User's Guide

Interfacing the TI AFE7769DEVM With the Altera Arria 10 FPGA

ABSTRACT

This user's guide provides a walkthrough of the hardware and software setup of two evaluation modules (EVMs): Texas Instrument’s AFE7769DEVM transceiver and Altera's Arria™ 10 field-programmable gated array (FPGA). Additionally, the user guide outlines four test cases, all prefaced with initial bringup steps, with respect to signal generation and the interfacing of both software's from TI and Altera™.

Table of Contents

List of Figures

Trademarks

Arria™ and Altera™ are trademarks of Intel Corporation in the U.S. and other countries.
All trademarks are the property of their respective owners.

1 Introduction

This user's guide introduces a wireless development platform using Texas Instruments' AFE7769DEVM in collaboration with Altera, a hardware company developed from Intel that provides FPGA-based evaluation modules to companies within the communications industry. This reference solution serves to help customers ramp up system integration of the analog front-end (AFE), and provides a quick evaluation and prototyping platform for 5G ORAN and wireless solutions, including small cell and macro cell solutions.

The AFE7769D is a 4T4R2F RF-sampling transceiver with integrated digital pre-distortion (DPD) that serves to linearize power amplifiers (PA's) for improved wireless coverage to the end customer. The wireless development platform utilizes the EVM version of the device, which interfaces with Altera's component of the solution via an FPGA mezzanine card (FMC+) connector. The Arria 10 is an FPGA module that interfaces with the AFE7769DEVM through the FMC+ connector. This hardware is based on Intel Agilex® 5SoC FPGA.

2 Hardware and Software Setup

2.1 Hardware Setup

This section discusses the initial setup of the test environment, as well as the necessary power connections.

1. Mount the FMC+ connector of the AFE7769DEVM to the receiving part on the Arria 10 FPGA (labeled J19), as shown in Figure 2-1.

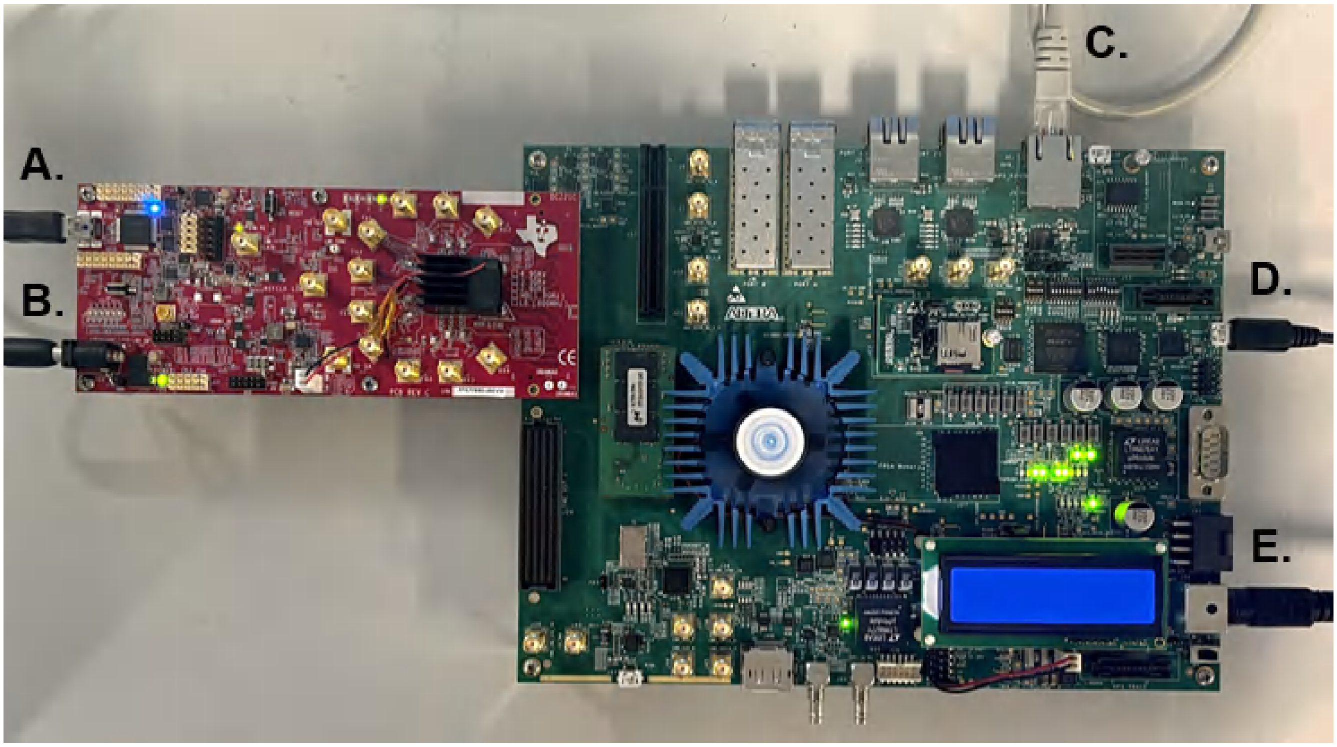


Figure 2-1. AFE7769DEVM and Arria 10 FPGA

2. Ensure the following connections are made to the necessary power supplies/ports, and that the Arria 10 is switched on, as shown above. The switch can be found in the bottom-right corner of the above figure.
   a. USB-C cable: connected to PC used to launch Latte GUI.
   b. AFE7769DEVM Power cable: connected to a power supply @ 5.5V
   c. Ethernet cable: connected to Ethernet port
   d. USB Blaster cable: connected to PC used to launch Quartus Programmer and SignalTap
   e. Arria 10 Power cable: connected to safe power outlet

For all other hardware integrations for the Arria 10, see the Intel® Arria® 10 SX SoC Development Kit.

