

Stellaris® ARM® Cortex™ M4 Training

Low Power

Section 2: Hands-on Lab

Stellaris® ARM® Cortex™ M4 Training Low Power

Low Power Modes & Hibernation Module

**Section 2 – Getting started with
Hibernation Module using EK-LM4F232**



1



Agenda

1. Lab Objectives	3
2. Setting Up Hardware	4
Connect your Evaluation Kit	4
3. Familiarizing with Evaluation Kit	5
Run Quick Start Application	5
4. Setting Up Software	6
a. Downloading CCS	6
b. Downloading LM4F Support Package	6
c. Downloading Drivers for ICDI Debug Port	6
5. Installing Drivers for Evaluation Kit	7
6. Example Hibernation Project	8
7. Getting Started with a Hibernation Project	9
a. Launching CCS	10
b. Importing a project	13
c. Building a project	16
d. Debugging a project	17
8. Source Code Explanation	19

Lab Objectives

Lab Objectives

- Gain familiarity with EK-LM4F232 evaluation board.
- Configure the device to enter in hibernation mode and wake-up from hibernation mode upon $\overline{\text{WAKE}}$ pin assertion.

3



Setting Up Hardware

Setting Up Hardware

- You will need: a computer with at least one USB port, a USB cable (type A-male to type B-male) and an EK-LM4F232 evaluation board.
- Connect the evaluation board to the computer using USB cable.



Connecting EK-LM4F232 to Your Computer

- Make sure that the *Power Select* jumper is connected at */CDI* position. The board will power up and a pre-programmed quick start application will launch.



Power Select Jumper Settings 4



Familiarizing with Evaluation Kit

Familiarizing with EK



- When the quick start application begins to execute, you will see the following on the screen.



Screen Shots from Quick Start Application

- Press ▲, ►, ▼, ◀ and SELECT/WAKE buttons to navigate through menus.
- Board Schematics: www.ti.com/ek-lm4f232

5



Note: Some weblinks may not be functional until the product launch.

Setting Up Software

Setting Up Software



- 1) You should have already downloaded & installed CCS v4.2.4.00033 & LM4F support package from the FTP site.
- 2) You should have already downloaded the drivers for EK-LM4F232 from the FTP website. Now we will install them on your computer.
- 3) You should already have the documentation from the FTP site on how to install the drivers.
 - 1) *Stellaris Virtual Serial Port*
 - 2) *Stellaris ICDI/SWD Drivers*
 - 3) *Stellaris DFU Drivers*

6



Code Composer Studio installation can take long time depending upon the components selected. It is recommended that you install the software before the workshop.

See FTP site for pre-workshop action items (Pre_workshop_actions.pdf)

FTP site <ftp://ftp.ti.com/pub/stellaris/>

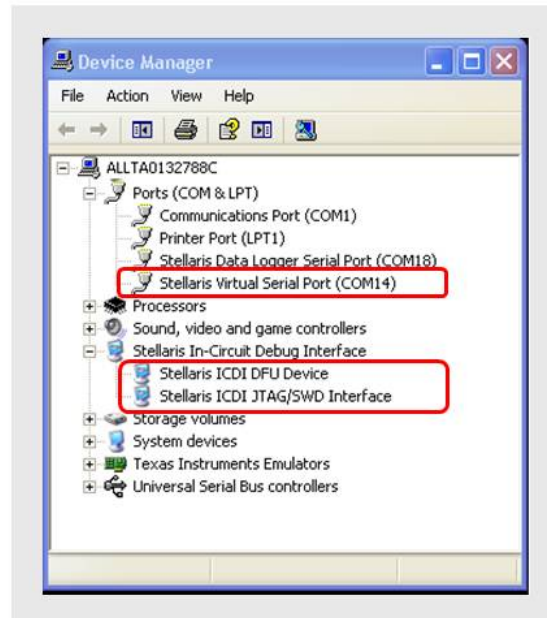
Installing Drivers for Evaluation Kit

Installing Drivers for EK

- When EK-LM4F232 is connected to the computer for the first time, *Windows New Hardware Found* wizard will appear.
- The wizard should automatically locate and install the required drivers for you.

