

RELEASENOTES

DSP/BIOS™ LINK

LNK 111 REL

Version 1.65.00.02

MAR 30, 2010

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

Copyright ©. 2003, Texas Instruments Incorporated

This page has been intentionally left blank.

TABLE OF CONTENTS

1	Introduction	6
	1.1 Text Conventions.....	6
2	Out-of-Box Contents.....	7
3	Minimum System Requirements	8
	3.1 Hardware	8
	3.2 Software	8
4	Functionality Supported	10
	4.1 Features.....	10
	4.2 Feature enhancements from previous release 1.65.00.01	13
	4.3 Generic Defect fixes from previous release 1.65.00.01	14
	4.4 Platform-specific defect fixes.....	16
	4.5 Operating System specific defect fixes	17
	4.6 Upgrade and compatibility information	19
5	Validation Status	21
6	Known Issues.....	22
	6.1 Generic	22
	6.2 Platform Specific Issues	22
	6.3 Operating System specific Issues	25
7	Guidelines and Limitations	26
8	Technical Support.....	27
	8.1 Support for DSP/BIOS™ LINK	27

1 Introduction

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 development¹.

As the name suggests, DSP/BIOS operating system is expected to be running on the DSP with all of these platforms.

The platforms in this release are:

- DRX45X
- DRA44x
- DM6437 connected to DRA44x over VLYNQ interface
- DRX416
- DM6446 with WinCE
- OMAP3530 with WinCE
- TNETV107XGEM with WinCE

Additionally, support for the following platforms is available in the release:

- OMAP3530 EVM
- OMAP370 EVM
- OMAP2530 EVM
- DM6446 (Davinci)
- DM6467 (Davinci HD)
- DM6467T
- DA8XX
- OMAPL1XX
- Linux PC + DM6437 over PCI
- Linux PC + DM648 over PCI
- DM357
- Multi-DSP configuration: Linux PC + two DM6437 over PCI
- OMAPL138
- DA850

1.1 TextConventions



This bullet indicates important information.

Please read such text carefully.

¹ Applications differentiate the products. The application developers would prefer to focus on the application rather than the IPC mechanism.

-
- This bullet indicates additional information.
-

2 Out-of-BoxContents

This release of DSP/BIOS™ LINK contains the following:

1. Installer package for DSP/BIOS™ LINK containing:
 - TAR.GZ file containing the DSP/BIOS™ LINK installation package.
 - Scripts, tools etc., to use on the debug and development host machines.
2. Release Notes (this document) providing an overview of this release.
3. Installation Guide, Platform Guide, User Guide, Programmer's Guide and other technical documents.

Complete list of the documents is available in the User Guide.

3 MinimumSystemRequirements

3.1 Hardware

Depending on the target platform configuration, DSP/BIOS™ Link sources can be compiled on a PC running either Microsoft Windows or Red Hat Linux.


3.1.1 Development/DebugHostMachine


Please refer to installation guide for the relevant platform.

3.2 Software

3.2.1 GenericSoftwareRequirement

- PERL Installation
- TeraTerm (or any other terminal emulation program)
- One of GNU make version 3.81, 3.81beta1, 3.90 or 3.92.

 *On hosts where the above mentioned make versions are not available, DSPLINK users can use gmake (from TI XDC tools) to build DSPLink.*


 *To build DSPLink using gmake (from TI XDC tools) run gmake as below:*


<Absolute path of gmake from TI xdc tools>\$ gmake -s [debug|release].

For example,

1) /opt/xdc/<xdctools install path>/gmake -s debug

2) /opt/ti-tools/<bios install path>/xdctools/gmake -s debug

 *GNU make 3.81 version can be downloaded and installed from <http://ftp.gnu.org/pub/gnu/make/make-3.81.tar.gz>*

 *The software listed above is not part of the DSP/BIOS™ LINK release. For all the TI products, please contact your TI representative.*

 *ActivePerl - a distribution of PERL from Active State - can be obtained from the URL <http://www.activestate.com/Products/ActivePerl/>*

3.2.2 PlatformSpecificSoftwareRequirement

 ***For all platforms, this release will not work with versions of DSP/BIOS earlier than that specified in the dependencies.***

3.2.2.1 ForWorkingwithDRA44x

- eBinder version ver 1.7 rev0 with valid licenses.
- JTAGJet emulator driver available from Signum Systems
- J1_BSP_v1.6.0beta.zip : Board Support Package(BSP)
- pre-drx40x-psp-rel0.1.0: TI PSP release.
- PrF2_ver1.3rev1.zip : PrFILE2 (including RAM disk driver) : "Board Support Package" and consists of board dependent modules.
- BGX_95V_100_000.zip : Patch for PrFILE2.

