Digital Front-End Radio Demonstration v9.1 User Guide

UG1163 (v9.1) October 18, 2023

AMD Adaptive Computing is creating an environment where employees, customers, and partners feel welcome and included. To that end, we're removing non-inclusive language from our products and related collateral. We've launched an internal initiative to remove language that could exclude people or reinforce historical biases, including terms embedded in our software and IPs. You may still find examples of non-inclusive language in our older products as we work to make these changes and align with evolving industry standards. Follow this link for more information.

Table of Contents

Chapter 1

Introduction

This guide describes how to use the AMD Digital Front-End (DFE) Crest Factor Reduction (CFR) + Digital Pre-Distortion (DPD) only radio demonstration platform and the Full DFE demonstration which is available on the AMD ZCU670 platform only. The full DFE Demo is used to demonstrate the full system including the Downlink chain from baseband signal to the radio front-end and the Uplink chain from ADC and RX equalizer to baseband and PRACH. The demonstration platform (demo) is built using the AMD DPD and AMD PC-CFR cores on an AMD Zynq™ UltraScale+™ MPSoC and Zynq UltraScale+ RFSoC devices. The demo uses standardized hardware platforms based on the ZCU102, ZCU111, ZCU208, and ZCU670 evaluation boards along with AMD RF Board v2.0, v3.0, v4.0, XM500, XM655, and XM755 (in short referred to as the XRF2, XRF3, XRF4, XM500, XM655, and XM755, respectively). The demo can be used in two modes:

- **Digital Loopback mode:** For basic demonstration and tutorial purposes, the demo can be run in a loopback mode using an internal digital Power Amplifier (PA) model. This mode is focused on CFR and DPD evaluation in real-time, exercising all features, without needing an actual PA and associated RF path circuitry.
- **RF mode:** For full demonstration purposes, the demo can be run with AMD RF Boards and full RF path circuitry including PA. In this mode, you have two types of designs to learn about different aspects of DFE functionality for Radio Unit applications. Downlink TX, Uplink RX, and Downlink Feedback processing is available in different combinations in various designs for evaluation.

The evaluation tool package includes:

- Various FPGA designs that focus on CFR + DPD along with other components for PA linearization correction. These designs are usable in both modes mentioned—Digital Loopback Mode (with digital model of a PA) and RF mode (where DAC and ADC are used to connect with RF lineup). These are available on all supported SoC boards.
- Various FPGA designs for RFSoC DFE board (ZCU670) that focus on full TX and RX DFE designs for evaluation in RF Mode only. Apart from CFR, DPD, DAC, and ADC, these designs include Digital Up/Down Conversion, TX/RX channel filters, TX/RX IFFT/FFT, and UL RX PRACH to support evaluating low-PHY and DFE in full system context.
- The MATLAB®-based evaluation tool (GUI and drivers) that allows interactions with the DFE designs to control the full operation of these designs. This includes DL TX and UL RX vector loading (for 4G, 5G signals), monitoring, and controlling for various blocks' functionality. The DFE Evaluation Tool Setup Wizard is now included in the package to support a step-by-step walk-through of PC software setup and ZCU* and RF board setup in the lab.

- The MATLAB Runtime based evaluation tools that allow you to run the demo GUI and DFE Evaluation Tool Setup Wizard without the MATLAB license.

Hardware Requirements

The DFE Radio evaluation tool works with following boards with varying capabilities:

- ZCU102, ZCU111, ZCU208, and ZCU670 standalone using digital loopback demo
- ZCU102 and XRF2 board for RF Demo mode
- ZCU111, ZCU208, or ZCU670 with XM500, XM655, XM755, XRF3, or XRF4 boards for RF Demo mode with full DFE design evaluation

As a summary, see the following table for the various configuration of the evaluation kit with the ZCU and RF board selection.

Table 1: **ZCU and RF Board Selection for DFE Demo**

	DLB	XRF2	XRF3	XRF4	XM500	XM655/XM755
ZCU102	x[1]	x	–	–	–	–
ZCU111	x[1]	–	x	–	x	–
ZCU208[3]	x[1]	–	x[2]	x	–	x
ZCU670[4]	x[1]	–	x[2]	x	–	x

Notes:

1. DLB support for all CFR + DPD only builds.
2. Adaptor card required.
3. CLK104 board is required for all ZCU208 combinations.
4. CLK104 is only needed for ZCU670 external clock option, otherwise, CLK104 is not needed for ZCU670 combinations.

The hardware required to support these combinations is as follows:

- Windows Personal Computer (Host PC) running Windows 10 64-bit Operating System (8 GB of RAM minimum recommended).
- AMD ZCU102, ZCU111, ZCU208, or ZCU670 evaluation board.
- XRF2, XRF3, XRF4, XM500, XM655, or XM755 board for use with a real Power Amplifier (PA). An adapter card is needed to use XRF3 with ZCU208 or ZCU670 board.
- CLK104 board is needed for ZCU208 board and optional for ZCU670 board if using on board reference clock. If external RF clock configuration is enabled on ZCU670 board, CLK104 board is also necessary.
- Power Supplies for each of the boards.
- Optional one A-to-micro-B USB cable used for JTAG (and for UART as well for ZCU208 and ZCU670 board).

- One microSD card with 4 GB or above is required for ZCU670 boot image and bit-file. If the XRF4 is being used for the same purpose, another microSD card with 4 GB or above is also required. Use the SD/microSD card included in the ZCU* board package/box, as that is the only one tested and supported.
- One CAT5 Ethernet patch cable for ZCU board and another CAT5 Ethernet patch cable for XRF4 (if used).
- Optional additional SMA cable for RF loopback testing.
- Optional RF attenuators for gain staging.
- Optional one A-to-mini-B USB cable for UART connection (ZCU102 only) and another optional one A-to-mini-B USB cable for XRF4 (if used) UART connection (if needed).
- Appropriate RF filters to suppress undesired DAC Nyquist images and the HD2 product generated from Power Amplifier.

Software Requirements

The DFE Radio Demo requires the following software for operation:

- Windows 10 64-bit Operating System with 8 GB RAM
- Optional AMD Vivado™ Lab Edition 2023.2 (Vivado Design Suite 2023.2 with AMD Vitis™ unified software platform can also be used, if you plan to perform FPGA/SoC design work on this same PC).
- The MathWorks, Inc MATLAB R2021b, R2022a/b, R2023a with Signal Processing Toolbox. If you do not have a license for the MATLAB versions, use the MATLAB Runtime 9.11.
- Python compatible with your MATLAB version; Pyro4 framework within Python. See this reference for MATLAB and Python compatibility https://www.mathworks.com/content/dam/mathworks/mathworks-dot-com/support/sysreq/files/python-compatibility.pdf.
- To support SD card boot mode, obtain the PC (does not have to be the same one running DFE package) privilege to format and copy boot contents into the SD card.

Related Documents

The DFE Radio Demo uses multiple components and configurations that require familiarity with related documents. Reading these related documents is considered essential for the correct use of this demo.

AMD DPD Product Documentation

The DPD product is fully described in the following documents:

1. *Digital Pre-Distortion LogiCORE IP Product Guide* (PG076) (registration required). This document describes the DPD product for Zynq devices.
2. *Digital Pre-Distortion Debug Interface User Guide* (UG989) (registration required). This document describes the operation of the DPD debug interface component of the AMD DPD product.

AMD DFE IP Product Documentation

Various DFE IP product are described in their corresponding product guide documents (available at RFSoC DFE Documents Lounge, registration required):

1. *RFSoC DFE Channel Filter LogiCORE IP Product Guide* (PG395) (registration required)
2. *RFSoC DFE DUC-DDC LogiCORE IP Product Guide* (PG393) (registration required)
3. *RFSoC DFE Resampler LogiCORE IP Product Guide* (PG392) (registration required)
4. *RFSoC DFE PRACH LogiCORE IP Product Guide* (PG391) (registration required)
5. *RFSoC DFE Fast Fourier Transform LogiCORE IP Product Guide* (PG390)

AMD PC-CFR Product Documentation

The PC-CFR product is fully described in the following documents:

1. *Peak Cancellation Crest Factor Reduction LogiCORE IP Product Guide* (PG097) (registration required). This document describes the PC-CFR product for 7 series, Zynq, and AMD UltraScale™ architecture devices.
2. *Peak Cancellation Crest Factor Reduction in a Multi-Standard Transmit System* (XAPP1174) (registration required).

AMD RF Board v2.0 Documentation

For a full understanding of the AMD RF board v2.0, see the *Xilinx RF Board v2.0 (XRF2) User Guide* (https://www.xilinx.com/member/dfe_demo.html#rfboards). This document describes the features, control, and configuration of the AMD RF Board.

AMD RF Board v3.0 Documentation

For a full understanding of the AMD RF board v3.0, see the *DFE Demo on ZCU111 and ZCU208 with Xilinx RF Board v3.0 (XRF3) User Guide* (`DFE_Demo_ZCU_and_RF_Boards_user_guide_vx_y.pdf`). This document describes the features, control, and configuration of the AMD RF Board.

AMD RF Board v4.0 Documentation

For a full understanding of the AMD RF board v4.0, see the *Xilinx RF Board v4.0 (XRF4) User Guide* (https://www.xilinx.com/member/dfe_demo.html#rfboards). This document describes the features, control, and configuration of the AMD RF Board.

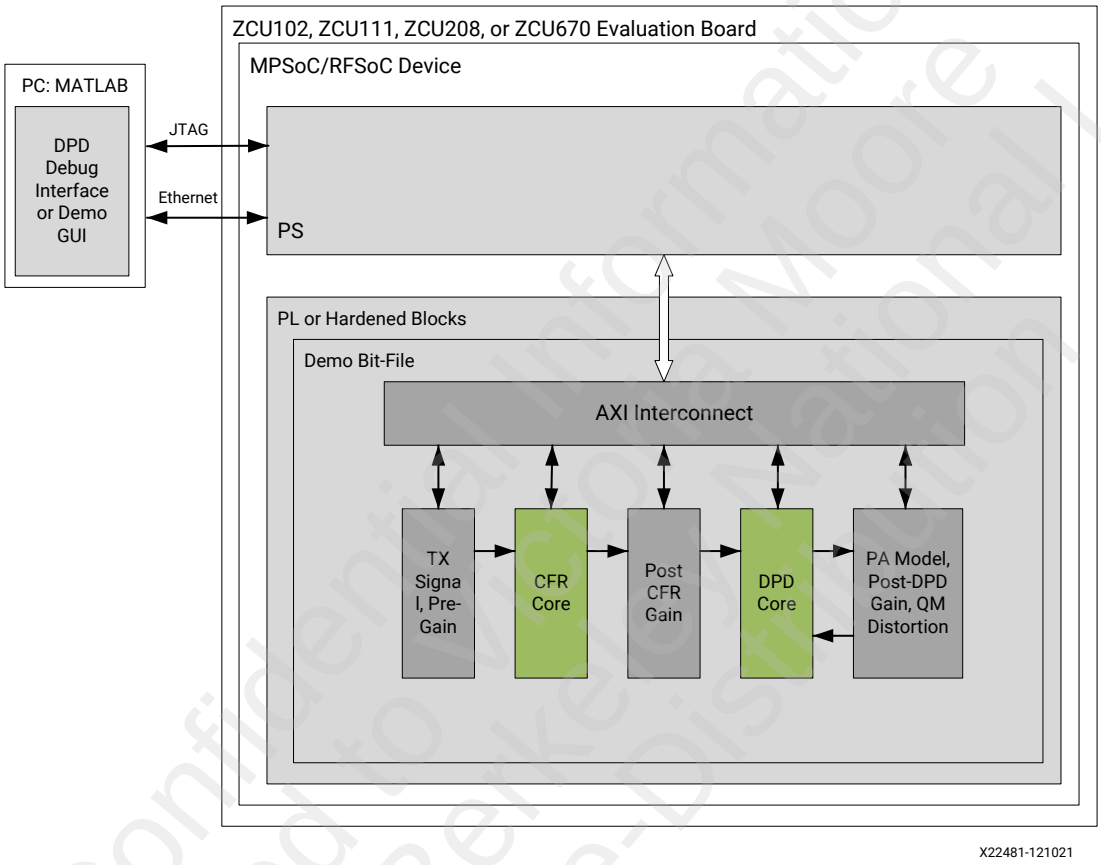
Overview

The DFE Radio Evaluation tool shows the capabilities of various DFE IP, CFR, and DPD IP along with various AMD Zynq™ UltraScale+™ MPSoC/RFSoC/RFSoC DFE devices.

Demo Modes of Operation

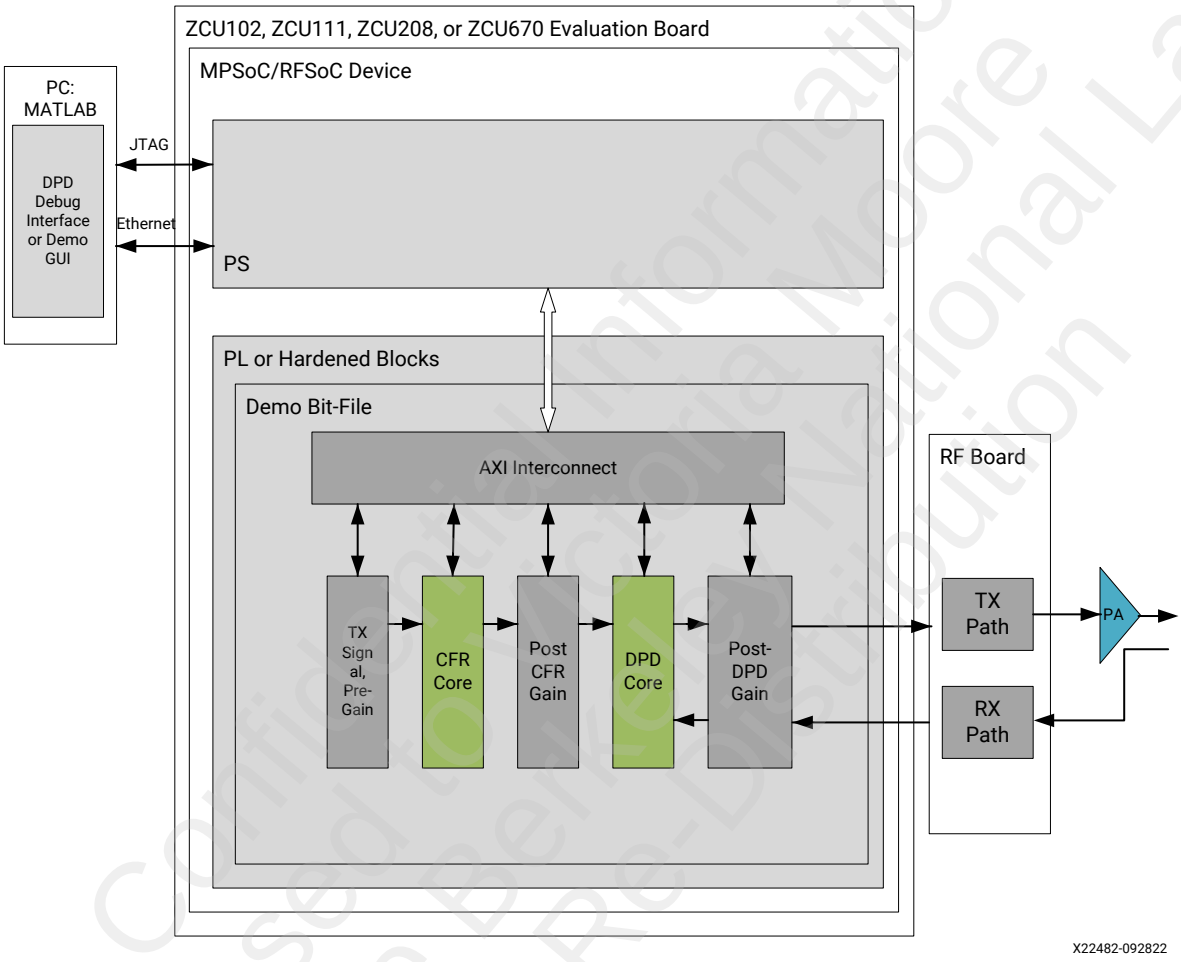
The DFE Radio evaluation tool can operate in two modes. In digital loopback mode, the design operates in an all-digital mode without the use of any RF analog circuitry. This mode uses a non-linear Power Amplifier (PA) distortion model and Quadrature Modulation (QM) imbalance model, both implemented in the FPGA logic to emulate the effects that transmitted signals suffer when passing through a PA and other RF analog circuitry. The figure shows the block diagram illustrating the different components used in this mode. This mode is only usable with designs that have CFR and DPD only functionality.

Figure 1: **Digital Loopback Mode**



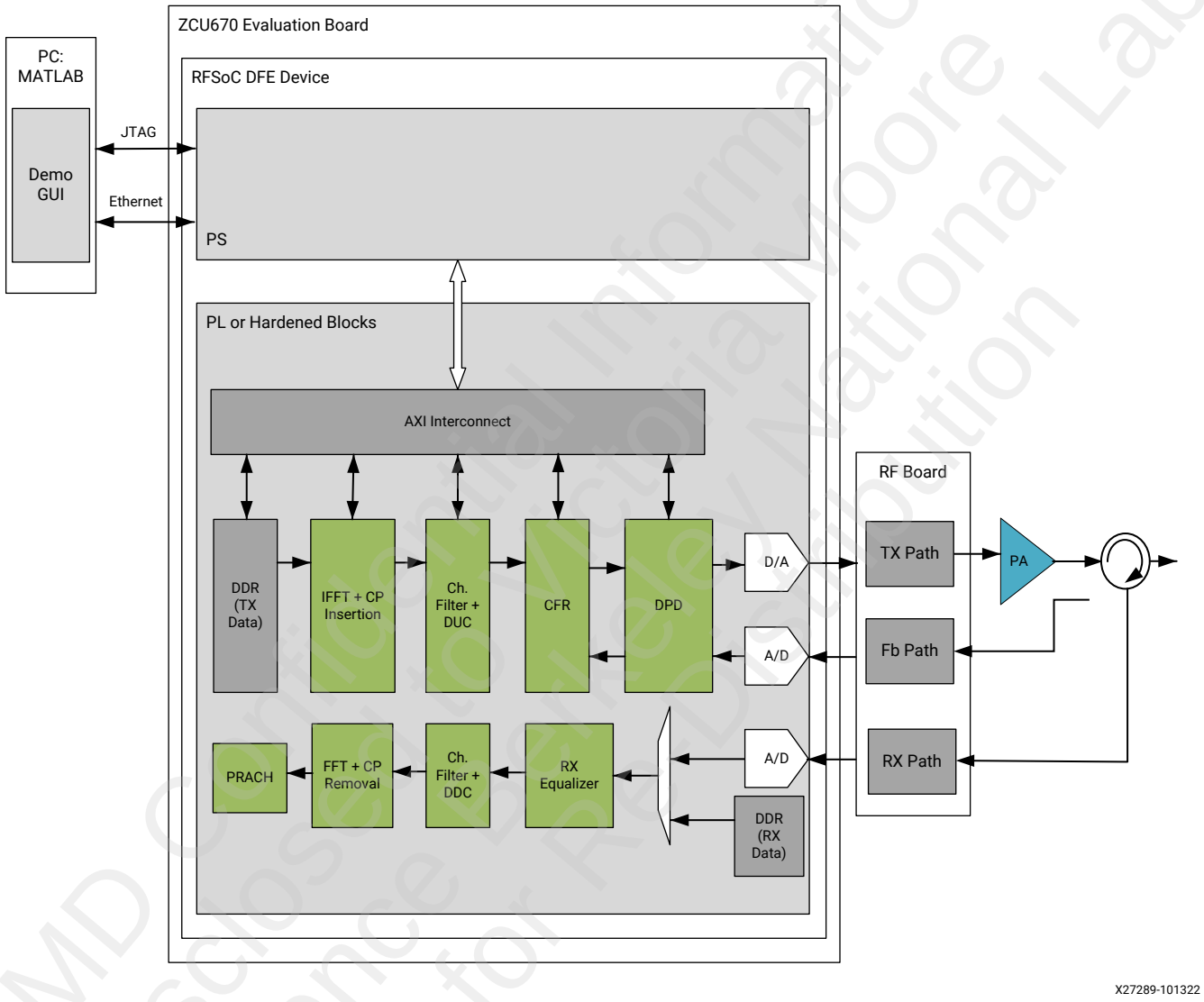
The CFR + DPD only designs can also be operated in RF mode with a supported AMD RF Board connected to a real RF lineup including PA. The following figure shows the block diagram for this mode of operation using the RF board.

Figure 2: **RF Mode (CFR + DPD Only Designs)**



In the RF Demo mode on ZCU670 board (RFSoC DFE device), you can load designs that allow evaluating full DFE chain as shown in the following figure.

Figure 3: **RF Mode (Full DFE Evaluation on ZCU670)**



Bit-File Configurations

The supplied CFR+DPD only bit-files uses a special super-set hardware configuration of DPD IP, to allow users to explore all available filter structures along with Long Term Memory (LTM) for that evaluation board. The supplied RFSoC DFE full-DFE functionality bit-files include a specific set of hardware configuration for each of the DFE functionality to enable evaluation of end-to-end DFE TX and RX chain. Specific details about various bit-files and their detailed configuration is available separately on DFE Demo Lounge (registration required) in:

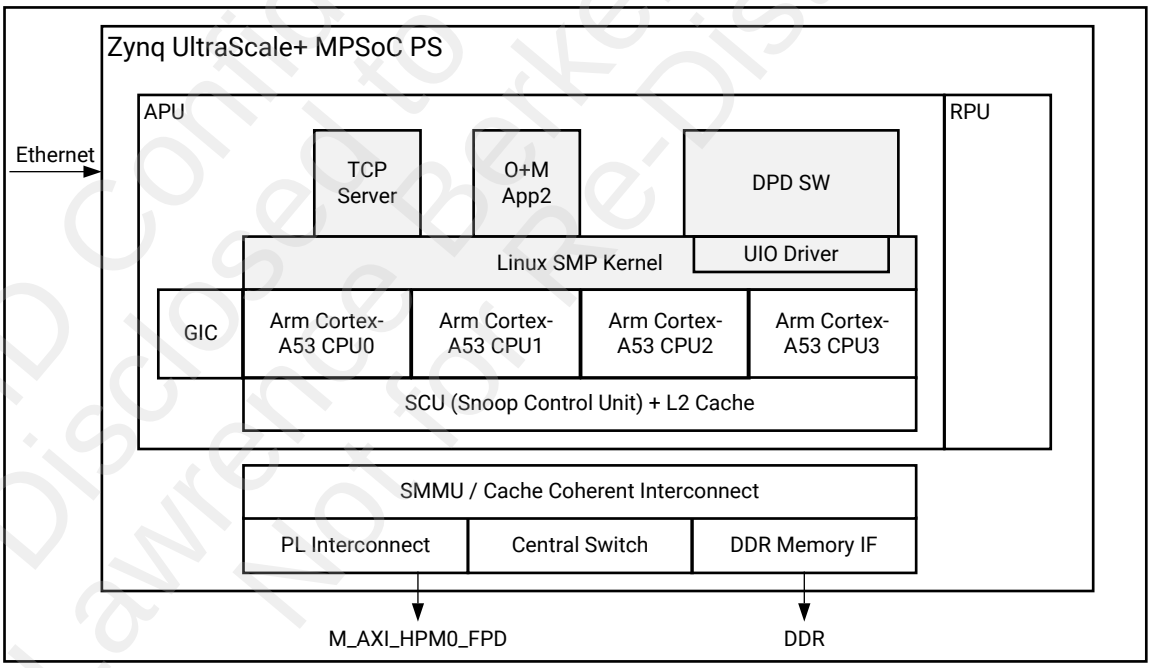
- `DFE_Demo_ZCU_and_RF_Boards_user_guide_vx_y.pdf` for RFSoC and RFSoC DFE bit-files
- `xrf2_user_guide_vx_y.pdf` for MPSoC bit-files

Supported Zynq UltraScale+ Processing System Configurations

The DFE Radio Demo uses DPD running as an application on Linux, operating in SMP configuration on all the available A53 (AMD Zynq™ UltraScale+™) Arm® processors. Linux also has TCP server software application running to support high speed packet transfers between MATLAB® and the DFE demo components.

