootloader. First you need to unplug the board from the USB port and then while you are pressing the reset button re-plug the board to one of your USB ports. This time a Bootloader drive will appear on your computer.

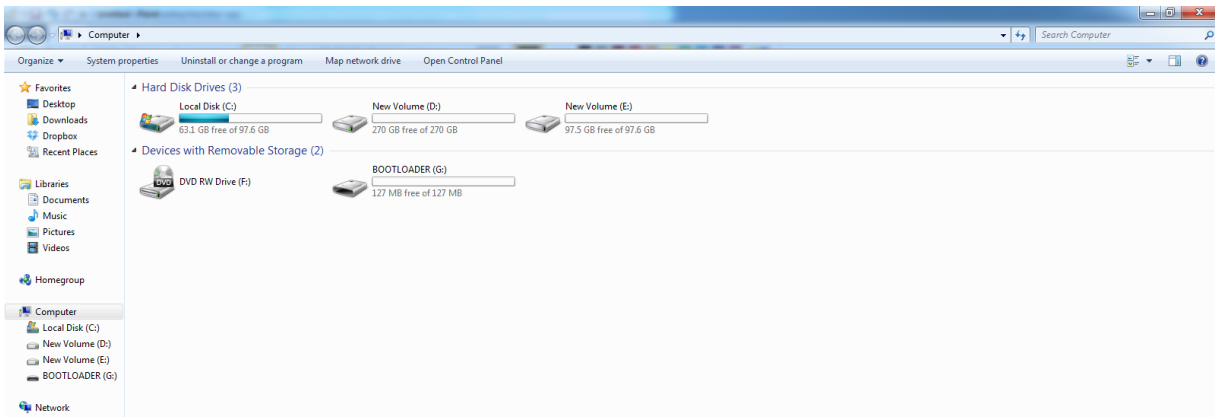


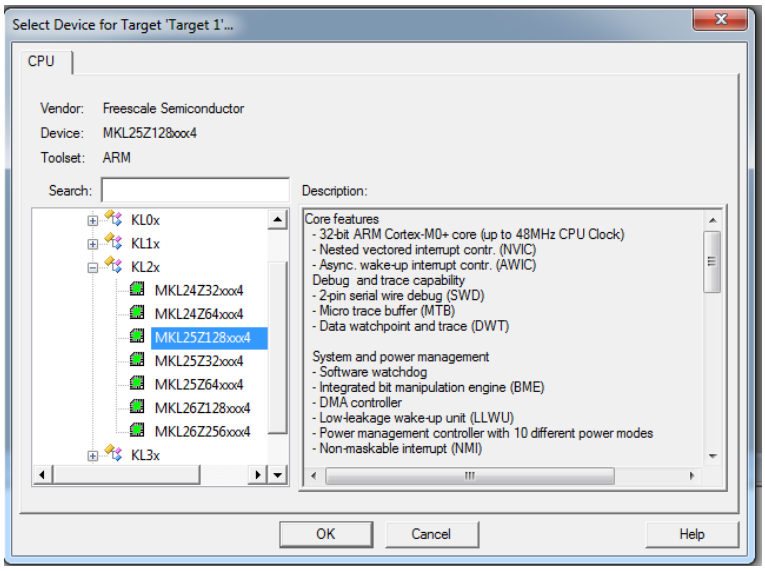
Figure 5: Driver software Installation

Now you just need to drag and drop the DEBUG-APP_Pemicro_v108.SDA on this folder so the next time you attach the board, you will be able to program it via KEIL MDK5.

