BBM 434 – Embedded Systems Lab
Lab 1 Instructions

Based on Lab 2 of the EDX course UT.6.01x Embedded Systems - Shape the World.

Lab Preparation

Watch the introduction to the board:

https://www.youtube.com/watch?v=SRr6-6Bwb00

Installation of the required software
In this lab, you will first need to install the required software and drivers.

Step 1) Installing Keil µVision for the ARM, MDK-Lite (32KB) Edition

We will be using Keil uVision IDE, and writing software in C. These programs only run on Windows (XP, Vista, 7, 8 or 10). There is a way to install software on a Macintosh so Windows and its applications will run on the Macintosh, see running Keil on a Macintosh. There are no solutions for Linux (note that if you are a Linux user you may use Texas Instruments Code Composer Studio™ (CCStudio) to develop software for Texax Instruments microcontrollers, but in this course we will be using Keil uVision). The Keil application does not require a serial number or license key.
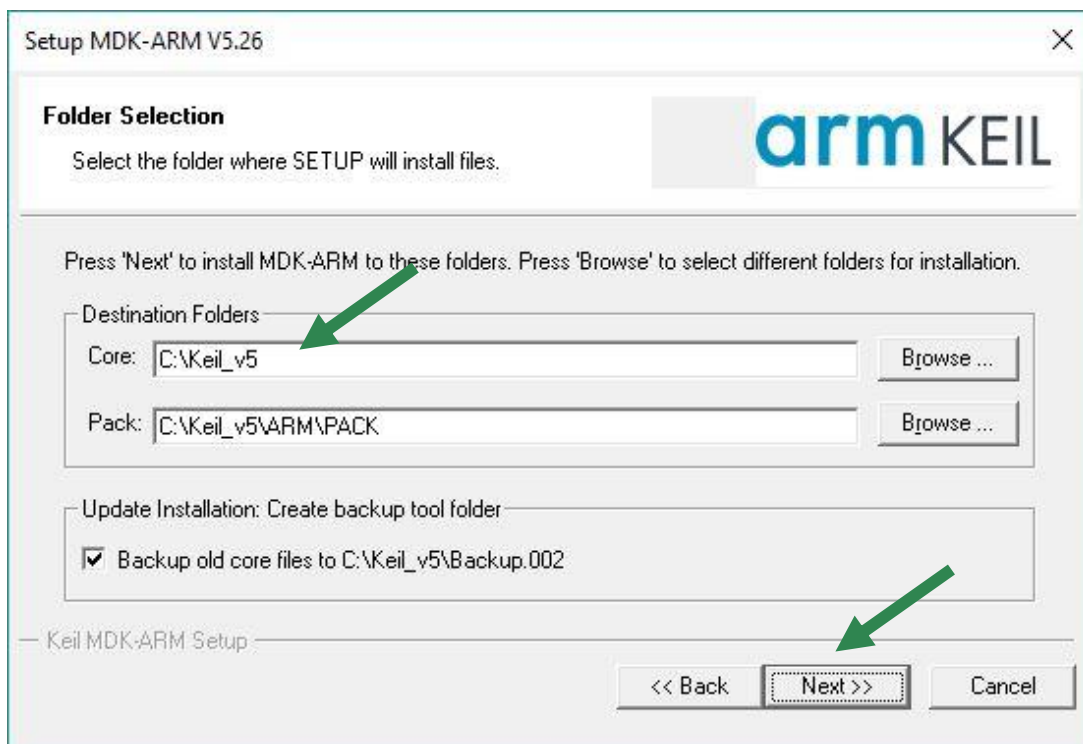
1) Go to https://www.keil.com/demo/eval/arm.htm and download mdk526.exe and download this 894MB file to your computer (or download it from https://web.cs.hacettepe.edu.tr/~bbm432/files/mdk526.exe).