The figure shows the block diagram of the PS portion of Zynq UltraScale+ (SMP) in DFE Demo with this mode of operation.

Figure 4: **Zynq UltraScale+ PS Configuration (SMP) in DFE Demo**

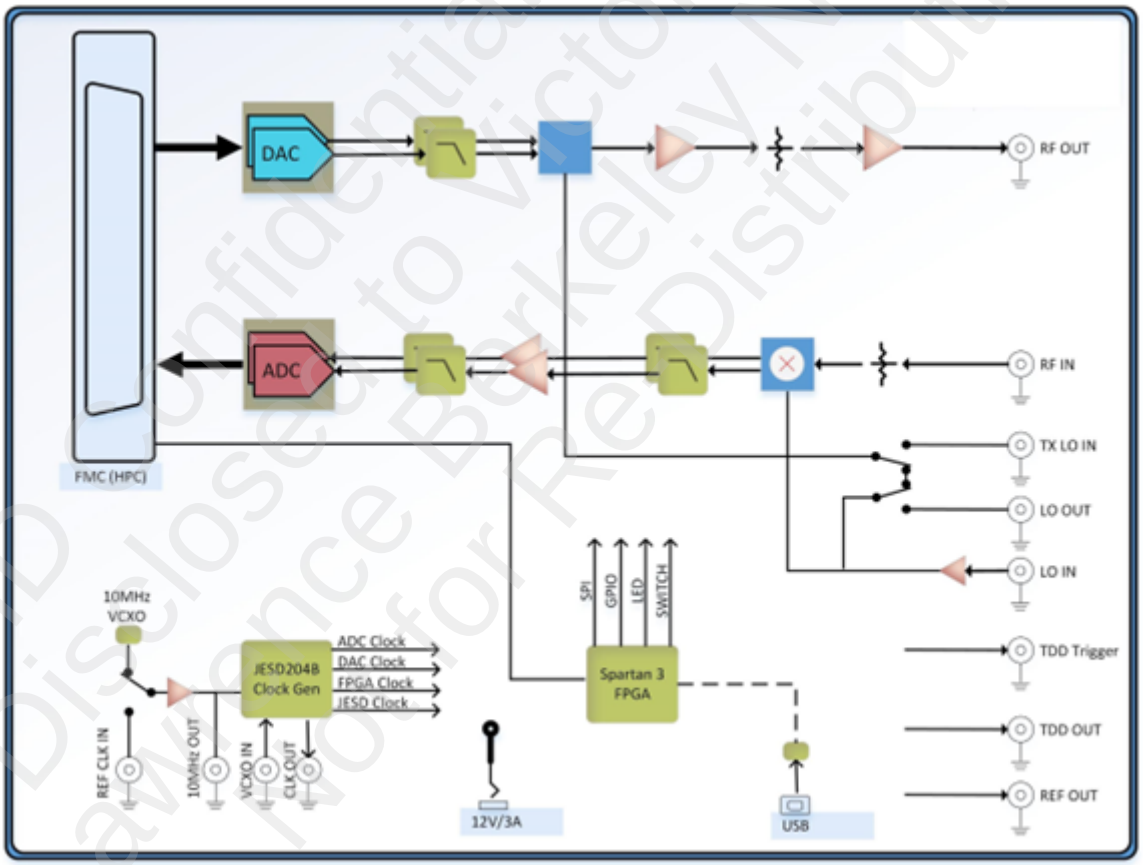


AMD RF Board Version 2.0

The AMD RF Board Version 2.0 has high-performance, wideband mixed-signal transmit and receive (observation) paths for DPD. The transceiver supports multiple cellular air interface standards including LTE, HSPA, CDMA, and Multi-Carrier GSM. This demo does not support GSM. For GSM demo, contact AMD Support.

The figure shows a high level block diagram of the AMD RF Board v2.0 with the FPGA Mezzanine Card (FMC) connector to allow connection to any AMD development boards with those FMC connectors.

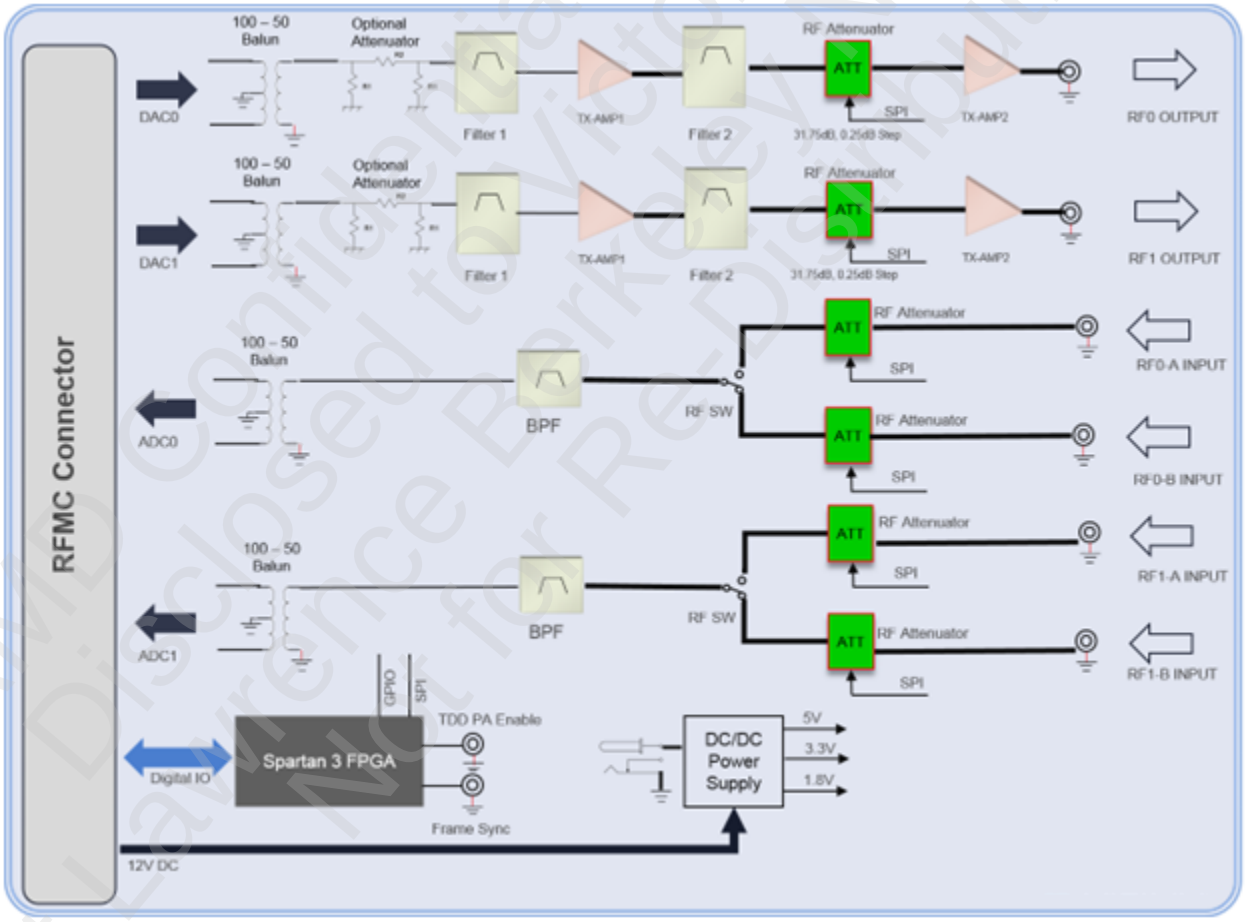
Figure 5: **AMD RF Board Version 2.0 Block Diagram**



AMD RF Board Version 3.0

The AMD RF Board Version 3.0 (XRF3) has high-performance, wideband mixed-signal transmit and receive (observation) paths for DPD on the ZCU111 directly, or ZCU208 board (with an XRF3-ZCU208 adapter card), and ZCU670 board (with an XRF3-ZCU670 adapter card). The figure shows a high level block diagram of the AMD RF Board v3.0 with the RF Mezzanine Card (RFMC) connector to allow connection to the AMD RFSoC development board. The XRF3 board has two transmit and two receive paths connected to two DAC and two ADCs respectively on the RFSoC device. The remaining six DACs and six ADCs are available on XRF3 as extended connections.

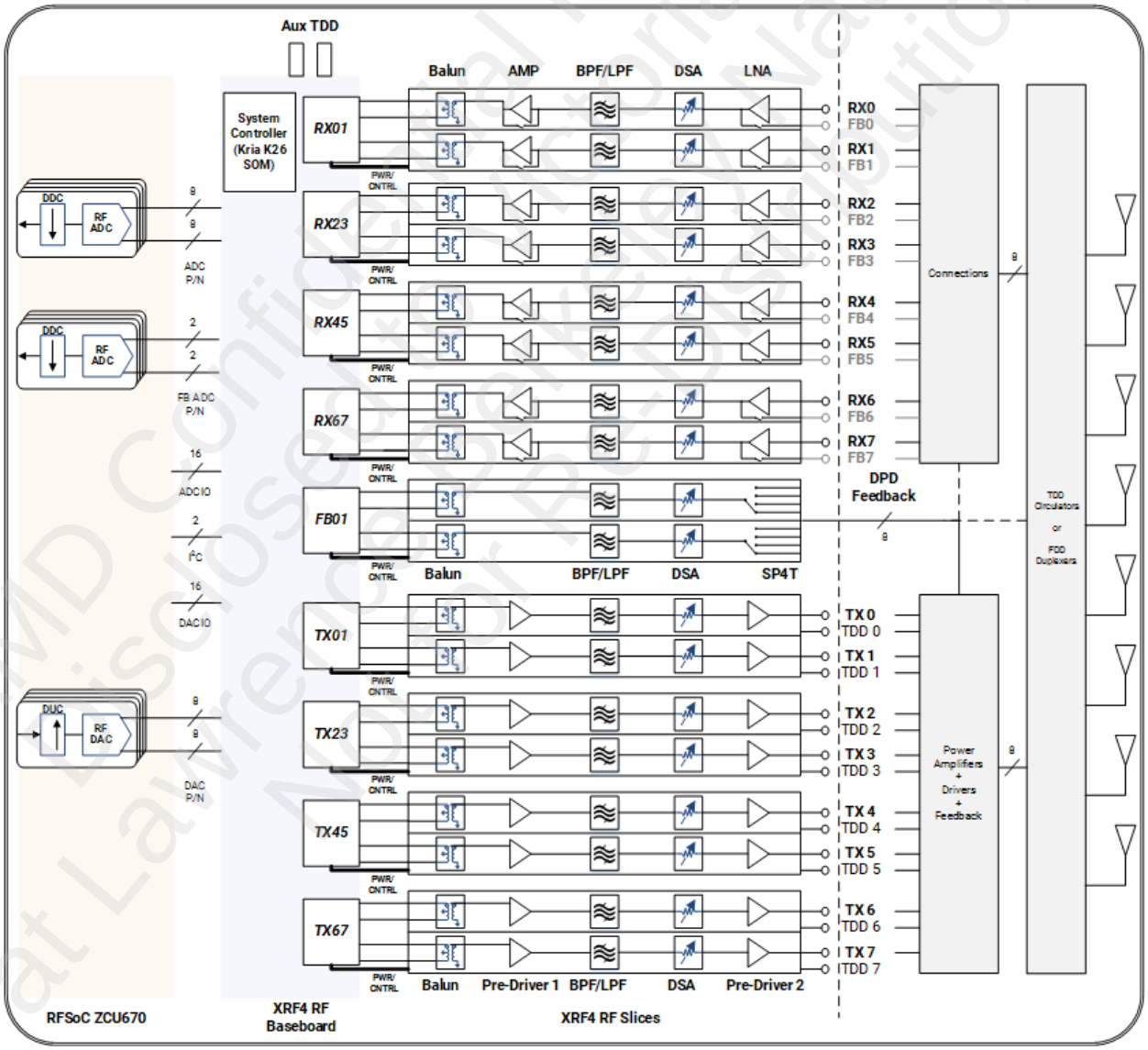
Figure 6: **AMD RF Board Version 3.0 Block Diagram**



AMD RF Board Version 4.0

The AMD RF Board Version 4.0 (XRF4) has high-performance, wideband mixed-signal, up to eight transmit and receive and up to two observation paths for DPD working with ZCU208 and ZCU670 boards. The figure shows a high level block diagram of the AMD RF Board v4.0 with the RF Mezzanine Card (RFMC) connector to allow connection to the AMD RFSoC development board. The XRF4 board setup includes the XRF4 Baseboard, the AMD Kria™ SOM card and up to nine XRF4 Slice cards (four TX Slice cards, four RX Slice cards, and one FB observation card).

Figure 7: **AMD RF Board Version 4.0 Setup Block Diagram**



Getting Started

ZIP File Description

The DFE Radio Demo is provided as a ZIP file.

IMPORTANT! *Unzip the ZIP files in a path on your Windows PC without spaces in any directory names.*

Table 2: **ZIP File Contents**

Directory	Description
dfe_demo_v9_1.zip	ZIP file name
-dfe_demo_v9_1/	Top-level directory
--matlab/	MATLAB® folder
---demo_lib/	Files for Evaluation Demo
----common_demo	Includes MATLAB files common to both demo modes
----dfe_rf_demo	Directory with files specific for RF mode with AMD RF boards configured for DAC operating at various sampling rates
--DFEEvalToolWizard/	Files for DFE Evaluation Tool Setup Wizard
---lib	Directory with lib files for DFE Evaluation Tool Setup Wizard
--dpd_v14_1_dbg/	Files for DPD debug tool
---dpd_drivers	Files for DPD drivers used in debug tool
---framework	Files for framework of GUI and hardware connection
---gui	Files for DPD debug tool GUI
---jtag_drivers	Files for JTAG connections
---tcl	Tcl script
---utilities	Files to support different functions
--SD_Images/	SD Images for all evaluation boards
---sdproducts_zcu102.zip	sdproducts ZIP file for ZCU102
---sdproducts_zcu111.zip	sdproducts ZIP file for ZCU111
---sdproducts_zcu208.zip	sdproducts ZIP file for ZCU208
---sdproducts_zcu670.zip	sdproducts ZIP file for ZCU670
--JTAG_Builds/	JTAG loading builds for all evaluation boards
---sw_products StandaloneExeFiles	Common software products for each evaluation board

Connecting the ZCU Board and Host PC

The ZCU development board must be connected to the host Windows PC for all demo modes. For the detailed connection, see Network Setup.

Connecting the ZCU102 and XRF2 Boards

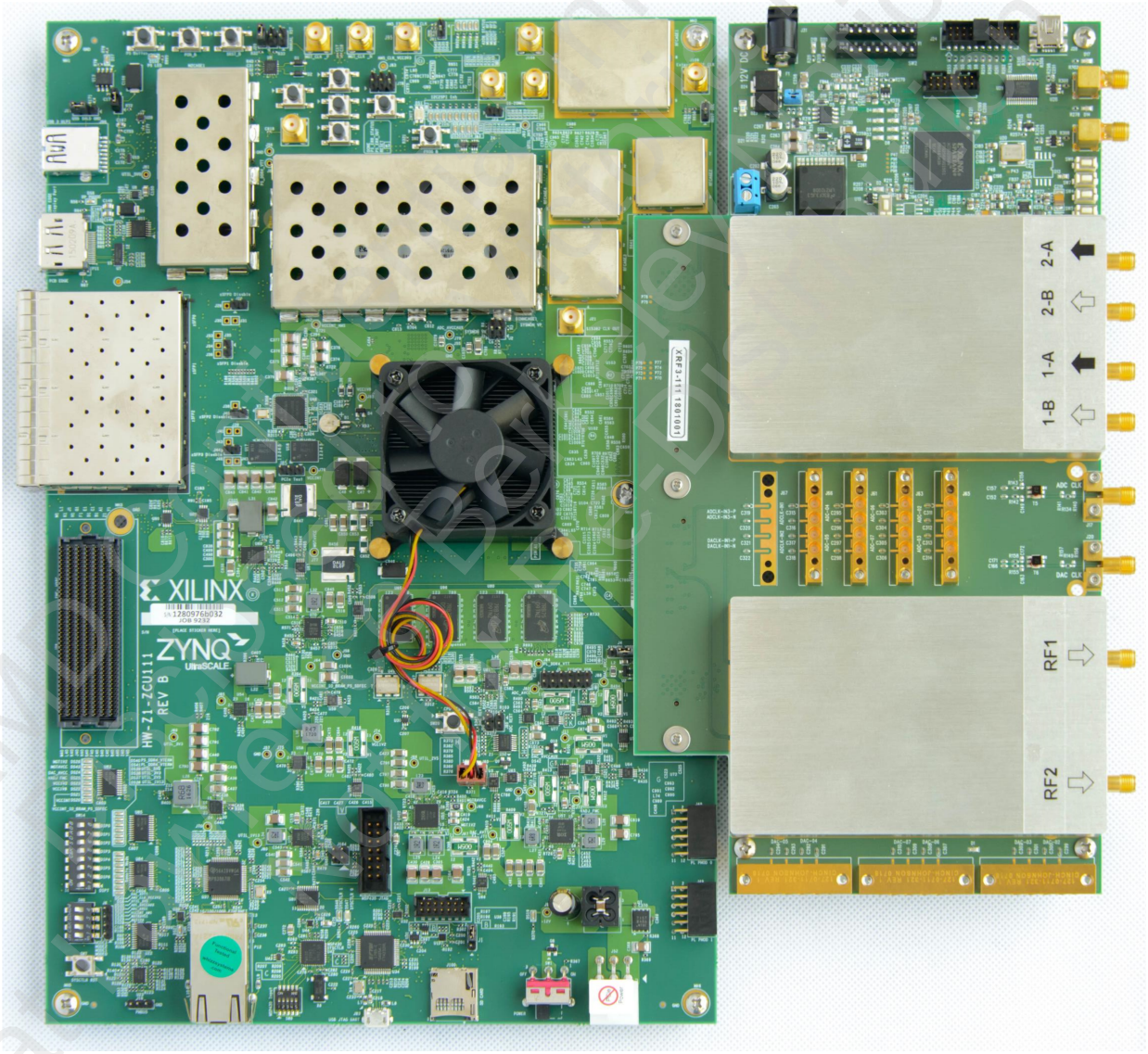
To operate the demo in RF mode, the RF board must be connected to the ZCU102 board. Connect the boards using the FPGA Mezzanine Card (FMC) connectors shown in the figure as an example with ZCU102.

Figure 8: **ZCU102 with AMD RF Kit v2.0**

Connecting the ZCU111 and XRF3 Boards

To operate the demo in RF mode using the ZCU111 board, use either XM500 (provisioned with ZCU111 kit) (see *Zynq UltraScale+ RFSoC ZCU111 Evaluation Kit Product Brief* (https://www.xilinx.com/publications/product-briefs/zcu111-product-brief.pdf)) or with the AMD RF Kit v3.0 as shown in the following figure.

