



## FFTC Driver

---

# Release Notes

Applies to Product Release: 2.02.00.07  
Publication Date: July 17, 2018

### 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) 2012-2018 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) .....   | 4 |
| Known Issues/Limitations.....          | 4 |
| Licensing .....                        | 4 |
| Delivery Package.....                  | 4 |
| Installation Instructions.....         | 4 |
| Test and Example .....                 | 5 |
| FFTC Test.....                         | 6 |
| FFTC Example.....                      | 6 |
| Customer Documentation List.....       | 7 |

# FFTC Driver version 2.02.00.07

## 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 2.02.00.07

- Updated buildlib.xs to add RULES\_MAKE macro to support build based on custom Rules.make

### Release 2.02.00.06

- Fix for IR SDOCM00120414 Error encountered during linking the fftcSimpleK2\*TestProject LE/BE for K2L,K2K,K2H(DSP) platform. Additional memory map changes were needed.

### Release 2.02.00.05

- Fix for IRs listed in Table 1 below.

**Table 1 Resolved IRs with Release 2.02.00.05**

| IR Parent/<br>Child Number | Severity<br>Level | IR Description  |
|----------------------------|-------------------|---|
| SDOCM00114155              | S2 -<br>Major     | fftcSimpleK2x test projects get linker error message: program will not fit into available memory: K2H,K2K,K2L. Linker cmd file updated. |

**Release 2.02.00.04**

- Fix for IRs listed in Table 2 below.

**Table 2 Resolved IRs with Release 2.02.00.04**

| IR Parent/<br>Child Number | Severity<br>Level | IR Description  |
|----------------------------|-------------------|---|
| SDOCM00112559              | Critical          | FFTC LLD cannot be used for more than 2 instances of FFTC and 3rd instance operation causes memory spill            |
| SDOCM00112754              | Minor             | FFTC LLD: K2H LLD should be limited to 4 instances  |
| SDOCM00112895              | Minor             | Incorrect Implementation of osalDeleteSem functions in FFTC and BCP PDK examples and tests causes small memory leak |
| SDOCM00113003              | Major             | Missing the extern "C" construct in fftc_osal.h   |

**Release 2.02.00.03**

- Fix for IRs listed in Table 3 below.

**Table 3 Resolved IRs with Release 2.02.00.03**

| IR Parent/<br>Child Number | Severity<br>Level | IR Description  |
|----------------------------|-------------------|---|
| SDOCM00108271              | Major             | All fftc unit tests do not finish when run  |
| SDOCM00108333              | Minor             | FFTC: Fftc_getDeviceAccumulatorConfig API parameter pAccRxQNum has wrong type         |
| SDOCM00107359              | Major             | [Wireless LLD] Need to build and test the wireless LLD to enable the call stack trace |

**Release 2.02.00.02**

- Incorporated new osal functions for QMSS accumulator queues for IR SDOCM00107127: Separate protection OSALs for QMSS LLD
- Fix for IRs listed in Table 4 below.

**Table 4 Resolved IRs with Release 2.02.00.02**

| IR Parent/<br>Child Number | Severity Level | IR Description  |
|----------------------------|----------------|---|
| SDOCM00108083              | Minor          | FFTC test project power domain enabling for K2L incorrect                           |
| SDOCM00108140              | Minor          | FFTC multiple instance test missing power enablement for FFTC 1 for K2H/K2K devices |
| SDOCM00108143              | Minor          | FFTC driver queue number correction for second FFTC instance use for K2L            |
| SDOCM00108144              | Minor          | FFTC driver queue number correction for second FFTC instance use for K2K and K2H    |

**Release 2.02.00.01**

- Merged the latest changes from Release 2.00.00.04 and 2.00.00.05
- Undo the updates per Release 2.02.00.00

**Release 2.02.00.00**

- Rename FFTC\_0 back to FFTC\_A for backwards compatibility

**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 5 provides information on IR resolutions incorporated into this release.

**Table 5 Resolved IRs for this Release**

| IR Parent/<br>Child Number | Severity<br>Level | IR Description  |
|----------------------------|-------------------|---|
| PRSDK-2194                 | NA                | Add RTOS Installer script to autoset SDK_INSTALL_PATH |

## **Known Issues/Limitations**

**Table 6 Known Issue IRs for this Release**

| IR Parent/<br>Child Number | Severity<br>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         | The top level directory contains the following:- <ol style="list-style-type: none"> <li><u>XDC Build and Package files</u><br/>These files (<code>config.bld</code>, <code>package.xdc</code> etc) are the XDC build files which are used to create the FFTC package.</li> <li><u>Exported Driver header file</u><br/>Header files which are provided by the FFTC driver and should be used by the application developers for driver customization and usage.</li> </ol> |
| 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>code/data size information</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

To see the test applications in action, please build and run the projects on all 4 cores in "synchronous" mode.

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

**Table 7      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 |