2.2 Software Setup

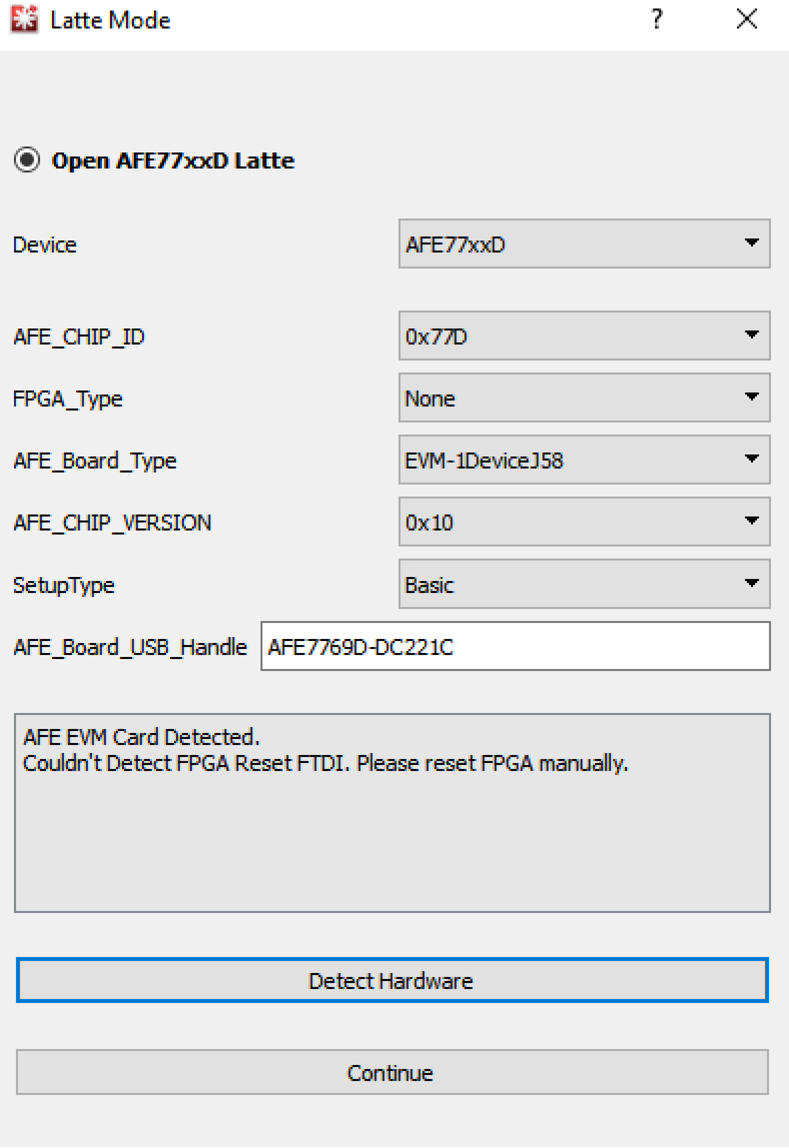
Ensure the following software are installed:

- AFE77xxD Latte GUI (version 1.1.1 or later)
- Intel Quartus Programmer and SignalTap GUI (version 22.2)

2.3 Test Cases

2.3.1 Initial Bringup for All Test Cases

1. Launch AFE77xxD Latte GUI.
2. An initial window pops up with the device setup information; verify the window has the exact same settings as shown in Figure 2-2.



Latte Mode

● Open AFE77xxD Latte

Device: AFE77xxD

AFE_CHIP_ID: 0x77D

FPGA_Type: None

AFE_Board_Type: EVM-1DeviceJ58

AFE_CHIP_VERSION: 0x10

SetupType: Basic

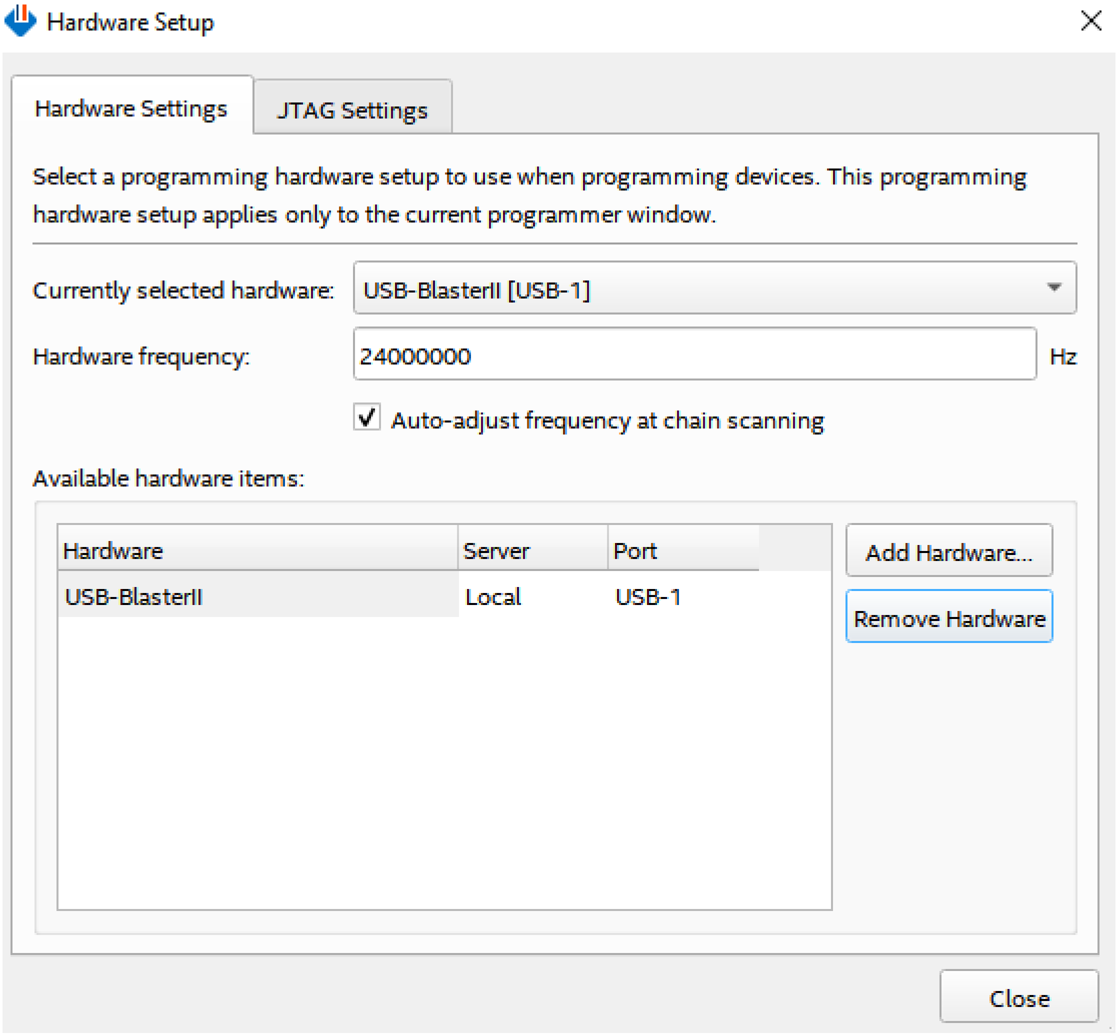
AFE_Board_USB_Handle: AFE7769D-DC221C

AFE EVM Card Detected.
Couldn't Detect FPGA Reset FTDI. Please reset FPGA manually.