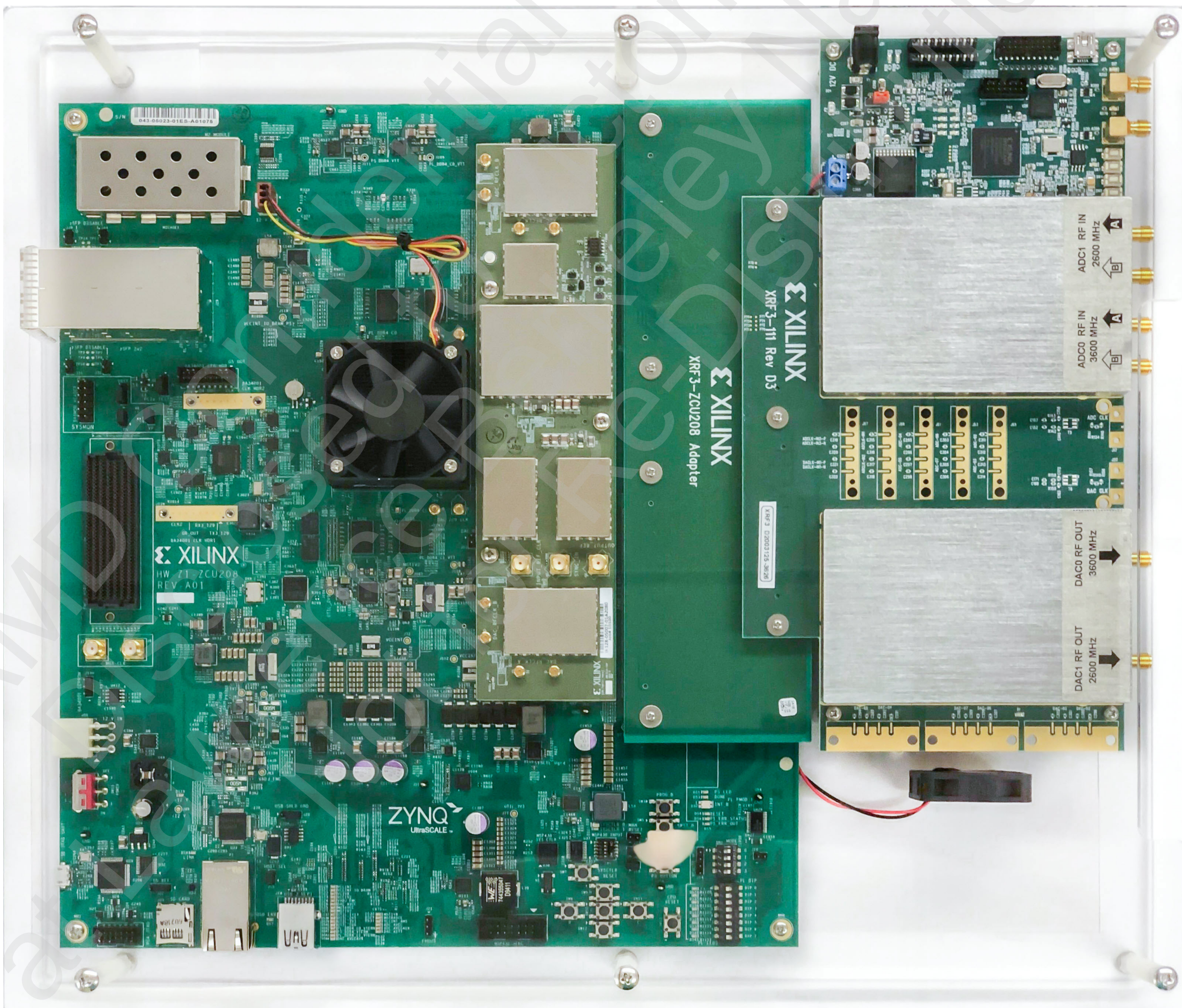
Figure 9: **ZCU111 with AMD RF Kit v3.0**



Connecting the ZCU208 and XRF3 Boards

To operate the demo in RF mode using the ZCU208 board, use either XM655 (provisioned with ZCU208 kit) (see *Zynq UltraScale+ RFSoC ZCU208 Evaluation Kit Product Brief* (https://www.xilinx.com/publications/product-briefs/xilinx-zcu208-product-brief.pdf)) or with the AMD RF Kit v3.0 (with XRF3-ZCU208 Adapter card) as shown in the following figure. Use default jumper and switch settings as described in Tables 3 and 4 of the *ZCU208 Evaluation Board User Guide* (UG1410) except for SW2 which should be set to SD2.0 boot mode when using the SD card Boot Setup.

Figure 10: **ZCU208 with AMD RF Kit v3.0 (with XRF3-ZCU208 Adapter Board)**



Connecting the ZCU670 with XRF3 or XRF4 Boards

To operate the demo in RF mode using the ZCU670 board, use either XM755 (provisioned with the ZCU670 kit) (see *Zynq RFSoC DFE ZCU670 Evaluation Kit Product Brief* (https://www.xilinx.com/publications/product-briefs/xilinx-zcu670-product-brief.pdf)), or AMD RF Kit v3.0 (with XRF3-ZCU670 adapter card) as shown in the following figure, or AMD RF Kit v4.0 (includes XRF4 Baseboard, AMD Kria™ SOM card, and RF slice card) in the next figure. Use default jumper and switch settings as described in Tables 3 and 4 of the *ZCU670 Evaluation Board User Guide* (UG1532) except for SW2 which should be set to SD2.0 boot mode when using the SD card Boot Setup.

Figure 11: **ZCU670 with AMD RF Kit v3.0 (with XRF3-ZCU670 Adapter Board)**

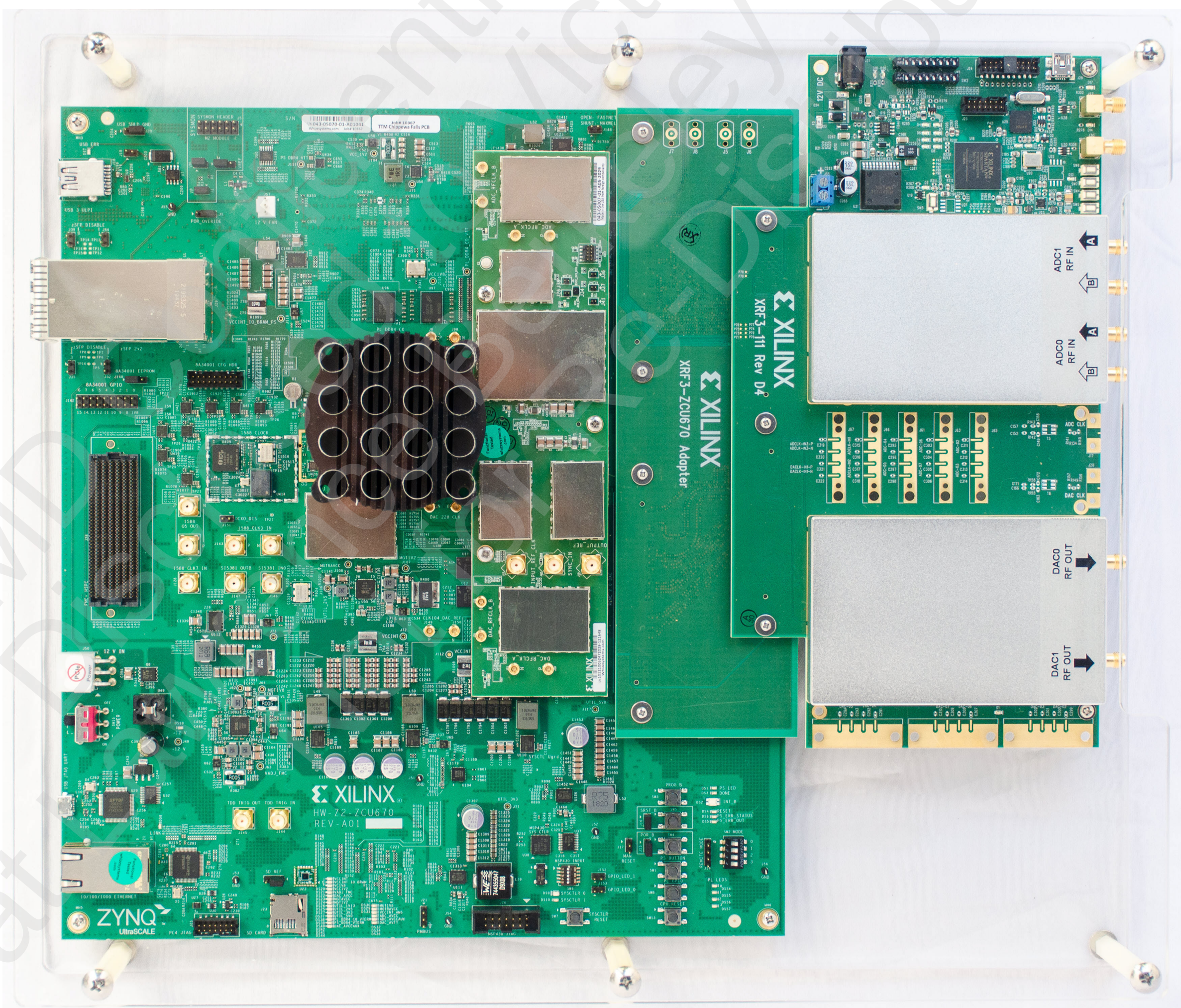
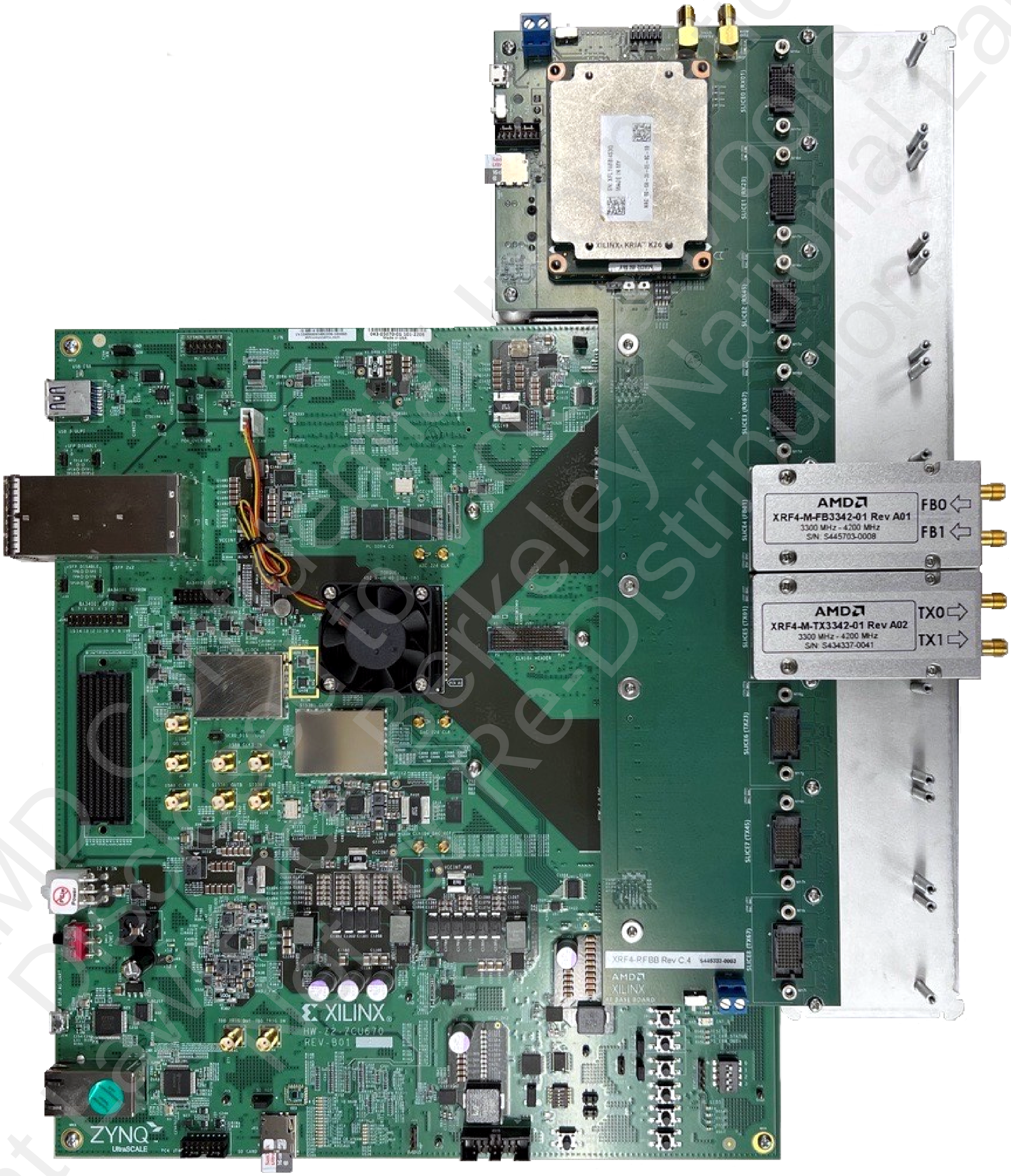


Figure 12: **ZCU670 with AMD RF Kit v4.0**



Using the DFE Evaluation Tool Setup Wizard

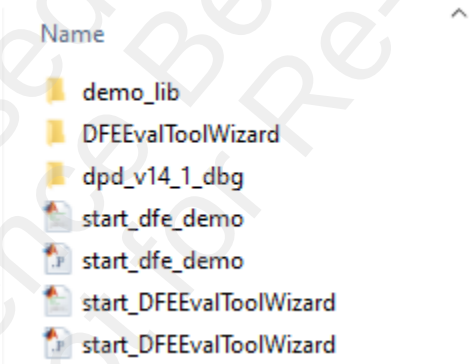
The DFE Evaluation Tool Setup Wizard is included in the package to support a step-by-step walk-through of PC software setup and ZCU* and RF board setup in the lab.

1. To get started, open a compatible version of MATLAB® on your PC/laptop.

   Note: If you have an incompatible MATLAB tool version, an error message is reported by this tool, when it is running.

2. Navigate to the `<unzipped area>/dfe_demo_v9_1/matlab` directory, and run `start_DFEEvalToolWizard` command on the MATLAB command prompt. When this tool is launched, a unique `<date>_<time>_check_log.txt` file is created for future debug reference.

Figure 13: **MATLAB Subdirectory Structure**

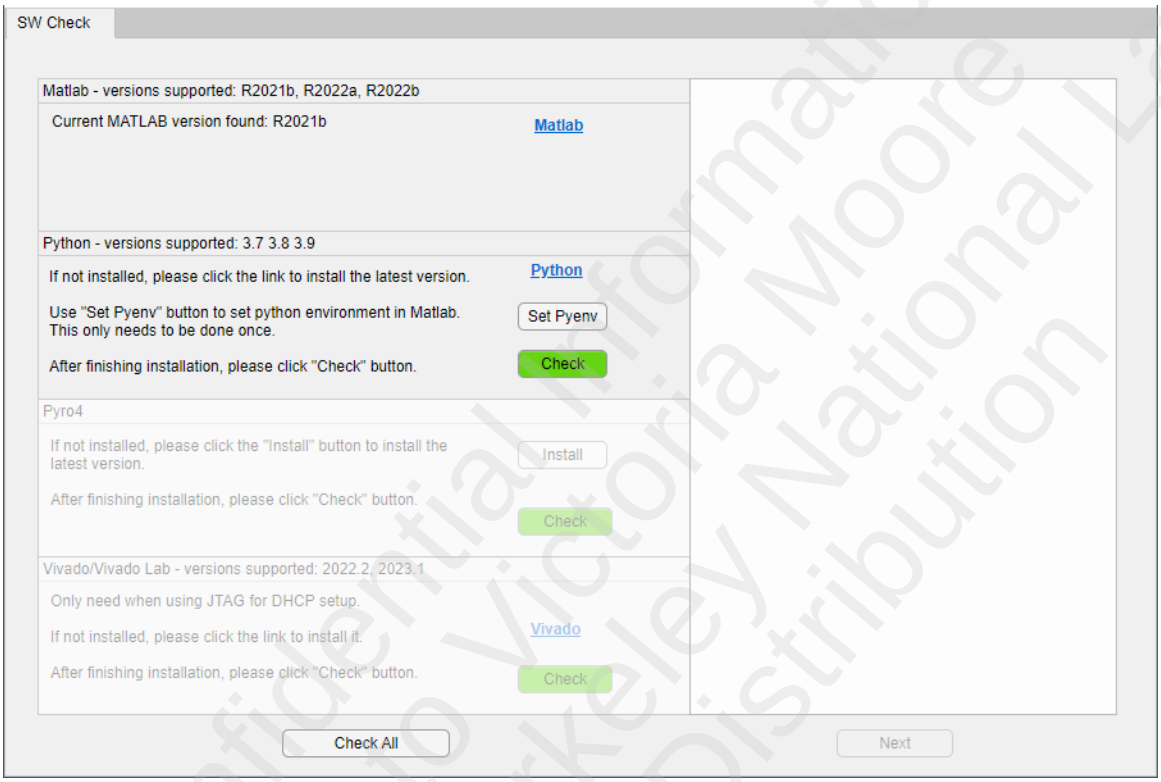


Dependency Software Check

1. Follow the steps shown on the GUI to check all software (SW) versions one by one using the visible **Check** button. The following figure shows a sample report of the compatible software installed.

   Note: AMD Vivado™/Vivado Lab is optional if you only need to set up static IP address for the evaluation board.

Figure 14: **DFE Evaluation Tool Preparation Setup – Successful SW Install/Check**



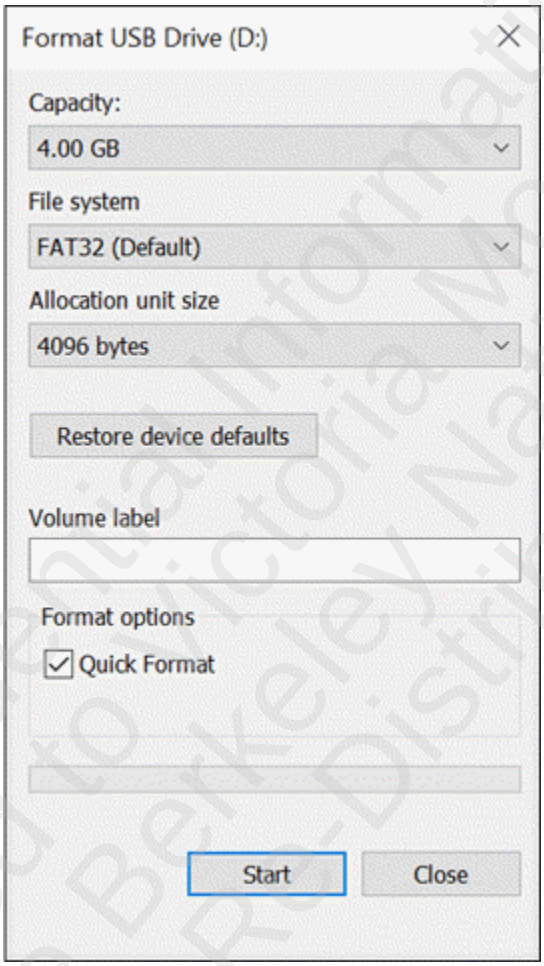
2. If the tool reports incompatible versions for any of the dependency software, follow the instructions on the GUI to update your PC.
3. When you have successfully completed this step, click the **Next** button to proceed to the next step.

Prepare the SD Card

Review the instructions on the GUI to prepare an SD card. The same details are captured here in the document as well.

1. Obtain a microSD card with >4 GB size (strongly recommend using the same one from the ZCU* board package/box as those are pre-qualified for the SD card boot on ZCU* board). Format the card in Windows, FAT32, 4 GB (see the following figure). If your laptop/PC does not have a built-in reader/writer for microSD card, use an appropriate microSD card adapter.

Figure 15: **Windows Drive Format**

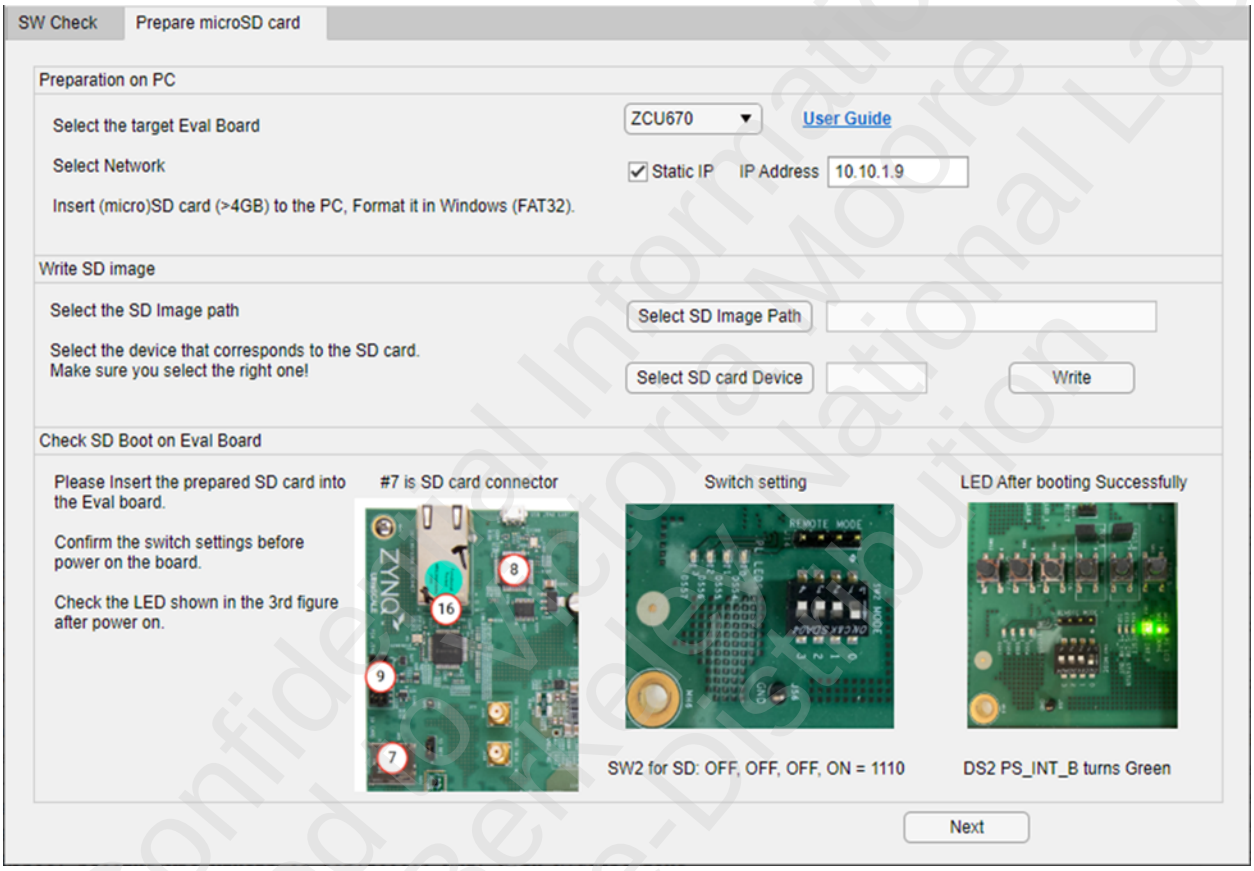


Note: For the Allocation unit size drop-down list, select the smallest size if you do not see 4096 bytes.

RECOMMENDED: *The microSD/SD card for each evaluation board is the card shipped with the ZCU* kit. They are 16 GB or 32 GB size cards from SanDisk. If you lost the card, contact the AMD support team to find out exact part to procure.*

2. Proceed with using the tool to write appropriate files into the microSD card.

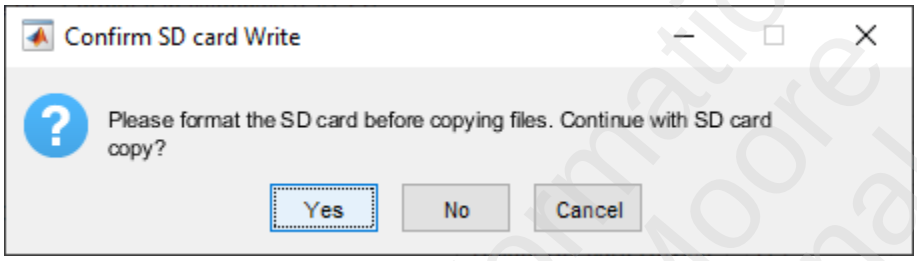
Figure 16: **Prepare SD Card**



3. Select the ZIP file for your evaluation board by clicking the **Select SD Image Path**. Then, select the **SD card Device** pull-down for your specific drive.
4. If selecting the **Static IP**, enter the desired IP Address for the board in the text field, so it can be included as part of the SD card image. If the **Static IP** is unchecked, **DHCP** is selected and you would leave the IP Address text field empty. Make a note of the static IP address you selected here, to refer in the next step.
5. After clicking **Write** and confirming the SD card write (see the following figure), the Preparation Tool unzips the SD images ZIP file for that board and writes the files into the microSD card.

