
CPPI LLD

Release Notes

Applies to Product Release: 02.00.00.13
Publication Date: August 20, 2013

Document License

This work is licensed under the Creative Commons Attribution-NoDerivs 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nd/3.0/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Contributors to this document

Copyright (C) 2013 Texas Instruments Incorporated - <http://www.ti.com/>



Texas Instruments, Incorporated
20450 Century Boulevard
Germantown, MD 20874 USA

Contents

- Overview 1
- LLD Dependencies 1
- New/Updated Features and Quality 1
- Resolved Incident Reports (IR) 3
- Known Issues/Limitations 4
- Licensing..... 4
- Delivery Package 4
- Installation Instructions..... 4
 - Directory structure..... 4
- Customer Documentation List 6

CPPI LLD version 02.00.00.13

Overview

This document provides the release information for the latest CPPI Low Level Driver which should be used by drivers and application that interface with CPPI IP.

CPPI LLD module includes:

- Compiled library (Big and Little) Endian of CPPI LLD.
- Source code.
- API reference guide
- Design Documentation

LLD Dependencies

LLD is dependent on following external components delivered in PDK package:

- CSL
- QMSS LLD
- RM LLD

New/Updated Features and Quality

This is an **engineering release**, tested by the development team for early integration effort

Release 2.0.0.13

- Added shared object library support for ARM user space applications

Release 2.0.0.12

- Add RM to cppiTest.out and cppiExample.out for the ARM. This requires starting rm server with following command line:
 - rmServer.out rm/device/k2h/global-resource-list.dtb
rm/device/k2h/policy_dsp_arm.dtb

Release 2.0.0.11

- Remove warnings for -Wunused-parameter for gcc (arm).
- Remove warnings from -Wall on ARM

- Add XGE to k2k but not k2h. Add support to reject DMAs not supported on device.
- Cppi_initDescriptor() API can returns incorrect queue for descriptor return queue

Release 2.0.0.10

- Update for Yocto Build Recipe support
- Update to makefiles to support Linaro tool chain: 2013.03 version
- Make example and unit test work on both DSP and ARM
 - Note: in order to run on ARM, it requires loading and activating kernel module which opens memory protection for the QM/CPPI to user space. This must be done once per boot of EVM.

```
insmod hplibmod.ko
cat /proc/netapi
```

- Make all QM and CPPI examples “restartable” from the ARM. This means each can be run sequentially (or multiple times) without rebooting the EVM.

Release 2.0.0.9

- Fix Cppi_initDescriptor() calling Qmss_getMemRegDescSize() with backwards arguments.
- Fully integrated Keystone II RM. If RM Server Handle equals NULL the LLD will operate in backwards compatibility mode as if RM does not exist. The CPPI test and example projects have been updated to use RM.

Release 2.0.0.8

- Aligned Resource Manager callouts with new Keystone II RM APIs. Only RM Service Handle equals NULL has been tested with LLD.

Release 2.0.0.7

- Synchronize with keystone 1. Rebase to 1.0.2.4 from 1.0.2.3 (see below).
- Remove cppi_types.h/qmss_types.h since LLDs use c99 types. No longer need to add ti/drv/qmss and ti/drv/cppi as include paths.

Release 2.0.0.6

- Fixed errors found in user mode LLDs example/test projects building

Release 2.0.0.5

- Bug fixes.

- Renamed the device specific folders as per new naming conventions.
- Support for TCI6636K2H device (k2h).

Release 2.0.0.4

- Updates for using auto-generated csLr_device.h and csL_device_interrupt.h files.

Release 2.0.0.3

- Modification for single LLD library to work for all platforms. Moved the default location of C66x libraries to lib\c66x inside component directory
- Build support for ARMv7 user mode target. Limited build verification in this release.

Release 2.0.0.2

- Add missing peripherals to cpPi_device.c
- Add consistency check for cpPi_device.c, that causes CpPi_init() to return error if the device table is bad.