If that does not happen, point the wizard to the location where the drivers are located.

- Upon successful installation, the drivers will be listed in *Windows® Device Manager*.



Stellaris Drivers listed in Device Manager

7



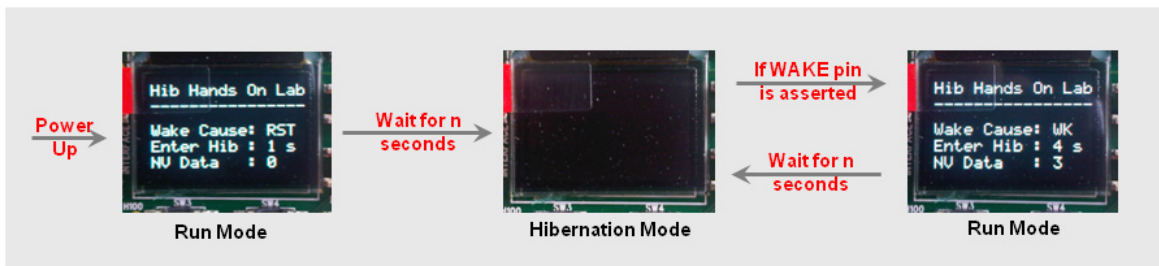
Before advancing to the next steps, it is important make sure the that the computer can recognize the evaluation board and all the required drivers are installed.

See document [how_to_install_drivers.pdf](#) for more details.

Example Hibernation Project

Example Hibernation Project

- After building & downloading an example Hibernation project to the evaluation board, the following should appear on the screen.



Screen Shots: Example Hibernation Project

8

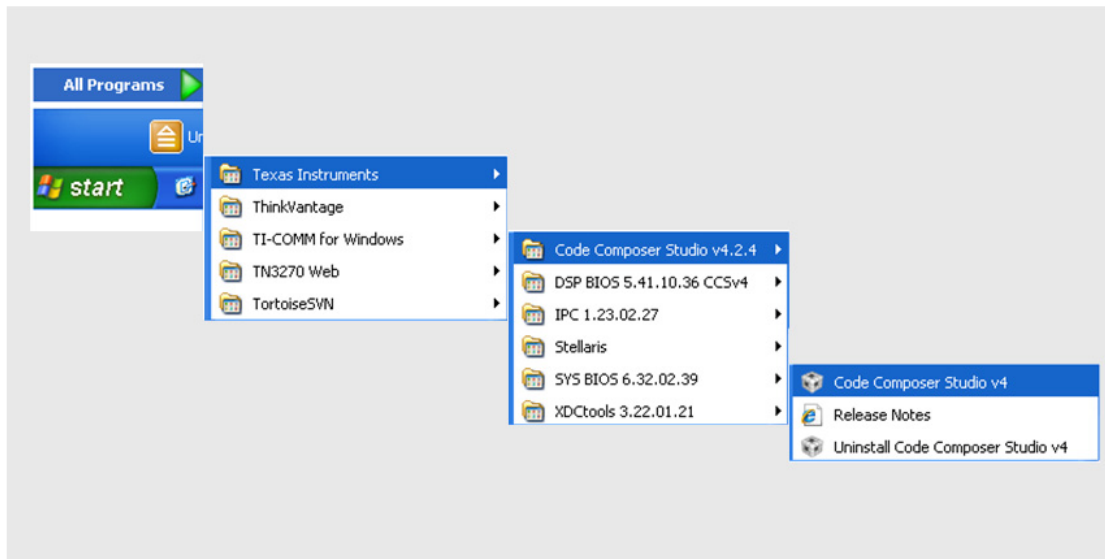


In the remainder of section 2, we will take an existing project that uses hibernation module, build it using CCS and program it to the EK-LM4F232 evaluation board. Subsequently, in Section 3 we will work on exercises where we will modify this project to learn several advanced concepts.

Getting Started with a Hibernation Project

Getting Started with a Hibernation Project

- Launch CCS IDE from Windows Start Menu.