3.2.2.2 *ForWorkingwithDRX416*

- eBinder version ver 1.7 with valid licenses.
- JTAGJet emulator driver available from Signum Systems
- eSOL Board Support Package(BSP)
- DSP/BIOS patch for J2 (TI internal).
- TI PSP release REL_DRX40X_PSP_0.6.0.zip (TI internal) supporting eSOL release for DRX41x.
- TI EDMA release EDMA3_LLD_setupwin32_1_05_00_01.exe
- PrF2_ver1.3rev1.zip: PrFILE2 (including RAM disk driver): "Board Support Package" and consists of board dependent modules.
- BGX_95V_100_000.zip: Patch for PrFILE2.
BGX_963_100_000.zip.

3.2.2.3 *ForWorkingwithDRX45X*

- For Standalone DSP/BIOS™ 5.33.05
 - CCS 3.3 IDE
 - DSP CG Tools version v6.0.14 or greater
- PROS Kernel release (REL_DRX45X_03.00.00.03.01.zip)
 - Toolchain (MS470 C/C++ CODE GENERATION TOOLS Release 4.1.4)

3.2.2.4 *ForWorkingwithDM6446*

- DSP/BIOS™ version can be 5.32.04 or 6.20.00.13.2 (one of the below)
- For Standalone DSP/BIOS™ 5.32.04.
 - DSP CG Tools version v6.0.18 or greater
 - CCS 3.3 IDE for debugging purposes.
- For Standalone DSP/BIOS™ bios 6.20.00.13.2.
- XDC release version xdc 3.15.00.13
- CCS 3.3.38.2 for debugging purposes.
- DSP CG tools version v6.1.5.
- Windows CE release
- Visual Studio 2005 with Platform Builder R2.
- With Davinci BSP for WinCE 6.0.

3.2.2.5 *ForworkingwithOMAP3530*

- Standalone DSP/BIOS™ 5.32.04.
- DSP CG Tools version v6.0.18 or greater
- CCS 3.3 IDE
- Windows CE release
- Visual Studio 2005 with Platform Builder R2.
- With OMAP3530 BSP v 6.11 for WinCE 6.0.

3.2.2.6 *ForworkingwithTNETV10*

- Standalone DSP/BIOS™ 5.41.00.06
- DSP CG Tools version v6.0.18 or greater
- CCS 3.3 IDE
- Windows CE release
- Visual Studio 2005 with Platform Builder R2.
- With TNETV10 BSP for WinCE 6.0.

4 FunctionalitySupported

4.1 Features

4.1.1 GenericFeatures

4.1.1.1 *Supportformultipleplatforms*

DSPLink supports multiple platforms in specific configurations as mentioned in this document. The standard features of DSP/BIOS™ LINK are supported with this release. For additional information, please refer to the User Guide and Programmer's Guide documents.

4.1.1.2 *Multi-processandmulti-applicationsupport*

Multiple applications or processes can directly call DSP/BIOS LINK PROC APIs to setup, control the DSP and destroy the DSPLink driver. They do not need to coordinate between themselves at run-time to access the DSPLink driver. The only thing they need to ensure is:

1. The same DSP executable is used by all applications simultaneously using DSPLink
2. The same dynamic configuration is used by all applications simultaneously using DSPLink

4.1.1.3 *SupportformultipleDSPsconnectedtoGPP instar-topology*

DSPLink supports multiple DSPs connected to the master GPP processor in a star-topology. With this feature, it is possible to connect multiple DSPs to the GPP in the system, over heterogeneous physical links. Example reference ports have also be included for multi-DSP configurations:

- o Linux PC connected to two DM6437 devices over PCI
- o DRA44x connected to external DM6437 over VLYNQ

The DSPLink static build configuration supports choosing the desired device and build configuration. The static build configuration supports command-line based usage to enable scripting.

