

U-Boot Debug using CodeWarrior for QorIQ LS series – ARM V7 ISA

1. Introduction

This document describes the steps required for U-Boot debugging using the CodeWarrior for QorIQ LS series – ARM V7 ISA.

This document includes the following sections:

- Build the U-Boot sources.
- Perform U-Boot debug in CodeWarrior for QorIQ LS series – ARM V7 ISA.

Contents

1. Introduction.....	1
2. Preliminary background.....	1
3. Create ARMv7 project	2
4. U-Boot debug support.....	7

2. Preliminary background

The following are the steps required to compile LS1021A U-Boot for the LS1021AQDS board.

2.1. Downloads

Before U-Boot debug, following downloads are necessary:

- Linaro GCC 4.8 (Aarch32) 4.8-2013.12 toolchain
- U-Boot source code

You can gcc-linaro 4.8 (Aarch32) 4.8-2013.12 toolchain from <http://www.linaro.org/downloads/> or you can use the one installed with Linux version of CodeWarrior for QorIQ LS series – ARM V7 ISA.

U-boot source code will be provided together with SDK for LS1021AQDS board.

2.2. Compiling U-Boot

To compile U-Boot, perform these steps (UBUNTU OS was used to build U-Boot):

1. Go to **U-Boot** folder
2. Set the ARCH and CROSS_COMPILE environmental variables and build the U-Boot:

```
make ARCH=arm CROSS_COMPILE=<path_to_toolchain>/arm-linux-gnueabihf- ls1021aqds_nor
```

or

```
export ARCH=arm
export CROSS_COMPILE=<path_to_toolchain>/arm-linux-gnueabihf-
make ls1021aqds_nor
```

3. U-Boot image and U-Boot binary will be placed in **U-Boot** folder.

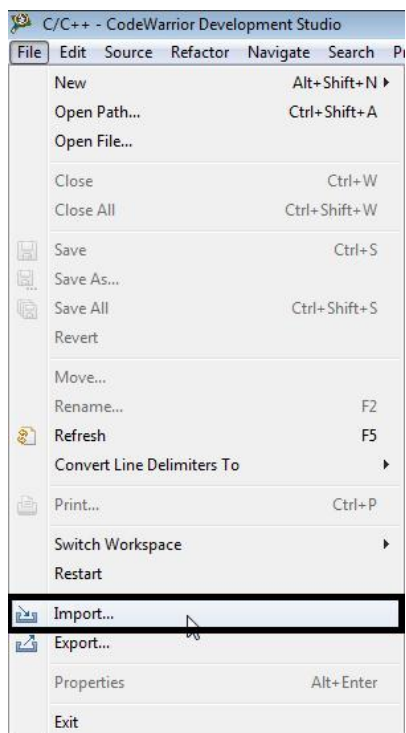
NOTE Default U-Boot will be built with dwarf-2 debug format.

3. Create ARMv7 project

To create an ARMv7 bare metal project for U-Boot debug, follow these steps:

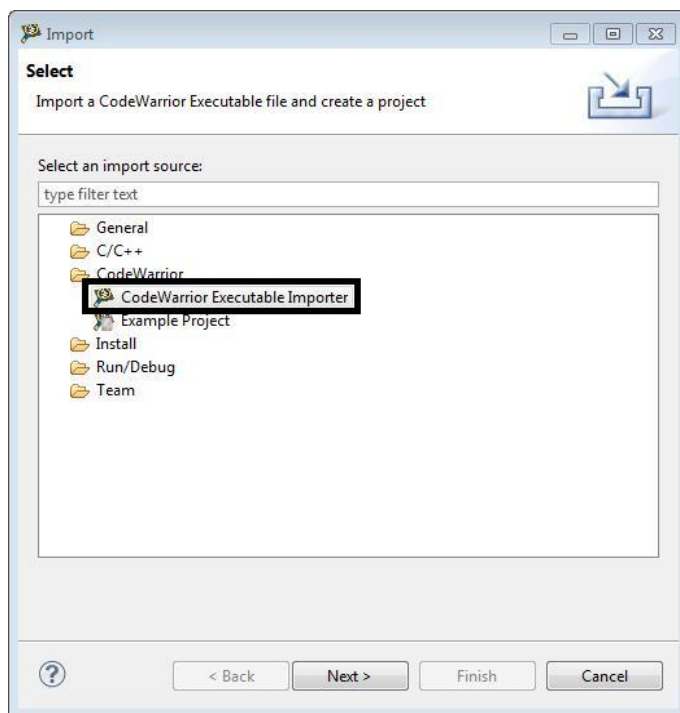
1. Start CodeWarrior for QorIQ LS series – ARM V7 ISA.
2. Choose **File > Import** to import the U-Boot executable file generated during the U-Boot compilation. It can be found in U-Boot folder.