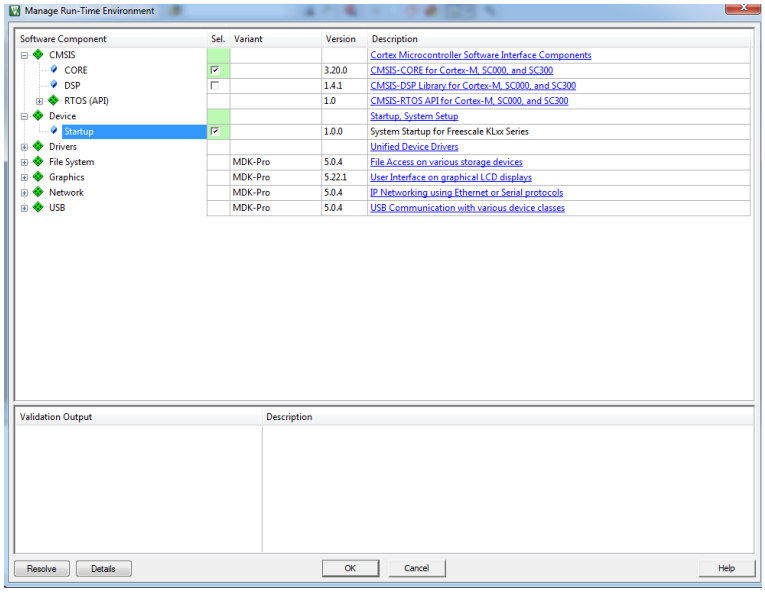
5 Step4: Blinky Project

5.1 Create a μVision Project

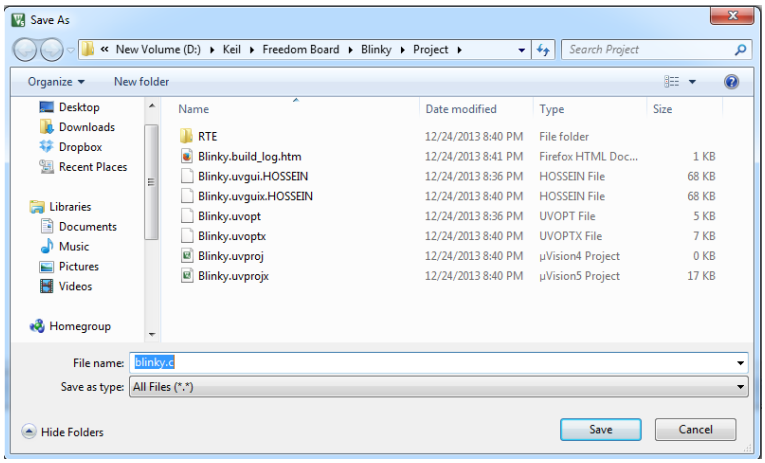
In this section you can find how to make a simple project in Keil-MDK. In Keil-MDK Select Project> New μVision project. Find MKL25ZL128xxx4 in the target device and click OK.



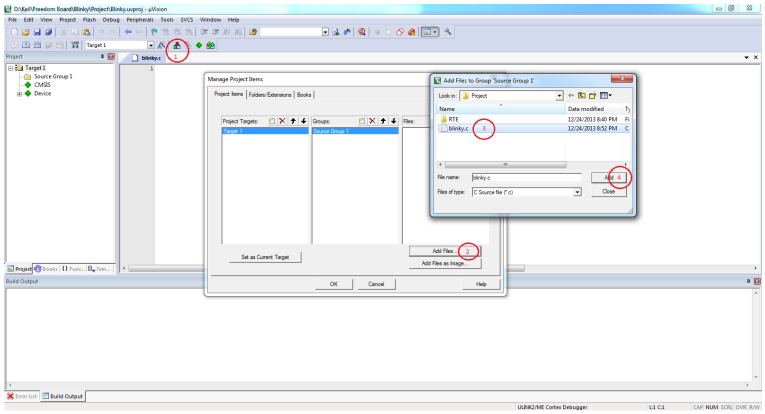
In the new window, check the startup option under the Device tab and Core option under the CMSIS tab and then select OK.



Now we need to write a C code to control the RGB LED on the board. Create a new file by selecting New under the File tab. Now save the file blinky with .C extension. Make sure that you have added the .c extension at the end of your file name.