4.1.2 PROCmodule

4.1.2.1 *DSPboot-load*

This release supports boot-loading the DSP from the ARM. This includes:

- Initializing the DSP & making it available for access from the GPP.
- On OMAP platforms: Configuring the DSP MMU to allow straight-through access for specified memory regions.
- Loading code on the DSP. Multiple types of loaders can be used for this. DSPLink provides a few standard loaders as part of the release package. Additional loaders can be added and plugged into DSPLink.
- Starting execution from the run address specified in the executable.
- Reading from or writing to DSP memory.

- Stopping DSP execution.
- DSP/BIOS LINK for the OMAP platforms does not perform any power management of the DSP. This is expected to be done by u-boot or other power-management utilities.

If PROC-only scalability configuration build is used, DSPLink does not perform driver handshake and interrupts initialization. Due to this, DSPLink can be used in PROC-only configuration to load and start a non-DSPLink DSP executable, i.e. a DSP executable that has no DSPLink content or shared memory usage within it.

Support for multiple boot modes

DSPLink supports three different types of DSP boot modes:

- DSPLINK_BOOT_MODE: Default
 - GPP boots first
 - Uses DSPLink to load the DSP
 - Uses DSPLink to start the DSP running
- DSPLINK_NOLOAD_MODE: External DSP load
 - GPP boots first
 - Application/GPP boot-loader pre-loads the DSP
 - Uses DSPLink to optionally power up the DSP
 - Uses DSPLink to start the DSP running
- DSPLINK_NOBOOT_MODE: External DSP start. This supports two scenarios
 - GPP-based load
 - GPP boots first
 - Application/GPP boot-loader pre-loads the DSP
 - Application/GPP boot-loader starts the DSP running
 - Uses DSPLink only for IPC with the DSP

OR

 - DSP-based load
 - DSP boots first, starts running an application
 - GPP comes up later and sets up DSPLink, which initializes shared memory
 - DSPLink is not used to load or start the DSP
 - Uses DSPLink only for IPC with the DSP

For complete details on the boot modes along with information on configuration and application usage, please refer to the Programmer's Guide document.

4.1.2.2 Dynamic configuration

DSP/BIOS™ LINK can be dynamically configured with parameters specific to the system configuration. This can be decided by the system integrator. Usage of a new dynamic configuration does not require re-build of DSPLink libraries.

4.1.3 Inter-processor communication modules

4.1.3.1 POOL

The POOL component supports configuring and using shared memory buffers across processors. This includes

- Configuring the shared memory region through open & close calls.
- Allocating and freeing buffers from the shared memory region.
- Translating address of a buffer allocated to different address spaces (e.g. GPP to DSP)
- Synchronizing contents of memory as seen by the different CPU cores (Not applicable for the OMAP3530 platform)

4.1.3.2 NOTIFY

The NOTIFY component allows applications to register for notification of events occurring on the remote processor and send event notification to the remote processor.

It uses the physical interrupt present between the processors and provides the facility of multiple prioritized events over the same physical interrupt.

4.1.3.3 MPCS

The MPCS component allows applications to achieve mutually exclusive access to shared data structures through a multi-processor critical section (MPCS) between GPP and DSP.

4.1.3.4 MPLIST

The MPLIST component provides a doubly-linked circular linked list based transport mechanism between GPP and DSP.

It provides APIs for pushing and popping buffers from the list, traversing the list, inserting and removing elements and other services. The list is shared between the two processors.

4.1.3.5 CHNL

The CHNL component provides issue-reclaim based data streaming. It is based on the SIO module in DSP/BIOS™.

4.1.3.6 MSGQ

The MSGQ component provides queue based messaging. It is an acronym for 'message queue'.

This component is responsible for exchanging short messages of variable length between the GPP and DSP clients. It is based on the MSGQ module in DSP/BIOS™.

The messages are sent and received through message queues.

This component is based on the MSGQ module in DSP/BIOS™.

4.1.3.7 RINGIO

The RingIO component provides circular ring buffer based data streaming. Each ring buffer can be operated on by a single reader and single writer.

4.2 Feature enhancements from previous release 1.65.00.01

This release is binary compatible to previous release 1.65.00.01

4.2.1 Move to Linux GIT kernel for DM6446

DSPLink has moved from Montavista Linux releases to GIT releases for Linux for DM6446. The tool chain used is arm 2009q1-203.