Figure 1. CodeWarrior File menu



3. Choose the source to **Import** and select **Next**.

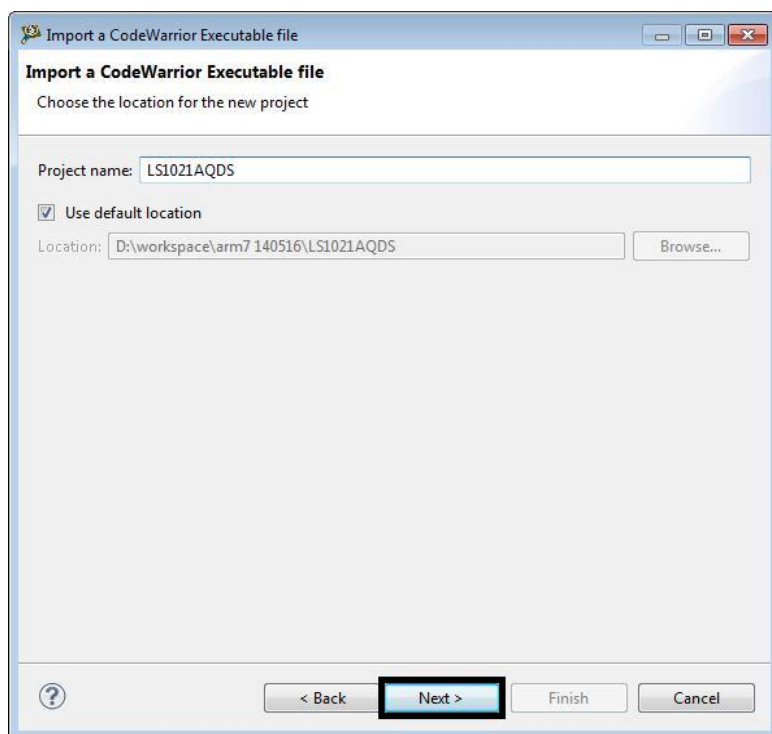
Figure 2. Importing dialog



Create ARMv7 project

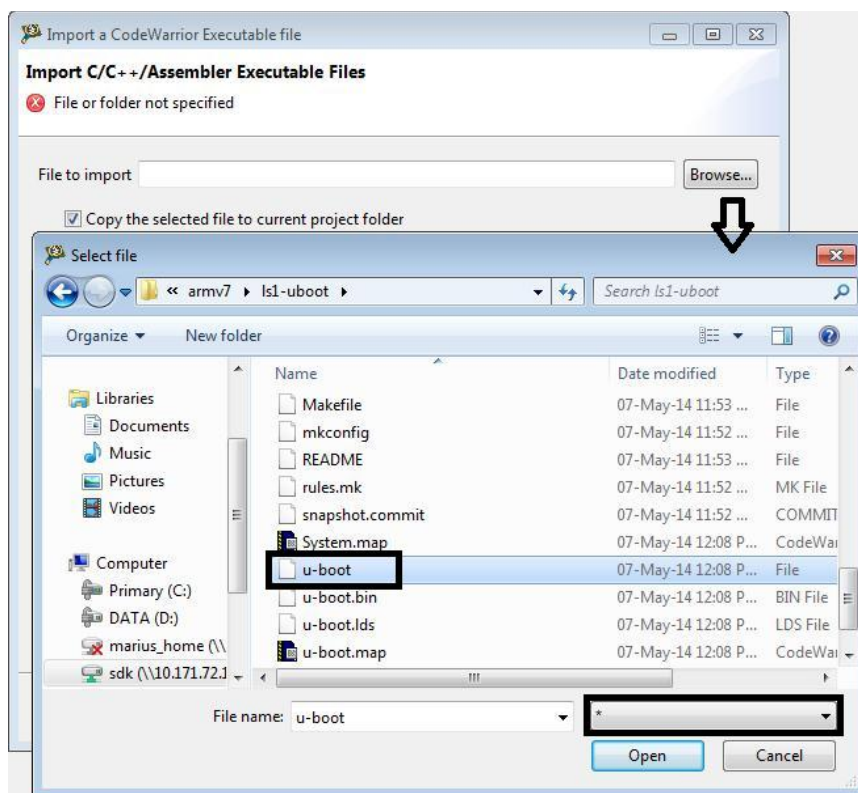
- Specify **Project name** and **Location**, or use the default location and select **Next**.