Figure 2-2. Initial Setup, AFE77xxD

3. Navigate to the AFE-Configuration tab on the left-hand side of the application, and “Load” the “AFE77xxD_8.1_9.1.xlsx” file.
4. Ensure the following message appears in the Log window once the configuration file is loaded:
   a. Loaded Configuration: ‘AFE77xxD_8.1_9.1.xlsx’
   b. Refreshed GUI.
5. Do not click "Run Device Bringup" in Latte yet.

6. Keep Latte open; launch Quartus Programmer.
7. Click “Hardware Setup” at the top of the pane, and ensure “USB-BlasterII [USB-1]” is selected.



Hardware Setup

Select a programming hardware setup to use when programming devices. This programming hardware setup applies only to the current programmer window.

Currently selected hardware: USB-BlasterII [USB-1]

Hardware frequency: 24000000 Hz

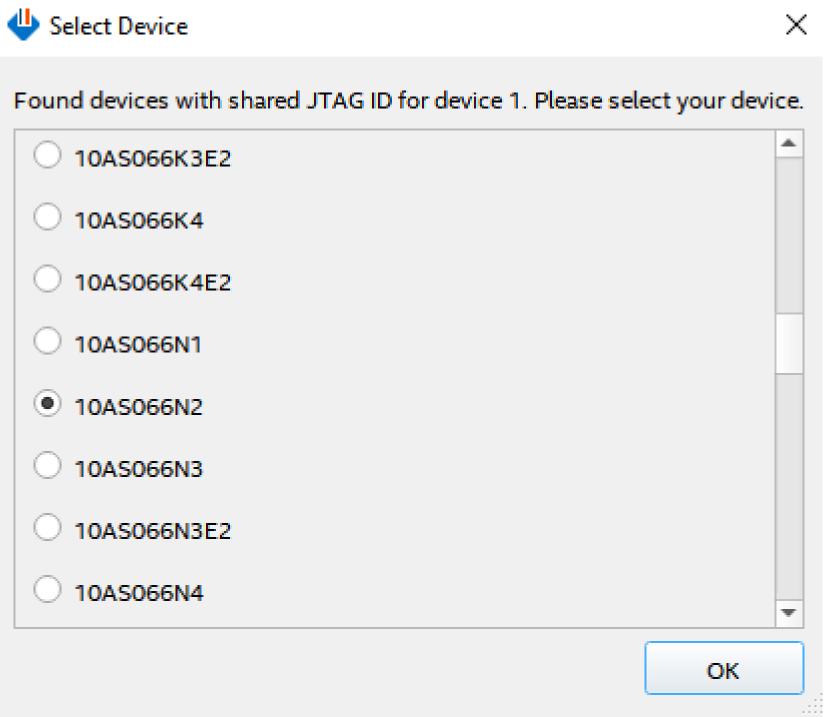
☑ Auto-adjust frequency at chain scanning

Available hardware items:

Hardware	Server	Port
USB-BlasterII	Local	USB-1

Figure 2-3. Hardware Setup Window

8. Click “Auto Detect” and a window of different FPGA names will pop up. Select 10AS066N2, and click “OK”.



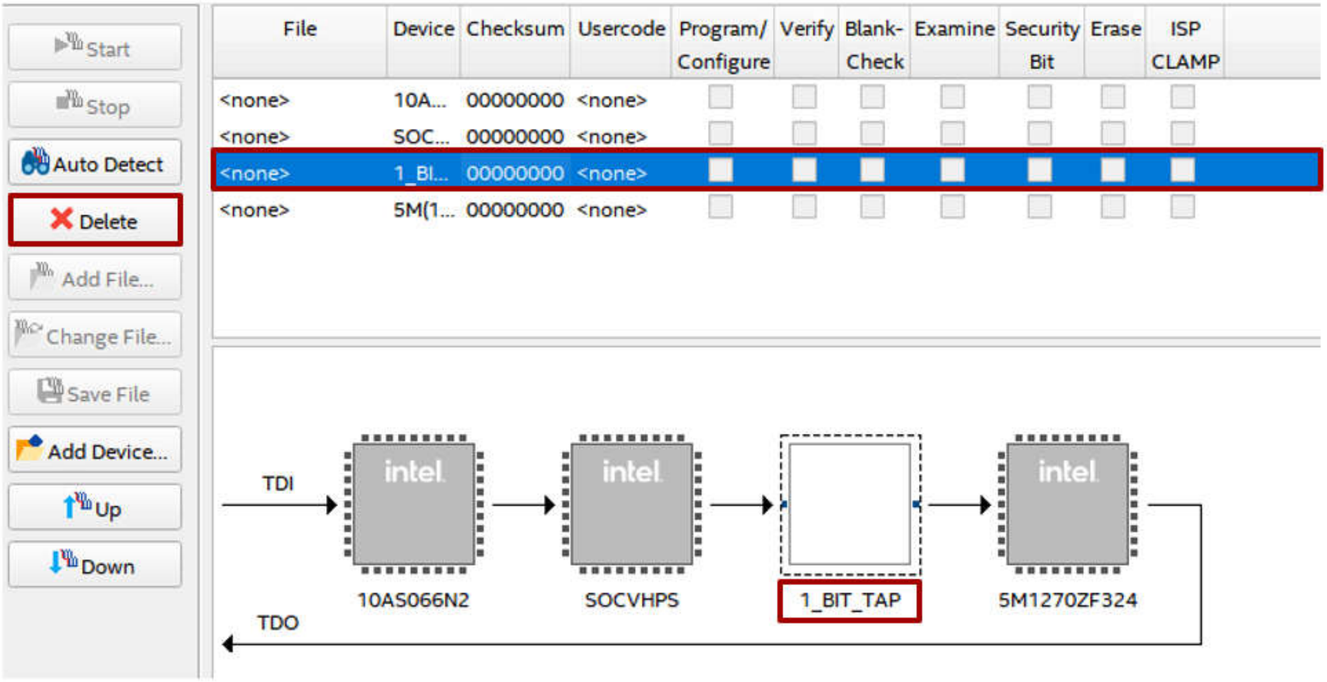
Select Device

Found devices with shared JTAG ID for device 1. Please select your device.