3) Execute the mdk526.exe file, installing the application to C:\Keil_v5.
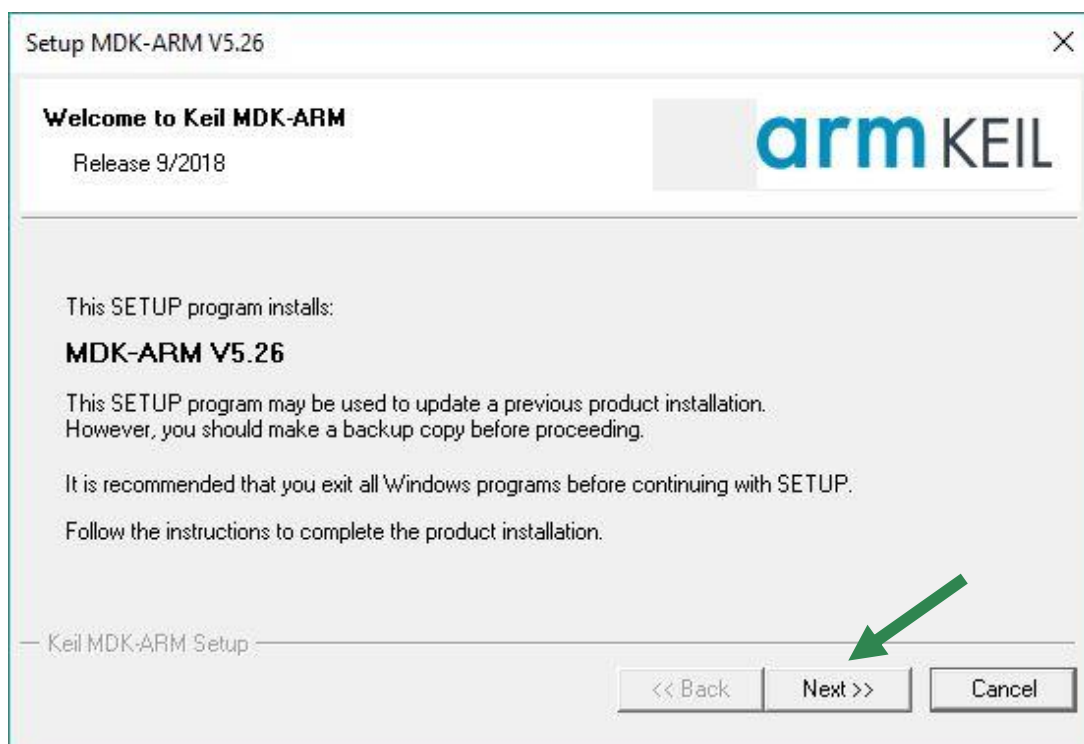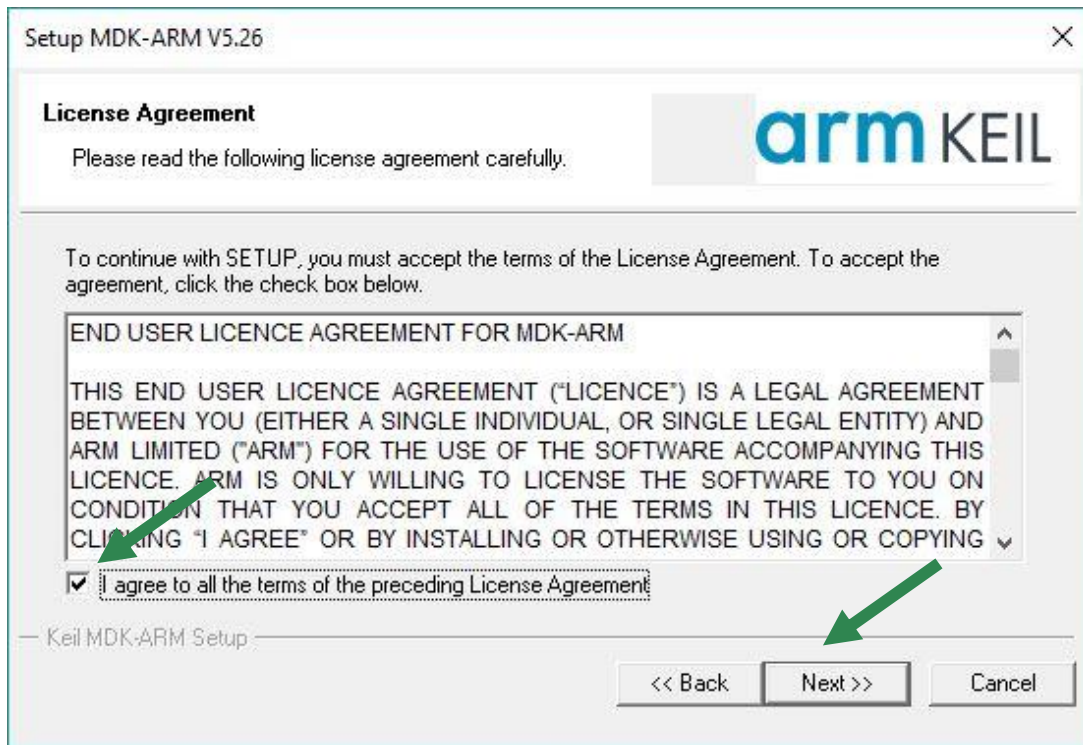
Install Keil to C:\Keil_v5, and click **Next**.