Figure 3. Importing executable file dialog



- Browse to the U-Boot executable file and select **Open**. By default, CodeWarrior looks for an `.elf` extension, so change the file type in the lower right corner of **Select File** dialog.

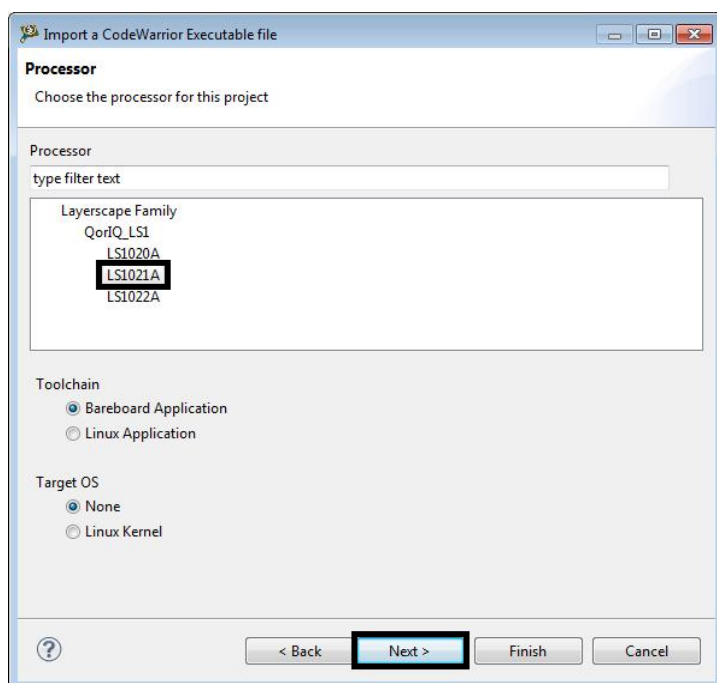
Figure 4. Select U-Boot executable file



6. Select **Processor** type for the project and select **Next**.

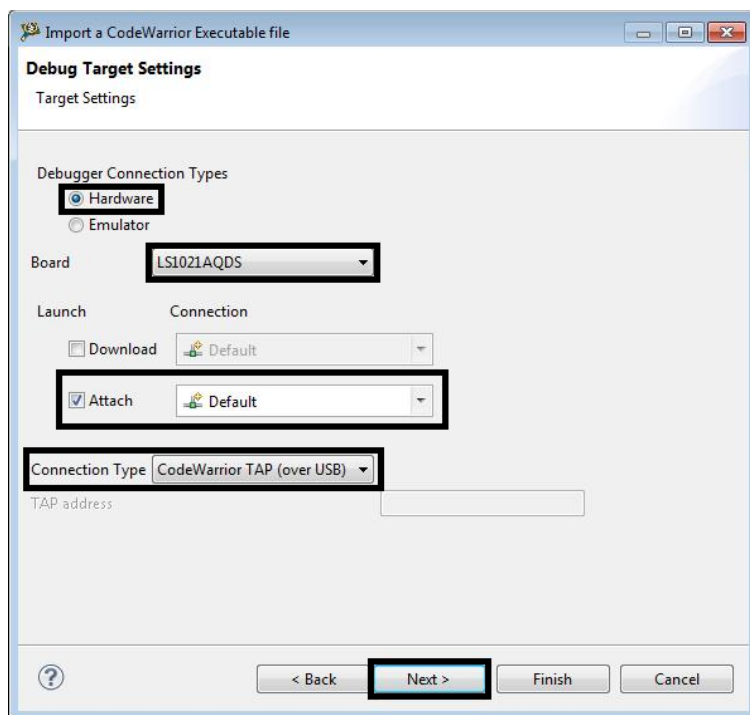
Create ARMv7 project

Figure 5. Select Processor type



7. Select **Debugger Connection Types**, **Board**, **Launch**, **Connection Type** and select **Next**.

Figure 6. Target Settings dialog

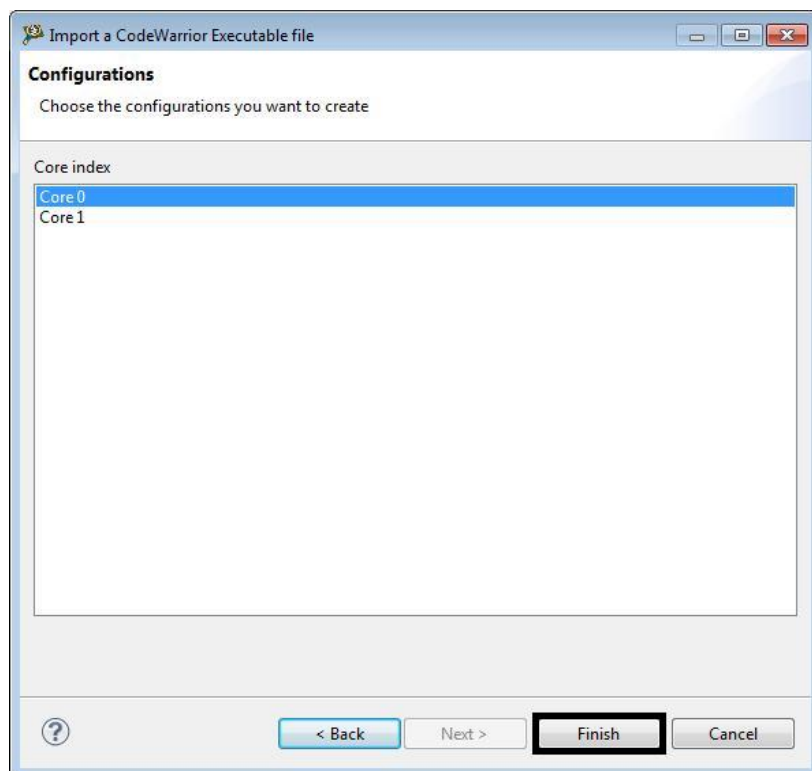


NOTE By default U-Boot will be generated as “Shared object file” and not as “Executable file”. Using Download Launch will not work in this case; Attach Launch will be used instead.

If U-Boot is not available on target board, Flash Programmer should be used to program U-Boot on target board.

8. Select the **Configurations** that you want to create and then, select **Finish** to close the wizard.

Figure 7. Select Configurations dialog



4. U-Boot debug support

4.1. Debug environment

Use the following setup for U-Boot debugging on ARMv7 core:

- LS1021AQDS board.
- Compiled U-Boot for the NOR FLASH target.
- Flash U-Boot on the target board (for more information on how to program the U-Boot to NOR flash, see SDK documentation).
- Switches set for NOR boot (for more information on how to set switches, see SDK documentation).
- Latest release of CodeWarrior for QorIQ LS series – ARM V7 ISA.

U-Boot debug support

- CWTAP probe.

