Creating a “Hello World” Sample Application in Code Composer Studio v5.1.0 for the DM3730/AM3703 SOM-LV, DM3730/AM3703 Torpedo SOM, and DM3730/AM3703 Torpedo + Wireless SOM

Application Note 529

Logic PD // Products
Published: March 2012
Last revised: June 2012

Revision History

REV	EDITOR	DESCRIPTION	APPROVAL	DATE
A	JO	-Initial Release	BSB, GJ	03/28/12
B	SO	-Throughout: Added language for AM3703 configuration of Torpedo + Wireless SOM; -Section 1.2: Removed statement about Blackhawk XDS 100v2 being included in DM3730 Development Kit	SO	06/20/12

Table of Contents

1 Introduction

The purpose of this document is to provide the necessary steps for creating a “Hello World” sample application in Code Composer Studio (CCS) v5.1.0 for the DM3730/AM3703 SOM-LV, DM3730/AM3703 Torpedo SOM, and DM3730/AM3703 Torpedo + Wireless SOM using the Blackhawk XDS100v2 model D USB JTAG controller.

Section 3 addresses how to create the sample application for the DM3730 SOM-LV, DM3730 Torpedo SOM, and DM3730 Torpedo + Wireless SOM. Section 4 addresses how to create the sample application for the AM3703 SOM-LV, AM3703 Torpedo SOM, and AM3703 Torpedo + Wireless SOM.

1.1 Nomenclature

- This document covers the DM3730 SOM-LV, DM3730 Torpedo SOM, and DM3730 Torpedo + Wireless SOM. Use of “DM3730 SOM” suggests text that applies to all three platforms; information specific to one platform will call out the precise name.
- This document also covers the AM3703 SOM-LV, AM3703 Torpedo SOM, and AM3703 Torpedo + Wireless SOM. Use of “AM3703 SOM” suggests text that applies to all three platforms; information specific to one platform will call out the precise name.
- Use of “DM3730 Development Kit” suggests text that applies to both the DM3730 SOM-LV Development Kit and DM3730 Torpedo SOM Development Kit; information specific to one development kit will call out the precise name.

1.2 Requirements

This application note assumes the following:

- Your DM3730 Development Kit or similar platform is properly assembled as described in the appropriate QuickStart Guide.
  - *DM3730 Torpedo Development Kit QuickStart Guide*[1]
  - *DM3730 SOM-LV Development Kit QuickStart Guide*[2]
- You are familiar with the User Guide associated with your DM3730 Development Kit or similar platform.
  - *DM3730 Torpedo Development Kit User Guide*[3]
  - *DM3730 SOM-LV Development Kit User Guide*[4]
- You have the Blackhawk XDS100v2 model D USB JTAG controller.[5](See Figure 1.1 below.) **IMPORTANT NOTE:** Do not attach the Blackhawk XDS100v2 until CCS v5.1.0 has been installed on your host PC.

[1] http://support.logicpd.com/downloads/1420/
[2] http://support.logicpd.com/downloads/1421/
[3] http://support.logicpd.com/downloads/1450/
[4] http://support.logicpd.com/downloads/1449/
[5] http://www.blackhawk-dsp.com/products/USB100.aspx

Figure 1.1: Blackhawk XDS100v2 with JTAG adapter

Your Blackhawk XDS100v2 must have the latest firmware. For information on how to update to the latest firmware, please refer to the Texas Instruments' (TI) *How Can I Update the CPLD on My XDS100v2* wiki article.[6]

- Your host PC has an available USB port and active Internet connection.

2 Setup

This section explains how to install CCS v5.1.0 on your DM3730 SOM or AM3703 SOM and connect the Blackhawk XDS100v2 to your DM3730 Development Kit.

2.1 Install CCS v5.1.0

Install CCS v5.1.0 if it is not already installed on your host PC. TI's *Code Composer Studio v5* wiki article[7] provides information for properly installing the software.

Again, this application note was tested on and written specifically for CCS v5.1.0.

2.2 Connect the Blackhawk XDS100v2

1. Ensure the DM3730 Development Kit is assembled as described in the appropriate QuickStart Guide.
2. Connect the Embedded Trace Macrocell™ (ETM) Adapter Board (included in the development kit) to the SOM as documented in the appropriate DM3730 Development Kit User Guide.
3. Connect the Blackhawk XDS100v2 to connector J3 on the ETM Adapter Board.
4. Using the USB cable provided, connect the Debug USB port on the baseboard to the USB port on your host PC. (See the appropriate DM3730 Development Kit User Guide for the location of this port on the baseboard.)
5. Power on the DM3730 Development Kit as documented in the QuickStart Guide.

3 Creating a "Hello World" Sample Application for the DM3730 SOM

This section will describe the steps to create a "Hello World" sample application in CCS v5.1.0 for the DM3730 SOM using the Blackhawk XDS100v2.

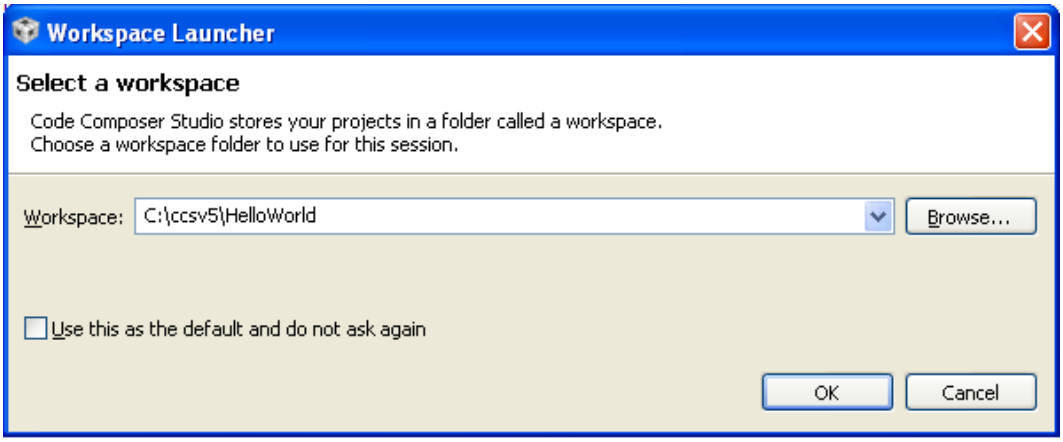
1. Download the necessary files:
   a. On TI's *Device Support files* wiki page,[8] download the *DaVinci (DMxxxx) v1.0.3* files and follow the directions in the "Usage" section on the same wiki page.