Update these fields with your correct information, and click **Next**.

Wait while it installs.

If prompted, select **Add example Projects to the recently used project list**, and click **Finish**.
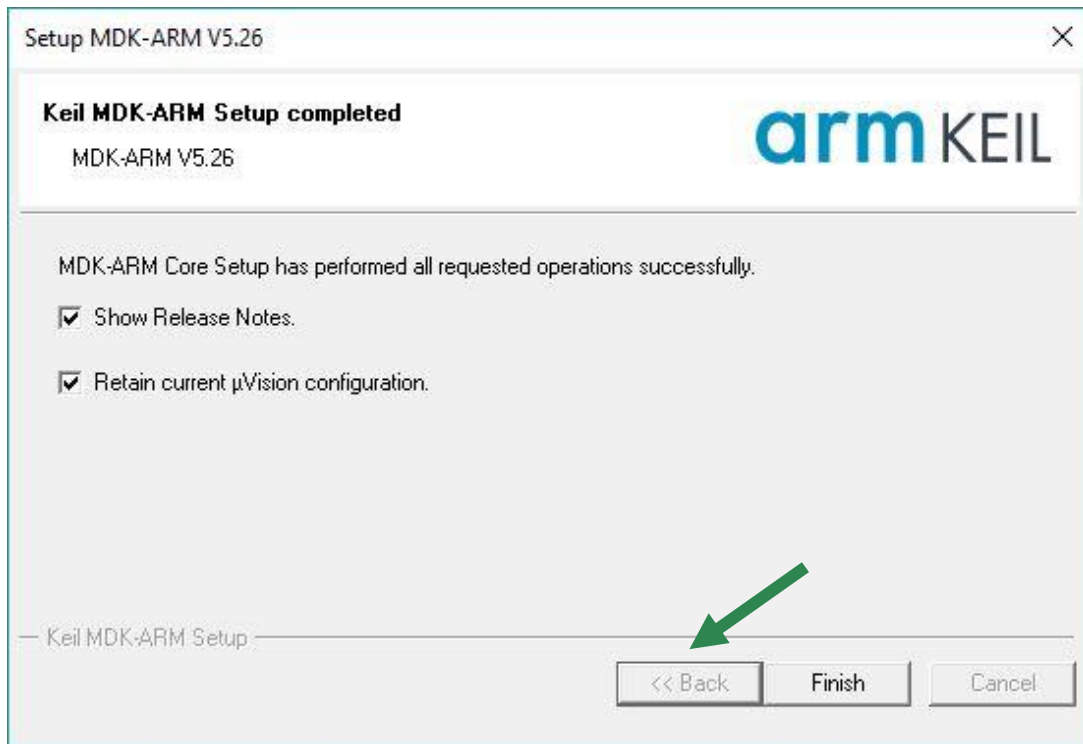
Step 2) Installing Windows drivers for the LaunchPad board

Once Keil is installed you will need to install the windows drivers for the Texas Instruments microcontroller board. The process is the same for both the LM4F120 and the TM4C123 LaunchPads. **You will need the actual physical LaunchPad board to complete this step. So if you have ordered your board and it hasn't yet arrived, come back to this step once your board arrives**.