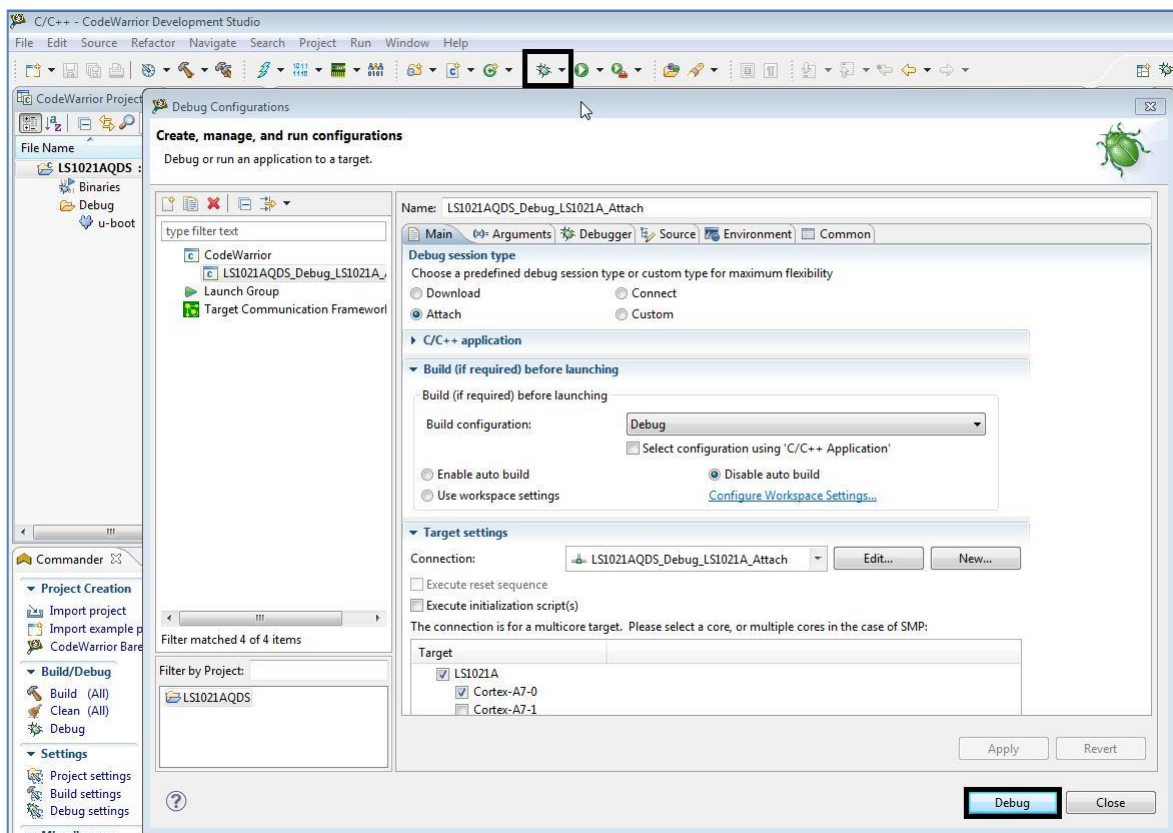
4.2. Start U-Boot debugging

The U-Boot executable file generated during the U-Boot compilation should be imported as CodeWarrior project (for more information, see [Create ARMv7 project](#)).

After the CodeWarrior project is created, perform these steps to start U-Boot debug:

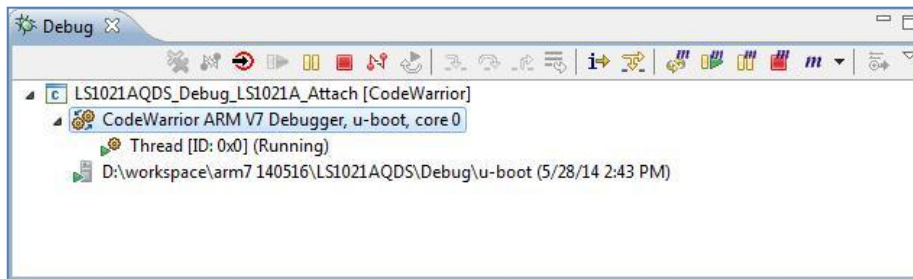
1. Choose **Run > Debug configurations**, to open **Debug configurations** dialog and select **Debug**.