[6] http://processors.wiki.ti.com/index.php/XDS100#Q:_How_can_I_update_the_CPLD_on_my_XDS100v2.3F
[7] http://processors.wiki.ti.com/index.php/Code_Composer_Studio_V5

b. On TI's *OMAP and Sitra CCS Support* wiki page,[9] download the GEL files for the DM3730 processor.

c. On TI's *Linker CMD Files for CCS* wiki page,[10] download the linker command file for the DM3730 processor.

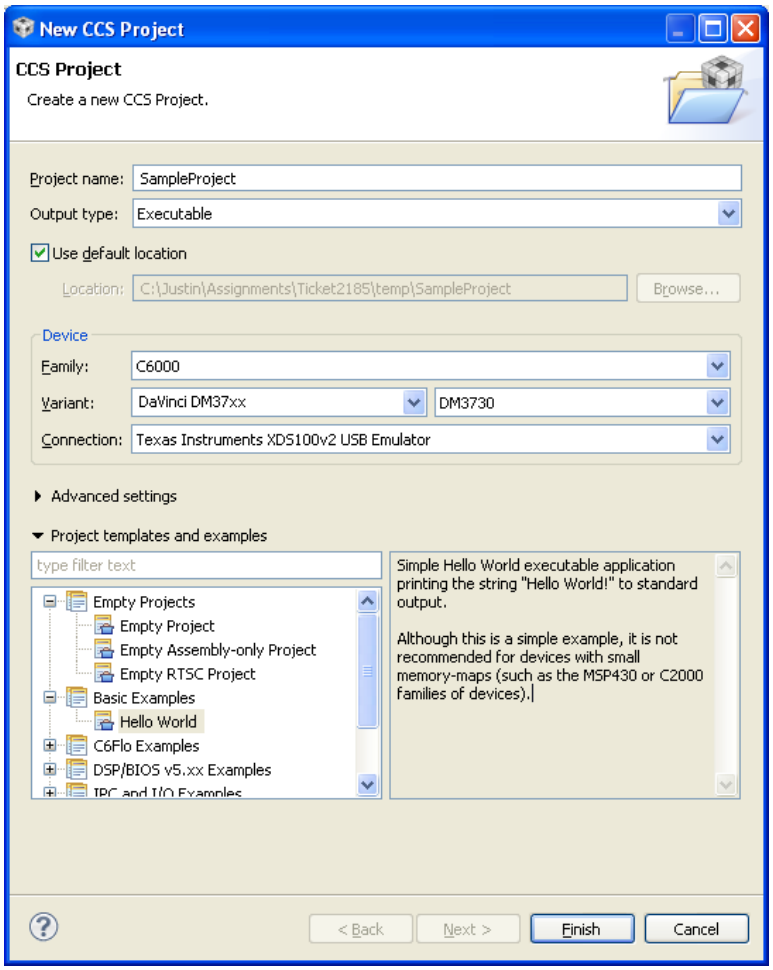
2. Create a temporary workspace in CCS v5.1.0 called *HelloWorld* as shown below and click OK.



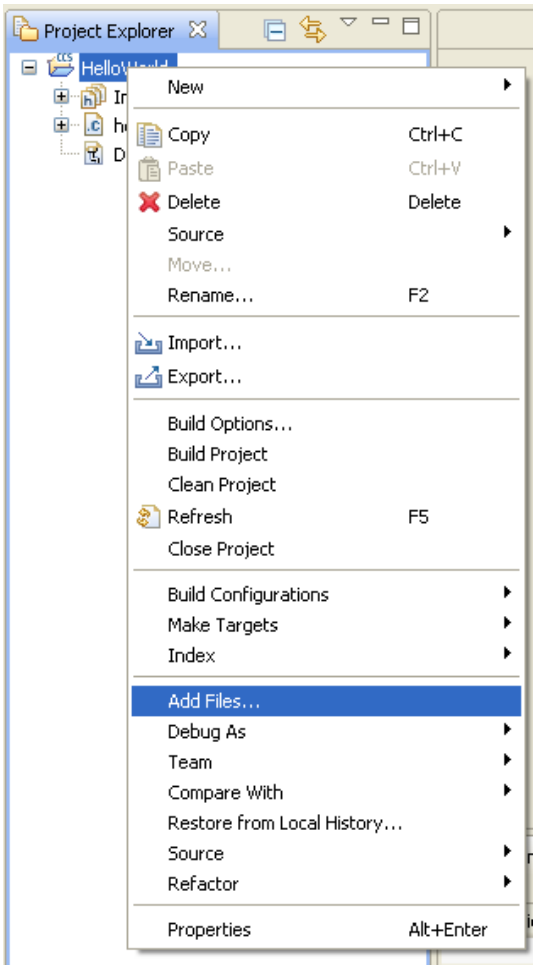
3. Close the TI Explorer page and create a new project in the *CCS Edit* perspective by selecting File > New > CCS Project.

[8] http://processors.wiki.ti.com/index.php/Device_support_files
[9] http://processors.wiki.ti.com/index.php/OMAP_and_Sitara_CCS_support#DM3730_and_DM3725
[10] http://processors.wiki.ti.com/index.php/Linker_CMD_Files_for_CCS#DaVinci"

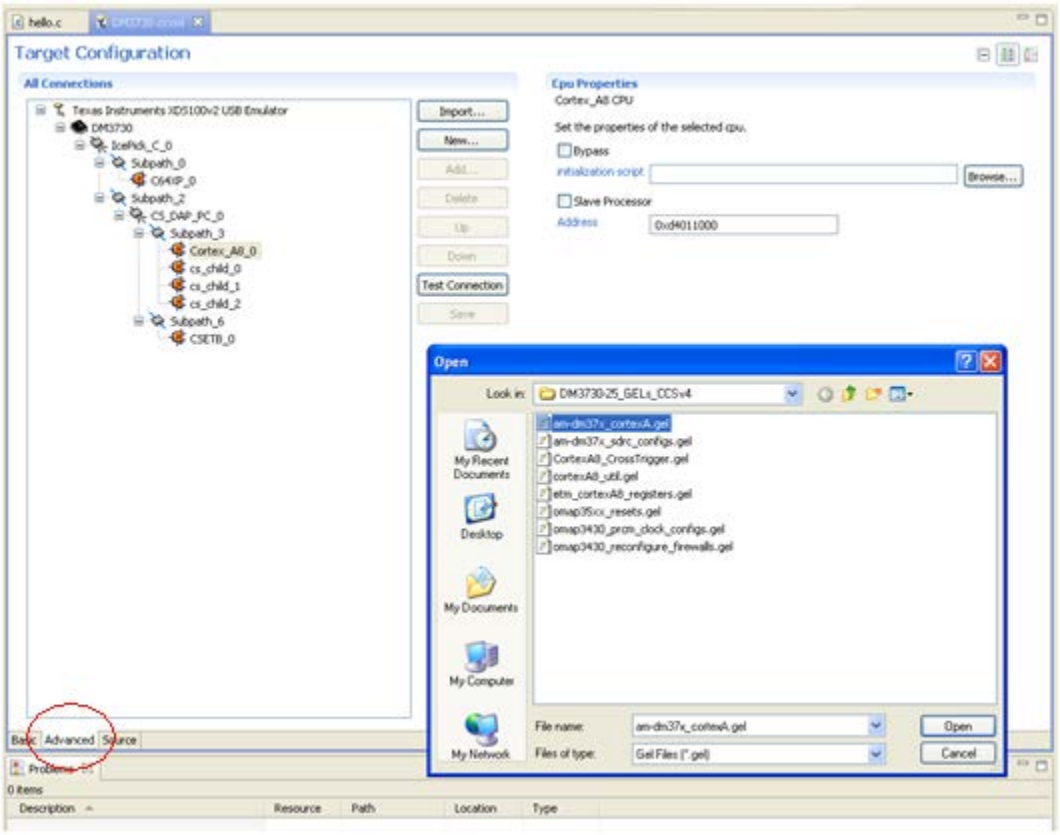
4. In the *New CCS Project* window, choose the settings that best match your configuration and click Finish. An example of one possible configuration is shown below.



