

# CC3100 SPI Debug Tool

---

## Overview

[Return to CC31xx & CC32xx Home Page](#)



This is a sample test application for verifying/validating the porting of CC3100 host-driver to a new MCU platform. This applications checks the SPI configuration with CC3100 and confirms the mapping of the SPI interface pins.

**Note:** This wiki page is only applicable for **CC3100-SDK v1.0.0** and upward releases. For documentation on older SDKs' examples, refer corresponding file in `<cc3100-sdk-installation-location>\cc3100-sdk\docs\examples\`

## Assumption and Knowledge base

- User will have to build his own project for the platform and need to add the provided files to use the tool.
- Sample project is provided with CCS for MSP430F5529 LaunchPad.

## Environment setup

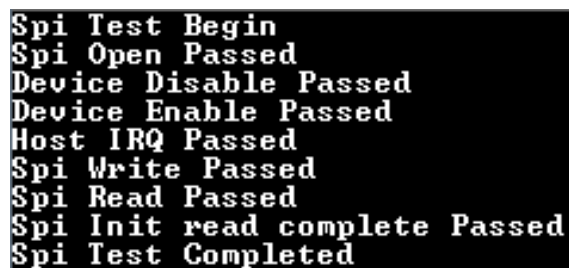
The user need to build their own project to use the tool to validate the SPI porting. Using the tool will require creating a new project and compiling it.

## Using tool with CCS or IAR

- Open the compiler and create a new project.
- Add Debugging tool files to the project.
  - Add "main.c" from "spi\_debug\_tool" folder.
- Write and add interface communication driver functions to "user.h".
  - `sl_DeviceEnable` : Enables the device by setting the appropriate GPIO pin high.
  - `sl_DeviceDisable` : Disables the device by setting the appropriate GPIO pin low.
  - `_SIFd_t` : Descriptor for SPI interface.
  - `sl_IfOpen` : Open a SPI interface to communicate with a simplelink device.
  - `sl_IfClose` : Close the opened SPI interface.
  - `sl_IfRead` : Read data from the opened SPI communication interface.
  - `sl_IfWrite` : Write data to opened SPI communication interface.
  - `sl_IfRegIntHdlr` : Register an interrupt handler routine for host IRQ.
- Write and add Board configuration function along with UART interface function to "daignostic.h"
  - `UartConfig` : Open the application UART channel.
  - `UartWrite` : Write data to opened UART channel.
  - `Init_Clk` : Initialize the system clock.
  - `StopWDT` : Stops the Watch Dog Timer.
- Add SPI, UART and board configuraton files to the project.
- Include header file path to the project.
  - Include SPI, UART and Board header file path to project.
  - Include "SimpleLink->Include" and "SimpleLink->Source" path to the project.

## Validating the SPI Configuration

- Connect the board to PC and configure the terminal program for seeing the logs - Detailed instructions are available at [http://processors.wiki.ti.com/index.php/CC31xx\\_&\\_CC32xx\\_Terminal\\_Setting](http://processors.wiki.ti.com/index.php/CC31xx_&_CC32xx_Terminal_Setting)
- Compile the run the project. On successful testing you will see the below output on the terminal.

A terminal window with a black background and white text. The text displays the results of an SPI test, with each step on a new line. The output is as follows:

```
Spi Test Begin
Spi Open Passed
Device Disable Passed
Device Enable Passed
Host IRQ Passed
Spi Write Passed
Spi Read Passed
Spi Init read complete Passed
Spi Test Completed
```

## Limitations/Known Issues

None

# Article Sources and Contributors

**CC3100 SPI Debug Tool** *Source:* <http://processors.wiki.ti.com/index.php?oldid=227214> *Contributors:* A0131814, A0132173, A0221015, Codycooke, Malokyle, SarahP

# Image Sources, Licenses and Contributors

**File:Cc31xx\_cc32xx\_return\_home.png** *Source:* [http://processors.wiki.ti.com/index.php?title=File:Cc31xx\\_cc32xx\\_return\\_home.png](http://processors.wiki.ti.com/index.php?title=File:Cc31xx_cc32xx_return_home.png) *License:* unknown *Contributors:* A0221015

**Image:SPI\_DiagnosticTool\_1.png** *Source:* [http://processors.wiki.ti.com/index.php?title=File:SPI\\_DiagnosticTool\\_1.png](http://processors.wiki.ti.com/index.php?title=File:SPI_DiagnosticTool_1.png) *License:* unknown *Contributors:* A0132173