Figure 8. Debug Configurations dialog



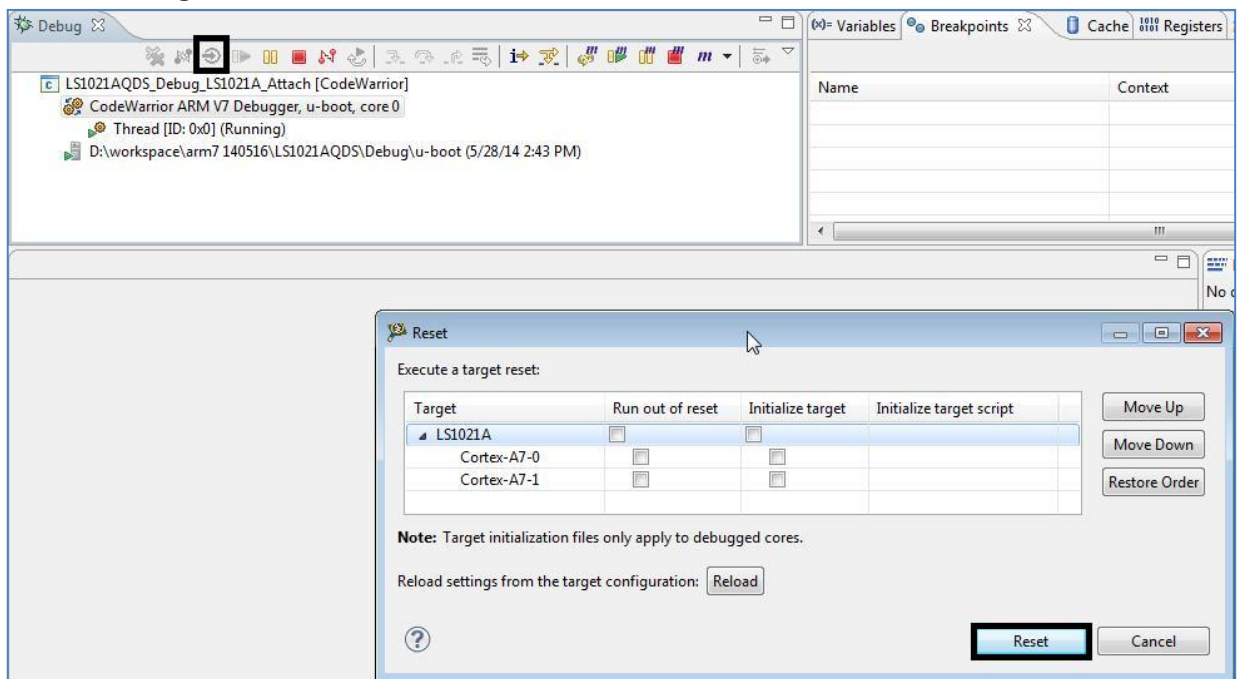
2. The connection initializes and configures the TAP, and then it will attach to board.

Figure 9. Debug view



3. To reinitialize the target from CodeWarrior, select **Reset** as shown in the figure.

Figure 10. Reset dialog



NOTE Make sure no initialization file is selected.

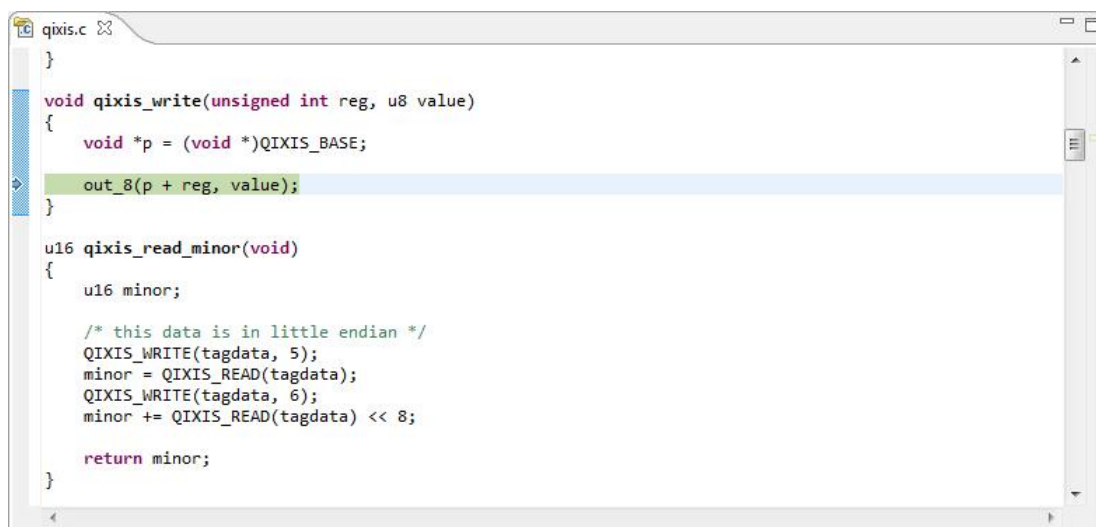
4. After reset debugger will prompt for source location .

Figure 11. File location dialog



5. After the path is provided, source will become available in CodeWarrior.

Figure 12. File editor



- Set a hardware breakpoint at `_start`, using **Debugger Shell** command `bp -hw _start`.