Download the current drivers from:
http://www.ti.com/tool/stellaris_icdi_drivers

Plug in the LaunchPad into a USB port. Pick the USB port you will most likely want to use. Plug the mini-USB into the socket labelled **Debug**. Make sure the LaunchPad power is on. There is a slider switch near the USB that should be in the **Debug** position. The green LED near the USB will illuminate when power is applied. Open the device manager program (click **Start**, and then in the **Search for files and programs** window type **device manager**) and you should see three In-Circuit Debug Interface devices that do not have drivers (yellow exclamation point).
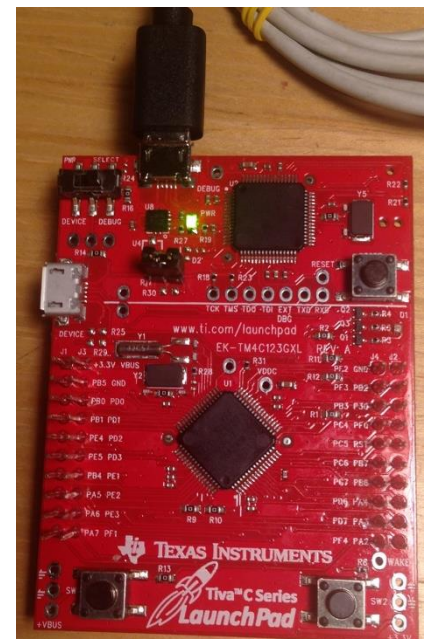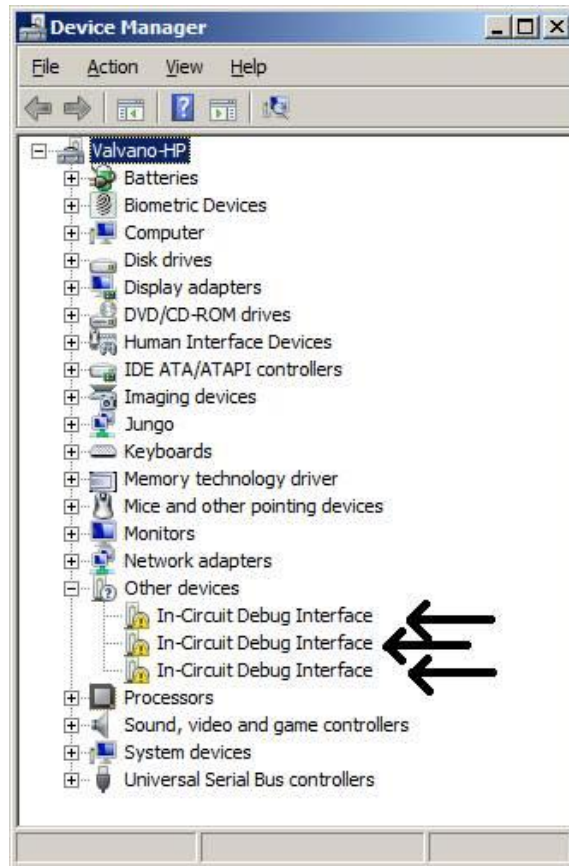
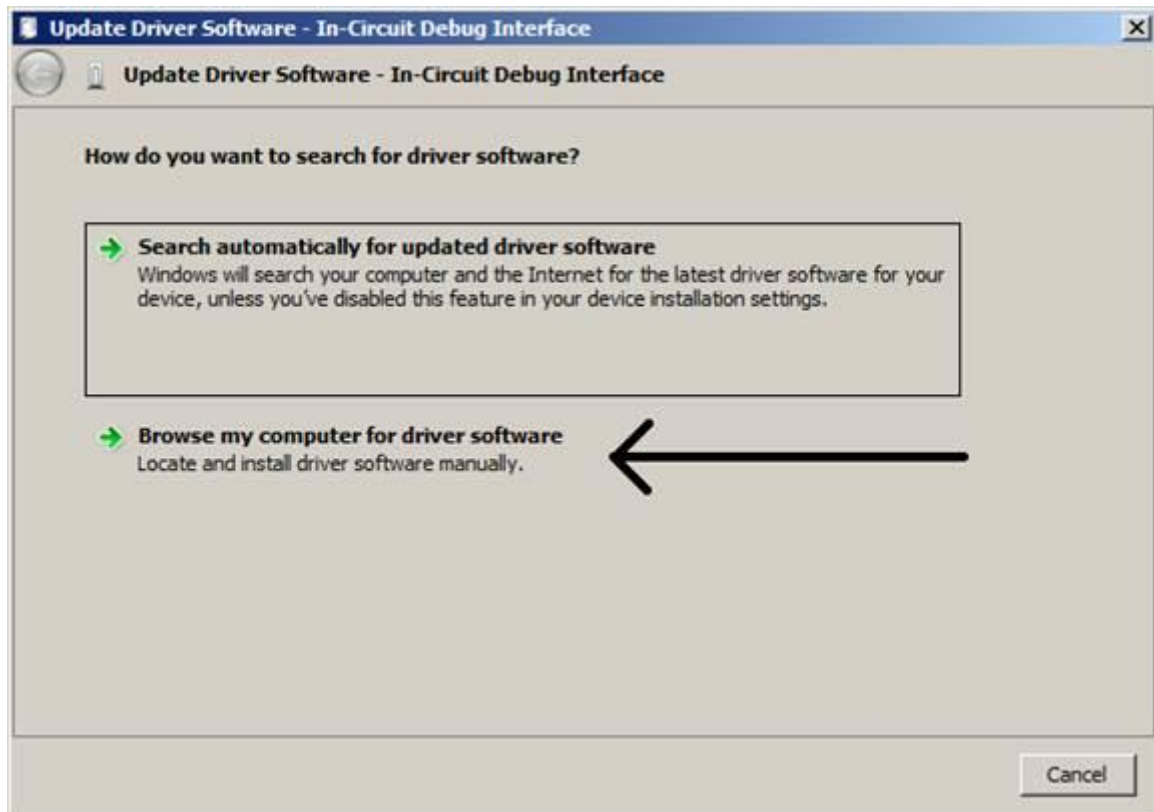For each device, first right-click on the device and then select **Update Driver Software.**
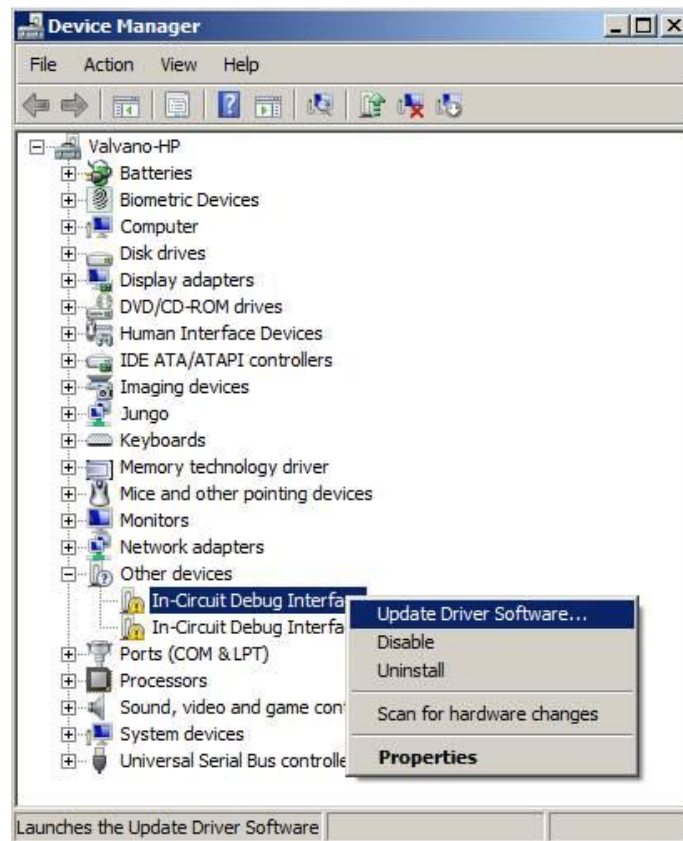
You will be given a choice to "Search automatically..." or to "Browse my computer...". Click **Browse my computer for driver software**.