The distribution files have been renamed to be consistent with the naming convention used for working with GIT based Linux kernels. These files are present in \$DSPLink/make folder.

These files are present in \$DSPLink/make folder.

- davinci_mvlpro5.0.mk renamed to -> davinci_2.6.mk
- davinci_uclibc5.0.mk renamed to -> davinci_uclibc.mk

The options to configure DSPLink for DM6446 using dsplinkcfg script have also changed.

For DM6446 -

```
perl dsplinkcfg.pl --platform=DAVINCI --nodsp=1 --dspcfg_0=DM6446GEMSHMEM --  
dspos_0=DSPBIOS5XX --gppos=DM6446LSP --comps=ponslrmc
```

or

```
perl dsplinkcfg.pl --platform=DAVINCI --nodsp=1 --dspcfg_0=DM6446GEMSHMEM --  
dspos_0=DSPBIOS5XX --gppos=DM6446LSPuc --comps=ponslrmc
```

4.2.2 Added support for OMAP3730

This release adds support for OMAP3730 as same device as OMAP3530. The same ARM/DSP binary executable will be run on both OMAP3730 and OMAP3530. Even though the memory map, clock rate may be different, no new platform configuration is exposed to the user. The user does not see a platform called OMAP3730 but updates the OMAP3530 configuration for OMAP3730. A single binary is generated for both devices in the same device directory. User will not have side-by-side libraries/binaries generated for the two devices for both ARM & DSP. This implies that DSPLink will not be able to provide different default memory maps for the two devices.

4.2.3 LSP version upgrade for OMAP3530

The LSP version has been upgraded to PSP release PSP SDK AM35x-OMAP35x-PSP-SDK-03.00.00.05.

4.2.4 LSP version upgrade for DA850/OMAPL138

The LSP version has been upgraded to LSP kernel release DaVinci PSP SDK 03.20.00.11 with GIT tool chain arm-2009q1-203.

4.2.5 LSP version upgrade for DA8xx/OMAPL13xx

The LSP version has been upgraded to LSP kernel release DaVinci PSP SDK 03.20.00.11 with GIT tool chain arm-2009q1-203.

4.3 GenericDefectfixesfrompreviousrelease1.65 .00.01

Identifier	Headline
SDOCM00064915	Update DSPLink CFG files to set POOLSIZE to POOLMEMORYSIZE for all the devices

Description:

Update DSPLink CFG files to set POOLSIZE to POOLMEMORYSIZE for all the devices

Update DSPLink CFG files to set POOLSIZE to POOLMEMORYSIZE for all the devices as the value 0x70000 confuses the application writers

Expected Result:

DSPLink CFG files should use POOLSIZE as POOLMEMORYSIZE and not 0x70000 for all the devices

Release Note:

DSPLink CFG files have been updated to use POOLSIZE as POOLMEMORYSIZE and not 0x70000 for all the devices.

Identifier	Headline
SDOCM00066145	Macro Definition of OPT (Optional Parameter in DSPLink) conflicting with OPT filed of EDMA PaRAM defined in CSL

Description:

A macro named OPT is defined to nothing in dsplink.h. There is a field called OPT in the EDMA PARAM structure of CSL definition. Since the name of the macro and the field are same, there are compilation errors when any application which is using DSPLink and CSL is trying to populate the OPT field. Since this field is set to nothing by the pre-processor as per the definition in dsplink.h, and at the compilation stage it is actually "nothing" being assigned to a value. Thus the compilation error.

Expected Result:

System integrator should be able to build an application with both DSPLink and CSL headers in it.

Release Note:

Macro definition of OPT has been removed from DSPLink source code for both gpp and dsp sources.

Identifier	Headline
SDOCM00065499	In-place operations on the Reader RingIO buffer require cache invalidation before releasing the buffer(RingIO_release)

Description:

ARM is writer and DSP is reader, in DSP we are performing some operation on the RingIO Buffer i.e. DSP manipulates the data and writes to the same buffer(RingIO acquired). As soon as this buffer is released from DSP, ARM writes data to this buffer, this data is not reflecting in the DSP when it is acquired, it is partially modified. Looks some cache issue while releasing the buffer.

Cache write back in DSP is happening for part of the memory after GPP RingIO release that's why data gets corrupted.