5. Add the *DM3730.cmd* file downloaded above in Step 1c by right clicking [Project Name] and then selecting Add Files… as shown below. Navigate to where you have the *DM3730.cmd* file saved, select Open, and choose to copy it into the workspace.

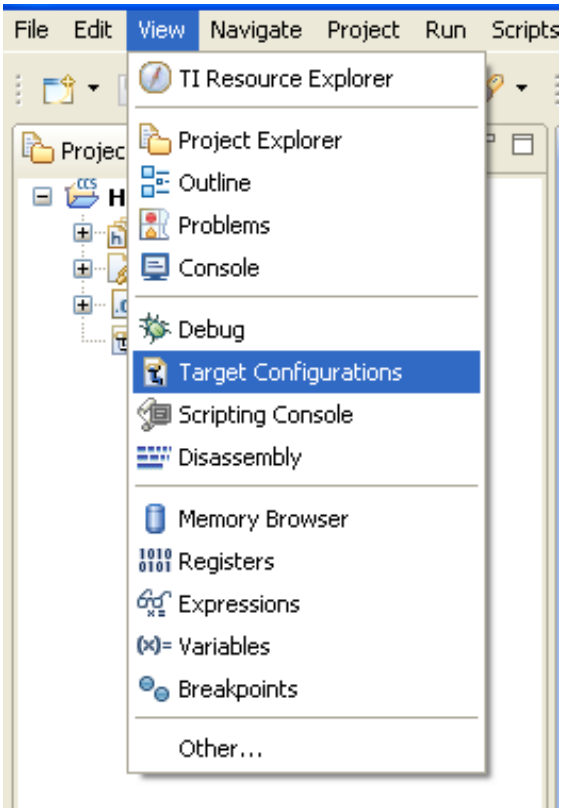


6. Edit your target configuration file by double clicking on *DM3730.ccxml* in your *Project* window. Choose the Advanced section (circled below in red) and click on the Cortex_A8_0 option. Browse to the folder that contains the GEL files downloaded in Step 1b. From those files, choose the *am-dm37x_cortexA.gel* file and click Open.

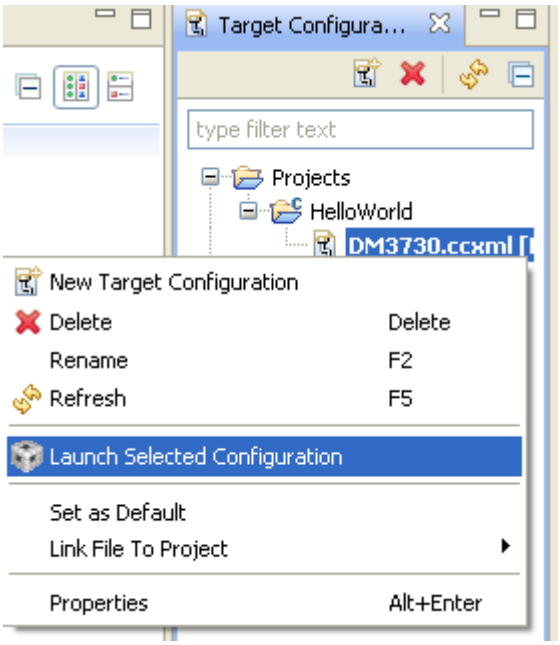


7. Build the project by right clicking on [Project Name] and selecting Build Project.

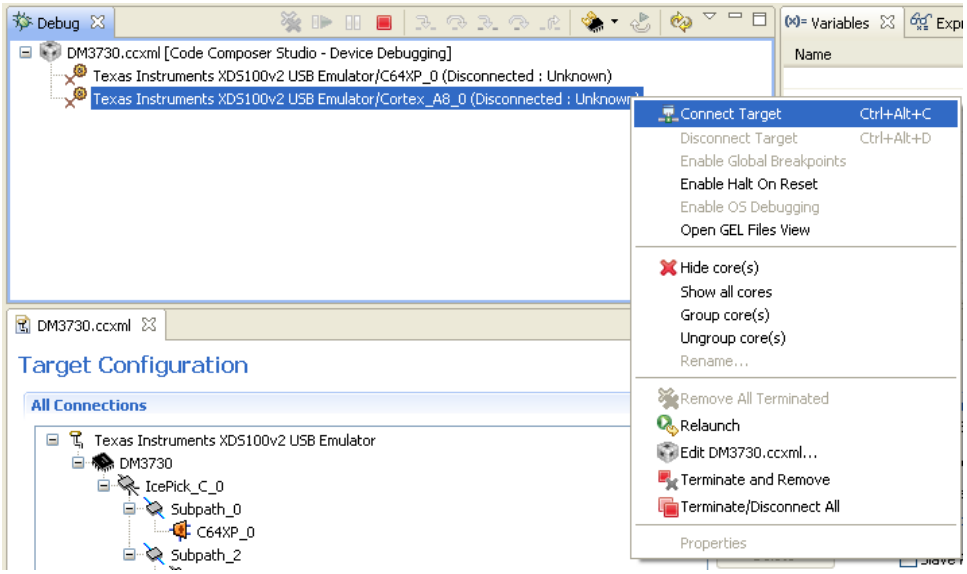
8. Now we want to launch the target configuration. To do this, select View > Target Configurations in the *CCS Edit* perspective.