   ***Note*:** Unzipping and copying files might take time depending on the file size.

Figure 17: **Confirm SD Card Write**



6. After confirming completion of the Write to microSD card step, remove the microSD card from the laptop/PC and insert it into the Eval board and set the proper switches based on the descriptions in the DFE Evaluation Tool. Additional information about those boards and its switches and LED can be found using the DFE Evaluation Tool Setup Wizard.

   Note: After booting successfully, the PS_INT_B or FPGA_INT_B LED turns Green.

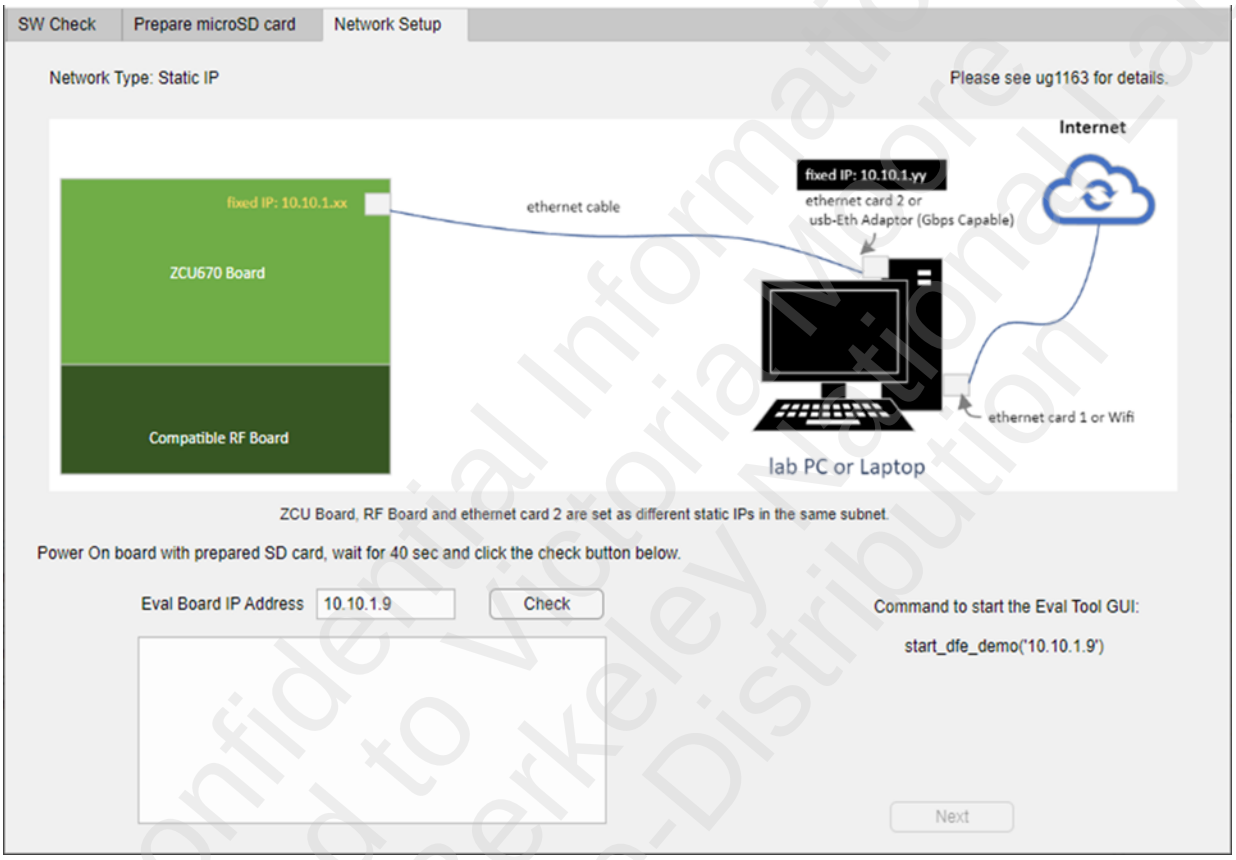
7. After successful boot up, click **Next** to configure the boards and PC for appropriate network connection.

Network Setup

If you select Static IP in the SD Card tab, this tab uses that information here and provide guidance for the Static IP configuration process. If Static IP was not selected, it provides the setup process for the DHCP method.

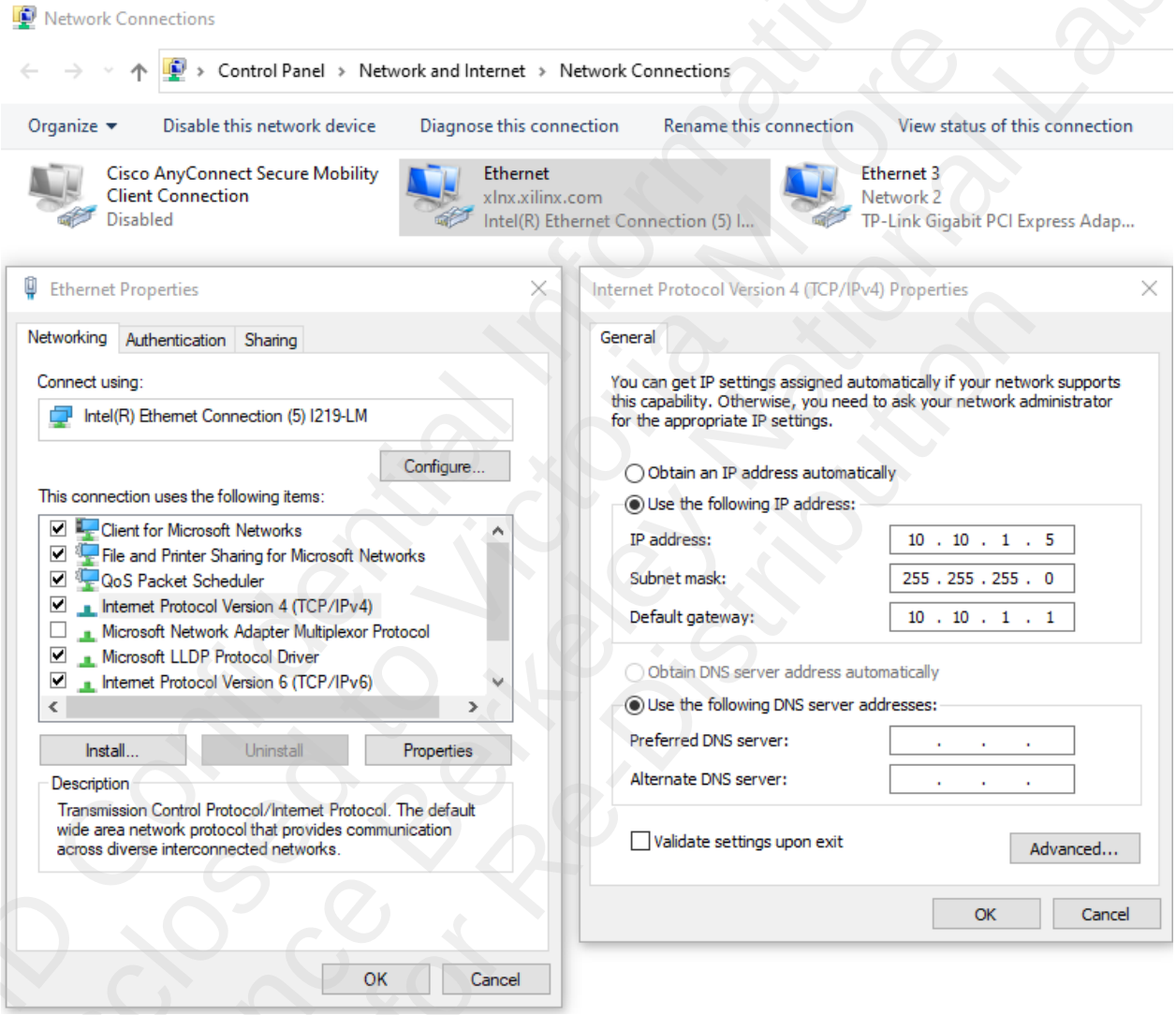
1. Follow the figure to hook up the board and PC directly for static IP.

Figure 18: **Static Network Setup**



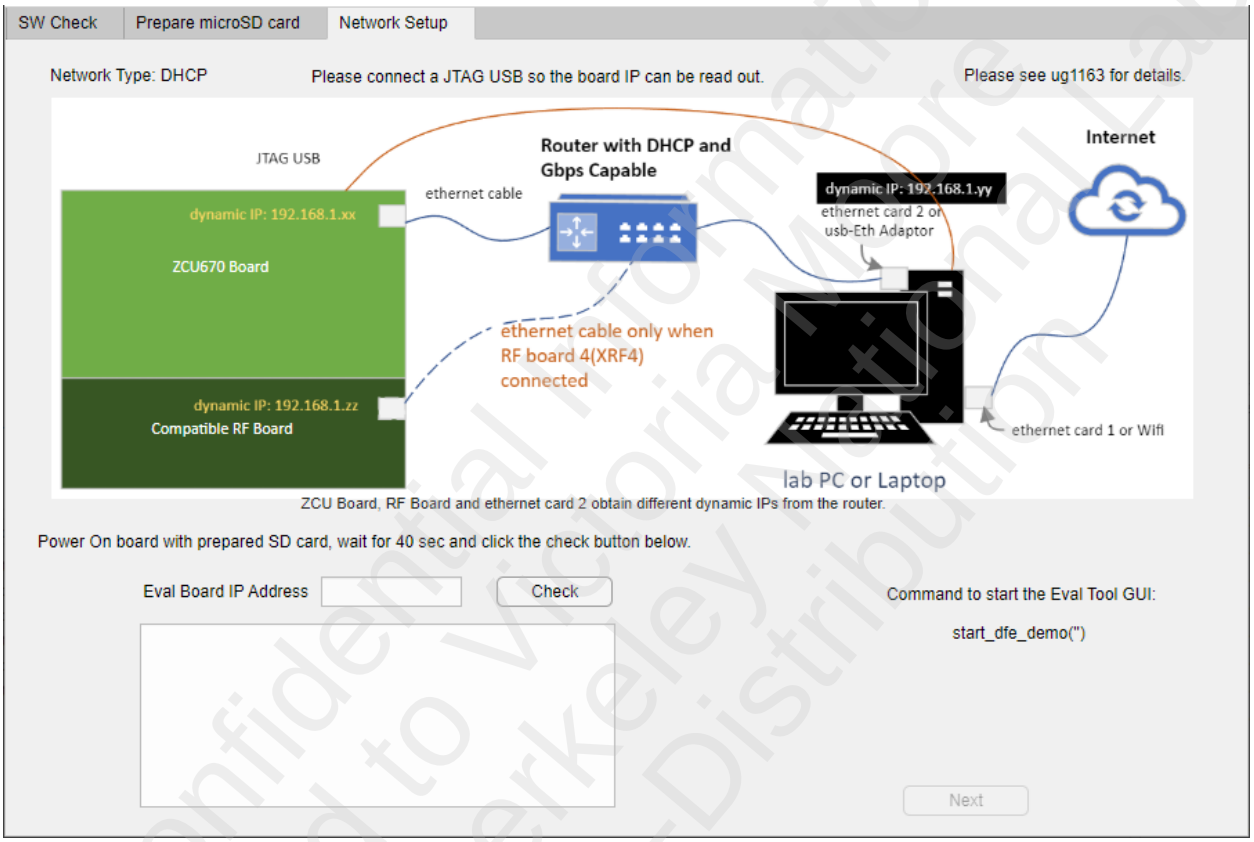
2. As per the diagram on the GUI, ensure to set (see the following figure) the lab PC's relevant Ethernet adapter with static IP address in the same subnet as the board's static IP that was written into the SD card image in the previous step.

Figure 19: **Windows Configuration to Manually Set Static TCP/IPv4 IP Address**



3. Click the **Check** button to test the IP address to confirm that the PC and board can communicate with each other. Make a note of the command line example for using the static IP address to launch the DFE Evaluation Tool later on.
4. If you select **DHCP** in the SD Card tab, the following figure shows how to set up the DHCP server. A USB cable to connect JTAG/UART port to PC is needed.

Figure 20: **DHCP Network Setup**

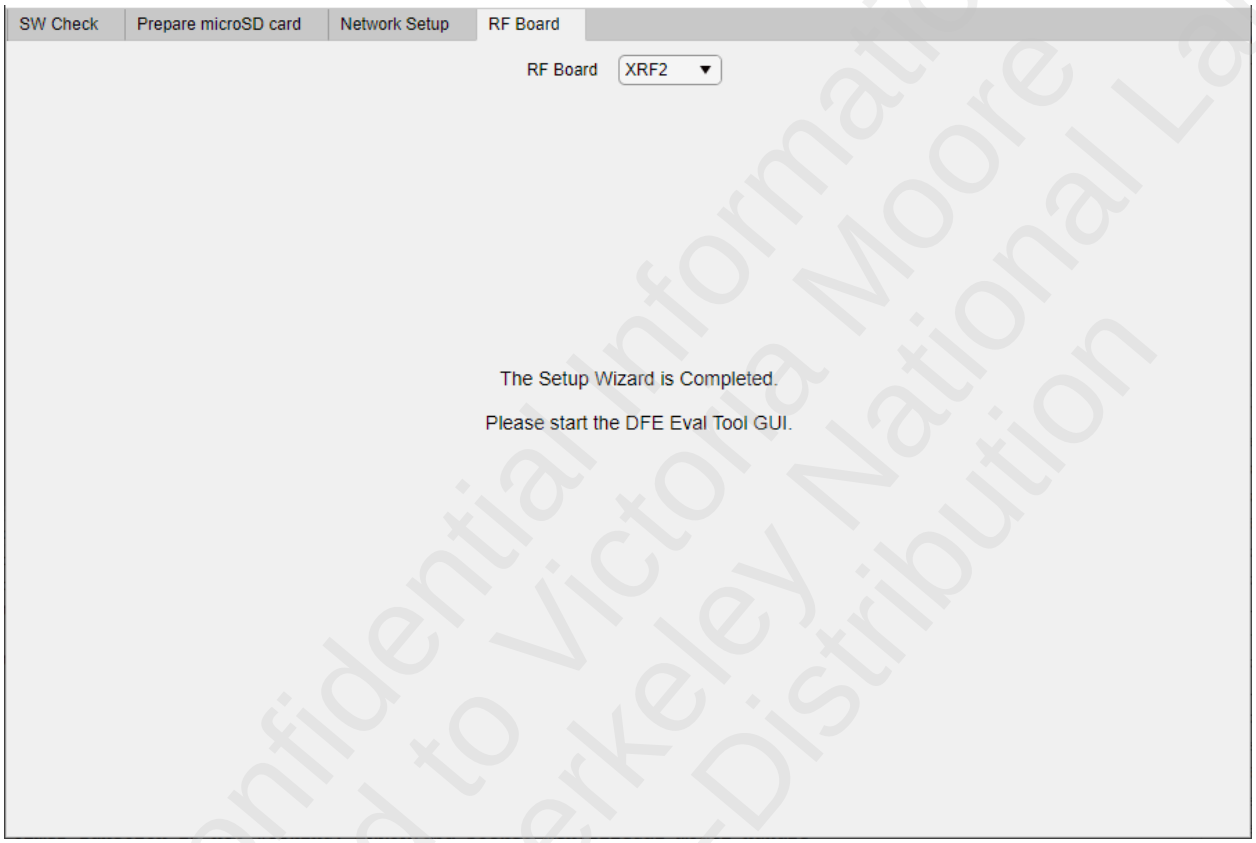


5. Click **Next** to configure the next tab.

Check the RF Board Network Connection

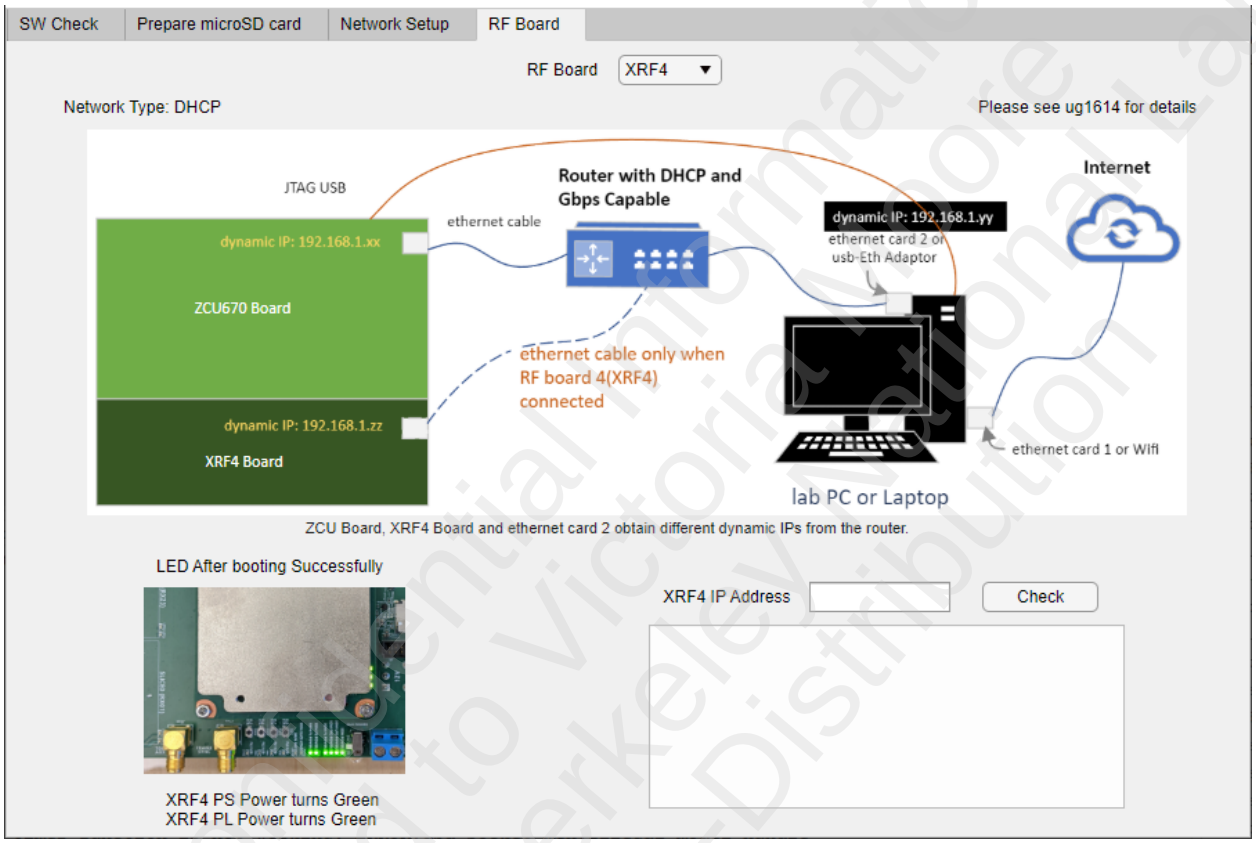
If the RF board is XRF2 or XRF3, the setup wizard is completed. If the XRF4 board is selected, the XRF4 board IP address needs to be checked.

Figure 21: **RF Board Setup Wizard**



The following figure shows a diagram on how to hook up the board and PC directly. Enter the IP address of the XRF4 board and click the **Check** button to verify the connectivity. For the DHCP setup, leave the XRF4 IP address empty and the system finds the IP address. For the static IP setup, the IP address is `10.10.1.7`.

Figure 22: **XRF4 Network Connectivity Configuration Tab**

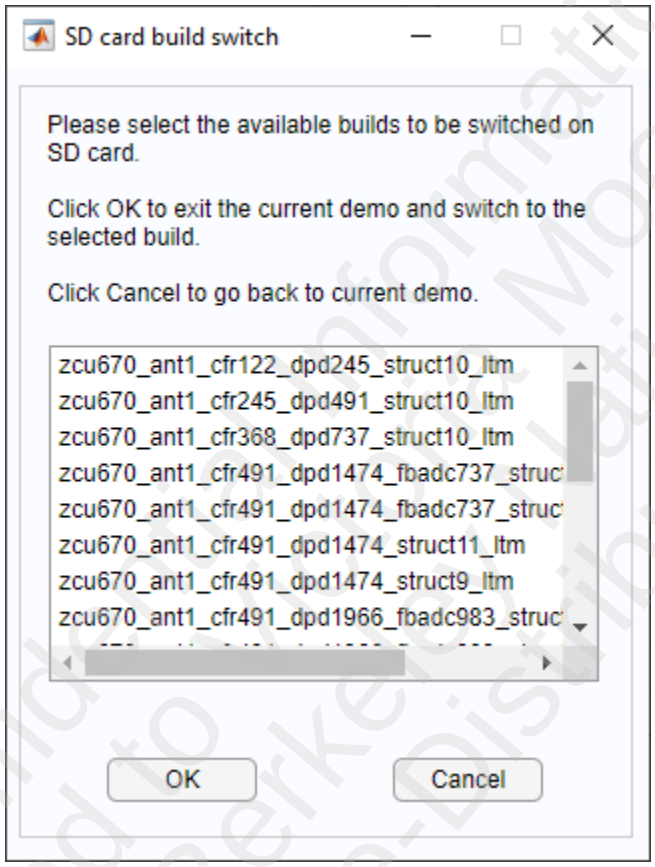


Switch Build on SD Card

The DFE Evaluation Tool GUI loads based on the default design in the `SD_Images` package. You can switch to other builds in the SD card by clicking the icon ( ) which is located at upper right corner of the DFE main GUI.

After clicking the **Builds Switch** button, a list of available builds is displayed.

Figure 23: **SD Card Build Switch**



Select your preferred build and click **OK** to continue. After the GUI pop-up window reports success, power cycle (OFF > ON) the ZCU evaluation board to boot this newly selected design.

Refer to the DFE Evaluation Tool usage guide from the DFE Evaluation Tool Lounge (registration required) to configure and use the selected design on the board.