- ○ 10AS066K3E2
- ○ 10AS066K4
- ○ 10AS066K4E2
- ○ 10AS066N1
- ● 10AS066N2
- ○ 10AS066N3
- ○ 10AS066N3E2
- ○ 10AS066N4

Figure 2-4. Auto Detect Window

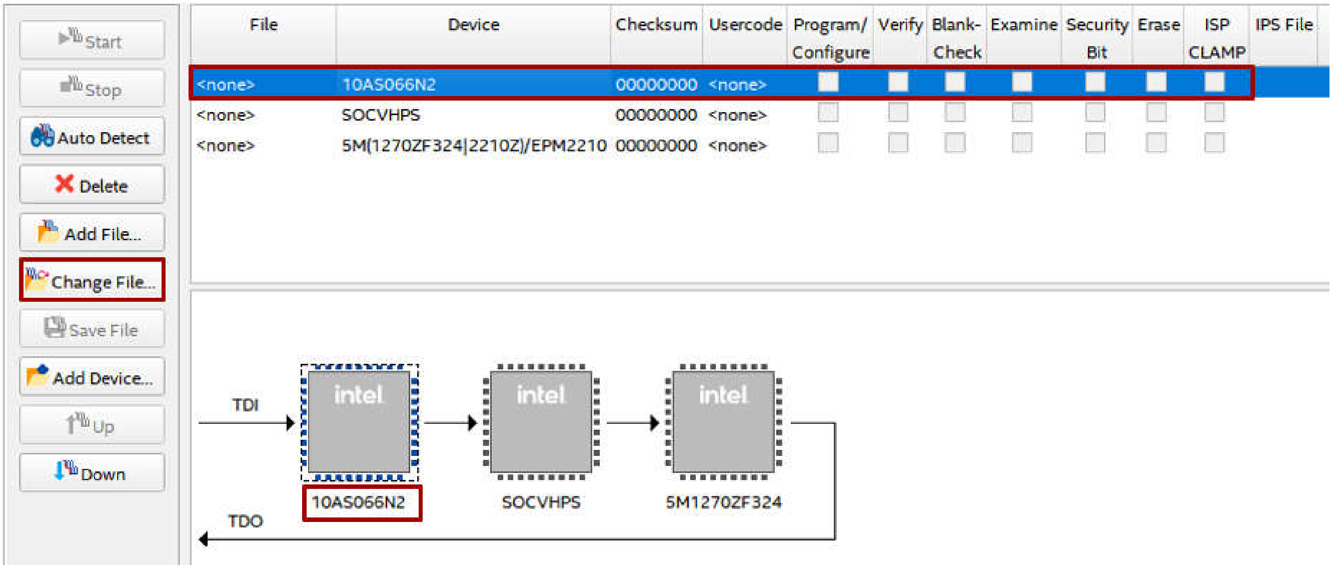
9. In the middle pane, a four-component block diagram will appear. Referencing the order of the blocks in the “File” column, select the third one (also known as the “1_BIT_TAP” block) and click “Delete”.



File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank-Check	Examine	Security Bit	Erase	ISP CLAMP
<none>	10A...	00000000	<none>	□	□	□	□	□	□	□
<none>	SOC...	00000000	<none>	□	□	□	□	□	□	□
<none>	1_BI...	00000000	<none>	□	□	□	□	□	□	□
<none>	5M(1...	00000000	<none>	□	□	□	□	□	□	□

Figure 2-5. Remove "1_BIT_TAP"

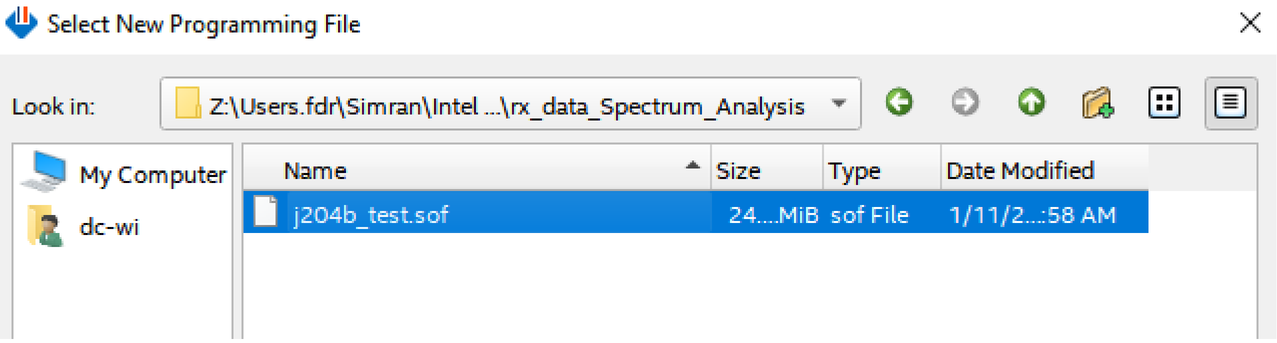
10. Next, select the first block in the same “File” column and click “Change File”.