BCACHE_inv() API is called before every RingIO release in DSP.
To use Reader RingIO buffer, is it required cache lines to write back before release

Expected Result:

RingIO should support in place buffer operations.

Release Note:

Cache invalidation has been added to the following RingIO API's to support in buffer operations

- RingIO_readerRelease
- RingIO_readerCancel
- RingIO_writerCancel

4.4 Platform-specific defect fixes

4.4.1 DM6446 specific

None.

4.4.2 DA8xx/OMAPL1XX specific

None.

4.4.3 OMAP2530 specific

None.

4.4.4 OMAP3530 specific

None.

4.4.5 DM6437 specific

None.

4.4.6 DM648 specific

None.

4.4.7 DM357

None.

4.4.8 DM6467T

None.

4.5 OperatingSystemsspecificdefectfixes

4.5.1 WinCESpecific

None.

4.5.2 PrOSSpecific

Identifier	Headline
SDOCM00065756	MPCS deadlock issue in Delphi system

Description:

Deadlock seen in the system when there are reversed priorities at ARM & DSP tasks. The system hangs and nothing happens afterwards. HMI freezes at customer site.

Expected Result:

No application deadlock should be seen. Application writer should not have to program task specific priority values. MPCS code should handle priority inversion related problems.

Release Note:

MPCS code has been updated to add TSK mode SWI mode on both gpp and dso sources.

Updated definition:

- Task mode:
 - DSP side: The local lock in individual MPCS object has been updated to use LCK instead of SEM module. The local lock in each individual MPCS object has been updated to use a single global lock that supports nesting.
 - GPP side: The local lock in individual MPCS object has been updated to use mutex module. The local lock in each individual MPCS object has been updated to use a single global lock that supports nesting.
- Swi mode:
 - DSP side: The scheduler is disabled to allow current task to run to completion.
 - GPP side: Priority boosting is done where in the current task priority is boosted to a high priority to allow current task to run to completion.

Identifier	Headline
SDOCM00060291	<Document> Update Guidelines section in Release Notes for multi-processor priority inversion issue due to MPCS protection

Description:

Latency seen in the system when there are reversed priorities at ARM & DSP tasks.

Expected Result:

No application deadlock should be seen. Application writer should not have to program task specific priority values. MPCS code should handle priority inversion related problems.

Release Note:

This has been fixed indirectly by fixing SDOCM00065756 Application writer should not have to program task specific priority values. MPCS code should handle priority inversion related problems

4.6 Upgradeandcompatibilityinformation

4.6.1 UpdationofDSPLinkDM6446port

DSPLink 1.65.00.02 is API and binary compatible with DSPLink 1.65.00.01. Other changes that are relevant to applications are listed below.

DSPLink has moved from Montavista Linux releases to GIT releases for Linux for DM6446. The tool chain used is arm 2009q1-203.

The distribution files have been renamed to be consistent with the naming convention used for working with GIT based Linux kernels. These files are present in \$DSPLink/make folder.

These files are present in \$DSPLink/make folder.

- davinci_mvlpro5.0.mk renamed to -> davinci_2.6.mk
- davinci_uclibc5.0.mk renamed to -> davinci_uclibc.mk

The options to configure DSPLink for DM6446 using dsplinkcfg script have also changed.

For DM6446 -

```
perl dsplinkcfg.pl --platform=DAVINCI --nodsp=1 --dspcfg_0=DM6446GEMSHMEM --
dspos_0=DSPBIOS5XX --gppos=DM6446LSP --comps=ponslrmc
```

or

```
perl dsplinkcfg.pl --platform=DAVINCI --nodsp=1 --dspcfg_0=DM6446GEMSHMEM --
dspos_0=DSPBIOS5XX --gppos=DM6446LSPuc --comps=ponslrmc
```

4.6.2 LSPversionupgradeforOMAP3530

The LSP version has been upgraded to PSP release PSP SDK AM35x-OMAP35x-PSP-SDK-03.00.00.05.

4.6.3 LSPversionupgradeforDA850/OMAPL138

The LSP version has been upgraded to LSP kernel release DaVinci PSP SDK 03.20.00.11 with GIT tool chain arm-2009q1-203.

4.6.4 LSPversionupgradeforDA8xx/OMAPL13xx

The LSP version has been upgraded to LSP kernel release DaVinci PSP SDK 03.20.00.11 with GIT tool chain arm-2009q1-203.

