DSP/BIOS™ LINK

DRX45X

LNK 186 USR

Version 1.65

This page has been intentionally left blank.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third–party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Mailing Address:

Texas Instruments

Post Office Box 655303

Dallas, Texas 75265

This page has been intentionally left blank.

TABLE OF CONTENTS

A. PLATFORM GUIDE

1 Purpose

DSP/BIOS™ LINK is foundation software for the inter-processor communication across the GPP-DSP boundary. It provides a generic API that abstracts the characteristics of the physical link connecting GPP and DSP from the applications. It eliminates the need for customers to develop such link from scratch and allows them to focus more on application development.

This document provides the users necessary information about usage of DSP/BIOS™ LINK on the DRX45X platform.

This document corresponds to the product release Version 1.65 .

2 Text Conventions

O	This bullet indicates important information. Please read such text carefully.
q	This bullet indicates additional information.
[arg1 | arg2]	In context of the commands, contents enclosed in square brackets are the optional arguments to the command. Different values of these arguments are separated by "|".

3 Terms & Abbreviations

CCS	Code Composer Studio
IPC	Inter Processor Communication
GPP	General Purpose e.g. ARM
DSP	Digital Signal Processor e.g. TMS320C5510
CGTools	Code Gen Tools, e.g. Compiler, Linker, Archiver

4 References

1.	User Guide	DSP/BIOS™ LINK user guide
2.	InstallGuide_<OS>_Jacinto.doc	Installation guide for relevant OS if present.
3.	Porting Guide	Porting guide for relevant OS if present.

5 Configuring CCS

5.1 DRX45X EVM

To use CCS for debugging the DSP side application, you will need to configure CCS to use both ARM and DSP with the EVM.

Following the procedure :-

1) To initialize the DDR, PLL and other components take the gel file from $HOME\REL_DRX45X_03.00.00.03.01\build\drx45x_evm_psp_sample.gel.
2) Open the project. ($HOME\ REL_DRX45X_03.00.00.03.01\ pspdrivers\system\drx45x\pros\evmdrx45x\build\ evm_drx45x_psp_pros_sample.pjt)
3) In the lnk.cmd ($HOME\REL_DRX45X_03.00.00.03.01\ pspdrivers\system\drx45x\pros\evmdrx45x\build)check the address and replace the SDRAM with DDR. These address should belongs to DDR.

```
EXCPT_VECT      :   org =   0x80000004    len =   0x0000003C
INT_VECT        :   org =   0x80000040    len =   0x00000104
DDR_INIT_SECT   :   org =   0x80100000    len =   0x04000000
DDR_UNINIT_SECT :   org =   0x86100000    len =   0x04000000
DDR_NON_CACHE   :   org =   0x8A100000    len =   0x02000000
DDR_NON_CACHE_APPS  :   org =   0x8CB00000    len =  0x00400000
SECTIONS
{

   .cdesc      :   load        =   DDR_INIT_SECT /* MMU PT */
   .vects      :   load        =   EXCPT_VECT    /* Exception vectors
to be located in internal RAM starting at 0x4 */
   .intvects   :   load        =   INT_VECT      /* Interrupt vector
numbers to be located in internal RAM starting at 0x40 */
   .start      :   load        =   DDR_INIT_SECT /* Start of PSP image
(kstart.asm in PrKernel) */
   .text       :   load        =   DDR_INIT_SECT
   .const      :   load        =   DDR_INIT_SECT
   .cinit      :   load        =   DDR_INIT_SECT
   .bss        :   load        =   DDR_UNINIT_SECT
   .data       :   load        =   DDR_UNINIT_SECT
   .sysmem     :   load        =   DDR_UNINIT_SECT
   .nocache    :   load        =   DDR_NON_CACHE
   .stack      :   load        =   DDR_UNINIT_SECT
   .text:pagetable : load      =   DDR_INIT_SECT
}
```

4) Include the libraries.(Build the DSPLINK libraries and include)
5) Modify the psp_pros_main_sample.c file within the psp package to call samples with the appropriate arguments.
6) Build the project.
7) After loading the gel file press alt-c to connect the target.
8) Load the program.
9) After the loading the object file, press F10.

10) If you see Disconnect message thrown by CCS, Disconnect the CCS and reconnect again.

11) Press CTRL+SHIFT+L to reload the program binary.

12) Repeat step 9 and Press F5

13) Verify the correct execution of the samples through prints observed on the CCS output window.

- Follow the steps given in the PSP release notes to run the PSP sample project.
- CCS can attach to only ARM in the beginning. It can attach to the DSP only after the ARM-side application releases it from reset through a call to `PROC_Start ()`.

6 Platform specific information

6.1 Readwrite sample

The addresses to be passed as parameters for readwrite samples are platform specific.

Read write sample can be used for addresses in DDR, GEM L1D RAM and L2 RAM on DRX45x platform.

e.g :-

```
RDWR_Main ("/opt/readwrite.out", "2281308160", 2281308160, "1024",
1024, "1000", 1000,0) ;
```