File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine	Security Bit	Erase	ISP CLAMP	IPS File
<none>	10AS066N2	00000000	<none>	□	□	□	□	□	□	□	
<none>	SOCVHPS	00000000	<none>	□	□	□	□	□	□	□	
<none>	5M(1270ZF324|2210Z)/EPM2210	00000000	<none>	□	□	□	□	□	□	□	

Figure 2-6. Change "10AS066N2"

11. In File Explorer, select the “j204b_test.sof” file and click “Open”.



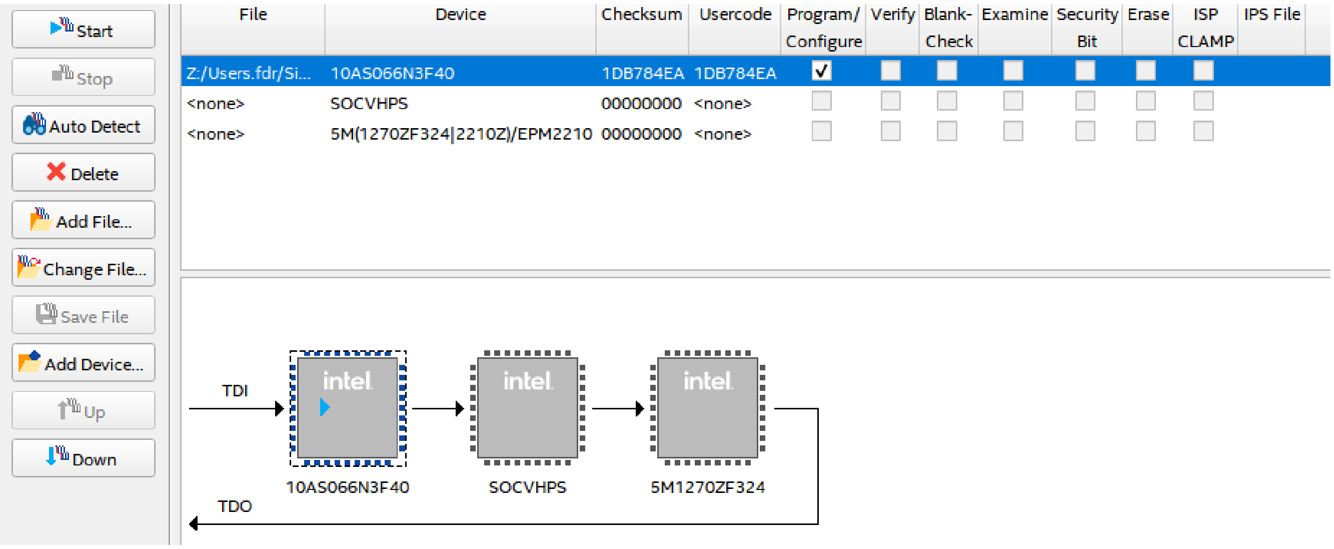
Select New Programming File

Look in: Z:\Users.fdr\Simran\Intel ...\rx_data_Spectrum_Analysis

Name	Size	Type	Date Modified
j204b_test.sof	24....MiB	sof File	1/11/2...:58 AM

Figure 2-7. Insert .sof File

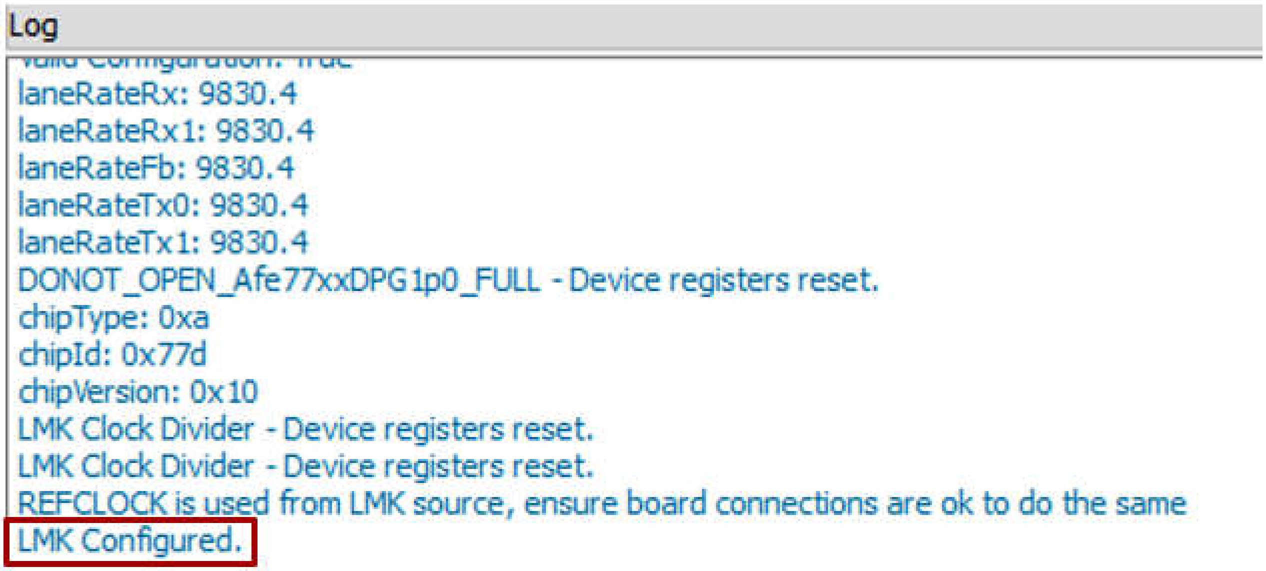
12. Verify the block diagram looks exactly like the following, especially from an order perspective.



File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank-Check	Examine	Security Bit	Erase	ISP CLAMP	IPS File
Z:/Users.fdr/Si...	10AS066N3F40	1DB784EA	1DB784EA	☑	□	□	□	□	□	□	
<none>	SOCVHPS	00000000	<none>	□	□	□	□	□	□	□	
<none>	5M(1270ZF324|2210Z)/EPM2210	00000000	<none>	□	□	□	□	□	□	□	

Figure 2-8. Updated Block Diagram

13. Navigate back to Latte and click "Run Device Bringup".
14. Monitor the Log window. Once you see the “LMK Configured” message, navigate back to Quartus Programmer and click “Start”.