4.6.5 Pros Specific: Application changes related to SDOCM00065756: MPCSDeadlock issue in Delphi system

MPCS code has been updated to add TSK mode SWI mode on both gpp and dso sources.

Updated definition

- Tsk mode:
 - DSP side: The local lock in individual MPCS object has been updated to use LCK instead of SEM module. The local lock in each individual MPCS object has been updated to use a single global lock that supports nesting.

- GPP side: The local lock in individual MPCS object has been updated to use mutex module. The local lock in each individual MPCS object has been updated to use a single global lock that supports nesting.
- Swi mode:
 - DSP side: Protection between tasks is done by disabling the scheduler.
 - GPP side: : Protection between tasks is done by boosting the priority where in the current task priority is boosted to a high priority to allow current task to run to completion.

Change in dsplinkcfg script

- Tsk mode: No change
- Swi mode:
 - A extra input for Swi mode to accept priority to which current task should be boosted in MPCS_enter has been added

To maintain backward compatibility, no input is mandatory. The default value which corresponds to the highest task priority supported by ProS will be used.

```
perl dsplinkcfg.pl --platform=JACINTO1 --nodsp=1 --
dspcfg_0=DRA44XGEMSHMEM --dspos_0=DSPBIOS5XX --gppos=PROS --
comps=ponslrm --fs=PSEUDOFs
```

To specify priority

```
perl dsplinkcfg.pl --platform=JACINTO1 --nodsp=1 --
dspcfg_0=DRA44XGEMSHMEM --dspos_0=DSPBIOS5XX --gppos=PROS --
comps=ponslrm --fs=PSEUDOFs -swipriority=99
```

Addition of GPP build specific defines

To match task mode and swi mode on DSP, corresponding build flags have been added to GPP specific defines as well.

- GPP_TSK_MODE – Task mode
- GPP_SWI_MODE – Swi mode

4.6.6 WinCE Specific: Application changes related to SDOCM00065756: MPCS deadlock issue in Delphi system

To match task mode and swi mode on DSP, corresponding build flags have been

added to GPP specific defines as well.

GPP_TSK_MODE – Task mode

GPP_SWI_MODE – Swi mode

However, they are not used in the code.

5 ValidationStatus

This engineering release has been validated for basic sanity using the following configurations:

Sr. No	Platform	GPP-side			DSP-side	
		OS distribution	Tool-chain	Base port	Tool-chain	DSP/BIOS
1	DRA4xx	eSOL PrKernel v4 ver3.1 rev5	RVCT toolchain 2.2	pre-drx40xpsp-0.4.0	CGTOOLS v6.0.8 with CCS3.3	BIOS - 5.32.04
2	DM6446 Davinci BSP	for WinCE	6.0	Visual Studio	2005 with	Platform

Distribution files for Uclibc tool-chains are provided for OMAP2530, OMAP3530, DM6446, DM6467, DM6467T, DA850, DA8xx, OMAPL1xx and OMAPL138.

6 Known Issues

6.1 Generic

Identifier	Headline
SDOCM00035970	DSPLINK does not build correctly in CYGWIN environment

Description :

DSPLink make system does not build DSP-side sample applications correctly in CYGWIN environment. However, DSPLink libraries on DSP-side and GPP-side build can be completed successfully in CYGWIN environment.

Workaround:

Remove CYGWIN from the path on MSDOS prompt and build DSP-side sample applications using gmake.

Identifier	Headline
SDOCM00056316	Intermittent crashes are seen when running DSPLink sample applications on Montavista Linux based devices

Description :

On devices running Montavista Linux on the ARM (DM6446, DM6467, DM357, DA8xx, OMAP-L1xx), intermittent crashes are seen when running the sample applications. Crash is not seen on devices where GIT kernel is used (OMAP3530, OMAP2530, DM6437, DM648).

Observation:

The crash appears to happen when interrupts are sent very frequently to the ARM from the DSP.

Workaround:

None.