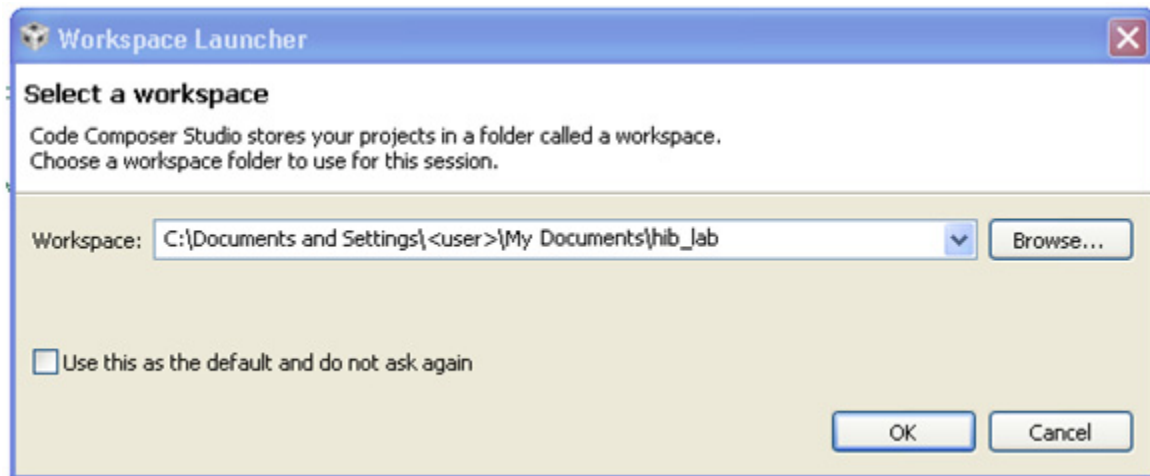
Launching CCS from Windows Start Menu

9




Launching CCS

1. When the “Select a workspace” dialog appears, specify the location for a new workspace. Do not check the “Use this as the default and do not ask again” checkbox. Click on “OK” button.



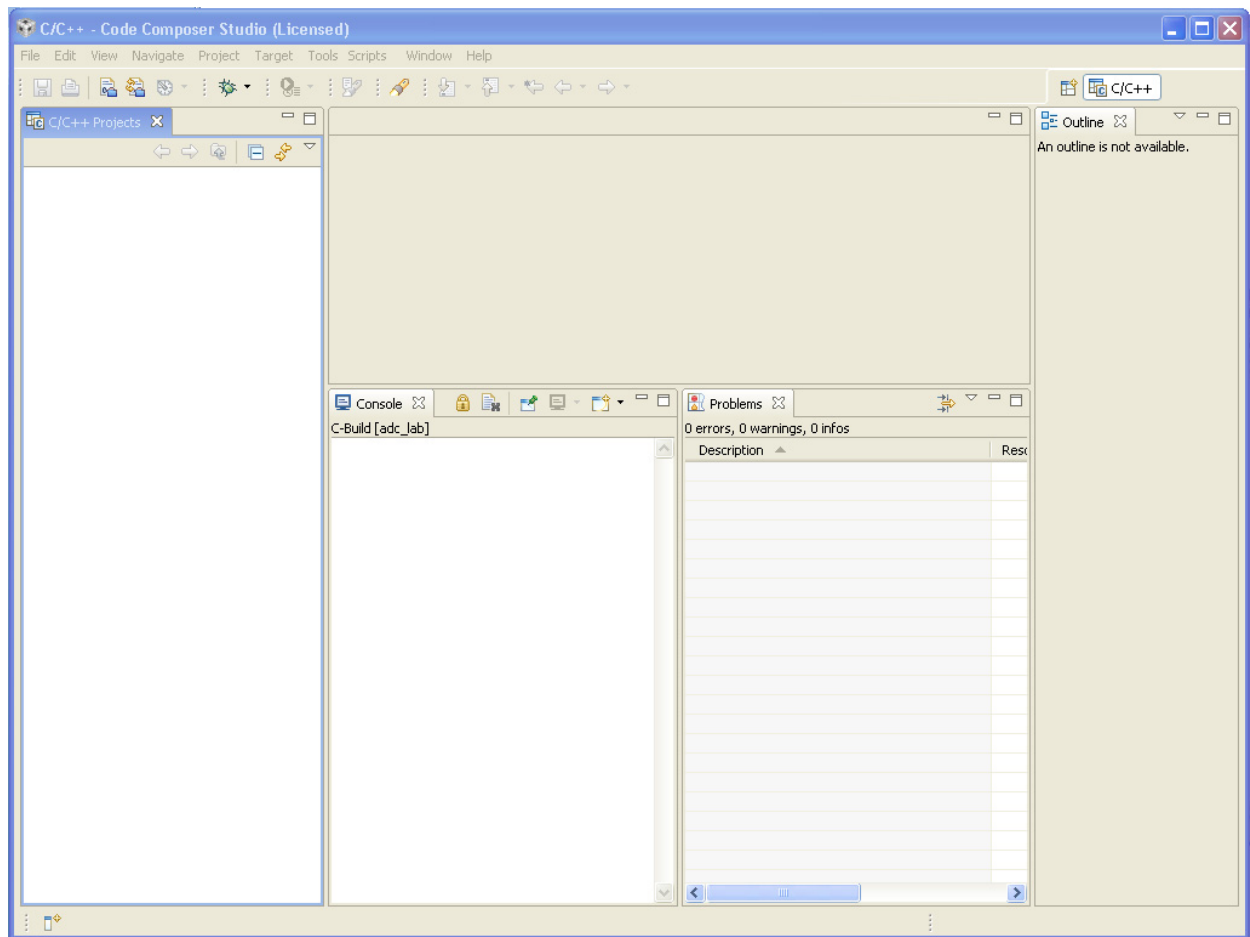
2. The following screen will appear. CCS takes a while to launch.