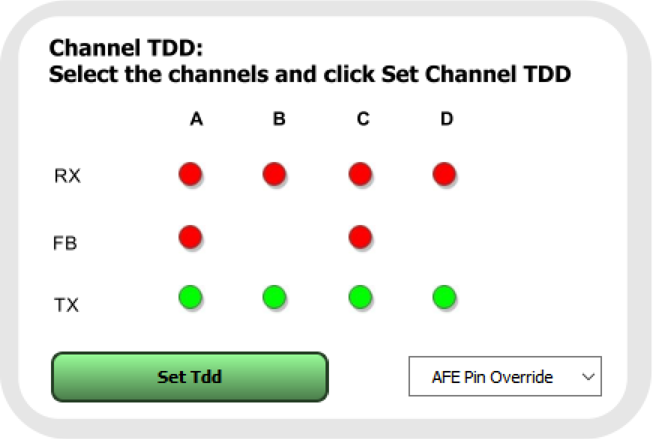
```
valid Configuration: True
laneRateRx: 9830.4
laneRateRx1: 9830.4
laneRateFb: 9830.4
laneRateTx0: 9830.4
laneRateTx1: 9830.4
DONOT_OPEN_Afe77xxDPG1p0_FULL - Device registers reset.
chipType: 0xa
chipId: 0x77d
chipVersion: 0x10
LMK Clock Divider - Device registers reset.
LMK Clock Divider - Device registers reset.
REFCLOCK is used from LMK source, ensure board connections are ok to do the same
LMK Configured.
```

Figure 2-9. "LMK Configured" Indicator

15. In Quartus Programmer, monitor the Progress bar on the top right corner of the window. It will progress to “100% (Successful)” within a minute.
16. Return to Latte GUI and monitor Device Bringup until complete. This is indicated by the final message in the Log window displaying “AFE configuration Complete”, followed by a record of the number of errors and warnings.
17. At the end of bringup, the Log window notifies the user of only two errors (no warnings), which includes “FPGA Reset device not found” twice. This is normal and nothing to worry about as Latte currently only recognizes the TSW14J58EVM, a TI FPGA, by name.

2.3.1.1 Test Case 1: Generate Sinusoidal tx_data From NCO @ 5MHz

1. Once the steps in "Initial bringup for all test cases" section are complete, navigate to the Channel-Controls tab in Latte and in the first section of this window, click the buttons under TX TDD to turn them green and click “Set Tdd”.

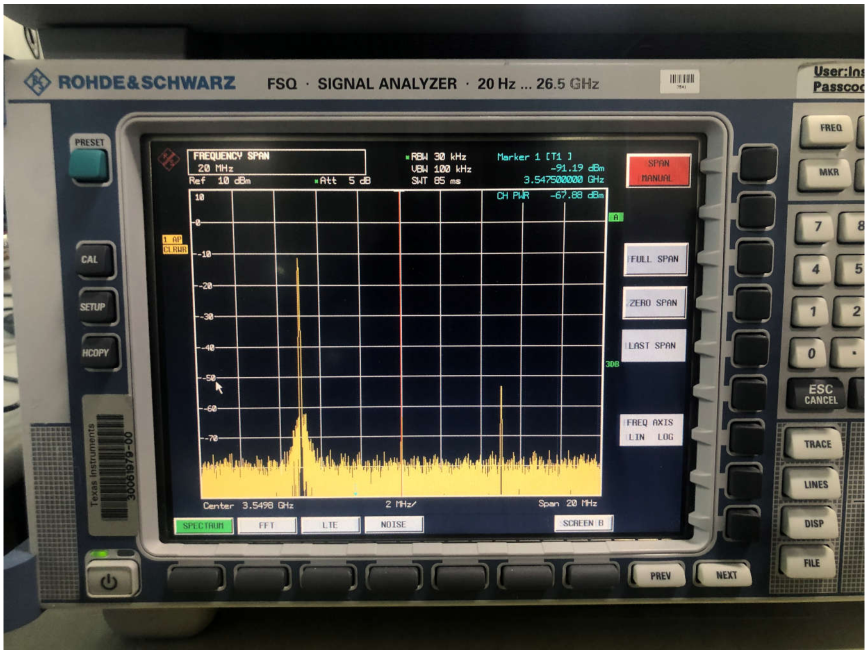


Channel TDD:
Select the channels and click Set Channel TDD

	A	B	C	D
RX	🔴	🔴	🔴	🔴
FB	🔴		🔴	
TX	🟢	🟢	🟢	🟢

Figure 2-10. Enable TX TDD

2. Monitor the power supply connected to the AFE. The current should increase from approximately 1.2A post-Device Bring up to 2A when TDD is enabled.
3. Connect any of the TX ports (TX1 = J7, TX2 = J8, TX3 = J9, TX4 = J10) from the AFE7769DEVM to the spectrum analyzer.
4. Figure 2-11 represents the expected result for this test case.

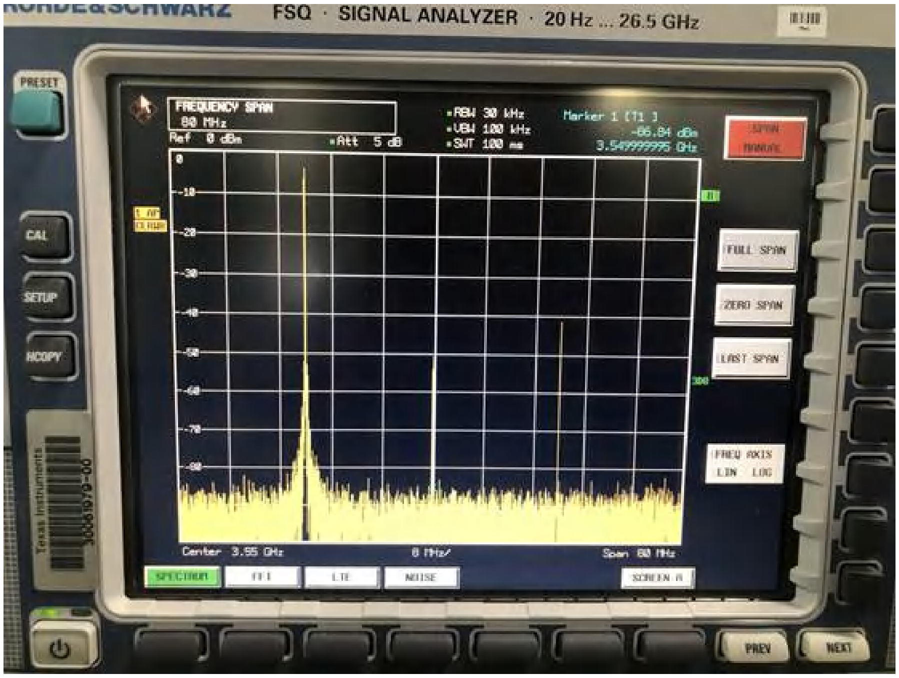


FREQUENCY SPAN 20 MHz
RBW 30 kHz
VBW 100 kHz
SWT 85 ms
Ref 10 dBm
Att 5 dB
Marker 1 [T1] -91.19 dBm 3.547500000 GHz
CH PWR -67.88 dBm
Center 3.5498 GHz
2 MHz/
Span 20 MHz