Using the DFE Evaluation Tool for CFR + DPD Only Designs

Digital Loopback Mode

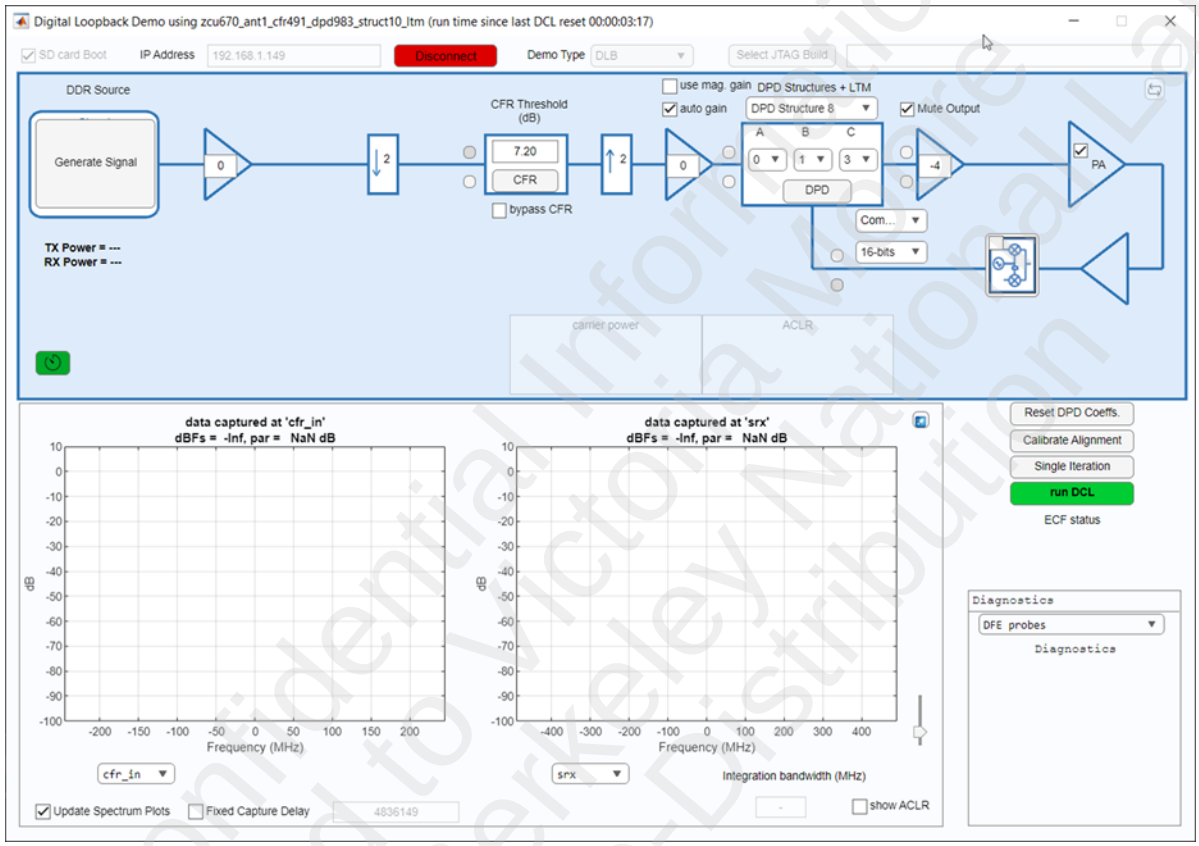
This section describes the operation of the DFE Evaluation Tool in digital loopback mode. All supported AMD evaluation boards can be used in this mode.

1. Set up the hardware and software as per instructions in Chapter 3: Getting Started.
2. Follow Chapter 4: Using the DFE Evaluation Tool Setup Wizard to copy SD images to the microSD card. Then, power up the evaluation board and check the Network settings.
3. Open a compatible MATLAB® tool and navigate to the `<unzipped area>/matlab`. Run `start_dfe_demo()` on the command window.

```
>> start_dfe_demo('Board IP', 'DLB') %% for static IP
% OR
>> start_dfe_demo('', 'DLB') %% for DHCP
```

When launched for the first time, the DFE Evaluation Tool GUI starts up based on the default design in the SD_Images package. You can switch to other builds in the SD card using the icon highlighted in Switch Build on SD Card at the bottom right of the Evaluation GUI with CFR/DPD only design.

Figure 24: **Demo GUI with Digital Loopback**

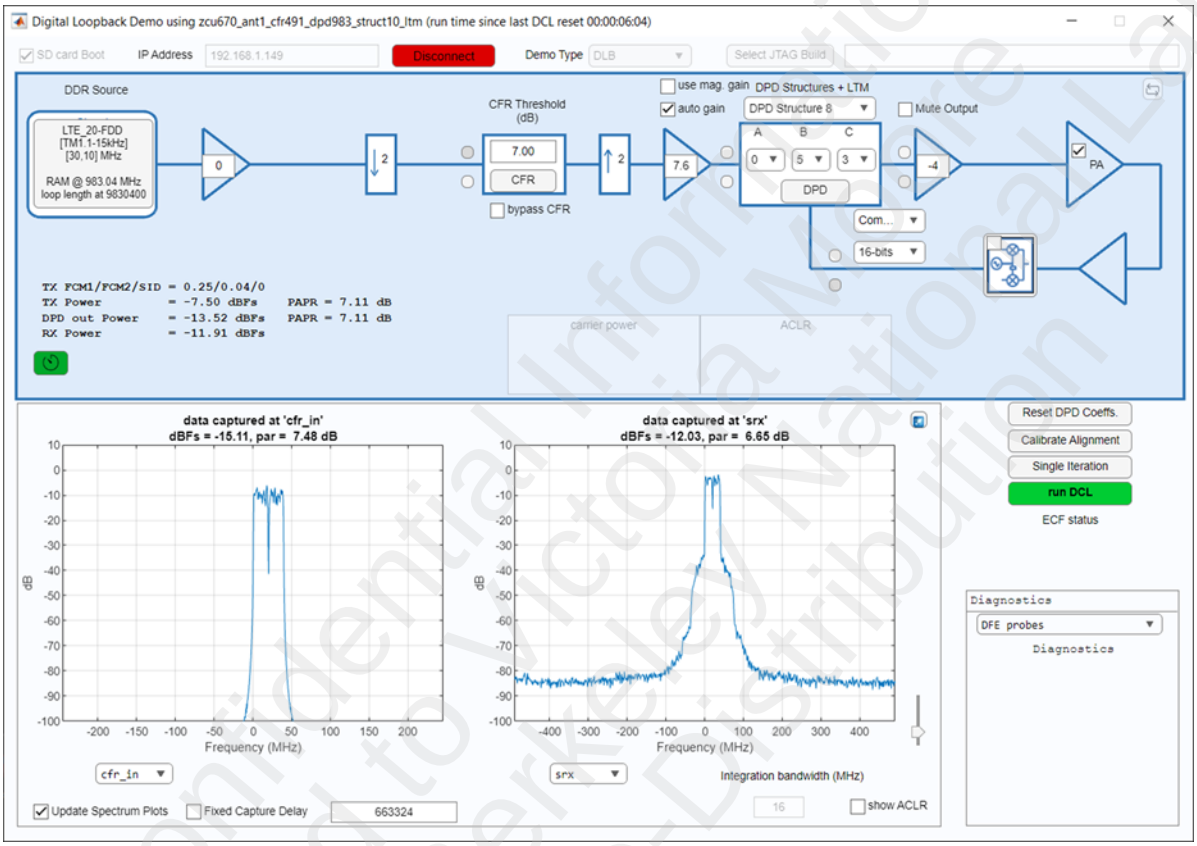


4. At this stage you are ready to load a signal (see Generating TX Signal Data for more information).
5. Select LTE_20MHz Test Mode 1.1(TM1.1) with two carriers located at 10 MHz and 30 MHz offset.

   Note: The same data is sent to all antenna paths in multi-antenna builds.
6. Uncheck the **Mute Output** box to see the receive waveform with PA and optional receive side I/Q imbalance and DC offset distortions. By default, the demo assumes a Complex receive IF architecture. A real receive IF architecture can be selected using the drop-down box above the feedback signal. When using the real receive IF, the I/Q imbalance and DC offsets can be applied on the transmit side. When changing the receive architecture, the DPD is reset and reconfigured appropriately. If desired, the quadrature error can be removed by unchecking the associated box.

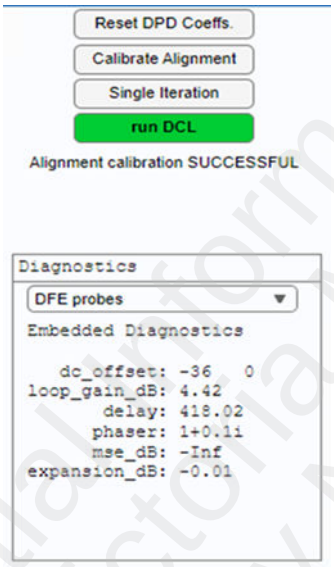
   The number of bits (precision) in the observation path that can be controlled by a drop-down box is shown in the figure.

Figure 25: **Demo GUI with Generated Signal**



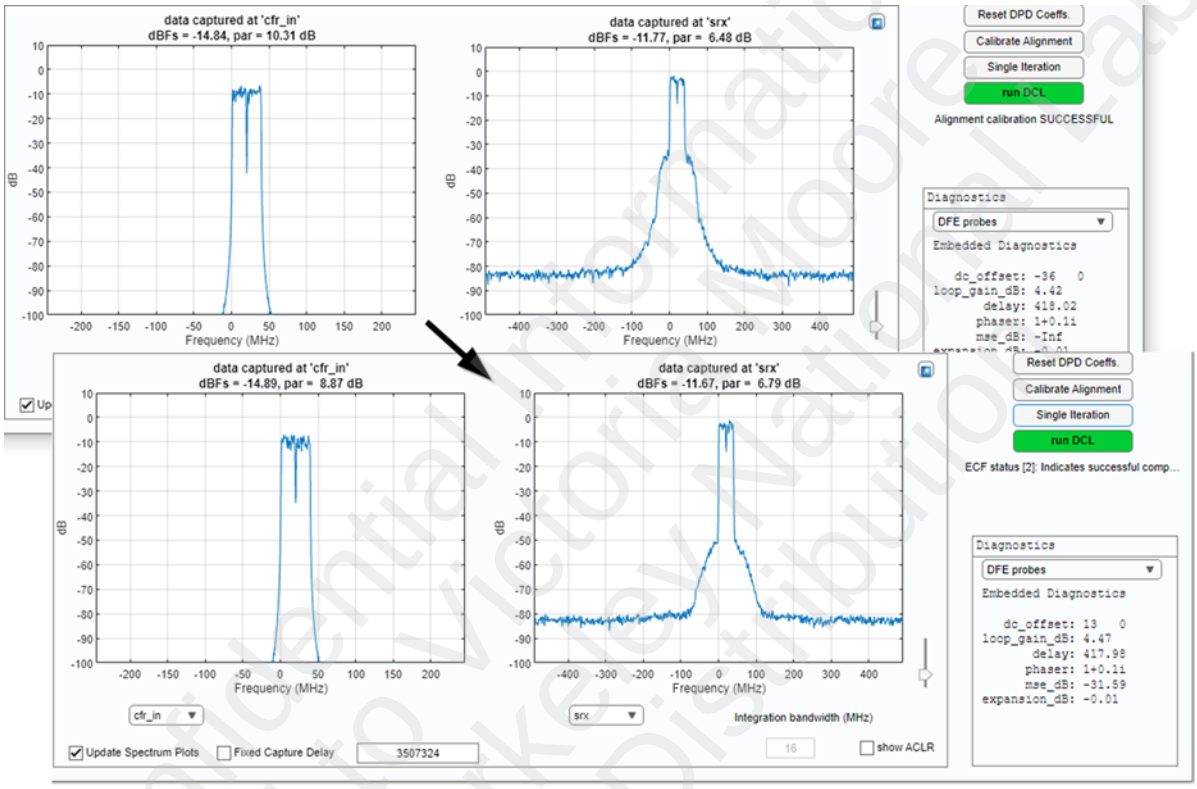
7. Set up the DFE operational parameters with the appropriate GUI controls or from the command line using the supplied `dfe_drivers` (type `help dfe_drivers` on the MATLAB console for a full list of available drivers). The controls available in digital loopback mode are described in DFE Radio Demo Parameters.
8. Calibrate Alignment: When the design line-up is customized, press **Calibrate Alignment** to allow the DPD core to check the RX data and report alignment diagnostics for the active port. The figure shows a successful alignment. Any errors found are also reported.

Figure 26: **Calibration Alignment Diagnostics**



9. Running DPD filter updates: The DPD filter correction capabilities can be demonstrated with the **Single Iteration** and the **Reset DPD Coeffs.** controls.
   a. Press the **Single Iteration** control to perform a single DPD filter coefficient update. This action causes DPD to update its filter coefficients to correct the non-linearities introduced by the PA model. The effect of linearization correction can be seen in the figure before and after the successful completion of an update. Multiple iterations can be run by pressing the **Single Iteration** sequentially, after the previous iteration is complete.

Figure 27: **Linearization with Single Iteration**

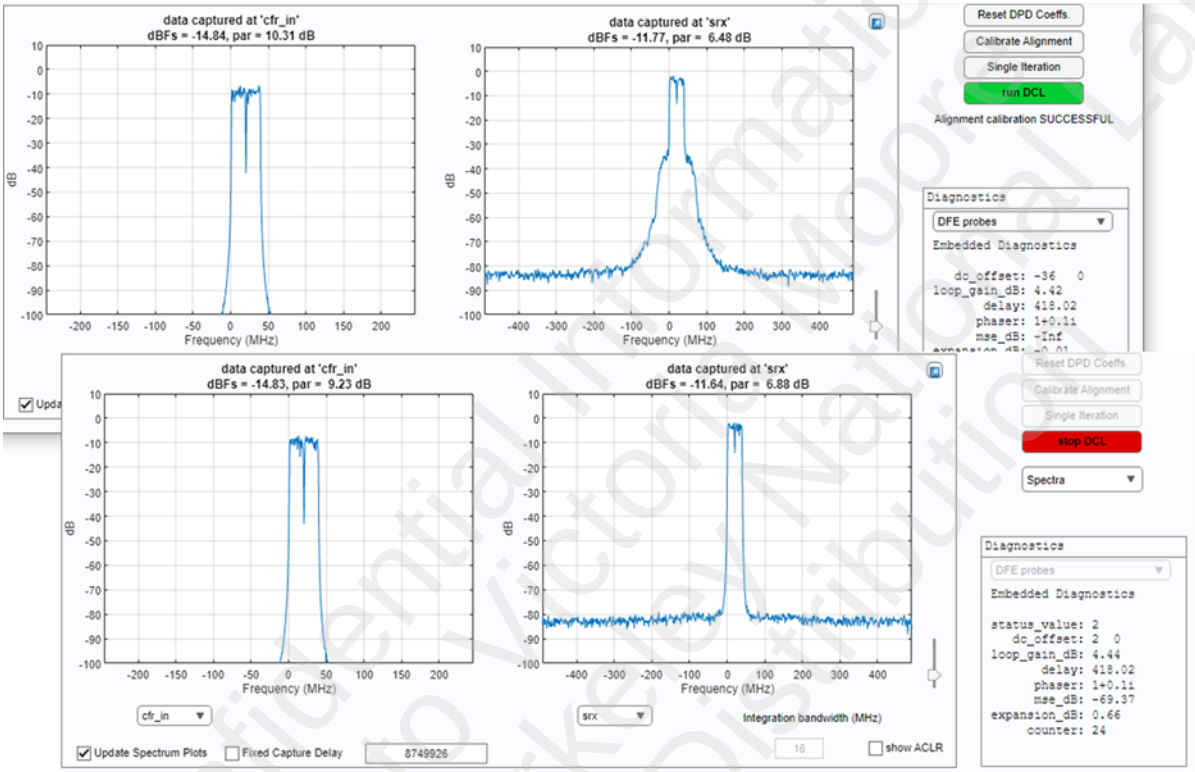


b. Press the Reset DPD Coeffs. to reset the DPD filter coefficients to their initial, pass-through state. This results in the sRX spectrum going back to its original shape including the spectrum regrowth near the carriers that result from the PA non-linearities.

***Note*:** For multi-antenna Digital Loopback demos; repeat steps 11 to 13 for each antenna. The active antenna is selected using the antenna drop-down box in the lower left of the GUI.

10. Running the Dynamic Control Layer (DCL): Press the green **run DCL** to activate the DCL; this becomes red while the DCL is active. See the *Digital Pre-Distortion LogiCORE IP Product Guide* (PG076) (registration required) for more details on the DCL operation. The figure shows the effect of activating the DCL. The effect can be clearly seen comparing the sRX signal spectrum before and after the DCL is active.

Figure 28: **DPD Updates with DCL Active**



11. For multi-antenna demonstrations with the DCL enabled, the main demo GUI displays the SRX spectrum from different ports randomly. This occurs because the DCL is controlling which antenna is currently active and the frequency of the GUI update is not co-related to the DCL port controller.
12. An additional method is available for measuring the quality of the correction, by allowing ACLR measurements on the signal at various points in the datapath. This is described in Spectrum Plots.
13. Additional diagnostic scripts are available in the **Diagnostics** drop-down menu. You can plot AM/AM expansion plot and TX/RX time aligned data. These can be used to detect issues, specifically when working with real PAs.

IMPORTANT! *All DPD power reporting is in dBFS (dB relative to digital Full Scale). Because of the limiting effect of CFR to peaks in the transmit signal, the effective power at the output of the CFR block can be a fraction of a dB below that measured at the input of the CFR block.*

RF Mode for CFR + DPD Only Designs

The AMD RF board-based DFE demonstration allows you to evaluate the CFR and DPD performance with amplifiers that operate in RF frequency bands that a particular RF board can support. The procedure to set up the boards, PA, lab equipment, and take measurements is described in the following documents:

- *ZCU111 Evaluation Board User Guide* (UG1271)
- *Xilinx RF Board v2.0 (XRF2) User Guide* (https://www.xilinx.com/member/dfe_demo.html#rfboards)
- *DFE Demo with ZCU and RF Board User Guide* `DFE_Demo_ZCU_and_RF_Boards_user_guide_vx_y.pdf` (https://www.xilinx.com/member/rfsoc-dfe-eval-platform.html)
- *Xilinx RF Board v4.0 (XRF4) User Guide* (https://www.xilinx.com/member/dfe_demo.html#rfboards)
- *XRF4 RF Accessory Kit Data Sheet* (DS1012)
- *XRF4 RF Accessory Kit User Guide* (UG1614)

The procedure to select and load appropriate designs on the AMD ZCU102, ZCU111, ZCU208, or ZCU670 boards is similar to the one described in Digital Loopback Mode. For specific details, see the above list of the AMD RF board user guides.

DFE Radio Demo Parameters

- **Pre-CFR gain:** Gain applied to the signal before the CFR stage in dB. The CFR core expects an input average power of -15 dBFS for full power signals—this gain is used to set the desired power level into the CFR. This gain is port specific in multi-antenna builds and is set from the GUI or from the command line with the `dfe_pre_cfr_gain` driver.
- **Post-CFR gain:** Gain applied to the signal after the CFR stage in dB.
  - Full-Scale DPD input mode (use mag gain control is unchecked): For optimal DPD performance in this mode, the output of CFR should be scaled such that the peak value is close to full scale at the DPD input (~ -0.5 to -1 dBFS peak is recommended). When auto gain is checked, this gain is automatically computed by measuring the maximum peak seen in the DPD histogram.

- -15 dBFS DPD input mode (use mag gain control is checked): In this mode, the gain required to make the highest peak output of CFR close to full scale is supplied to the DPD IP as a software parameter (see DATAPATH_GAIN). When auto gain is checked, the post-CFR gain is set to 0 dB and the required gain is automatically computed by measuring the maximum peak seen in the DPD histogram. The magnitude gain and datapath gain are programmed into DPD using the DATAPATH_GAIN parameter.

Any change to a CFR parameter causes the gain to be updated. This gain is port specific in multi-antenna builds and is set from the GUI or from the command line with the `dfe_post_cfr_gain` driver.

- **DPD Architecture:** This control allows the selection of the DPD filter architecture and the DPD filter structure used in a demonstration. For more details about the architectures and structures supported, see the *Digital Pre-Distortion LogiCORE IP Product Guide* (PG076) (registration required) and Appendix A: Equivalent DPD Architectures for Different Filter Memory Depth and Filter Structure Builds.
- **Post Gain:** This controls the scaling (in dB) applied to the transmit signal after DPD correction. This gain is port specific in multi-antenna builds and is set from the GUI or from the command line with the `dfe_post_dpd_gain` driver.
- **CFR threshold:** This control sets the CFR threshold to achieve a desired output PAR assuming an input of -15 dBFS (see PC-CFR Configuration and Performance for more details). This threshold is port specific in multi-antenna builds and is set from the GUI or from the command line with the `dfe_cfr_parameters` driver.

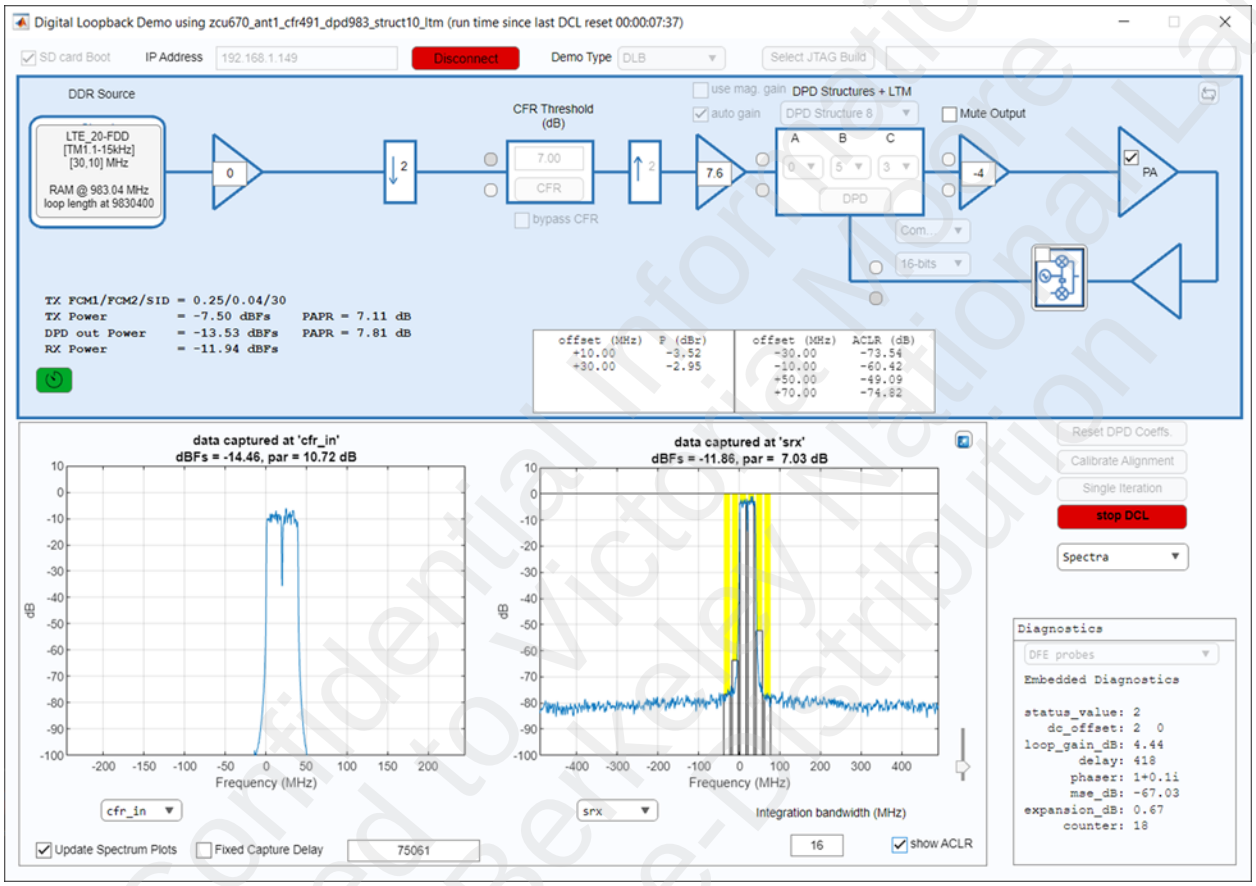
Digital Loopback Only

- **PA model enable/disable:** This control allows enabling/disabling of the PA.
- **I/Q imbalance and DC offset enable/disable:** These controls allow enabling/disabling of the Quadrature Modulator distortions.
- **Feedback path format:** This control allows for the selection of either Real or Complex feedback signal architectures.