Browse your computer and find the location of the windows drivers you downloaded (you may get a security warning about Windows not being able to verify the publisher. If so, click **Install this driver anyway**).
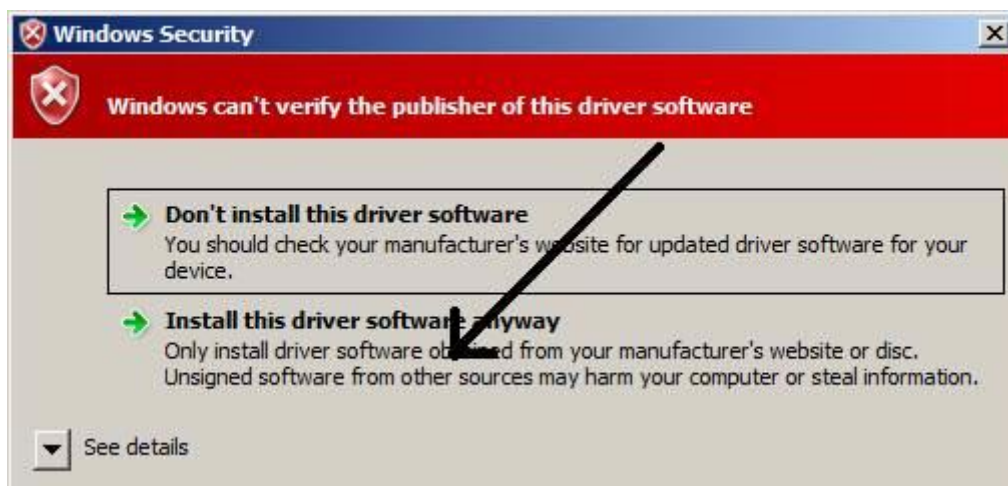
Repeat these steps for all devices with yellow exclamation points. After you have successfully installed the drivers, there should be no yellow exclamation points. However, if you do see one of Stellaris devices with a driver error, try right-clicking on the device with error and select **Update driver software**, and try the process again. Sometimes it doesn't install on the first try, but does install on the second try.
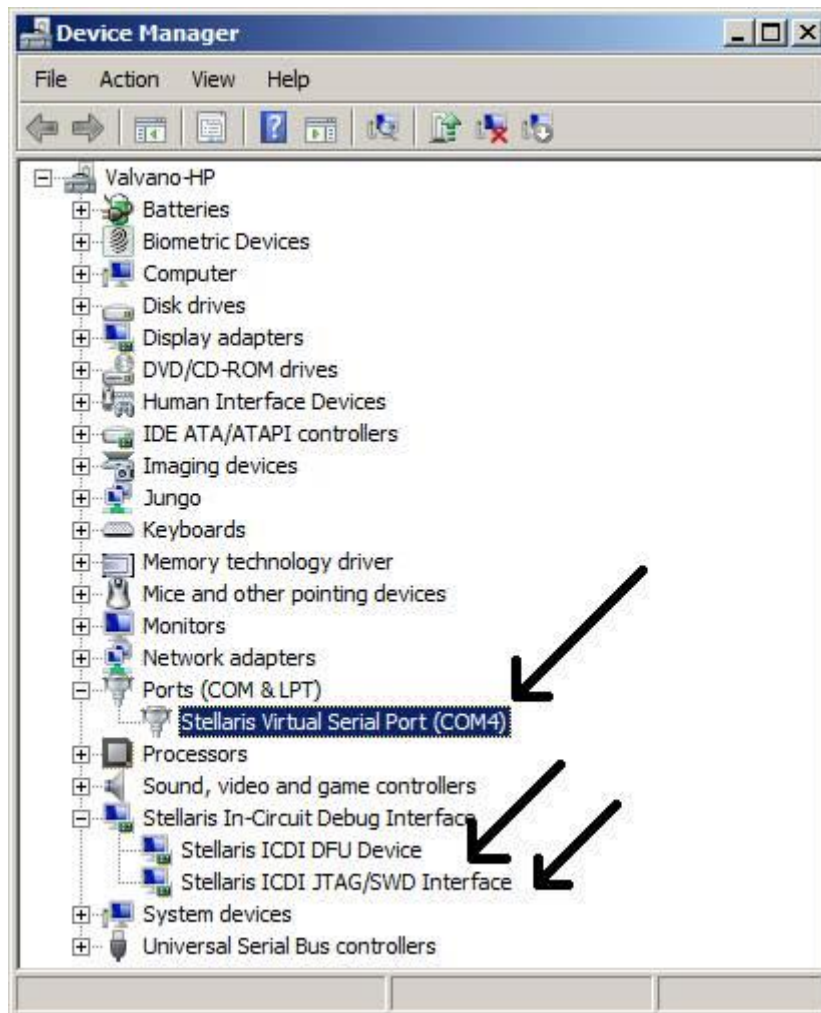
You may also check the Driver Installation Instructions for Stellaris® In-Circuit Debug Interface (ICDI) and Virtual COM Port for more details at http://www.ti.com/lit/ml/spmu287c/spmu287c.pdf

