

FFTC Driver

Release Notes

Applies to Product Release: 2.00.00.05
Publication Date: July 22, 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) 2012-2013 Texas Instruments Incorporated - <http://www.ti.com/>



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

VP00102-Form-1
Revision D

Contents

Overview.....	1
LLD Dependencies	1
New/Updated Features and Quality	1
Resolved Incident Reports (IR)	2
Known Issues/Limitations.....	2
Licensing	2
Delivery Package.....	2
Installation Instructions.....	2
Test and Example	3
FFTC Test.....	4
FFTC Example.....	4
Customer Documentation List.....	5

FFTC Driver version 2.00.00.05

Overview

This document provides the release information for the latest FFTC driver which should be used by drivers and application that interface with FFTC IP.

FFTC Driver module includes:

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

LLD Dependencies

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

- CSL
- CPPI LLD
- QMSS LLD

New/Updated Features and Quality

This is an **engineering release**, tested by the development team. New and updated features are in reference to version FFTC Driver 01.00.00.

Release 2.00.00.05

- Fix for IR SDOCM00102292: FFTC driver not compatible with QMSS using RMv2

Release 2.00.00.04

- Fix for IR SDOCM00100872. Example and test failure

Release 2.00.00.03

- Fix for IR SDOCM00098765. Update during EVM bringup.

Release 2.00.00.02

- Renamed the device specific folders tci6634 to k2k as per new naming conventions.
- Support for TCI6636K2H device (k2h).

Release 2.00.00.01

- Updates to work with auto generated CSL device file.

Release 2.00.00.00

- One library to support multiple devices and updated RTSC scripts accordingly.
- One fix to the test OSAL related to L2 cache protection.
- Updated test and example bios projects for testing with changes in the dependent LLDs.
- Fixed an issue with Multiple Instance test project.
- Tested with new CSL, QMSS and CPPI.

Release 2.00.00.1000

- Driver directory structure modified to support Keystone-II platforms to support multiple devices. This release supports C6634 device only.
- Tested with new CSL, QMSS, CPPI, SRIO, BIOS & XDC tools which has support for C6634 device.
- Test code is migrated, so runs only in first 4 cores for this release.
- Release notes document is added to the release.

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
SDOCM00100 872	Major	fftcSimpleK2KC66BiosTestProject does not complete

Known Issues/Limitations

Table 2 Known Issue IRs for this Release

IR Parent/ Child Number	Severity Level	IR Description

Licensing

Please refer to the software Manifest document for the details.

Delivery Package

There is no separate delivery package. The FFTC Driver 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 FFTC driver package has been installed:



The following table explains the contents of the FFTC package:-

Directory Name	Description
ti/drv/fftc	<p>The top level directory contains the following:-</p> <ol style="list-style-type: none"> <u><i>XDC Build and Package files</i></u> These files (config.bld, package.xdc etc) are the XDC build files which are used to create the FFTC package. <u><i>Exported Driver header file</i></u> Header files which are provided by the FFTC driver and should be used by the application developers for driver customization and usage.
ti/drv/fftc/build	The directory contains internal XDC build related files which are used to create the FFTC Driver package.
ti/drv/fftc/device	The directory contains the device specific files for the FFTC device driver.
ti/drv/fftc/docs	The directory contains the FFTC driver documentation.
ti/drv/fftc/example	The “example” directory in the FFTC driver has a usage example which explains how the FFTC driver API’s are used to send and receive data.
ti/drv/fftc/include	The “include” directory has private FFTC driver header files. These files should not be used by application developers.
ti/drv/fftc/lib	The “lib” folder has pre-built Big and Little Endian libraries for the FFTC driver along with their <u><i>code/data size information</i></u> .
ti/drv/fftc/package	Internal FFTC driver package files.
ti/drv/fftc/src	Source code for the FFTC Driver.

Test and Example

The section documents information about the test and example code located in the FFTC driver.

FFTC Test

The FFTC Driver Unit Test Suite is aimed at testing all the layers of FFTC driver, i.e, CSLR, LLD and the FFTC higher layer APIs.

The Unit tests here test the following:

- i. Verify the defaults for all FFTC Registers defined in FFTC CSLR at reset against the spec.
- ii. Test all the FFTC Low Level driver (LLD) APIs working for FFTC_A and FFTC_B instances.
- iii. Test all the FFTC Higher layer APIs using CPPI and QMSS libraries. Run tests to verify various use cases. Specifically, the tests here test for:
 - Various sample sizes (16, 48, 540, 2048) with multiple FFT blocks
 - Single core testing
 - Multi core testing
 - Host mode and Monolithic descriptors
 - Protocol specific pass through Info (PS Info) in SOP and Descriptor
 - Various FFT Configurations such as Zero padding, Variable shifting, cyclic prefix addition, mixed size DFT list configuration.
 - Polling mode and High Priority Accumulation interrupts to retrieve results from the engine
 - Multiple FFTC instances

FFTC Example

The FFTC Multi core example demonstrates the use of FFTC APIs to submit multi-block FFT requests and retrieve their results on various cores simultaneously.

The example uses the following configuration:

- FFTC_A instance
- 1 packet each with 16 DFT sample size * 5 blocks
- Host mode CPPI descriptors
- No PS Info
- High priority accumulation interrupts
- with Rx object in blocking mode
- all cores using FFTC Tx queue 0
- uses "fftc_cfg_16.h" for DFT input and output reference data.

The example code is setup such that the result corresponding to an FFT request submitted by one core is processed by another core on the device. The following figure demonstrates the example's multi-core setup:

core 0 --> core 1 --> core 2 --> core 3 --> core 0

Core 0 acts as a master core and performs the system initialization. After which all the cores continue to setup the FFTC driver, Tx objects, Rx objects, submit FFT request and verify result received.

To see the multicore example application in action, please build and run the multi core example project on all 4 cores in "synchronous" mode.

Customer Documentation List

Table 3 lists the documents that are accessible through the **/docs** folder on the product installation CD or in the delivery package.

Table 3 Product Documentation included with this Release

Document #	Document Title	File Name
1	API documentation (generated by Doxygen)	docs/fftcDocs.chm
2	Design Document	docs/FFTC_SDS.pdf