3. When the following screen appears, click on the  icon to start using CCS.

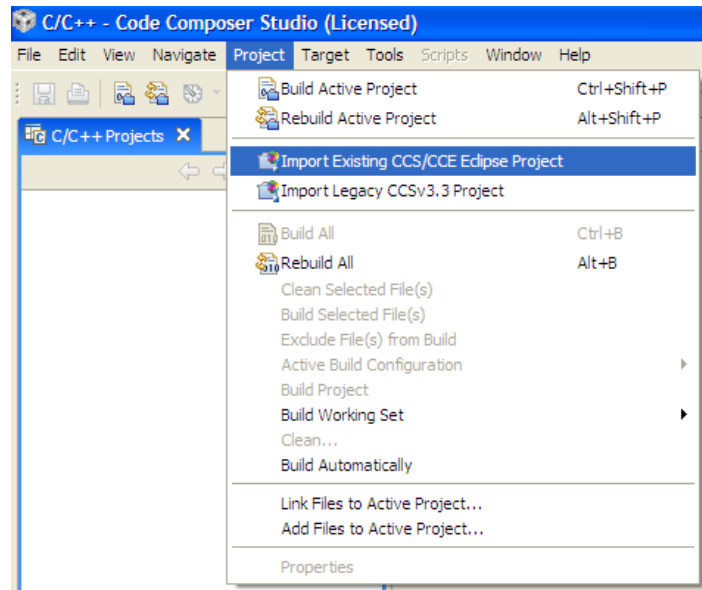


4. CCS main screen will appear as follows.

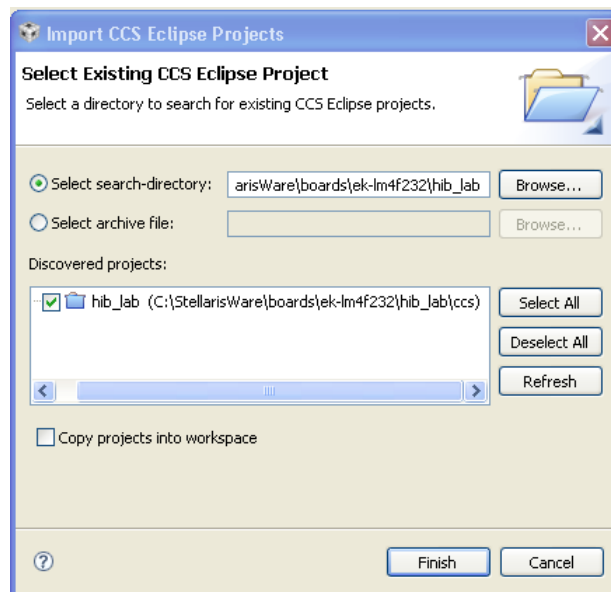


Importing a Project

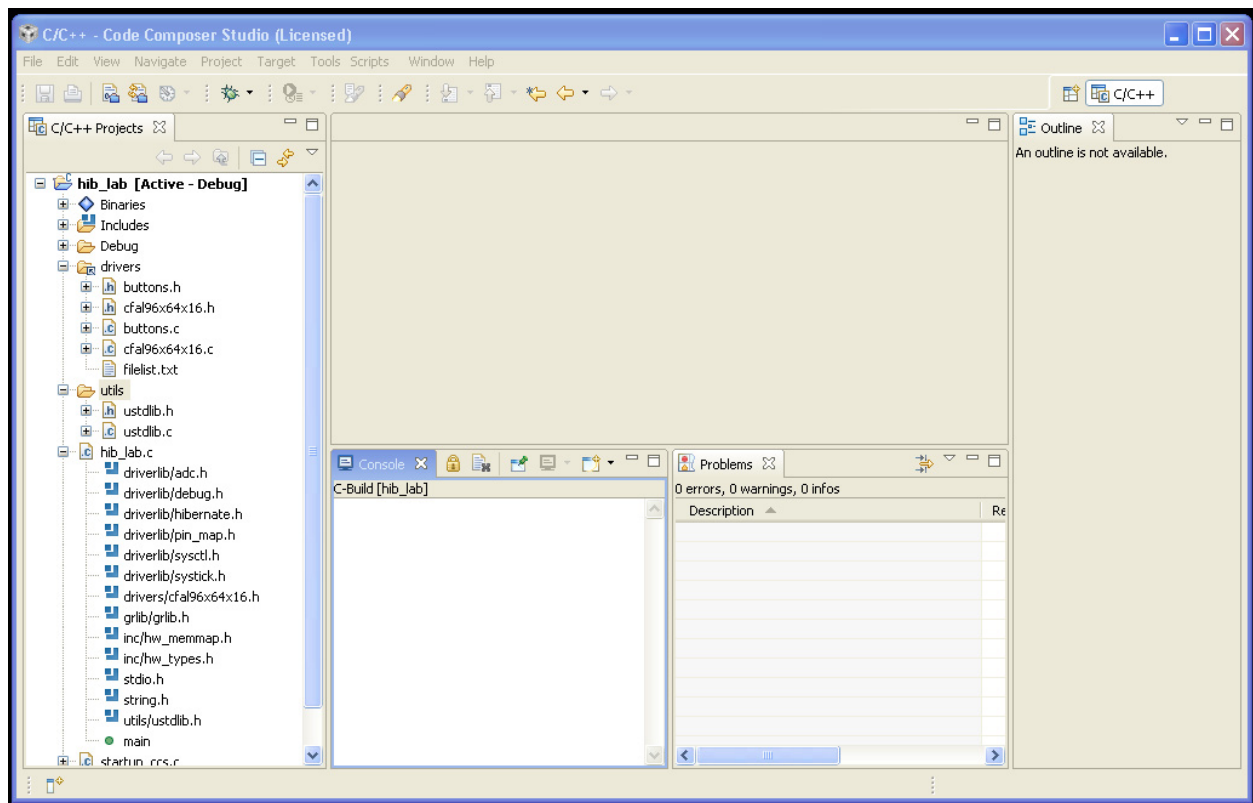
5. Click on “Project” menu, and select “Import Existing CCS/CCE Eclipse Project” option” as shown below.



6. The “Import” dialog box appears. Browse to the directory “C:\StellarisWare\boards\ek-lm4f232\hib_lab” where “hib_lab” project is located. Subsequently, click on “Finish.”