Lab Experiment Instructions

Part 1) Run the lab in the simulator

For each lab we will first design, develop, and debug in simulation. Since Lab 1 does not require you to build any hardware or software, all you will do here is run the example program in simulation mode. This step also verifies you have properly installed **Keil uVision**. Verify that you have the file LaunchPadDLL.dll TExaS simulator is in your **\Keil_v5\ARM\BIN** folder.

You will find the example files and starter codes for lab 1 in the **434_Lab1_Toggle_PF1** folder. Download from here: https://web.cs.hacettepe.edu.tr/~bbm432/files/434_Lab1_Toggle_PF1.zip

1) Open the Lab1 project file **C2_Toggle_PF1.uvproj** in the **434_Lab1_Toggle_PF1** directory.

2) Verify it is configured to run in the simulator (see Figure below).
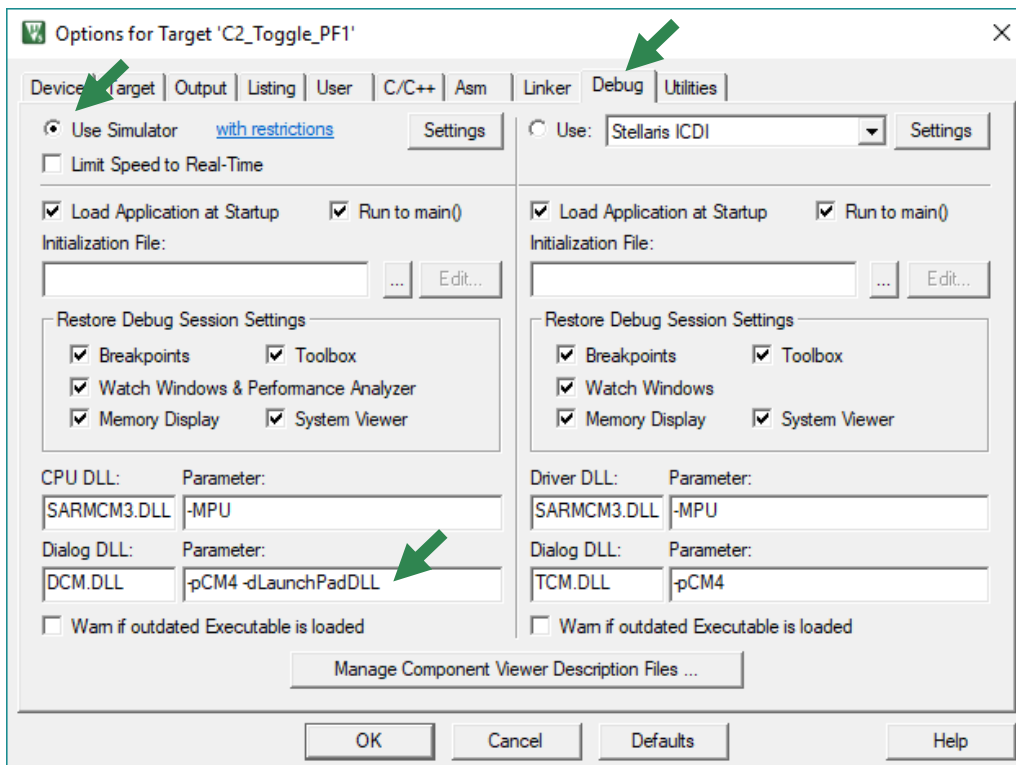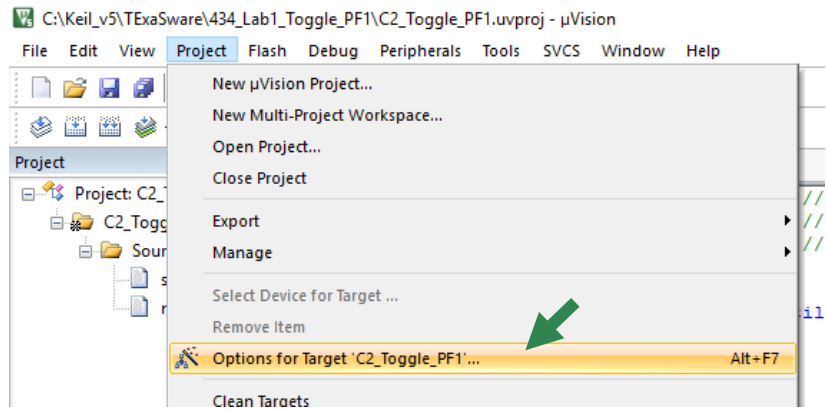
3) Build the project by executing **Project->Build Target**.