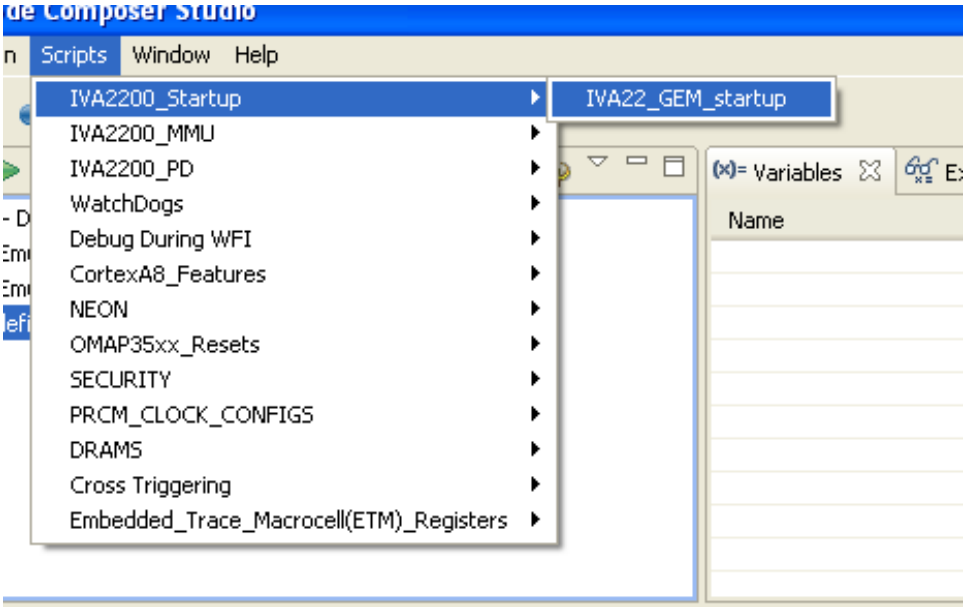
9. In the *Target Configuration* window, navigate to your target configuration file by selecting Projects > Hello World. Right click on the *DM3730.ccxml* file and choose Launch Selected Configuration.



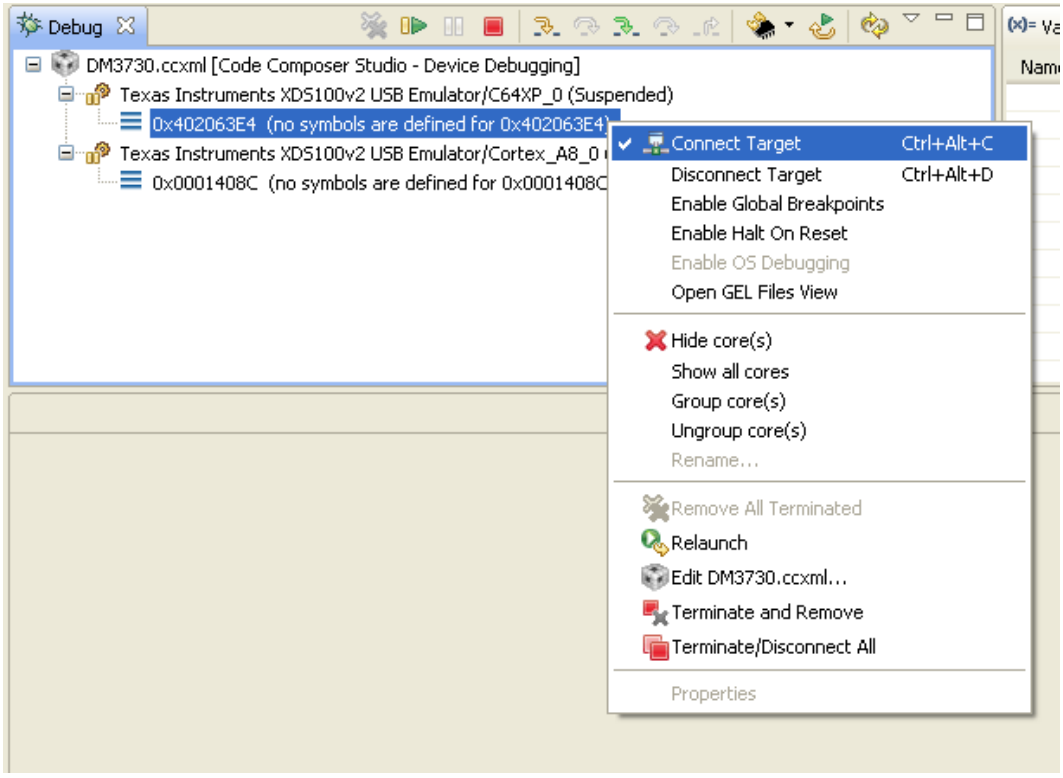
10. To connect to the ARM core, right-click on Texas Instruments XDS100v2 USB Emulator/Cortex_A8_0 and select Connect Target.



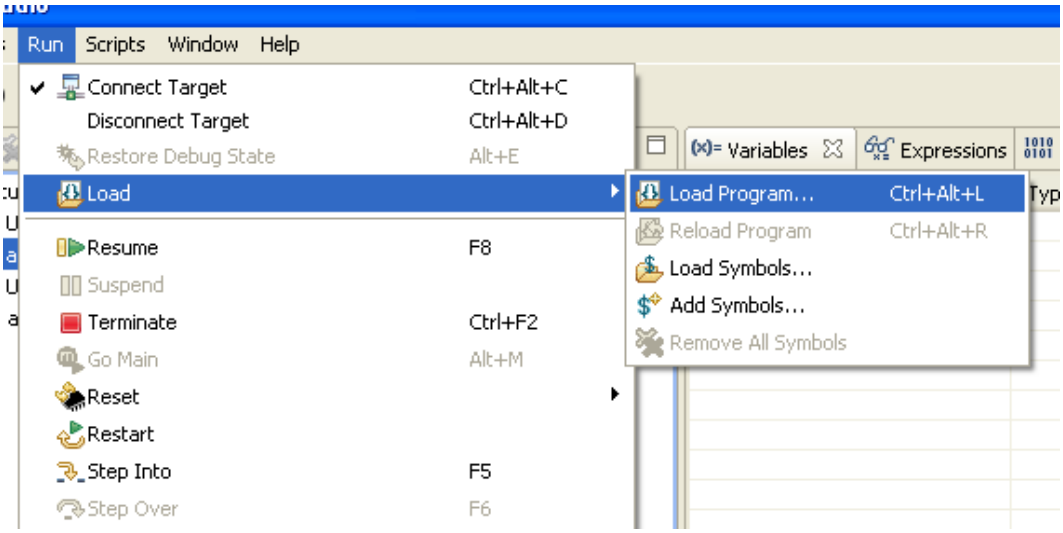
11. Run a script to wake up the DSP by selecting Scripts > IVA2200_Startup > IVA22_GEM_startup.