Figure 13. Debugger Shell view



```
CodeWarrior Debugger Shell v1.0
%>bp -hw _start
id instance      address  type  enabled?  process
description
#3      #1  x:0x67f80000 -hw    ENABLED    $0 start.S,
line 23 [u-boot]
%>
```

- Resume using F8 or **Debugger Shell** command `go`.

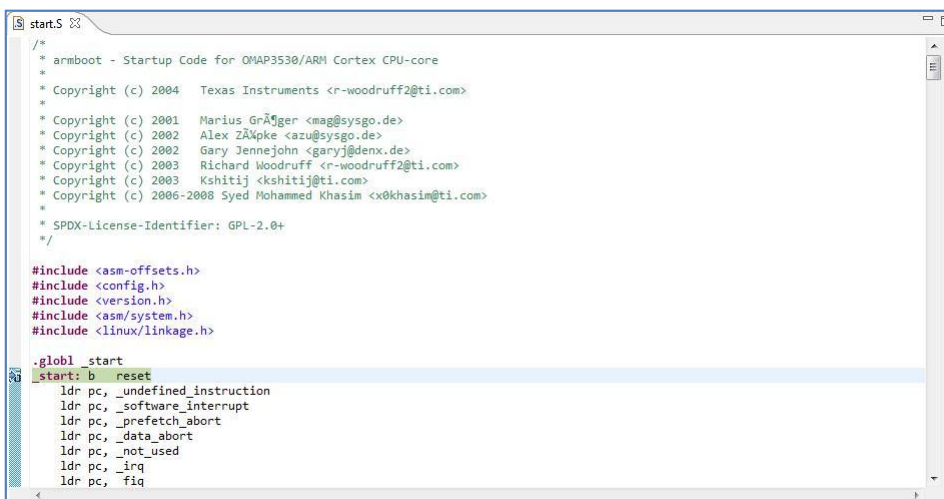
Figure 14. Debugger Shell view



```
CodeWarrior Debugger Shell v1.0
%>bp -hw _start
id instance      address  type  enabled?  process
description
#3      #1  x:0x67f80000 -hw    ENABLED    $0 start.S,
line 23 [u-boot]
%>go
%>
```

- Breakpoint will be hit and U-Boot debugging can be performed from `_start`.

Figure 15. File editor



```
start.S
/*
 * armboot - Startup Code for OMAP3530/ARM Cortex CPU-core
 *
 * Copyright (c) 2004 Texas Instruments <r-woodruff2@ti.com>
 *
 * Copyright (c) 2001 Marius Gräßler <mag@sysgo.de>
 * Copyright (c) 2002 Alex ZÃ¼pke <azu@sysgo.de>
 * Copyright (c) 2002 Gary Jennejohn <garyj@denx.de>
 * Copyright (c) 2003 Richard Woodruff <r-woodruff2@ti.com>
 * Copyright (c) 2003 Kshitij <kshitij@ti.com>
 * Copyright (c) 2006-2008 Syed Mohammed Khasim <x0khasim@ti.com>
 *
 * SPDX-License-Identifier: GPL-2.0+
 */

#include <asm-offsets.h>
#include <config.h>
#include <version.h>
#include <asm/system.h>
#include <linux/linkage.h>

.globl _start
_start: b reset
        ldr pc, _undefined_instruction
        ldr pc, _software_interrupt
        ldr pc, _prefetch_abort
        ldr pc, _data_abort
        ldr pc, _not_used
        ldr pc, _irq
        ldr pc, fiq
```

9. Debugging (step, run, or breakpoint) can be done till the U-Boot boot up.

NOTE If you encounter reset skid issue, the program will not stop at `_start` symbol. As a workaround you can set a hardware breakpoint at `_start` and move PC to `_start` symbol address. This issue has been resolved in FPGA v11 image.

How to Reach Us:

Home Page:

www.freescale.com

E-mail:

support@freescale.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: freescale.com/SalesTermsandConditions.

Freescale, the Freescale logo, CodeWarrior, and QorIQ are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. Layerscape is trademark of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. ARM, Cortex and TrustZone are trademarks or registered trademarks of ARM Ltd or its subsidiaries in the EU and/or elsewhere. All rights reserved.

© 2014 Freescale Semiconductor, Inc.