This C file will be later compiled so we need to add it to our project:

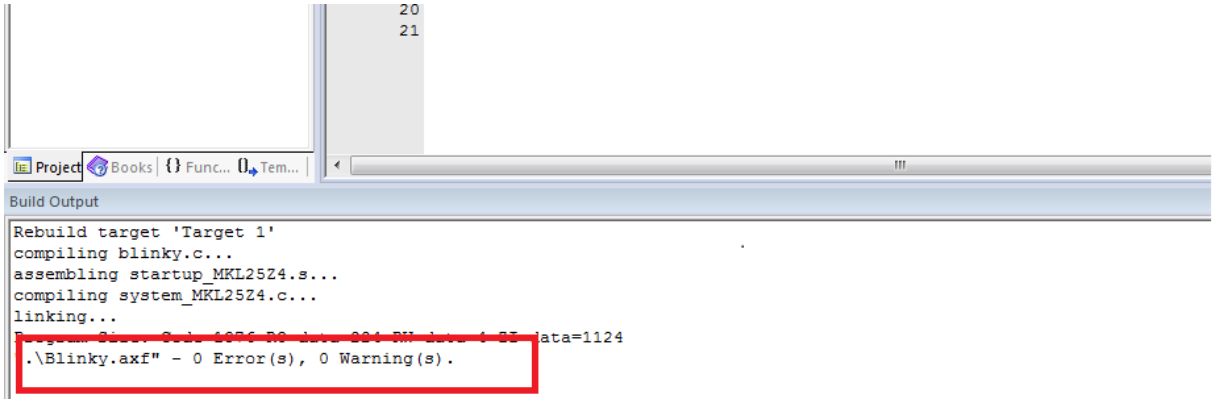


Now you should be able to find your C file under the Source Group 1 tab. Copy and paste the following code into blinky.c. Make sure that you have Delay.h file in your project folder.

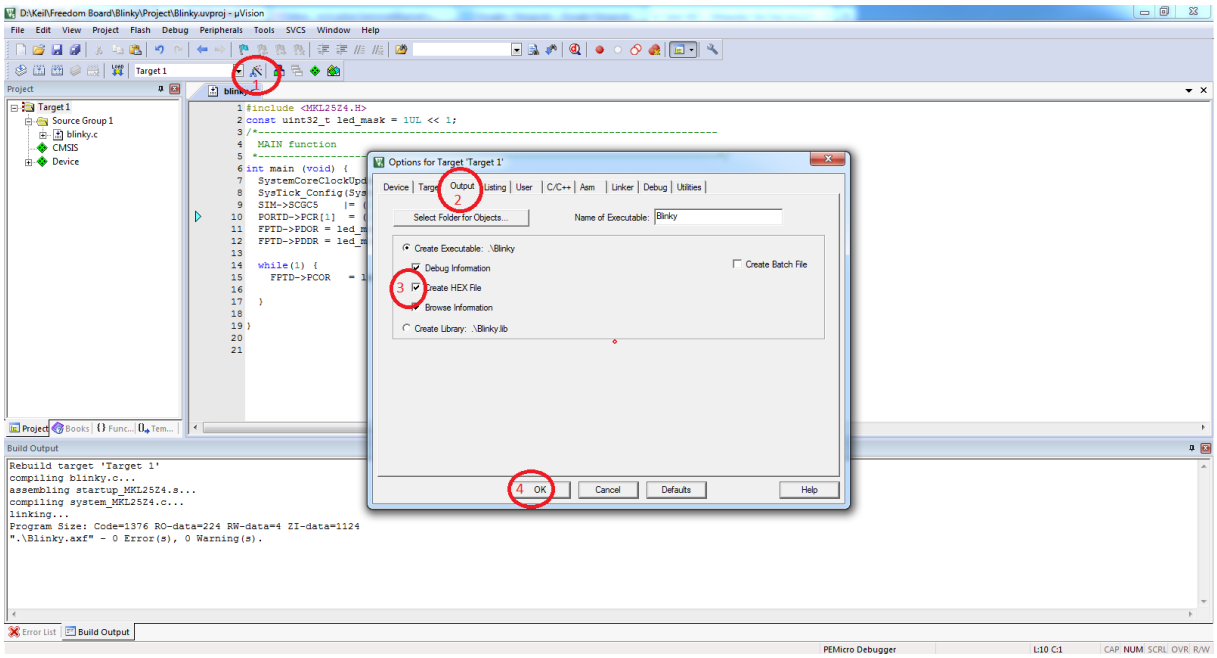
```
#include <MKL25Z4.H>
#include "Delay.h"
const uint32_t led_mask = 1UL << 1;
/*----------------------------------------------------------------------------
  MAIN function
 *----------------------------------------------------------------------------*/
int main (void) {
  SystemCoreClockUpdate();                  /* Get Core Clock Frequency */
  SysTick_Config(SystemCoreClock/1000);     /* Generate interrupt each 1 ms */
  SIM->SCGC5   |= (1UL << 12);      /* Enable Clock to Port D */
  PORTD->PCR[1] = (1UL << 8);                   /* Pin PTD1 is GPIO */
  FPTD->PDOR = led_mask;          /* switch Blue LED off */
  FPTD->PDDR = led_mask;          /* enable PTD1 as Output */

  while(1) {
                FPTD->PCOR  = led_mask;
    Delay(500);
                FPTD->PDOR  = led_mask;
    Delay(500);
  }
}
```