6.2 Platform Specific Issues

6.2.1 DM6446

Identifier	Headline
SDOCM00056114	CE appln fails if user calls the sequence: 1) eng = Engine_open("myEng",.); 2) Engine_close(eng); 3) eng = Engine_open("myEng

Description :

An application will fail if it tries to do the following:

```
eng = Engine_open("myEngine", ...);
Engine_close(eng);
eng = Engine_open("myEngine", ...);
```

As far as DSPLink is concerned, this boils down to:

```

Engine_open:
    LAD_startupDsp() =====switch-to-LAD-process=====>
PROC_setup(); PROC_attach(); etc.
    PROC_attach()
Engine_close:
    LAD_releaseDsp() =====switch-to-LAD-process=====>
PROC_destroy();
    
```

Workaround:

None.

6.2.2 DM6467

Identifier	Headline
DSPLN00000734	<PROC> DSP-side configuration mismatch/failure is thrown if a test is run infinite number of times

Description :

<PROC> DSP-side configuration mismatch/failure is thrown if a test is run infinite number of times

Workaround:

Power cycle the EVM and re-try.

6.2.3 DM648

None.

6.2.4 DM6437

Identifier	Headline
SDOCM00044596	<sample> scale sample hangs in SWI mode on both debug and release modes.

Description :

Observation :

scale sample hangs only in SWI mode on both debug and release modes.

Same sample works fine on TSK mode.

Workaround:

None.

6.2.5 OMAP2530

Identifier	Headline
SDOCM00035953	OMAP2530: First run of any application times out in PROC_start.

Description :

The first run of any DSPLink application times out in PROC_start. The subsequent application runs without re-booting the board in between work correctly.

Workaround:

The GPP sources have been updated to reset and release the DSP twice as a workaround to ensure that DSP starts correctly in the first run. This workaround is present in the DSPLink release.

6.2.6 DM357

None.

6.2.7 OMAPL1XX
6.2.7.1 OMAPL1XX

DIO and SIO create calls are not supported in DSPLink SWI Mode for DSP_BootMode_NoBoot boot mode because of a DSP/BIOS restriction in calling these API's before main. Sample applications i.e. loop, scale with SWI's will not work in DSP_BootMode_NoBoot mode. DSPLink supports SWI based sample applications in the other boot modes DSP_BootMode_Boot_NoPwr, DSP_BootMode_Boot_Pwr, DSP_BootMode_NoLoad_NoPwr, DSP_BootMode_NoLoad_Pwr

6.2.7.2 Defects

Identifier	Headline
DOCM00051547	<Samples> Loop sample stops execution in-between on swi release mode.

Description :
Observation :

Loop sample stops execution in-between on swi release mode on both DA8xx and OMAPL1xx platforms.

The sample works fine on swi debug mode.

The sample won't hang, after ctrl+c able to run other samples without rebooting EVM.

Workaround:

None.

6.2.8 DA8XX/DA850/OMAPL138

DIO and SIO create calls are not supported in DSPLink SWI Mode for DSP_BootMode_NoBoot boot mode because of a DSP/BIOS restriction in calling these API's before main. Sample applications i.e. loop, scale with SWI's will not work in DSP_BootMode_NoBoot mode. DSPLink supports SWI based sample applications in the other boot modes DSP_BootMode_Boot_NoPwr, DSP_BootMode_Boot_Pwr, DSP_BootMode_NoLoad_NoPwr, DSP_BootMode_NoLoad_Pwr

6.3 OperatingSystemspecificIssues

6.3.1 PrOS

None.

6.3.2 WinCE

Known issues in the DM6446 Debug package of WinCE BSP prevent us from using Debug build on DM6446.

6.3.3 DSP/BIOS

Identifier	Headline
CQ17490	BIOS/XDC conf tool breaks DSP build, if the DSPLink installation directory contains `.` Character.

Description :

DSP Side build system generates all files in the build folder instead of sample directories. This would cause error if the DSPLink installation directory has `.` Character. For example, if the installation path is

/home/user/dsplink.160/

then DSP Side build will break.

Workaround:

Install DSPLink without a `.` In the path.

7 Guidelines and Limitations

None.

8 TechnicalSupport

8.1 SupportforDSP/BIOS™LINK

8.1.1 TIExternalwiki

A significant amount of information about DSPLink including FAQs and troubleshooting guides are present and the TI external wiki. These can be used as a first-level source of information for DSPLink:

<http://www.tiexpressdsp.com>

8.1.2 SupportforTIproducts

Procedure to report problems/issues for TI products is available at the TI support page:

External forums at: <http://community.ti.com>

Or: <http://support.ti.com>