12. Now connect to the DSP core the same way you connected to the ARM core by right-clicking on 0x402063E4 and selecting Connect Target.



13. Download the program onto the DSP by selecting Run > Load > Load Program….



14. Choose Browse Project and select the *HelloWorld.out* file. Click OK twice.

15. Run the program. You should see “[C64XP_0] Hello World!” in the console window.

4 Creating a “Hello World” Sample Application for the AM3703 SOM

This section will describe the steps to create a “Hello World” sample application in CCS v5.1.0 for the AM3703 SOM using the Blackhawk XDS100v2.

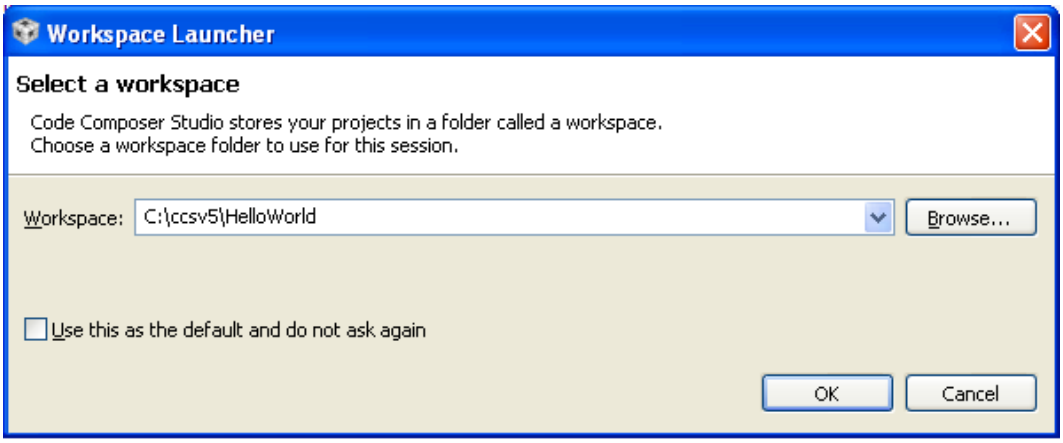
1. Download the necessary files:

   a. On TI’s *Device Support files* wiki page, [11] download the *Sitra (AMxxxx) v1.0.3* files and follow the directions in the “Usage” section on the same wiki page.

   b. On TI’s *OMAP and Sitra CCS Support* wiki page,[12] download the GEL files for the DM3730 processor.

   c. On TI’s *Linker CMD Files for CCS* wiki page,[13] download the linker command file for the DM3730processor.

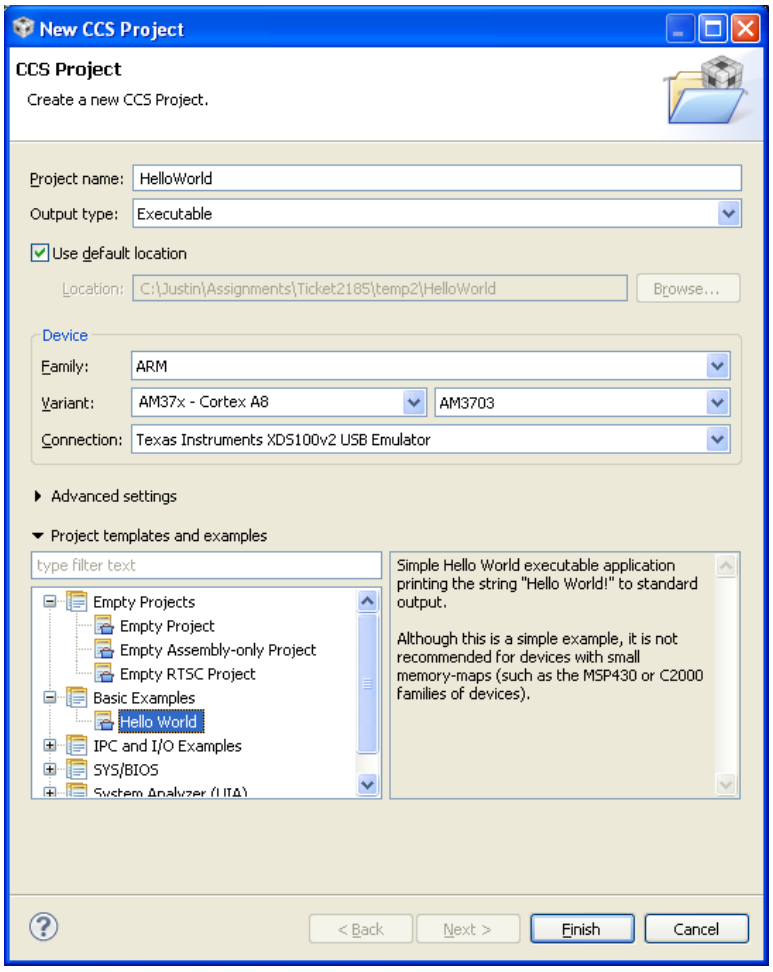
2. Create a temporary workspace in CCS v5.1.0 called HelloWorld as shown below and click OK.



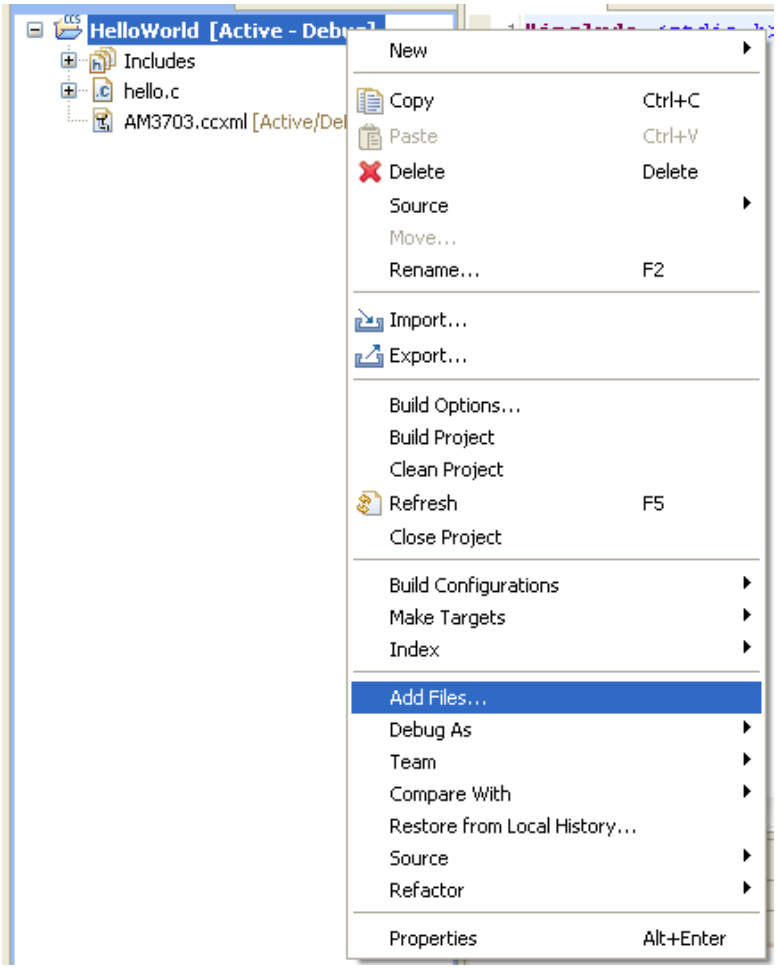
3. Close the TI Explorer page and create a new project in the *CCS Edit* perspective by selecting File > New > CCS Project.