5.2 Compiling and Programming

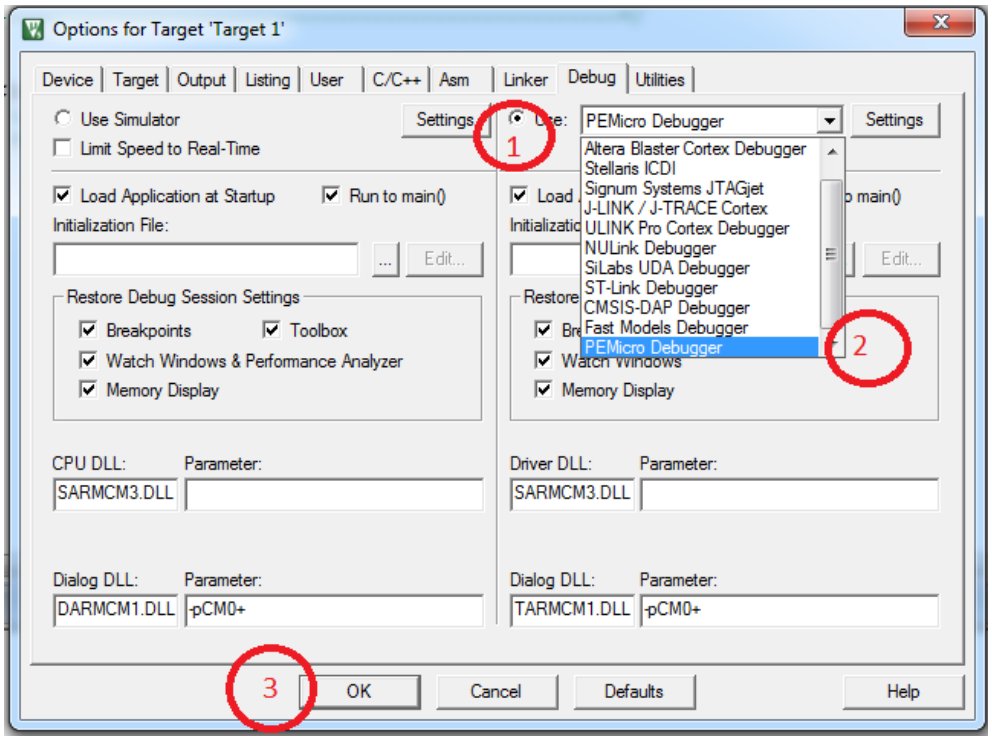
In order to compile your code, simply click on ”Rebuild all the target files” under the Project Tab. You should be able to compile the code without any errors or warnings.



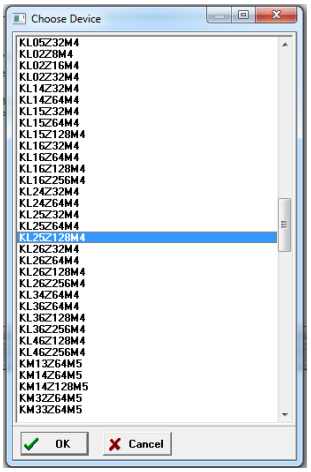
This C code still cannot be run on the microcontroller. We need to generate the Intel HEX file out of this code:



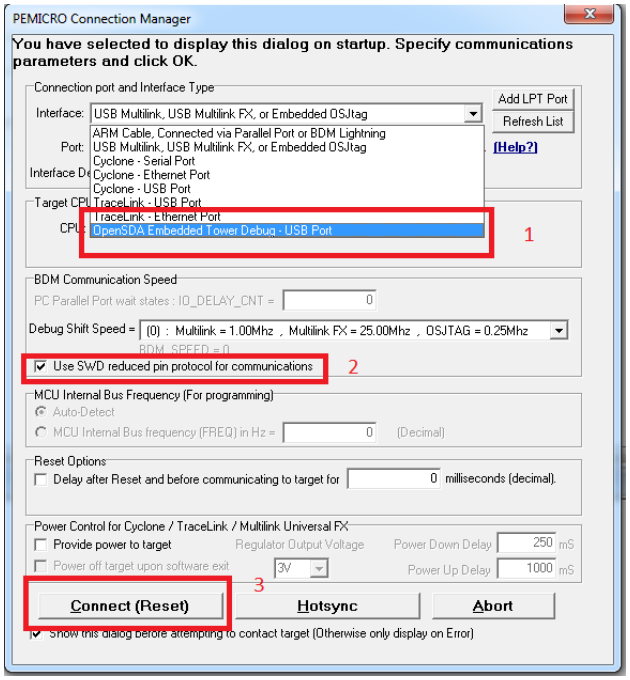
We need to set the OpenSDA as the programming and debugging tool. In the same window under the Debug tab select PEMicro Debugger and then select OK.



Now we just need to program the board with the generated hex file. Under the Flash tab, select Download option. You need to specify the target device that you are going to program. Select KL25Z128M4 from the "Choose Device" window and then click OK.

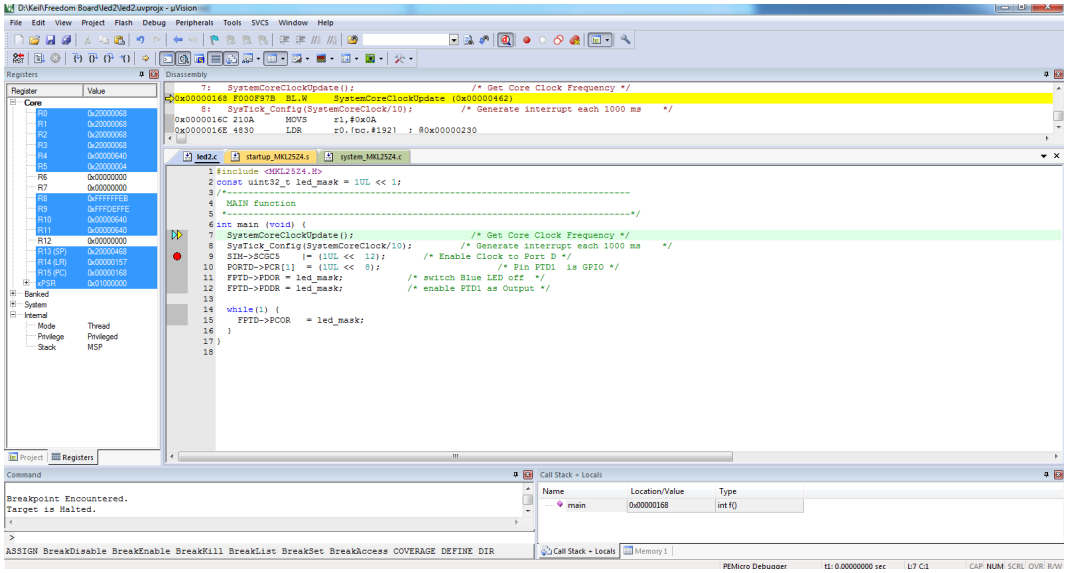


On the opened window (PEMICRO Connection Manager) make sure that you have selected ”OpenSDA Embedded Tower Debug USB-Port” from the Interface box. Also make sure that you have selected ”Use SWD reduced pin protocol for communications” and then hit the Connect button in order to program the board. Press reset button on the board to let the program execute.



5.3 Debugging

To debug your code while it is executing on the board press Ctrl+F5 or selet start/stop under the Debug tab.