Using the Demo GUI

This section describes some of the common elements of the demo GUI. For all screen captures, the Digital Loopback version of the demo has been used with a few (noted) exceptions. Therefore the top block diagram of the signal chain changes depending on the specific demo hardware configuration in use but the described functionality is common across each demo.

Figure 29: **Demo GUI with Digital Loopback Showing DPD Operation**



Generating TX Signal Data

Clicking within the **Signal Standard** box in the main GUI brings up a secondary GUI that is used to select or specify the desired signal.

The demonstration platform supports multiple options for signal generation and loading.

1. The demo GUI allows for the generation of a single frame of multi-carrier single RAT waveforms. 5G_NR and LTE waveforms are currently supported.
2. The demo also allows for the loading of custom signal vectors. The custom flow supports loading of `*.dsf` (created by the demo) and user supplied `.mat` files. When `.mat` is selected, the user is prompted to select the variable in the `.mat` file that contains the signal vector and optionally the CFR filter coefficients to be loaded into the demo. The CFR pulse must be saved in the `.mat` file in a format suitable for direct loading into the CFR engine. For more information, see the *Peak Cancellation Crest Factor Reduction LogiCORE IP Product Guide* (PG097) (registration required). For details on the CFR pulse requirements, see the *Peak Cancellation Crest Factor Reduction LogiCORE IP Product Guide* (PG097) (registration required).

   The format of the signal stored in the .mat files is as follows:

- Baseband complex I/Q data vector. The required sampling rate of the I/Q vector can be determined using the `dfe_get_configuration()` driver. The vector playback rate is reported in the `vec.fs` field in MHz.
- Integer values, typically scaled to -15 dBFS (for example, data = round(data × 10^(-15/20) × 2^15/std(data));).

3. The demo allows for the scripted generation and loading of multi-frame and multi-RAT signals. The scripting environment is described in `Xilinx DFE demo.pdf` and `DFE demo vector playback overview.pdf` located in the `.\common_demo\signal_generation\scripting` directory.

When selected and generated, the signal is loaded onto the DDR3 chips on ZCU102, ZCU111, ZCU208, and ZCU670. They are continuously looped through.

PC-CFR Configuration and Performance

This section describes the PC-CFR configuration and performance. Configuration is performed using AXI4-Lite registers. On the demo GUI, the **PC-CFR Configuration and WCFR Configuration** box (Figure 30) provides GUI control to program these registers. **Threshold (dB)** is the desired output PAR in dB assuming CFR input is at -15 dBFS RMS. The bit-files provided with the demo can have CFR in one of the two configurations, that is, CFR with smart peak processing enabled or CFR with smart peak processing disabled. The Post-Processing stage can either be Hard Clipper or Window Crest Factor Reduction (WCFR). CFR Hardware can be implemented as either DFE-CFR Primitive as provided on AMD Zynq™ UltraScale+™ RFSoC DFE devices or PL Only IP which is supported by all devices.

In the bit-files with CFR and DPD operating at the same sampling rate, the smart peak processing option has been disabled and the post-processing stage after PC-CFR can either be "Hard Clipper" or "WCFR".

***Note*:** The bit-files packaged with the demo have different combinations of sampling rates for CFR and DPD. This discussion is to illustrate CFR features for all bit-files and is not specific to the bit-files specifically mentioned in this section.

In the bit-files with DPD operating at double or higher sampling rate when compared to CFR, the smart peak processing option has been enabled. In those cases usually, CFR has three iterations with CPGs = [8,6,4]. The post-processing stage is WCFR with a window length set to 384.

There are per-iteration thresholds available for finer control of peak cancellation as shown in the CFR performance GUI screen capture (Figure 31). The snapshots of CFR Configuration GUI for both Smart Peak Processing Enabled and Non-Smart Peak Processing configurations are shown in the following figures.

Figure 30: **PC-CFR Configuration with Smart Peak Processing Enabled in Static Mode**

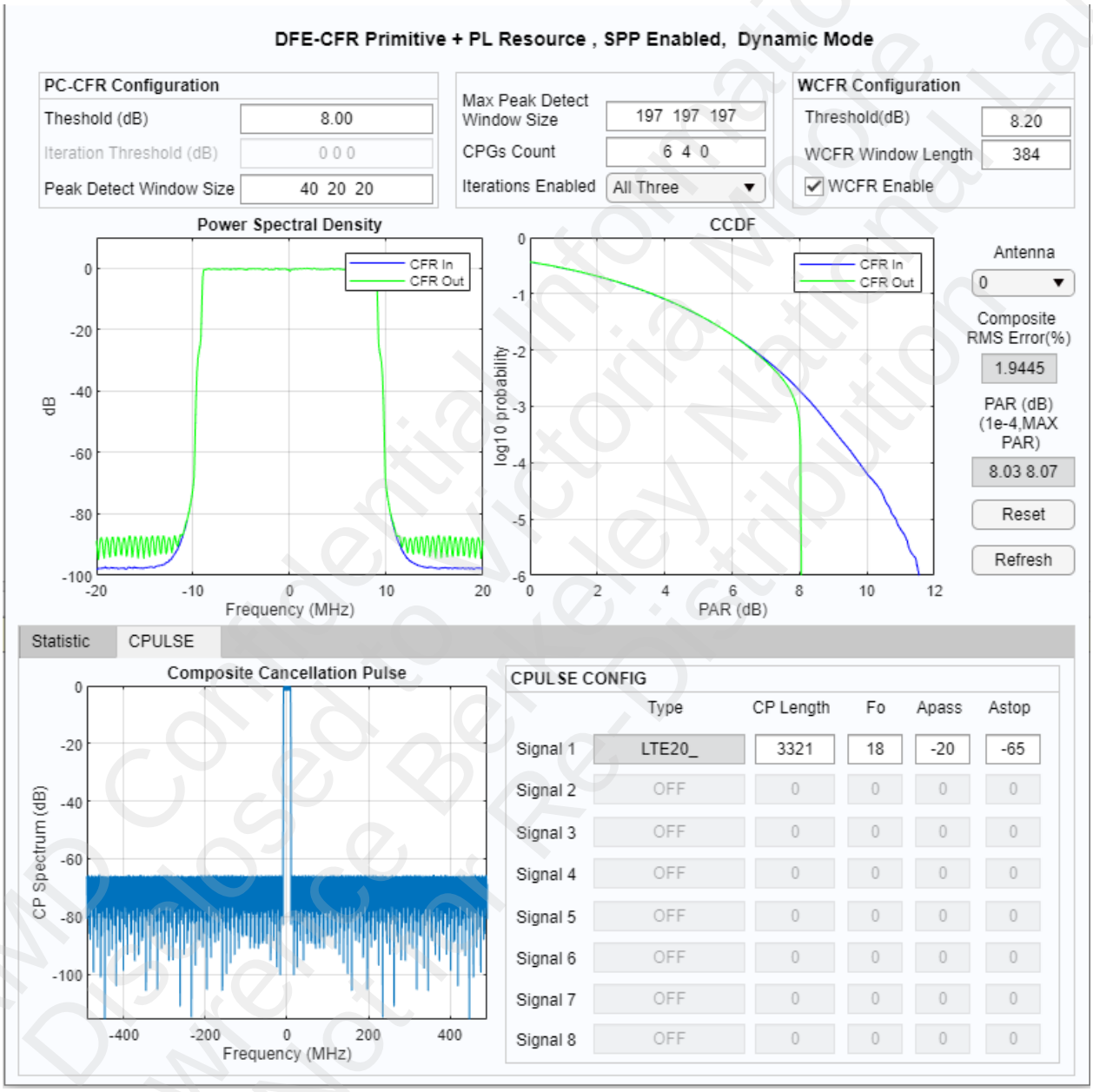
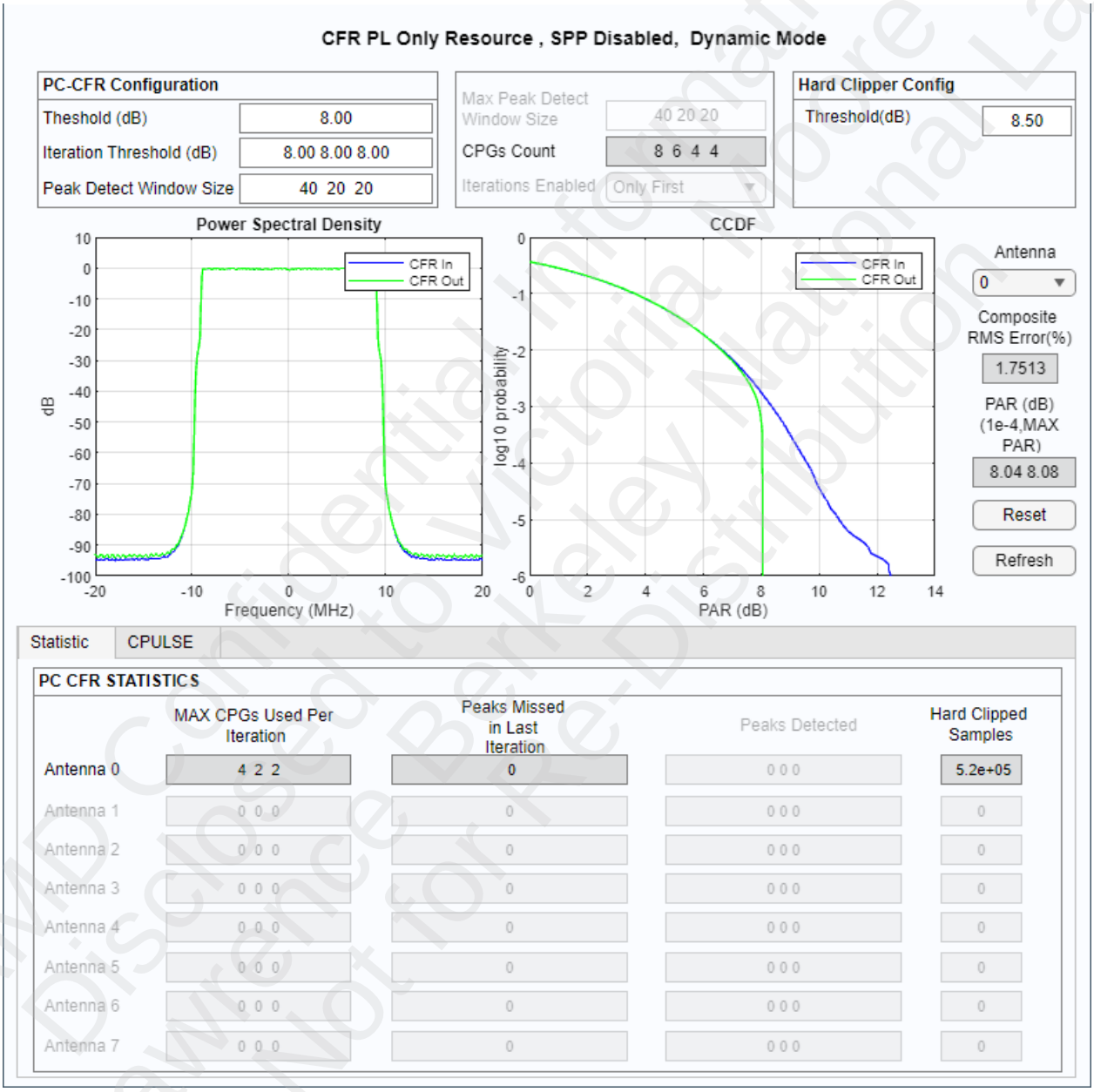


Figure 31: **PC-CFR Configuration with Smart Peak Processing Disabled in Dynamic Mode**



These thresholds default to the main threshold (**CFR Threshold**) whenever the main threshold value is updated in the GUI. These values can be modified individually after the main threshold is programmed. Hard clipper threshold is set to control peak PAR. Its default value is 0.5 dB higher than the main threshold. **Peak Detection Window Size** is used to qualify peaks to be canceled by picking the highest peak over those many samples. This helps in better peak cancellation with a multi-carrier input signal, (see the "Peak Detection" section in *Peak Cancellation Crest Factor Reduction in a Multi-Standard Transmit System* (XAPP1174) (registration required) for more details).

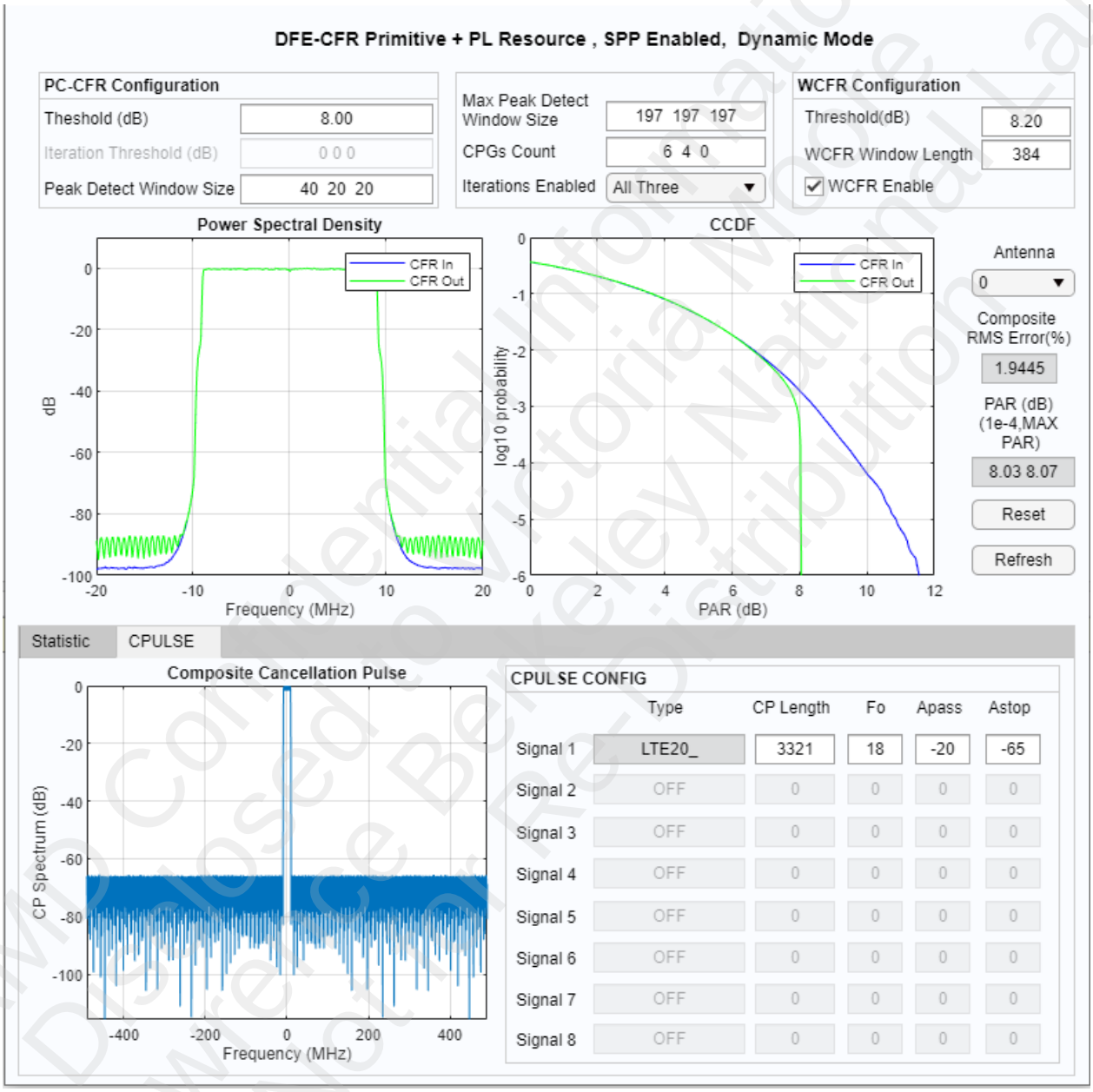
Actual output PAR could be 0.1 to 0.3 dB higher than the set target because cancellation reduces CFR output RMS. You can lower the CFR threshold to compensate this growth.

The **PC CFR Statistic** box reports various CFR performance parameters read from the AXI4-Lite register space. The field **WCFR Peaks Detected Per Iteration Per Antenna** is shown in the Statistic box (Figure 32). Maximum CPG utilization, Peaks Missed, and Peaks Detected are reported for each of the iterations per antenna. This helps in deciding correct number of CPGs and iterations needed for any given input signal.

Peaks missed (in the last iteration) is an indication if the number of iterations and CPGs chosen are adequate. Ideally the number of peaks missed should be ~1e-5 (expressed as clipping/miss rate) or less. Similarly peaks processed by the hard clipper should be ~1e-6 or less for the best CFR performance. **Peaks Detected per Antenna** is also reported in the **Statistics** box. This field is only available on DFE-CFR Primitive IP type and indicates peaks found by PC-CFR for each iteration. **Composite RMS Error (%)** is a metric which is a very good indicator of EVM and measured in runtime as RMS error between the captured CFR output and the corresponding input. Output **PAR** is also reported at two points, 1e-4 clipping probability and max PAR measured on the captured data.

The **Reset** clears all the reported parameters, while the **Refresh** updates all the fields with the latest register values and a fresh capture for EVM and PAR. You should press the **Reset** after any register programming to flush out the previous values of the reported parameters, especially the RMS error.

Figure 32: **PC-CFR Configuration with Smart Peak Processing Enabled in Static Mode**



With the bit-file having CFR with smart peak processing enabled, the field **Threshold (dB)** in the **WCFR Configuration** section and **WCFR Peaks Detected** in the **Statistics** box appear on the GUI as shown in Figure 32. WCFR threshold is used to control peak PAR. A key advantage of WCFR is it guarantees that no peaks are passed on unlike PC-CFR. The default value of WCFR threshold is 0.2 dB higher than the main PC-CFR threshold. Figure 32 shows the GUI with these changes.

The CPULSE tab in Figure 30 shows the composite cancellation pulse (CP) spectrum, which has to match the PSD of the input signal. Various control parameters are provided for adjusting the cutoff frequency, passband ripple, and stopband attenuation as used in the MATLAB `firceqrip` function. Stopband attenuation has to be at least 65 dB or more for good ACLR performance. Increase the CP length for better band-edge SEM performance. However, higher CP length increases the CFR latency and would need more CPGs for the cancellation of all qualified peaks.

***Note*:** CP is computed at 4 × CFR F_s when smart peak processing is enabled.

The right most plot in Figure 30 shows the CCDF of the input signal (in blue) and the CFR output (in green). The plot uses a whole RF frame (10 ms of data) capture, to estimate the CCDF.

Similarly, the left most Power Spectral Density plot in Figure 30 shows the input (in blue) and output (in green) power spectra. The refresh updates these plots (CCDF and PSD) with a fresh capture.

PC-CFR with Dynamics

The PC CFR core supports frequency dynamic and power dynamic modes. The frequency dynamic mode also called Frequency-Hop (FH) mode is aimed at FH MC-GSM. In this mode the CP has to be computed every slot, that is 577 µs, based on the frequency and power information fed over a separate AXI4-Stream channel.

The power dynamic mode assumes that carrier configurations do not change on-the-fly and carrier frequencies are programmed through an AXI4-Lite memory mapped interface. The number of RATs and carriers for each RAT are configured during core creation in the Vivado IP catalog. A power marker signal along with 8-bit relative carrier powers for three RATs are fed as sideband signals to the core and are part of AXI4-Stream `tuser` bus. The power marker indicates when to sample the relative carrier powers and triggers a new CP computation.

There are two options in power dynamic mode, slow and fast computation. In slow computation, the compute resource is shared across the carriers, whereas in fast computation, each carrier has a dedicated compute engine and therefore requires more DSP48s. Assuming a CP length of L, slow computation requires approximately NL/2 cycles (where N is the number of carriers), while fast computation needs L/2 cycles because N compute engines process the CP updates concurrently.

In some of the bit-files provided with the demo, the CFR core is configured to support power dynamic mode using a single RAT and a maximum of eight carriers supporting a wide variety of power dynamics.