7. The project will load in CCS as shown below.

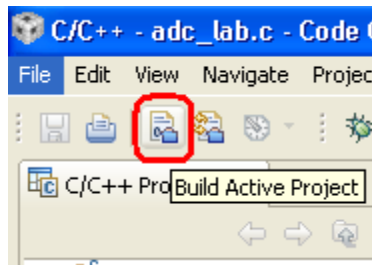


8. Double-click on “hib_lab.c” file to view the source code as shown below:

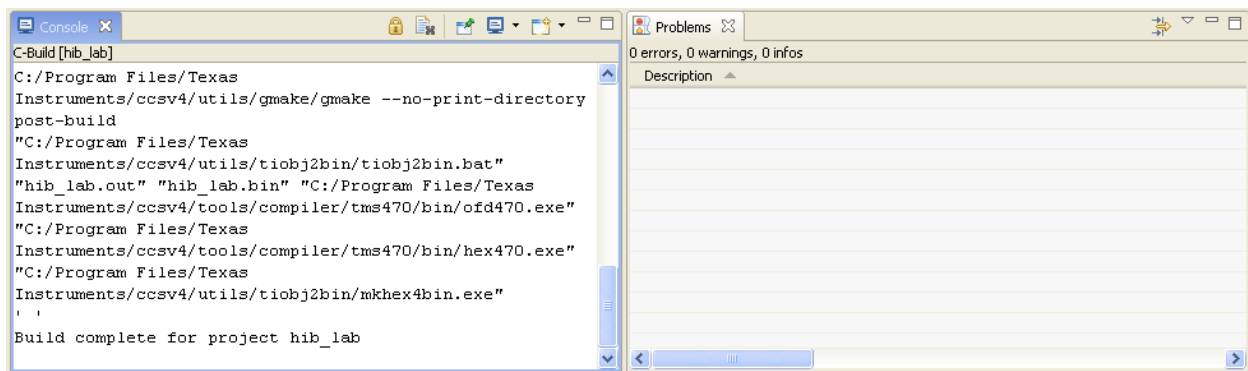
```
hib_lab.c
1 //*****
2 //
3 // hibernation_lab.c - Hibernation Hands-on lab.
4 //
5 // Copyright (c) 2011 Texas Instruments Incorporated. All rights reserved.
6 // TI Information - Selective Disclosure
7 //
8 //*****
9
10 #include "utils/ustdlib.h"
11 #include "inc/hw_types.h"
12 #include "inc/hw_memmap.h"
13 #include "driverlib/sysctl.h"
14 #include "driverlib/pin_map.h"
15 #include "griib/griib.h"
16 #include "drivers/cfal96x64x16.h"
17 #include "driverlib/debug.h"
18 #include "driverlib/hibernate.h"
19 #include "driverlib/systick.h"
20
21 //*****
22 //
23 //! \addtogroup example_list
24 //! <h1>Hibernation Lab (adc)</h1>
25 //!
26 //! Hibernation Lab has following three objectives:
27 //! 1) Configure hibernation module and wake-up the processor from hibernation
28 //! mode when nWAKE pins is asserted.
29 //! 2) Configure Real Time Clock and wake-up the processor from hibernation mode
30 //! when RTC match occurs.
31 //! 3) Measure & compare the current consumption in:
32 //! (1) Active mode & Hibernation mode without RTC.
33 //! (2) Hibernation mode with RTC.
34 //!
35 //! The device goes in hibernation mode upon power-on-reset. After a delay of
36 //! ~5 seconds, the device goes in hibernation mode, saves user data in non-
37 //! volatile/ battery backed memory. The hibernation module looks for nWAKE
38 //! assertion or RTC match.
39 //!
40 //*****
41
42 //*****
43 //
44 // The error routine that is called if the driver library encounters an error.
45 //
46 //*****
47 #ifdef DEBUG
48 void
49 __error__(char *pcFilename, unsigned long ulLine)
50 {
51 }
52 #endif
53
54 int
```

Building a Project

9. All the required libraries & files are already included in the project. Click on the “Build Active Project” icon to build (compile & link) the active project.