[11] http://processors.wiki.ti.com/index.php/Device_support_files
[12] http://processors.wiki.ti.com/index.php/OMAP_and_Sitara_CCS_support#DM3730_and_DM3725
[13] http://processors.wiki.ti.com/index.php/Linker_CMD_Files_for_CCS#DaVinci"

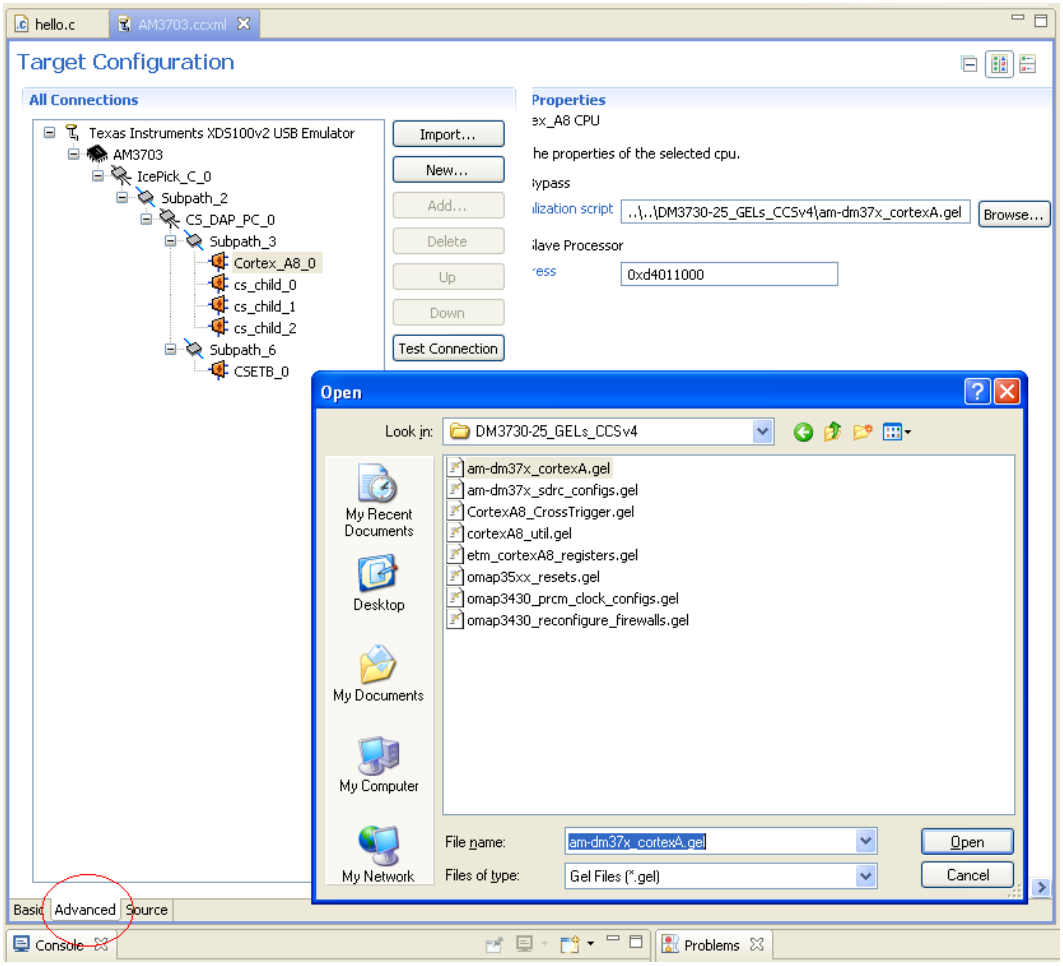
4. In the *New CCS Project* window, choose the settings that best match your configuration and click Finish. An example of one possible configuration is shown below.



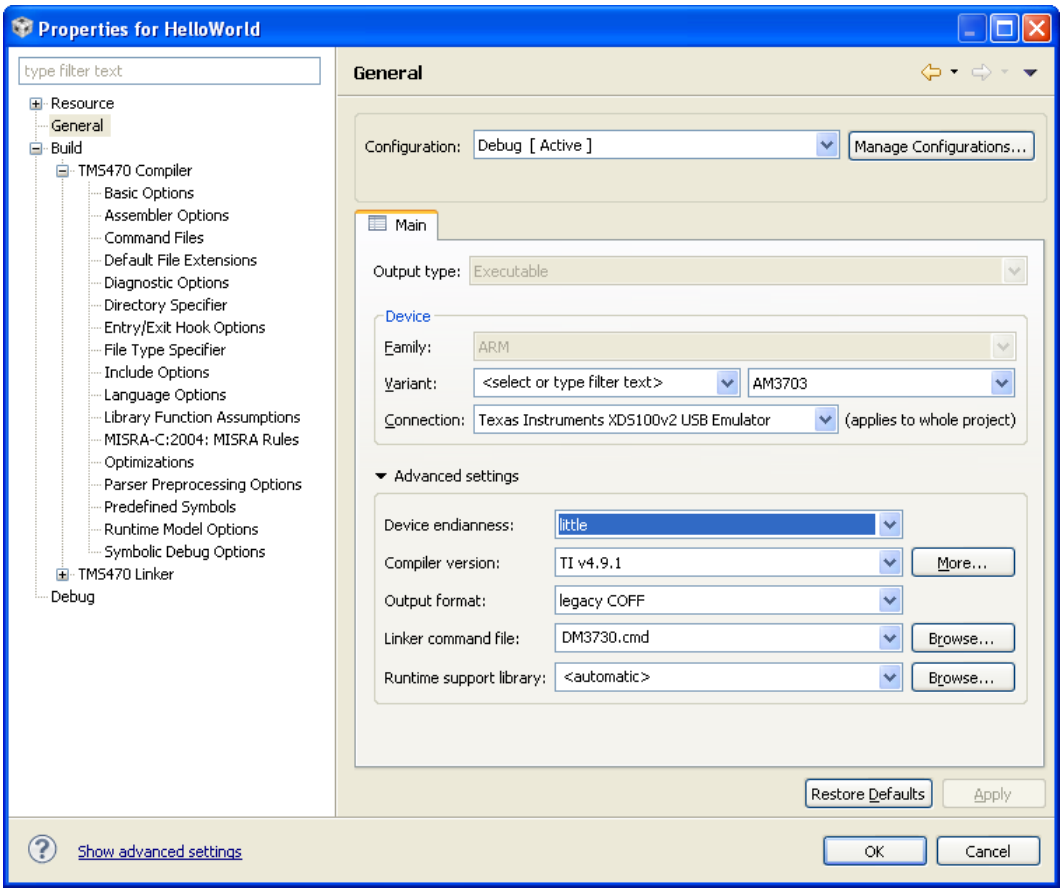
5. Add the *DM3730.cmd* file downloaded above in Step 1c by right clicking on [Project Name] and selecting Add Files… as shown below. Navigate to where you have the *DM3730.cmd* file saved, select Open, and choose to copy it into the workspace.