Figure 2-11. Spectrum Analyzer Results - 5MHz NCO

2.3.1.2 Test case 2: Generate Sinusoidal tx_data From NCO @ 20MHz

1. Once the steps in "Initial bringup for all test cases" section are complete, navigate to the Channel-Controls tab in Latte and under the “Channel TDD” box, click the buttons for TX to turn them green and “Set Tdd”, as shown in Figure 2-10.
2. Monitor the power supply connected to the AFE. The current should increase from approximately 1.2A post-Device Bring up to 2A when TDD is enabled.
3. Connect any of the TX ports (TX1 = J7, TX2 = J8, TX3 = J9, TX4 = J10) from the AFE7769DEVM to the spectrum analyzer.
4. Figure 2-12 represents the expected result for this test case.

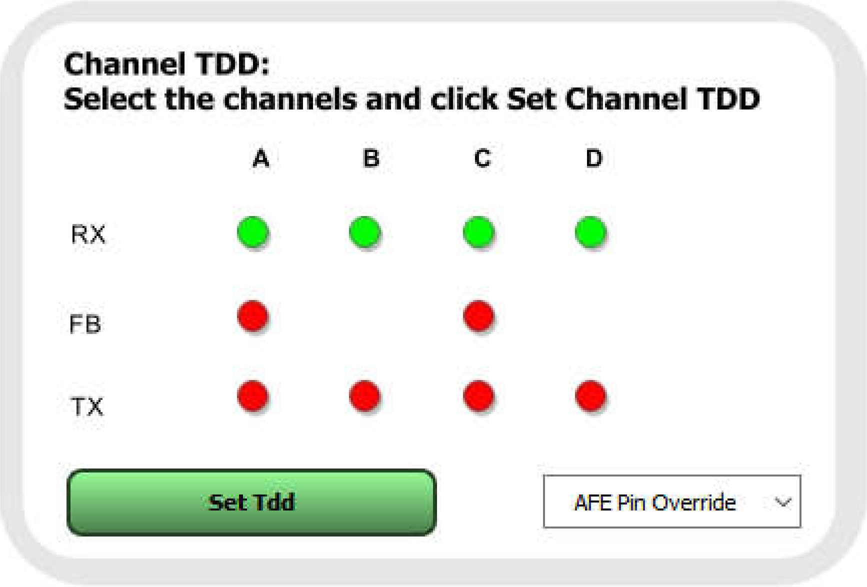


FREQUENCY SPAN
80 MHz
Ref 0 dBm
Att 5 dB
RBW 30 kHz
VBW 100 kHz
SWT 100 ms
Marker 1 [T1]
-66.84 dBm
3.549999995 GHz
Center 3.55 GHz
8 MHz/
Span 80 MHz

Figure 2-12. Spectrum analyzer results, 20MHz NCO

2.3.1.3 Test Case 3: Connect Signal Generator Output Port to RX of the AFE7769DEVM (at output power, -13dBm)

1. Once the steps in "Initial bringup for all test cases" section are complete, navigate to the Channel-Controls tab in Latte and under the “Channel TDD” box, click the buttons for RX to turn them green and “Set Tdd”.

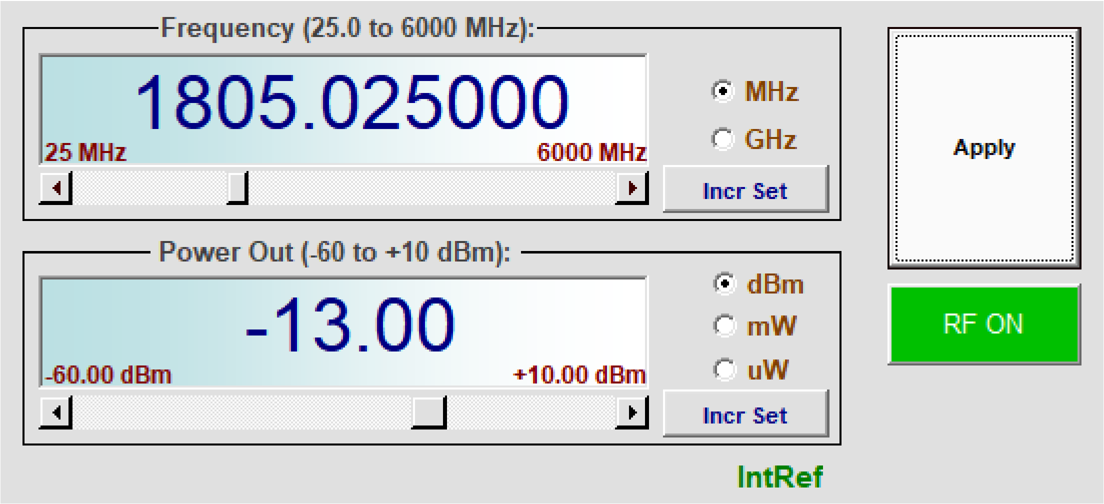


Channel TDD:
Select the channels and click Set Channel TDD

	A	B	C	D
RX	🟢	🟢	🟢	🟢
FB	🔴		🔴	
TX	🔴	🔴	🔴	🔴

Figure 2-13. Enable RX TDD

2.
3. Connect any of the RX ports (RX1 = J1, RX2 = J2, RX3 = J3, TX4 = J4) from the AFE7769DEVM to the RF output of the signal generator.
4. Ensure the signal generator is set to a frequency and output power of 1805.025 MHz and -13dBm, respectively.