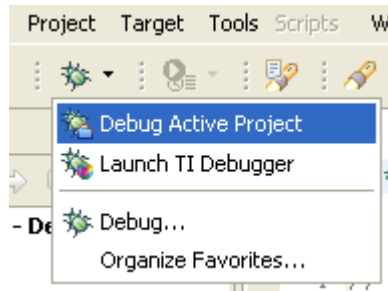


10. The project should build successfully without errors. The “Console” window will display the build results. Should any error/warnings occur, they will be displayed in the “Problems” window.



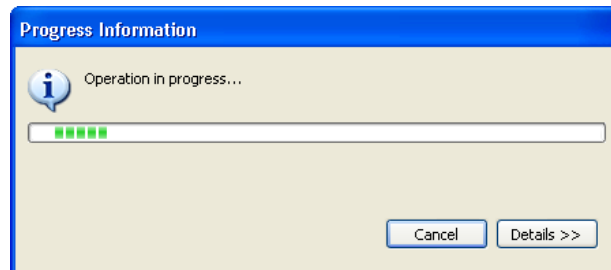
Debugging a Project

11. After the project has been successfully built, now we can download the application into the flash of the microcontroller using CCS debugger. Click on the “Bug” icon to download and debug the active project.

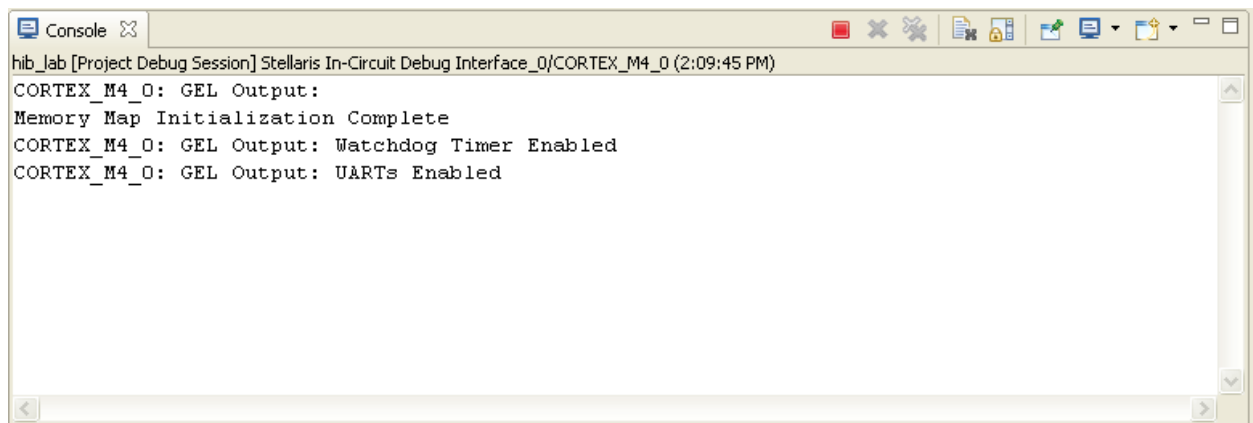


Note: As an alternative to procedure mentioned in Step 11 above, users can also use LM Flash Programmer to program the binary file “hib_lab.bin” located in “C:\StellarisWare-HIB-HandsOn\boards\ek-lm4f232\hib_lab\ccs\Debug”

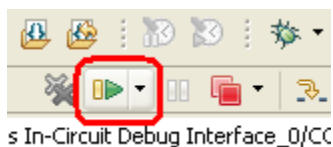
12. Upon clicking the “Bug” icon, the following window will appear indicating that the debugger is launching and is connecting to the board.