6. Edit your target configuration file by double clicking on the *AM3703.ccxml* file in your *Project* window. Choose the Advanced section (circled below in red) and click on the Cortex_A8_0 option. Browse to the folder that contains the GEL files downloaded in Step 1b. From those files, choose the *am-dm37x_cortexA.gel* file and click Open.

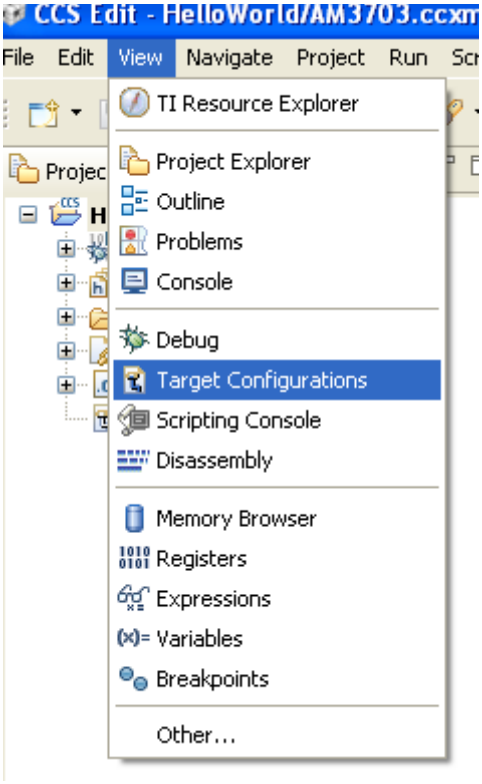


7. Change the *Device endianness* of the project by right clicking on [Project Name] and selecting Properties. In the *General* tab, change the *Device endianness* to *little* and click OK.

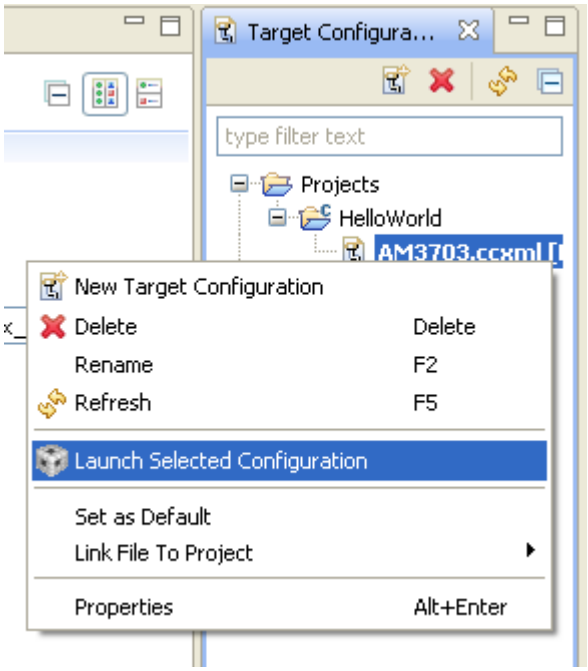


8. Build the project by right clicking on [Project Name] and selecting Build Project.

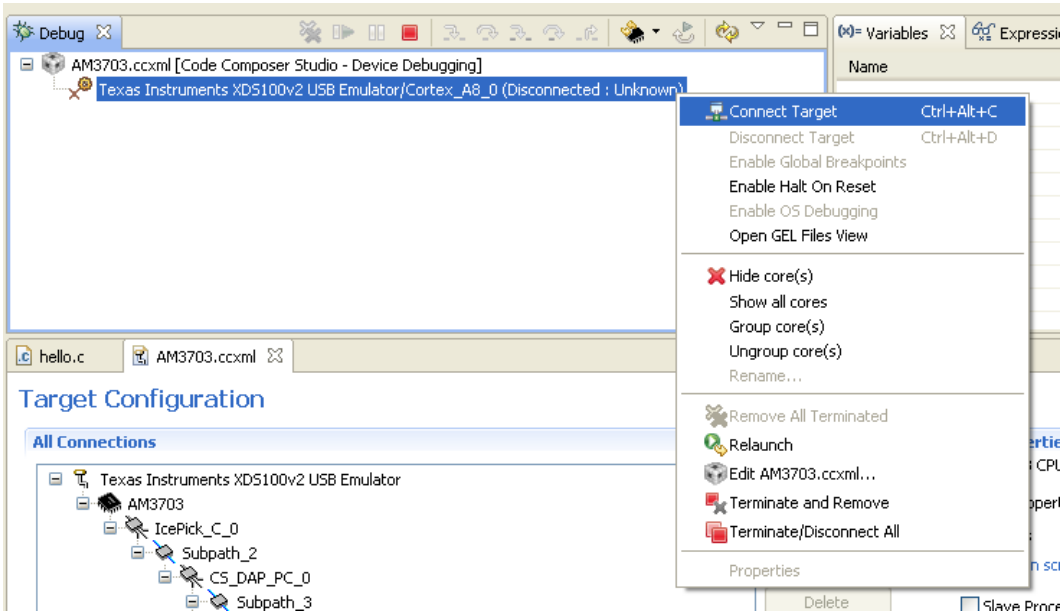
9. Now we want to launch the target configuration. To do this, select View > Target Configurations in the *CCS Edit* perspective.