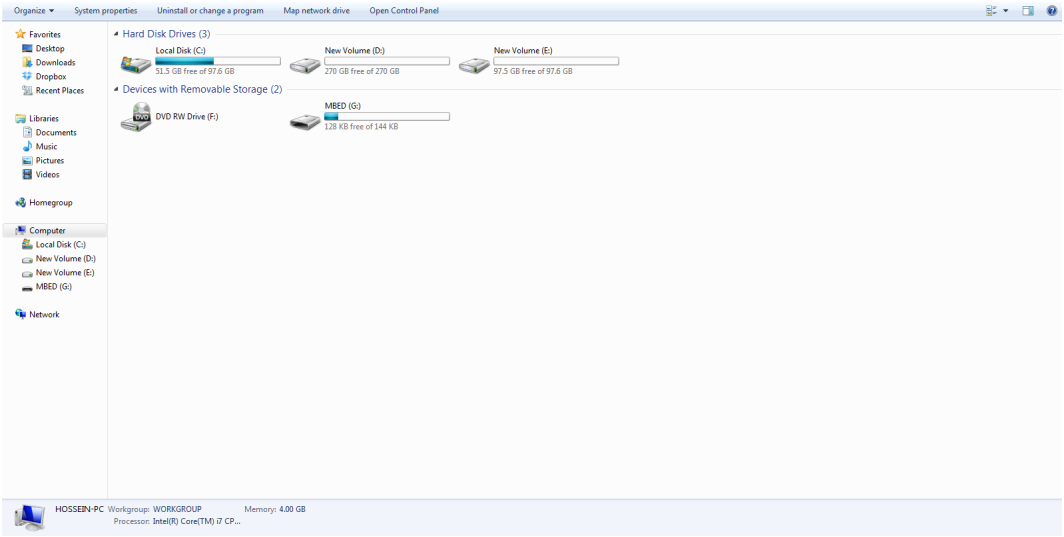
Use F11 and F5 to execute the program step by step. This program will blink LED

6 mbed development platform:

mbed development platform is an online development kit which supports many ARM based platforms from different vendors. Fortunately, FRDM is one of the well supported platform by mbed. Before going to the next section, make sure that you have created an account on the mbed website .

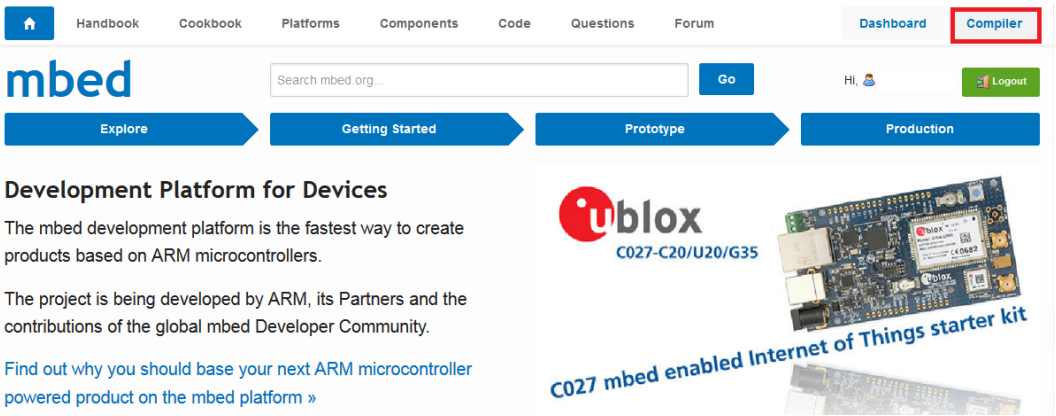
6.1 Getting started:

To get started, you need to re-program the BOOTLOADER of the microcontroller in order to be able to program the board using the .bin files. You can download this BOOTLOADER from here. Using this new BOOTLOADER, re-boot the board according to the section four of this document. You can find more information on how to re-boot the board in here. If you re-boot the board properly, you should be able to see the ”MBED” Folder on your computer after you re-plugged the board.