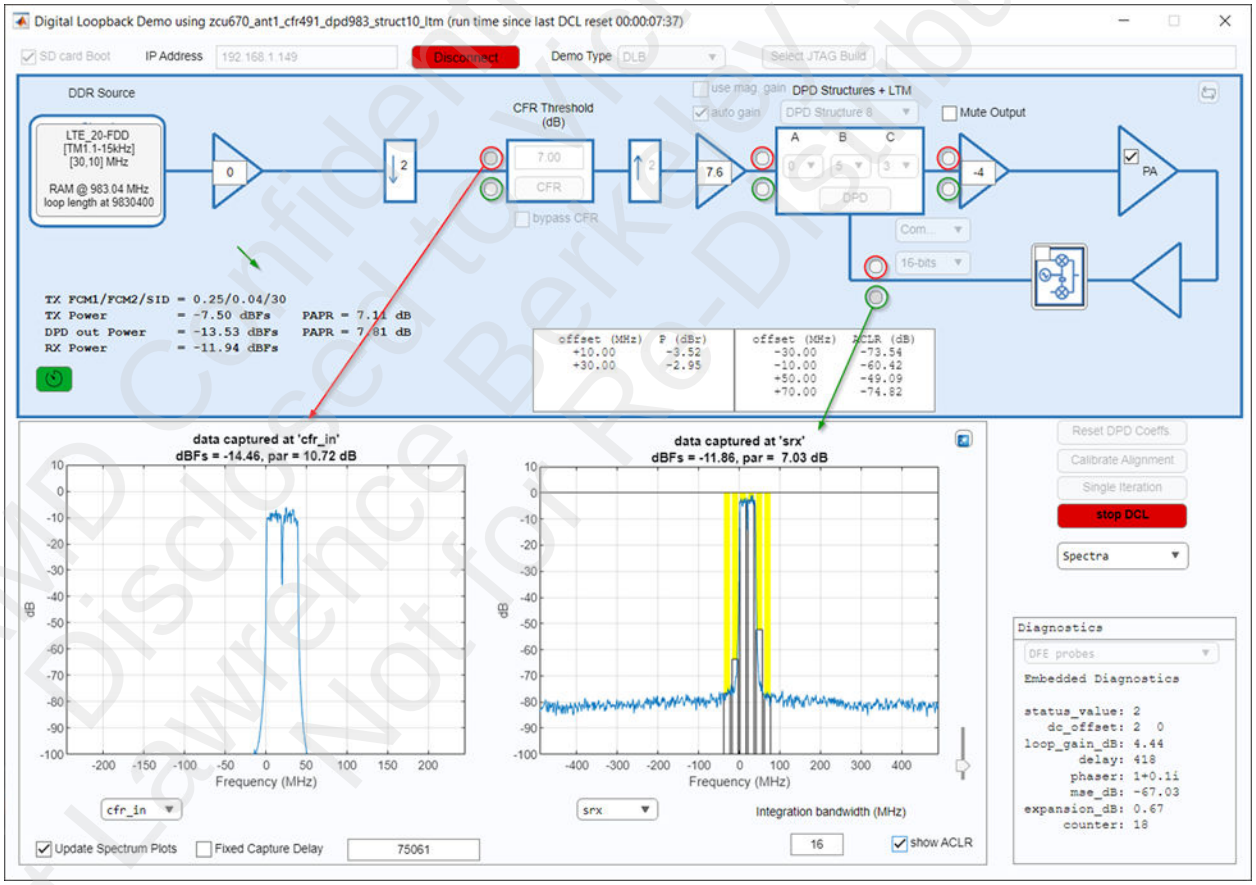
IMPORTANT! *To operate CFR with dynamic signals, see Contacting Support.*

Spectrum Plots

When the Update Spectrum Plots (see the following figure) is enabled the spectrum plots are updated every couple of seconds with the spectrum of selected signal locations using a random sampling delay. To control this delay, select the check box labeled Fixed Capture Delay. This allows a fixed sample delay to be entered into the edit box.

The GUI allows display of the waveform at various points through the signal chain. The capture location is selected using radio buttons located within the block diagram in the GUI. The upper radio button (marked in red in the figure) selects the point in the chain as the source for the left-hand plot, the lower radio button (marked in green) selects the source for the right-hand plot. The plot titles are annotated with the capture location, the signal power in dBFS and Peak to Average ratio of the signals.

Figure 33: **Selecting Spectra Plots in the GUI**



The ACLR of the spectrum in the right-hand axis can be estimated by selecting the **show ACLR** check box. To perform this:

1. Select the green radio button.
2. Edit the integration bandwidth for the signal being transmitted.

3. For most waveforms the carrier frequencies and adjacent channel frequencies are pre-defined. These values can be modified by clicking in the edit box and setting the new desired frequencies. After the new values are entered click outside the edit box to enable the new value.

***Note*:** The spectrum update should be disabled while the ACLR or carrier power boxes are being edited. Re-enable the update when edits are complete.

IMPORTANT! *Continuously capturing spectrum data incurs an overhead due to the large amount of traffic transferred over the JTAG/Ethernet connection. Uncheck the* ***Update Spectrum Plots*** *radio button to stop the plots being automatically updated and free up the MATLAB computing resources. This is useful for custom MATLAB scripting/testing and is particularly important if a custom script accesses the capture spectrum data directly as the GUI might overwrite the intended capture data with a periodic capture.*

Gain Controls

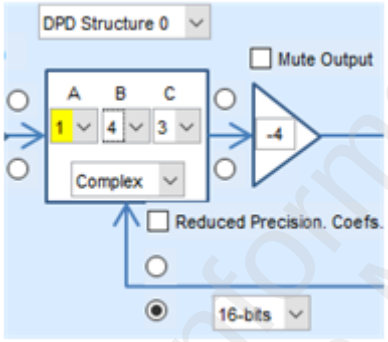
The **Post Gain** box applies the dB value in the edit box to the gain of the signal after DPD. This should be close to 0, to take advantage of the dynamic range of the DAC. This might also be used for small adjustments to demonstrate the recovery of DPD from RF gain variations by applying a step up or down of up to 1 dB.

DPD Parameter Selection

The **DPD Architecture** and **DPD Filter Structure** drop-down menus allow selection of the specified Pre-Distortion Correction Architecture. See the *Digital Pre-Distortion LogiCORE IP Product Guide* (PG076) (registration required) and Appendix A: Equivalent DPD Architectures for Different Filter Memory Depth and Filter Structure Builds for the architectures supported in this version of DPD.

The architecture values for A, B, C influence DPD pre-distortion quality and the speed of convergence. When starting to evaluate with a PA, start with B=1 and C=3 (A is automatically set to a value that gives an equivalent performance of a build using a MEMORY_DEPTH of B). A memoryless model is used when B is set to 0. Higher B values increases the memory handling capabilities of the DPD datapath. If the performance is not acceptable, try increasing B. C=3 is recommended, however if exploring B is exhausted, reducing C to 1 or 2 can be tried, to simplify the complexity of the DPD model. To use the highest order of the model, you can set A=0. You can increase A and/or decrease C, after you see reasonably good ACLR performance, to fine tune the performance and get better convergence. This allows you to experiment with some performance trade-offs.

Figure 34: **DPD Architecture Selection**



DPD Control and Status

The **Calibrate Alignment** (see figure) runs the alignment algorithm and reports various status information in the status reporting region below all the buttons. This should be selected again each time there is a change in the RF line-up. The change might include changing or reconnecting PA, attenuators and cables. If the TX or RX attenuator settings on the RF board is varied, it is advisable to re-calibrate.

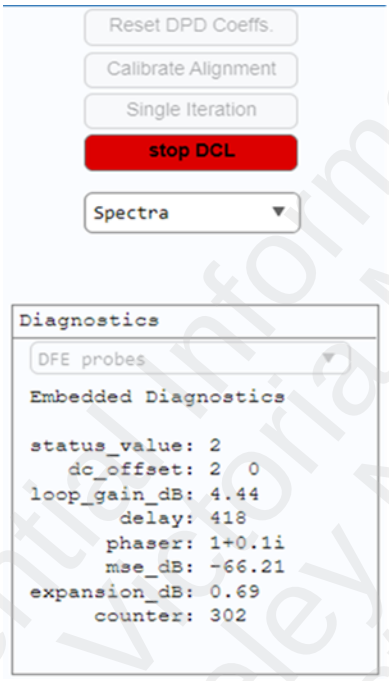
The **Single Iteration** (see figure) runs a single pre-distortion filter update iteration with the returned status and update time displayed in the ECF status line when completed.

The **Reset DPD** resets the coefficients and the DCL state. It does not impact the calibrated alignment state.

Figure 35: **DPD Modes**

The **run DCL** starts the DCL. When running, this changes to red and the label changes to stop DCL as shown in the following figure.

Figure 36: **DCL Button During DCL Mode Run**



While running, various status information is displayed:

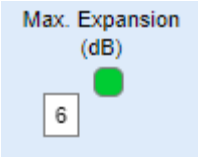
- The counter line displays the **DCL_counter: n** where n is the number of estimations performed.
- Additional status information about the Mean Square Error (MSE) in dB of the RX signal and DPD output signal expansion in dB are also reported.

Glitch Protection

The bit-files for the RF demo include a mechanism to protect against glitches in the transmit signal that can be potentially harmful if applied to a PA. Glitch protection works as follows:

- Define the maximum expansion expected to be applied by DPD to the transmit signal by setting a value (in dB) in the Max. Expansion box (see figure).

Figure 37: **Glitch Protection**



- A glitch power threshold is computed by the demo software based on the maximum power level of the input to DPD and the post-gain setting, and this threshold is supplied to the FPGA.

- The maximum instantaneous signal power is monitored after all the pre-gain, CFR, DPD, and post-gain processing in the FPGA and before the digital signal is output to the RF board. If this power does NOT exceed the glitch power threshold, the **Glitch Reset** stays green as shown in the figure and the transmit signal is passed to the RF board. However, if the power exceeds the glitch threshold:
  - The glitch detector is triggered and the **Glitch Reset** changes color from green to red to indicate the triggered condition.
  - Zeros are sent to the RF board instead of the transmit signal.
  - Monitoring the expansion value reported on the GUI allows you to observe how non-linearities are being pre-distorted and when the glitch detector triggers.
- To recover from the glitch detection triggered condition, click the **Glitch Reset**. This clears the triggered condition and changes the color of the **Glitch Reset** to green. If the triggered condition cannot be cleared because more glitches are detected immediately, the demo software provides the following message in the MATLAB console to indicate possible causes of the persistent condition:

  Glitch detection has not been cleared. Ensure threshold and gains are set appropriately.

  IMPORTANT! *It is important to understand the reasons that might cause triggering of glitch detection, and to ensure that the Max. Expansion box is set to the appropriate value to avoid potential damage to a PA.*
- For example, if an LTE20 signal is used and its RMS power at the DPD output is measured of -12.26 dBFS/ 0 dB gain and PAR is srt to 7.16 dB, and post gain value is set to 0 dB, the peak power at glitch protection logic in the FPGA is (-12.26+7.16+0 =) -6.1 dBFS. The glitch threshold can be set as high as 5.5 dB (keeping 0.6 dB headroom for measurement errors).

Command Line Operation

For a normal demo situation the GUI should allow sufficient control over the demonstration environment. However, all operations are also available from the command line. Contact AMD Technical Support with your requirements or see the "Writing an Advanced Script" section of the *Digital Pre-Distortion Debug Interface User Guide* (UG989) (registration required) if there is a need to create custom testing/demonstration scripts.

MATLAB function `dfe_signal_capture` can be used to capture signals at various capture points in DFE datapath.

MATLAB function `dfe_cfr_sweep_par` can be used to plot PAPR versus RMS Error for CFR IP in the DFE Demo environment. For more information on this script, see the `DFE CFR SWEEP Scripts Overview.pdf` file included with the package under `common_demo/signal_generation/scripting/` subdirectory.

External PA Enable Signal Support

The DFE Radio Demo includes functionality to generate an external signal suitable to drive enabling/disabling of a PA for testing in TDD operation. The generation of the PA enable signal takes place when a new waveform is downloaded for playback using the signal generation GUI. If the functionality is enabled, the new waveform is analyzed for extended periods of zero magnitude samples. If these periods are long enough, they are deemed to correspond to Up-Link (UL) transmission and therefore, suitable for disabling the PA. The PA enable signal is generated according to the detection of off periods. The PA enable signal is outputted through a connector on AMD RF Boards.

Enable the functionality to generate an external PA enable signal calling the MATLAB function:

```
enable_PA_enable_generation(enable, polarity, delay, off_min_len,
off_guard, on_guard, off_mag_threshold)
```

Where:

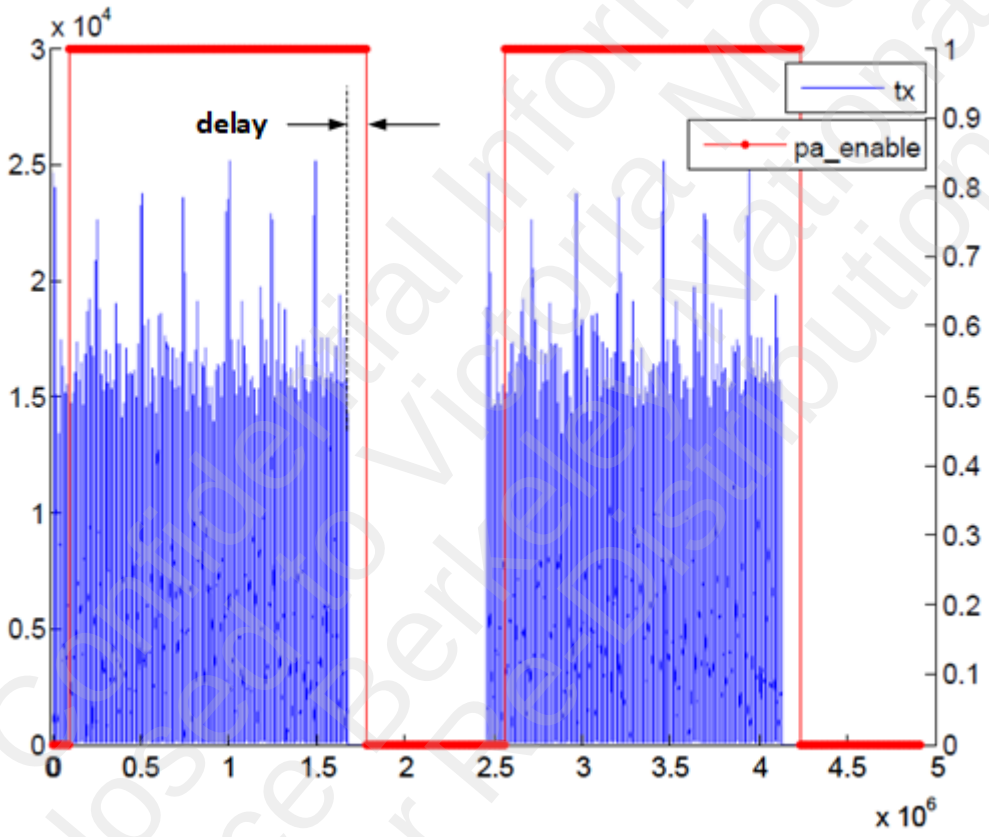
- **enable:** An integer value 1 or 0 to enable or disable the generation of the PA enable signal. Set this to 1 to enable the generation of the PA enable signal.
- **polarity:** An integer value 0 or 1 depending on the desired active-High or active-Low polarity of the PA enable signal respectively.
- **delay:** An integer value representing the desired number of samples to delay the PA enable signal relative to the TX waveform.
- **off_min_len:** An integer value that defines the minimum number of consecutive samples to determine an off period in the transmitted waveform.
- **off_guard:** A positive integer value representing the number of samples to delay the down-link (DL) to up-link (UL) transition turn-off point.
- **on_guard:** A positive integer value representing the number of samples to advance the UL to DL transition turn-on point.
- **off_mag_threshold:** A positive integer value used to determine when the transmit waveform is off. For some applications, the TX UL waveform into DPD might not be truly zero magnitude (for example, when the signal comes from an RF/ADC). This requires changing the default magnitude threshold from 0 to a higher value to accurately determine the off (UL) periods.

The figure shows a plot of the PA enable signal with a delay relative to the TX signal. It is important to note that:

- The `enable_PA_enable_generation` function should be called prior to the downloading of a desired waveform for TDD testing. Once invoked as shown above, the TDD waveform should be selected and downloaded using the signal generation GUI of the DFE Radio Demo. This generates the corresponding PA enable signal and activates its output.

- The delay of the PA enable signal of a TDD waveform can be adjusted dynamically after the desired waveform has been downloaded. To do so, call the `enable_PA_enable_generation` function as above with the new adjusted delay. There is no need to re-download the TDD signal in this case.

Figure 38: **pa_enable Signal with Delay**



To disable the output of the PA enable signal, call the same MATLAB function parameterized as follows:

```
enable_PA_enable_generation(0)
```

Chapter 6

Using the ZCU670 DFE Evaluation Tool with Full DFE Design

On the AMD ZCU670, the full DFE Demo GUI is used to demonstrate the full system including the Downlink chain from baseband signal to the radio front end and the Uplink chain from ADC and RX equalizer to baseband and PRACH.

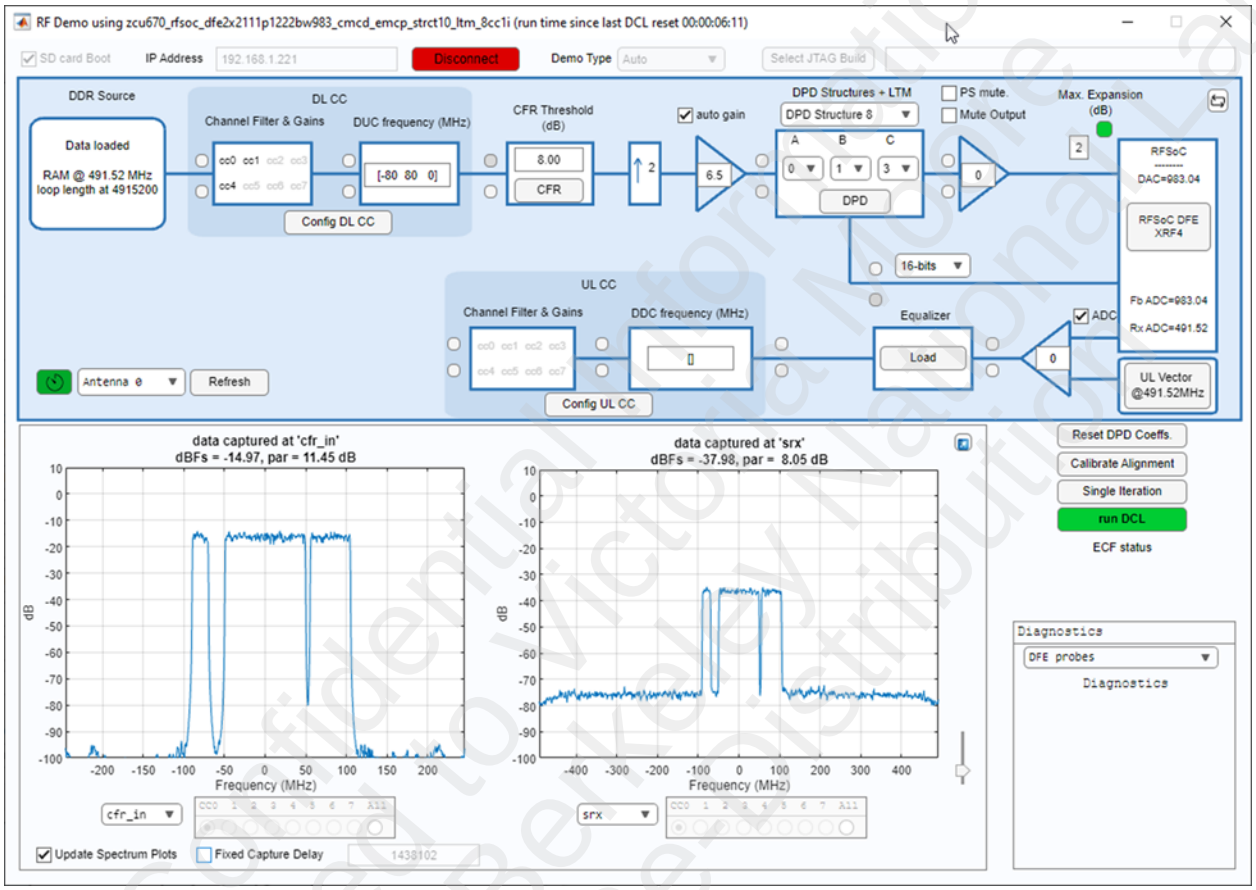
RF Mode with AMD RF Board v4.0 or XM755

In full DFE design, the system includes Downlink TX path, DPD feedback (FB) path, and Uplink RX path. Usually, Quad ADCs (F_s is up to 3 GSPS) are used for RX path and Dual ADCs (F_s is up to 6 GSPS) are used for FB path for DPD adaption only. In some cases, if the FB path and RX path have the same sample rates, select any ADC for FB path.

The procedure to set up the boards, PA, lab equipment, and take measurements is described in the *DFE Demo with ZCU and RF Board User Guide* `DFE_Demo_ZCU_and_RF_Boards_user_guide_vx_y.pdf` (https://www.xilinx.com/member/rfsoc-dfe-eval-platform.html). The process to select and load appropriate designs on the AMD ZCU670 boards is similar to the one described in Digital Loopback Mode. You need to select the preferred full DFE build and switch build on the SD card. After power cycling (OFF > ON) the ZCU evaluation board, the newly selected design starts up.

When the GUI is launched with the bit-file having all DFE functionality enabled (similar to Figure 3), it should look like the following figure.

Figure 39: **Main GUI for Full DFE Demo**



Using the Demo GUI

The actual Demo GUI layout is drawn based upon the functionality available in the bit-file selected for loading. It can include OFDM, Channel Filter, DDC/DUC, and PRACH along with CFR, DPD, and RF Data Converters. This section discusses the overall control and monitoring in the main GUI and reviews the Downlink and Uplink separately.

Controls and Monitor Points in the Main GUI

The Downlink (DL) and the Uplink (UL) chains can be configured independently.