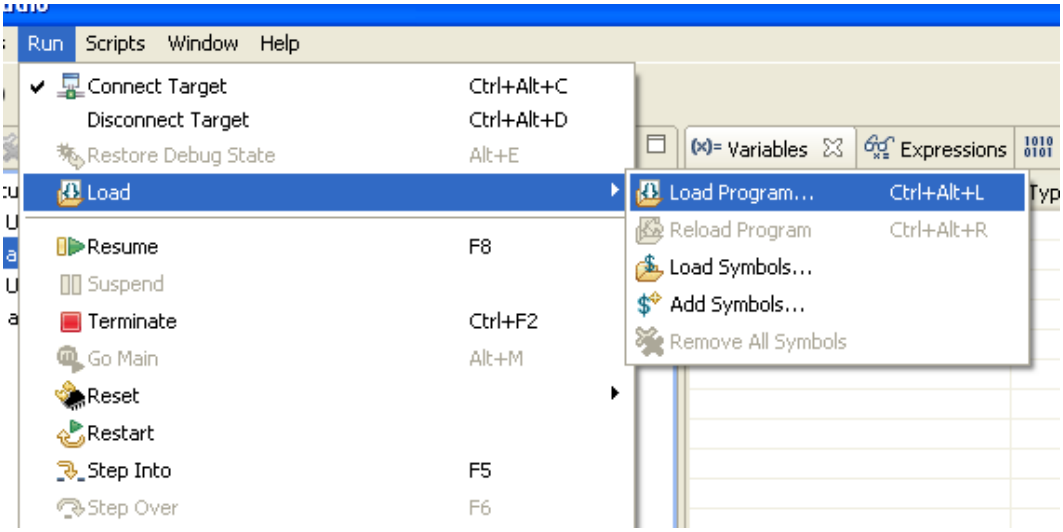
10. In the *Target Configuration* window, navigate to your target configuration file by selecting Projects > Hello World. Right click on the *AM3703.ccxml* file and choose Launch Selected Configuration.



11. To connect to the ARM core, right-click on Texas Instruments XDS100v2 USB Emulator/Cortex_A8_0 and select Connect Target.



12. Download the program onto the ARM core by selecting Run > Load > Load Program….



13. Choose Browse Project and select the *HelloWorld.out* file. Click OK twice.

14. Run the program. You should see “[Cortex_A8_0] Hello World!” in the console window.

5 Troubleshooting

This section shows what the correct output of the console should be when connecting to the DM3730 SOM and the AM3703 SOM.

5.1 DM3730 Console Expected Output

```
Cortex_A8_0: GEL Output: OMAP 32K Watchdog Timer is disable
Cortex_A8_0: GEL Output:  Putting DPLL into bypass before proceeding
Cortex_A8_0: GEL Output:  Putting CORE DPLL into bypass before
proceeding
Cortex_A8_0: GEL Output:  Locking CORE DPLL
Cortex_A8_0: GEL Output:  PRCM clock configuration IIA setup has been
completed
Cortex_A8_0: GEL Output:  SystemClock = 26.0 MHz
Cortex_A8_0: GEL Output:  DPLL_MULT_VALUE = 332
Cortex_A8_0: GEL Output:  DPLL_DIV_VALUE = 25
Cortex_A8_0: GEL Output:  CORE_DPLL_CLK = 664.0 MHz
Cortex_A8_0: GEL Output:  CORE_CLK = 332.0 MHz
Cortex_A8_0: GEL Output:  L3_CLK = 166.0 MHz
Cortex_A8_0: GEL Output: mDDR Hynix H8KDS0UN0MER - 2048 Mbit(256MB) on
CS0, 16M x 32bit x 4Banks
Cortex_A8_0: GEL Output: Waiting for SDRC DLL to lock...
Cortex_A8_0: GEL Output: SDRC DLL successfully locked
Cortex_A8_0: GEL Output: common_sdram_init() completed
Cortex_A8_0: GEL Output: SDRC initilization for mDDR Hynix H8KDS0UN0MER
completed
Cortex_A8_0: GEL Output: 26MHz clock configuration IIa
Cortex_A8_0: GEL Output: 26MHz clock configuration IIa
Cortex_A8_0: GEL Output:      CORTEXA8_CORE_VERSION = 0x413FC082
Cortex_A8_0: GEL Output: Target contains version r3p2 of the
CortexA8...
Cortex_A8_0: GEL Output:      Read the ETM_POWER_DOWN_STATUS
register...
Cortex_A8_0: GEL Output:      ETM_POWER_DOWN_STATUS = 0x00000001
Cortex_A8_0: GEL Output:      ETM Access is enabled!
Cortex_A8_0: GEL Output:      ETM_ID = 0x410CF236
Cortex_A8_0: result: C64x+ release from reset
```

5.2 AM3703 Console Expected Output

```
Cortex_A8_0: GEL Output: OMAP 32K Watchdog Timer is disable
Cortex_A8_0: GEL Output:  Putting DPLL into bypass before proceeding
Cortex_A8_0: GEL Output:  Putting CORE DPLL into bypass before
proceeding
Cortex_A8_0: GEL Output:  Locking CORE DPLL
Cortex_A8_0: GEL Output:  PRCM clock configuration IIA setup has been
completed
Cortex_A8_0: GEL Output:  SystemClock = 26.0 MHz
Cortex_A8_0: GEL Output:  DPLL_MULT_VALUE = 332
Cortex_A8_0: GEL Output:  DPLL_DIV_VALUE = 25
Cortex_A8_0: GEL Output:  CORE_DPLL_CLK = 664.0 MHz
```



```
Cortex_A8_0: GEL Output:  CORE_CLK = 332.0 MHz
Cortex_A8_0: GEL Output:  L3_CLK = 166.0 MHz
Cortex_A8_0: GEL Output: mDDR Hynix H8KDS0UN0MER - 2048 Mbit(256MB) on
CS0, 16M x 32bit x 4Banks
Cortex_A8_0: GEL Output: Waiting for SDRC DLL to lock...
Cortex_A8_0: GEL Output: SDRC DLL successfully locked
Cortex_A8_0: GEL Output: common_sdram_init() completed
Cortex_A8_0: GEL Output: SDRC initilization for mDDR Hynix H8KDS0UN0MER
completed
Cortex_A8_0: GEL Output: 26MHz clock configuration IIa
Cortex_A8_0: GEL Output: 26MHz clock configuration IIa
Cortex_A8_0: GEL Output:      CORTEXA8_CORE_VERSION = 0x413FC082
Cortex_A8_0: GEL Output: Target contains version r3p2 of the
CortexA8...
Cortex_A8_0: GEL Output:      Read the ETM_POWER_DOWN_STATUS
register...
Cortex_A8_0: GEL Output:      ETM_POWER_DOWN_STATUS = 0x00000001
Cortex_A8_0: GEL Output:      ETM Access is enabled!
Cortex_A8_0: GEL Output:      ETM_ID = 0x410CF236
```