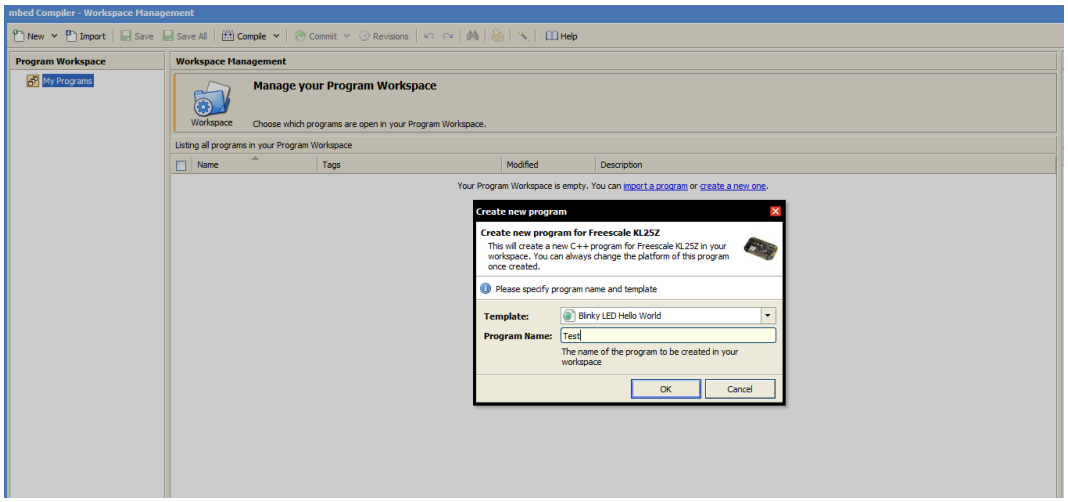


6.2 Make an mbed project:

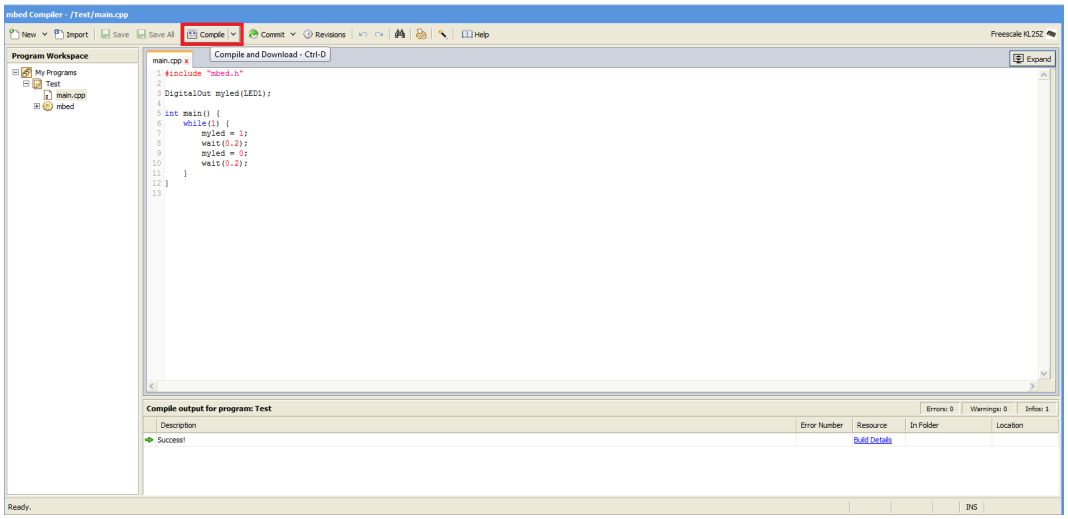
Once you log into your mbed account, click on the Compiler button in order to use the mbed Compiler:



In the new window, right click on ”My Programs” and select ”New Program”. In the ”Create New Program” window, make sure that you have selected the ”Blinky LED Hello World” option from the template menu. Provide a name on the Program Name box and click on OK:



Since you have selected a ”Blinky LED Hello World” template for your project, the main code is already generated by the program and you don’t need to add anything to the project. On the left hand side of the compiler, double click on the main.cpp file and review the generated code. To generate a .bin file from this project, you need to compile the code. Click on the ”Compile” button:



The software will compile your code and if the compilation was successful, it will provide you a link to download the .bin file. Save the .bin file on the ”MBED” folder on your computer and press the reset button on the board to allow the execution of your program on the microcontroller.

7 You can find more information on creating an mbed program in here.

8 Useful Links:

- Creating a μvision Projects
- MDK-ARM5
- FRDM-KL25Z Product Page
- MDK5 Software Packs
- FRDM-KL25Z User's Manual