Frequency (25.0 to 6000 MHz): 1805.025000

25 MHz 6000 MHz

◉ MHz ○ GHz

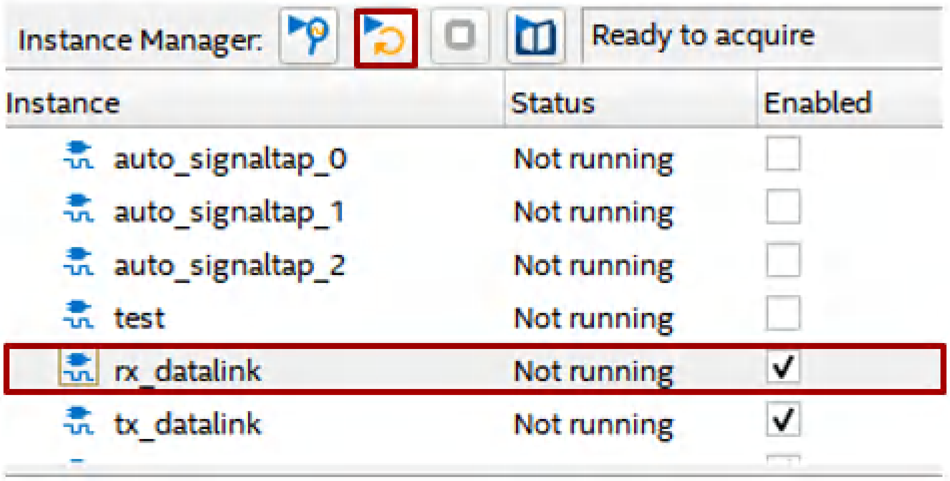
Power Out (-60 to +10 dBm): -13.00

-60.00 dBm +10.00 dBm

◉ dBm ○ mW ○ uW

Figure 2-14. Signal Generator Settings, Test Case 3

5. In File Explorer, navigate to the “test1122_txz_rxz.stp” file, and double-click to open it in Signal Tap.
6. Ensure “rx_datalink” is the selected tab and row; click the boxed icon to begin running data acquisition.



Instance Manager: Ready to acquire

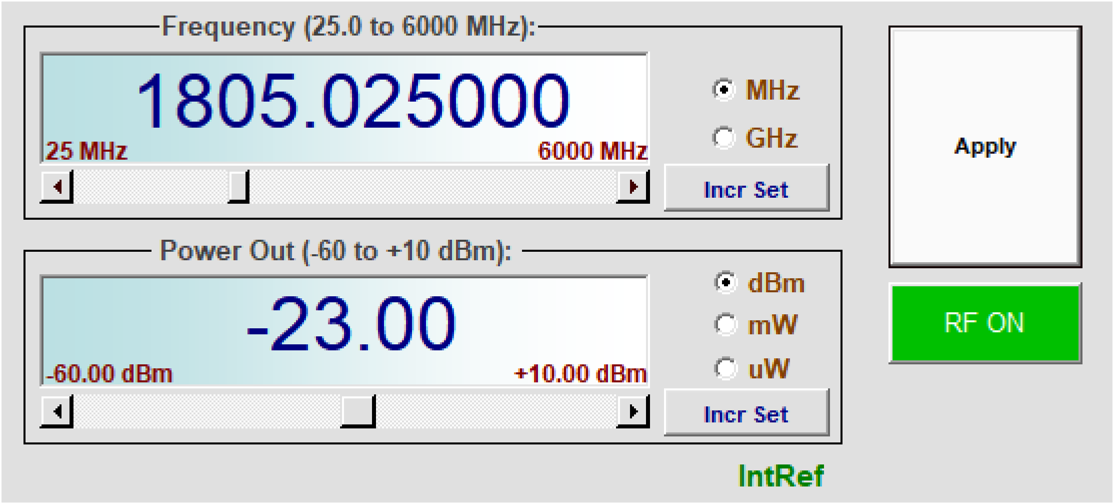
Instance	Status	Enabled
auto_signaltap_0	Not running	□
auto_signaltap_1	Not running	□
auto_signaltap_2	Not running	□
test	Not running	□
rx_datalink	Not running	☑
tx_datalink	Not running	☑

Figure 2-15. rx_datalink in SignalTap

7. Once acquisition is clear, pause the simulation with the same icon, this time a red square.
8. Right-click on the “uu00|jesd204_rx_link_data” section of the middle pane, choose “Create Signal Tap List File”, and save as a .txt file.
9. Decipher the data using the provided MATLAB script titled “rx_data_test.m”, and replace the sample .txt file name with the newly-created .txt file from the previous step.
10. After running the code, a figure with four graphs will appear.

2.3.1.4 Test Case 4: Connect Signal Generator Output Port to RX of the AFE7769DEVM (at output power, -23dBm)

1. Once the steps in "Initial bringup for all test cases" section are complete, navigate to the Channel-Controls tab in Latte and under the “Channel TDD” box, click the buttons for RX to turn them green and “Set Tdd”, as shown in Figure 2-13.
2. Connect any of the RX ports (RX1 = J1, RX2 = J2, RX3 = J3, TX4 = J4) from the AFE7769DEVM to the RF output of the signal generator.
3. Ensure the signal generator is set to a frequency and output power of 1805.025 MHz and -23dBm, respectively.



Frequency (25.0 to 6000 MHz): 1805.025000
25 MHz – 6000 MHz
● MHz ○ GHz

Power Out (-60 to +10 dBm): -23.00
-60.00 dBm – +10.00 dBm
● dBm ○ mW ○ uW

Figure 2-16. Signal Generator Settings, Test Case 4

4. In File Explorer, navigate to the “test1122_txz_rxz.stp” file, and double-click to open it in Signal Tap.

5. Ensure “rx_datalink” is the selected tab and row; click the boxed icon as shown in Figure 2-15 to begin running data acquisition.
6. Once acquisition is clear, pause the simulation with the same icon, this time a red square.
7. Right-click on the “uu00|jesd204_rx_link_data” section of the middle pane, choose “Create Signal Tap List File”, and save as a .txt file.
8. Decipher the data using the provided MATLAB script titled “rx_data_test.m”, and replace the sample .txt file name with the newly-created .txt file from the previous step.
9. After running the code, a figure with four graphs will appear.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265