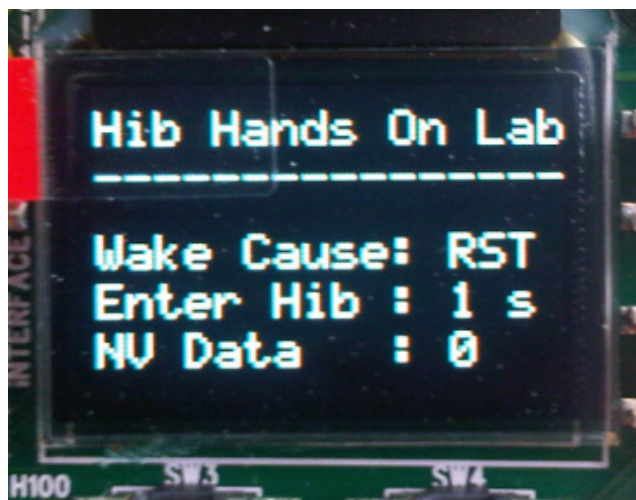
13. When the debugger is launched, the console window will show the following contents indicating that memory map has been initialized and microcontroller’s flash has been programmed.



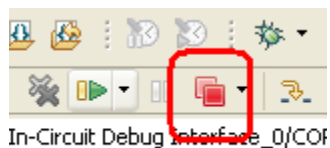
14. Press the “Run” button to start the code execution.



15. You will see the following on the screen of EK-LM4F232 evaluation board.



15. Pause the code execution by pressing the “Pause” button before the 5 seconds times runs out & device goes into hibernation mode. Press the “Terminate” button to disconnect from the device, close the debugger and switch back to CCS code development screen.



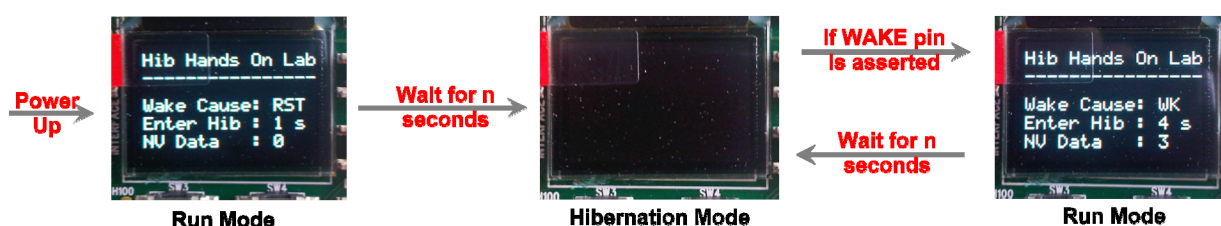
Source Code Explanation

Carefully examine how hibernation module had been initialized and configured in this example using various API function calls. In this application, after power up the processor waits for a finite duration of time (specified by SysCtkClkDelay()) before entering in to the hibernation mode. The device stays in hibernation mode untill WAKE pin is asserted. After the device wakes up from hibernation mode, normal code execution begins for next 'n' seconds.

Before the hibernation module can be used, it must be first enabled using HbernateEnableExpClk() function. Upon wake up from hibernation, HiobernateIntStatus() can be use to determine the cause of hibernation i.e. whether microcontroller woke up due to a RTC match, or WAKE pin assertion. During first power up, the wake cause is reset. Wake cause is displayed on the screen in “Wake Cause” field as RST, WK or RTC.

Battery Backed Memory

Everytime the device enters in hibernation mode, user data (a count variable, NV Data, in SRAM) is saved to the battery-backed memory in hibernation module. Upon next wake-up, the user data is read from the battery-backed memory, incremented and saved again when the micrcontroller enters hibernation mode again. The cycle continues as shown in the state diagram below.



User data can be read from and written to battery-backed memory using HibernateDataGet() and HibernateDataSet() functions respectively.

In the subsequent sections of this lab, we will take this example and modify it to learn how real time clock can be used in hibernation mode. We will also measure and compare the current consumed in different modes of operation (run mode with PLL and hibernation mode).

Function Protoyppte and Syntax

For more details on the API function calls, please refer to DriverLib documentation in StellarisWare at “C:\StellarisWare-HIB-HandsOn\docs.”

Function Definition

For API function definitions, please refer to the source code (.c) files for the corresponding peripheral. For example, function calls for Hibernation module are defined in the file “hibernate.c” and is located at “C:\StellarisWare-HIB-HandsOn\driverlib.”

16. With this you have successfully completed section 2 of Blizzard Training on Low Power!