4) Start the debugger in simulation mode.
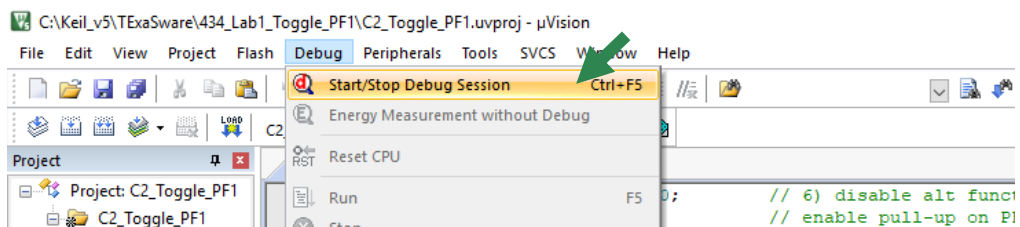
5) Run the program and notice the LED outputs.

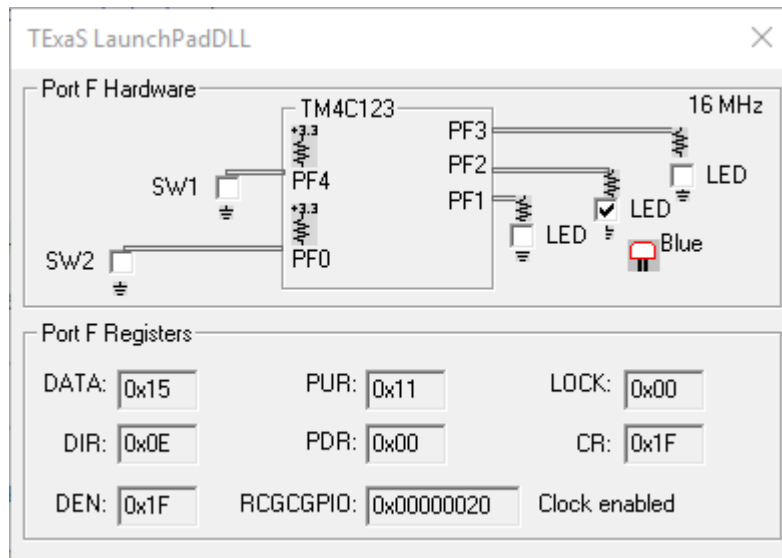If you do not see the window with the Port F input/output, execute **Peripherals->TExaSPortF.** Another trick to reset all the windows in their default position is to execute **Window->SetViewToDefault**.

Part 2) Run the lab on the LaunchPad

After you have completed the lab in simulation mode, you will switch over and debug it on the real board. You will simply run the same project on the actual LaunchPad.

1) In Keil, open **Project->OptionsForTarget**. Under the Device tab, search for your board and select it (**TM4C123GH6PM**). In the **Target** tab, select **TExaS** in the **Operating System** drop-down menu, and for the option **Floating Point Hardware** select **Not Used**. In the **Debug** tab, click on the **Use button on the right** (instead of Use Simulator) and select the **Stellaris ICDI**. Under the **Utilities** tab, click on **Use Target Driver for Flash Programming** and select **Stellaris ICDI**.
2) Compile the project by executing **Project->Rebuild all target files**
3) Download the object code into Flash EEPROM by executing **Flash->Download**
4) Start the debugger by executing **Debug->Start/Stop Debug Session**
5) Run your program executing by **Debug->Run**. At this point you can watch the LEDs on your board.

Part 3) Experiment with the code

Open the file **main.c** and have a look at the C code in the Keil IDE. Discuss the following:
1) Which part of the code turns on and off the LEDs?
2) What is the purpose of Delay() function?
3) How can you make the LED flash slower? Which part of the code needs to change and how?
4) From what you see in the code, how can you make the LED flash just one color, e.g. green?

You will be graded for both written report and lab demonstration performance.

How to Submit the Written Report

The deadline for the submission is **Friday 15 March 2019 until 13:00** (no further extensions will be allowed!). Submit your work through http://submit.cs.hacettepe.edu.tr/ (submission via e-mail will not be accepted!) in the following format:

- **b<studentID>.zip**
  - **report.pdf**