Release 2.0.0.1

- Modify the unit test to be device independent by removing hardcoded qm base address. Instead allow new feature to take base addresses from cpPi_device.c to take effect.
- Cache coherence fixes for refCnt.

Release 2.0.0.0

- Add configuration of the QM base addresses. Assuming the cpPi_device.c is used, this operation is completely transparent to the user.
- KeyStone2 devices have new directory structure for devices, example and test folders

Release 1.0.2.4

- Change cpPi_device.c to remove (void *) casts and replace with the actual types. This allows cpPi_device.c to be compiled with a c++ compiler.
- Clarify documentation for rx flow size thresholds (only comments changed, no functional change to code).

Resolved Incident Reports (IR)

Table 1 provides information on IR resolutions incorporated into this release.

Table 1 Resolved IRs for this Release

IR Parent/ Child Number	Severity Level	IR Description
SDOCM00102814	S2 - Major	LLD makefiles need to support shared libraries

Known Issues/Limitations

IR Parent/ Child Number	Severity Level	IR Description
BCG_IP_P.BT S_pa_cdmahp.4 01	Major	CDMAHP RX PS location bit is not updated in CPPI descriptor

Licensing

Please refer to the software Manifest document for the details.

Delivery Package

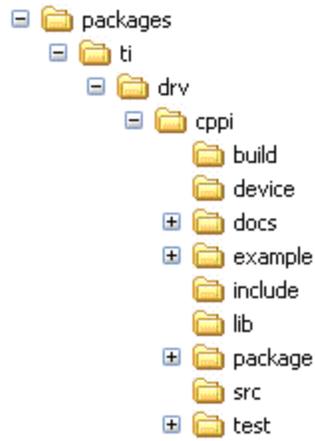
There is no separate delivery package. The CPPI LLD is being delivered as part of PDK.

Installation Instructions

The LLD is currently bundled as part of Platform Development Kit (PDK). Refer installation instruction to the release notes provided for PDK.

Directory structure

The following is the directory structure after the CPPI LLD package has been installed:



The following table explains each individual directory:

Directory Name	Description
ti/drv/cppi	The top level directory contains the following:- <ol style="list-style-type: none"> <u>Environment configuration batch file</u> The file “setupenv.bat” is used to configure the build environment for the CPPI low level driver. <u>XDC Build and Package files</u> These files (config.bld, package.xdc etc) are the XDC build files which are used to create the CPPI package. <u>Exported Driver header file</u> Header files which are provided by the CPPI low level driver and should be used by the application developers for driver customization and usage.
ti/drv/cppi/build	The directory contains internal XDC build related files which are used to create the CPPI low level driver package.
ti/drv/cppi/device	The directory contains the device specific files for the CPPI low level driver.
ti/drv/cppi/docs	The directory contains the CPPI low level driver documentation.
ti/drv/cppi/example	The “example” directory in the CPPI low level driver has the infrastructure mode example.
ti/drv/cppi/include	The “include” directory has private CPPI low level driver header files. These files should not be used by application developers.
ti/drv/cppi/lib	The “lib” folder has pre-built Big and Little Endian libraries for the QMSS low level driver along with their <u>code/data size information</u> .
ti/drv/cppi/package	Internal CPPI low level driver package files.
ti/drv/cppi/src	Source code for the CPPI low level driver.
ti/drv/cppi/test	The “test” directory in the CPPI low level driver has unit test cases which are used by the development team to test the CPPI low level driver.
eclipse	The “eclipse” directory has files required to integrate CPPI low level

driver documentation with Eclipse IDE's Help Menu.

Customer Documentation List

Table 2 lists the documents that are accessible through the **/docs** folder on the product installation CD or in the delivery package.

Table 2 Product Documentation included with this Release

Document #	Document Title	File Name
1	API documentation (generated by Doxygen)	docs/cppilldDocs.chm
2	Design Document	docs/CPPI_QMSS_LLD_SDS.pdf
3	Software Manifest	docs/CPPI_LLD_SoftwareManifest.pdf