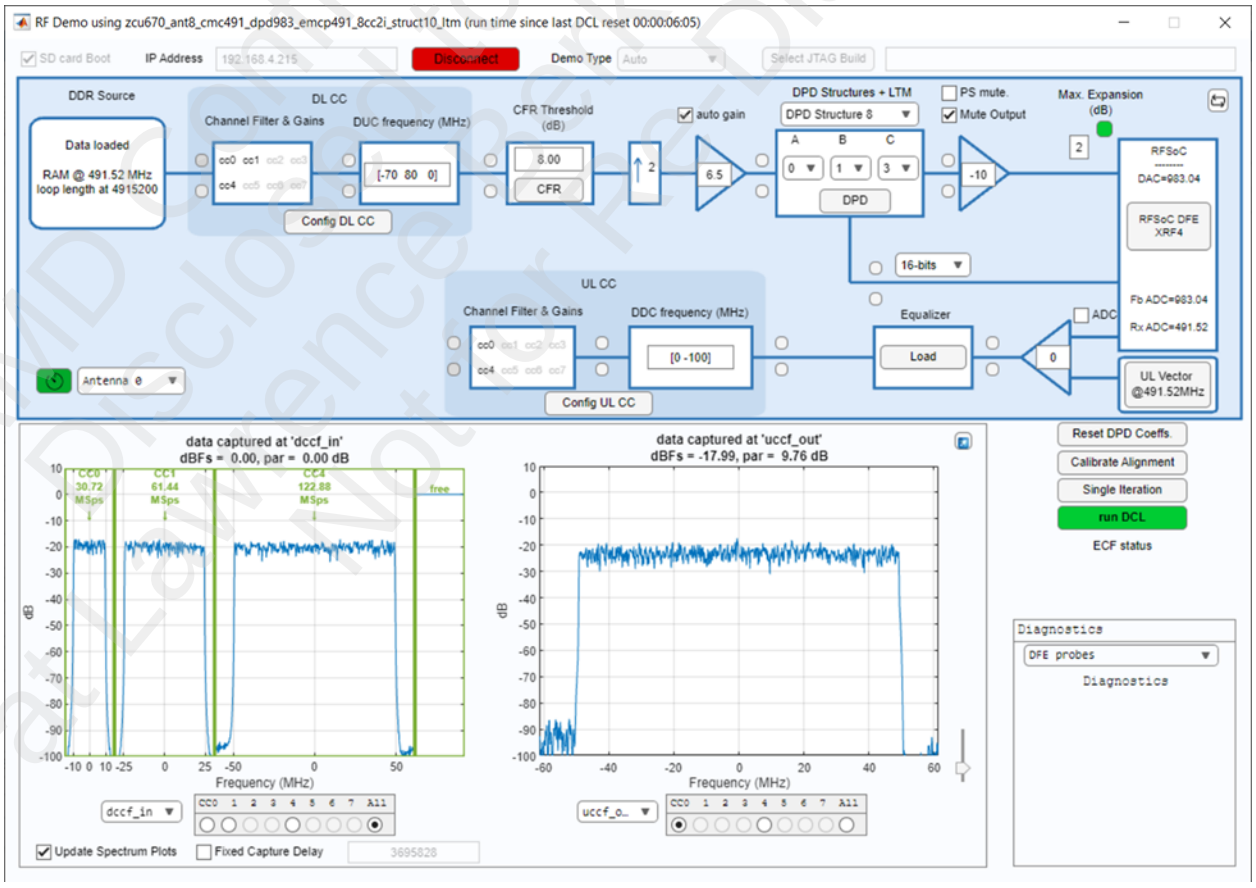
For the Downlink chain, the setup sequence is from baseband to the radio front-end. Click the **Config DL CC** button to load baseband data and configure each Downlink carrier. The default channel filter and CFR CPulse filter are applied. The initial CFR threshold is set and the post-CFR gain block is configured automatically if the auto gain check box is selected. You can further check CFR performance or adjust CFR parameters by clicking the **CFR** button to open the CFR

window. For more information, see the PC-CFR Configuration and Performance. DPD related parameters can also be changed by following "DPD" section in the *Digital Pre-Distortion Debug Interface User Guide* (UG989) (registration required). Clicking the **DPD** button opens the DPD debug interface GUI (see *Digital Pre-Distortion Debug Interface User Guide* (UG989) (registration required)).

For Uplink chain, the setup sequence is from the UL vector loading to PRACH. Load Uplink vector by deselecting the ADC check box and clicking the **UL Vector** button. By default, the UL subsystem is configured based on the UL vector. But you can also configure it manually by opening the **UL CC Subsystem** GUI. Each RX path has its own RX gain (up to 12 dB) and RX equalizer. The default coefficient for RX equalizer is pass-through. You can also load a file with 12 complex filter coefficients. Example `.mat` files (`eqcoeff_tilt.mat` and `eqcoeff_unity.mat`) are provided.

The main GUI also provides monitoring points at inputs and outputs of different blocks. You can select one of the antennas and check two monitoring points at the same time. For Downlink channel filter input, Downlink mixer input, Uplink mixer output, and Uplink channel filter output, the spectrum of each individual carrier or all carriers can be displayed. The following figure shows an example.

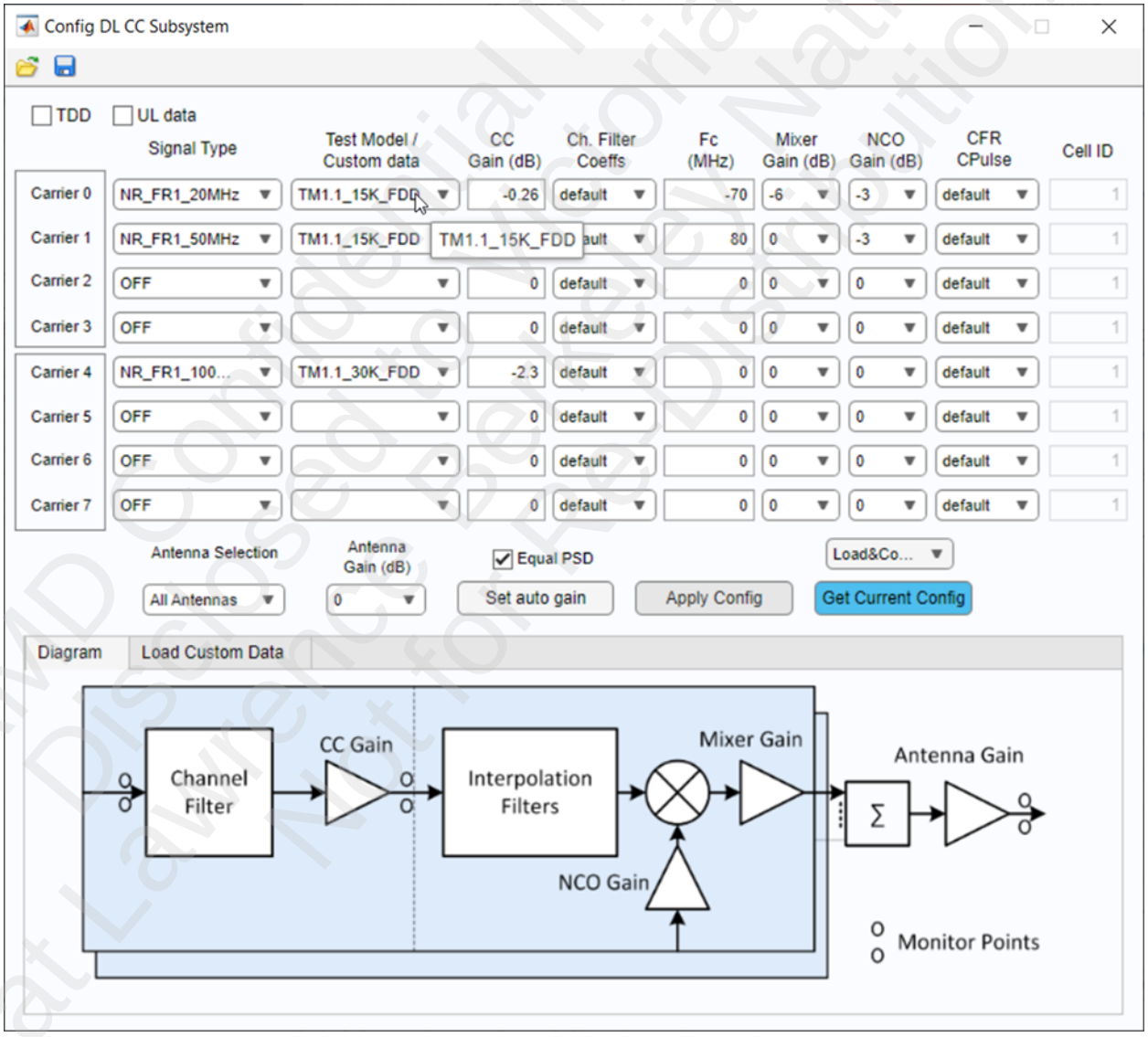
Figure 40: **Monitoring dccf_in and uccf_out for Antenna 0**



Data Loading and Downlink Subsystem Configuration

To load the data and configure the Downlink subsystem, click the **Config DL CC** button in the main GUI and a window displays the Config DL CC Subsystem (see figure). The upper part of the window is the configuration for each carrier. The lower part of the window shows the diagram of the parameters in the datapath. It also includes a second tab to load custom files. The figure shows a diagram to illustrate the location of some parameters.

Figure 41: **Config DL CC Subsystem**



The Downlink related parameters include the following:

1. Data related parameters:
   - **Signal Type:** Supported Signal Type, for example, NR_FR1_100MHz, LTE_20MHz, etc. It can also be custom data.
   - **Test Model/Custom data:** Supported Signal Test Model, TM3.1A_FDD, TM1.1_30K_TDD, etc. If Signal Type selects custom data, each field of custom data structure is listed in Test Model/Custom data.

   Each carrier data in time domain is sampled at 30.72 MHz, 61.44 MHz, or 122.88 MHz. For 8-carrier bit-files, the first or the second set of four carriers cannot exceed half of the maximum bandwidth. For example, if the design is 8-carrier 2-antenna interleaved and with channel filter operating at 491.52 MSPS, each antenna can have 245.76 MHz bandwidth. Carrier 0 to 3 or Carrier 4 to 7 can only occupy 122.88 MHz bandwidth.
2. Channel filter related parameters:
   - **CC Gain (dB):** Channel filter Gain in dB, up to 18 dB.
   - **Ch. Filter Coeffs:** Channel filter coefficients (256 taps), the demo applies default Downlink Channel filter coefficients based on the signal type and test model. If you load custom data, the custom channel filters are also needed.
3. Mixer related parameters:
   - **F_c (MHz):** Center frequency in MHz, the precision is 100 Hz.
   - **Mixer Gain (dB):** Mixer Gain in dB, choice includes 0, -6, -12, -18.
   - **NCO Gain (dB):** NCO Gain in dB, choice includes 0, -3, -6, -9.
   - **Antenna Gain (dB):** 0 or -6 dB. This is applied to all antennas in the design.
4. CFR Cancellation Pulse (CPulse):
   - **CFR CPulse:** The demo GUI applies the default CFR CPulse filter based on the signal type and test model. If you load custom data, the custom CPulse filter is also needed. The maximum Cancellation Pulse length supported is 8191 coefficients (at 4x cfr_fs rate).

The Config DL CC GUI also provides some additional features to help you set DL parameters. These features include the following:

- **Set auto gain Button:** For multiple carrier cases, the button is used to set gain of each carrier to meet the system requirements. The average power of each carrier at channel filter input is -15 dBFs when loading the existing data. The CFR input power is -15 dBFs for proper CFR performance.

  By default, the **Equal PSD** check box is selected, which indicates that all carriers have the same Power Spectrum Density when combined in CFR input. By clicking the **Set auto gain** button, the gains (CC Gain, Mixer Gain, and NCO Gain) of each carrier is set automatically to meet equal PSD.

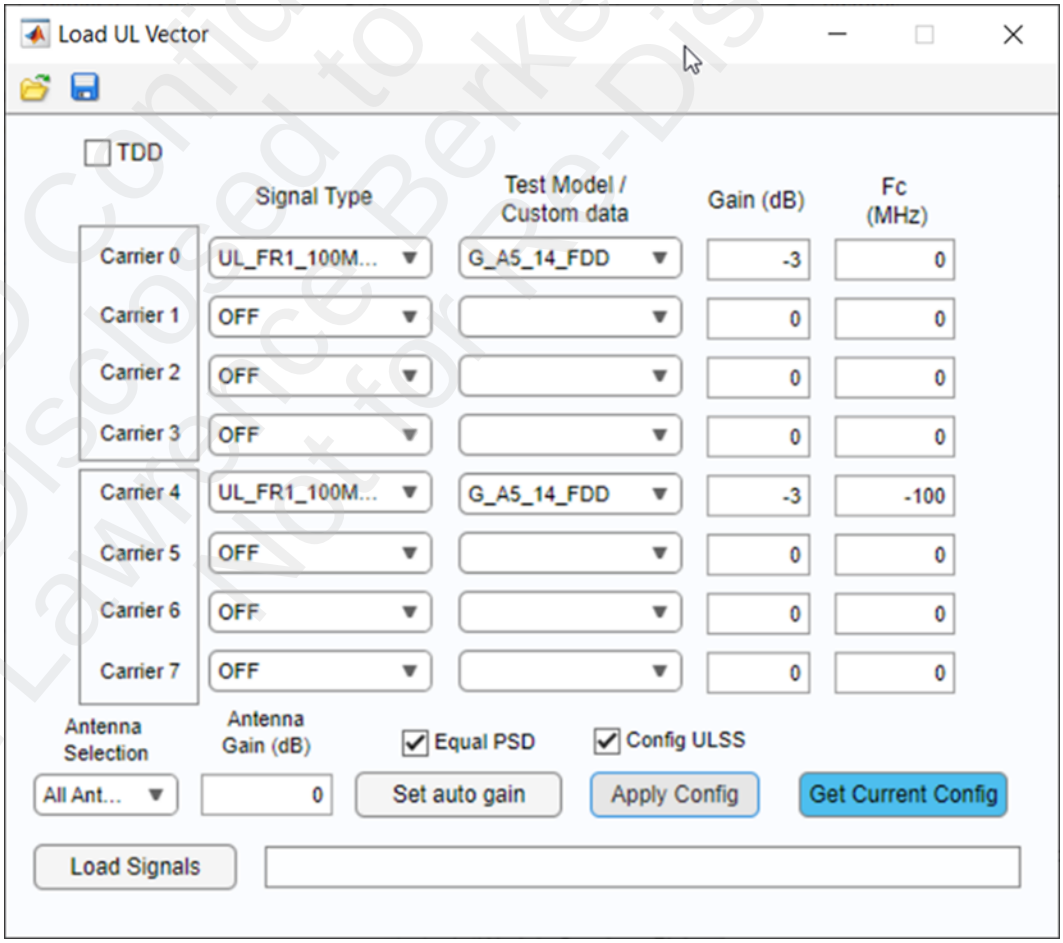
If the **Equal PSD** check box is deselected, enter the relative power for each carrier in the **CC Gain** edit fields. By clicking the **Set auto gain** button, the gains of each carrier is set to meet the relative gain following your settings while still maintaining -15 dBFs power at CFR input.

- **Load/Save Configuration Menu Items:** These two menu items can help load existing DL CC configuration files to the window or save the current configuration (which has been applied to hardware successfully) to files.
- **Load Custom Data Tab:** This tab includes Load signals, Load Channel filter, and Load CFR CPulse. When the files are loaded successfully, the custom options are listed in the drop-down menus. An example of custom data/filters file can be generated by running the script `dfe_create_custom_basebandData_CF_CPulse.m`.

Uplink Vector Loading Window

By default, Uplink RX is connected to the ADC output. By deselecting the **ADC** check box in the main GUI, Uplink RX is connected to the Uplink vector source (from DDR). Click the **UL Vector** button and the Load UL Vector window is displayed (see figure).

Figure 42: **Load UL Vector**



The Uplink related parameters include the following:

- **Signal Type:** Supported Signal Type, for example, NR-FR1-100MHz or LTE_20MHz, etc. It can also be a custom data.
- **Test Model/Custom Data:** Supported Signal Test Model, TM3.1A_FDD, TM1.1_30K_TDD, etc. If Signal Type selects custom data, each field of custom data structure is listed in Test Model/ Custom data.

  By default, Uplink FDD data is listed. When **TDD** check box is checked, TDD data examples are available. When unchecking UL check box, DL data options are available in the drop-down list.

  Similar to DL channel filter configuration limitation, UL channel filter configuration can have restrictions. Each carrier data in time domain is sampled at 30.72 MHz, 61.44 MHz, or 122.88 MHz. For 8-carrier designs, the first or the second set of the four carriers cannot exceed half of the maximum bandwidth. For example, if the build is 8-carrier 2-antenna interleaved, each antenna can have 245.76 MHz bandwidth. Carrier 0 to 3 or Carrier 4 to 7 can only occupy 122.88 MHz bandwidth. This restriction has to be followed when configuring the UL vector to load.
- **Gain (dB):** Gain value applied to each carrier, up to 18 dB.
- **F_c (MHz):** Center frequency in MHz, the precision is 100 Hz.

There are no CFR applied to UL vector data.

The UL Vector GUI also provides some additional features to help you set UL vector parameters. These features include the following:

- **Set auto gain Button:** For multiple carrier case, this button is used to set gain of each carrier to meet the system requirements.

  The average power of each carrier is -15 dBFs when loading the existing data. The target power level of the combined data at UL Vector output is -15 dBFs.

  By default, the **Equal PSD** check box is selected, which indicates that all carriers have the same Power Spectrum Density when combined in UL Vector output. By clicking the **Set auto gain** button, the gain of each carrier is set automatically to meet equal PSD.

  If the **Equal PSD** check box is deselected, you can enter the relative power for each carrier in **Gain** edit fields. By clicking the **Set auto gain** button, the gains of each carrier are set to meet the relative gain following your settings while still maintaining -15 dBFs power at UL Vector output.
- **Config UL SS Check Box:** Configure UL subsystem parameters when the check box is selected.

By default, **Config UL SS** check box is selected. So when applying the UL vector parameters to load UL vector data, the UL subsystem (including DDC, UL Channel filter, and UL OFDM) is configured based on UL vector data.

- **Config PRACH Check Box:** Configure PRACH Subsystem parameters when the check box is selected.

  By default, **Config PRACH** check box is selected. When loading PRACH example vector, the PRACH subsystem is configured.

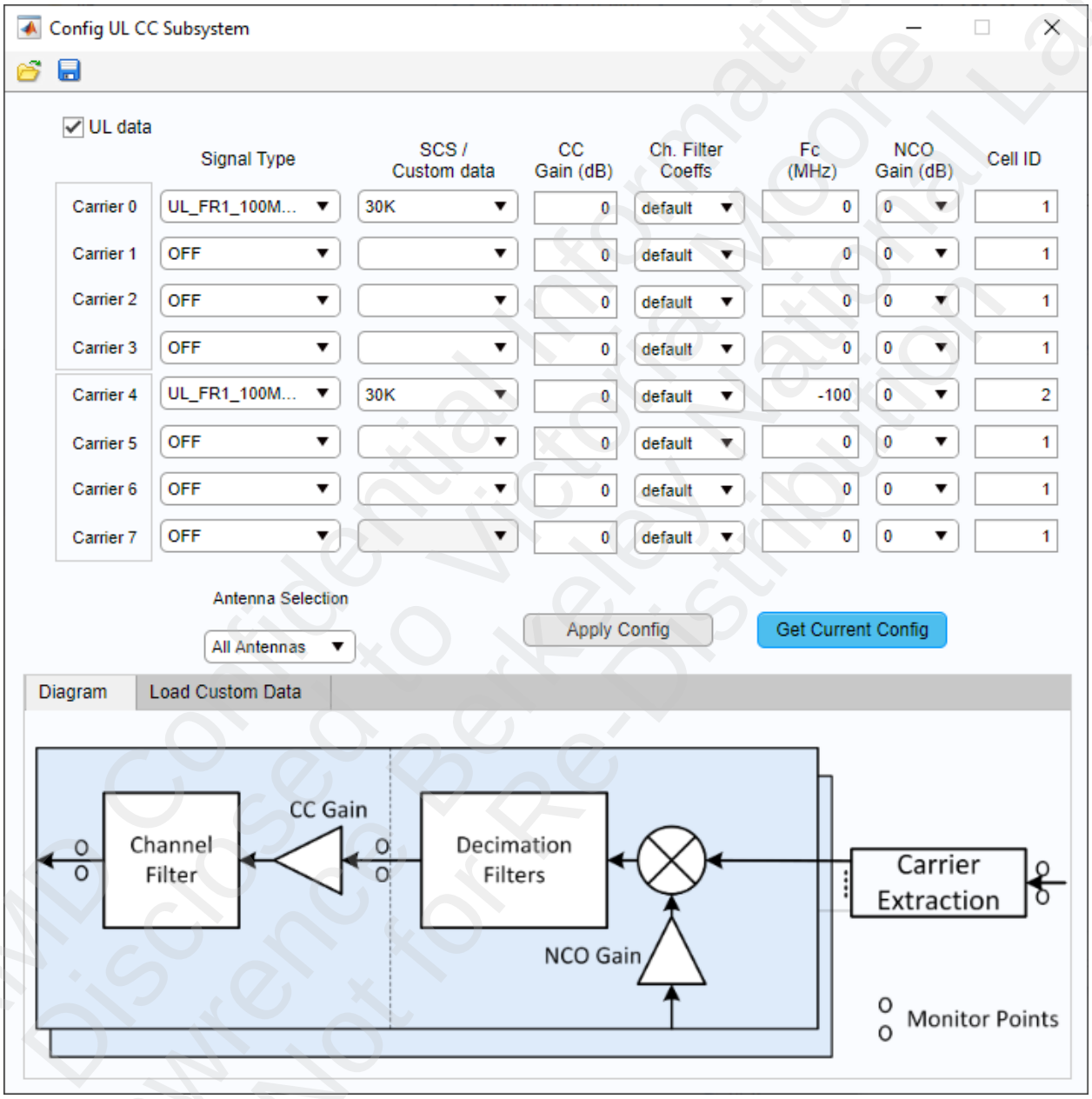
- **Load/Save Configuration Menu Items:** These two menu items can help load existing UL vector configuration files to the window or save the current configuration (which has been applied to hardware successfully) to files.

- **Load Signals:** You can load custom signals when clicking the **Load Signals** button. When the file is loaded successfully, the custom options are listed in the drop-down menus. An example of custom data/filters file can be generated by running the script `dfe_create_custom_basebandData_CF_CPulse.m`.

Uplink Subsystem Configuration

To configure the uplink subsystem, click the **Config UL CC** button in the main GUI and a window displays the Config UL CC Subsystem (see figure). The upper part of the window is the configurations for each carrier. The lower part of the GUI shows the diagram of the parameters in the datapath. It also includes a second Tab to load custom files. The figure shows a diagram to illustrate the location of some parameters.

Figure 43: **Config UL CC Subsystem**



The Uplink related parameters include the following:

1. Data related parameters:
   - **Signal Type:** Supported Signal Type, for example, NR_FR1_100MHz, or LTE_20MHz, etc. It can also be custom data.
   - **SCS/Custom data:** Supported SCS, 15 kHz or 30 kHz, etc. If Signal Type selects custom data, each field of custom data structure is listed in SCS/Custom data.

By default, Uplink LTE or NR FDD data is listed. When the **TDD** check box is checked, TDD data examples are available. When unchecking the **UL** check box, DL data options are available in the drop-down list.

Each carrier data in time domain is sampled at 30.72 MHz, 61.44 MHz, or 122.88 MHz. For 8-carrier design, the first or the second set of four carriers cannot exceed half of the maximum bandwidth. For example, if the build is 8-carrier 2-antenna interleaved, each antenna can have 245.76 MHz bandwidth. Carrier 0 to 3 or Carrier 4 to 7 can only occupy 122.88 MHz bandwidth.

2. Channel filter related parameters:
   - **CC Gain (dB):** Channel filter Gain in dB, up to 18 dB.
   - **Ch. Filter Coeffs:** Channel filter coefficients (256 taps), the demo applies default Uplink Channel filter coefficients based on the signal type and test model. If you load custom data, custom channel filters are also needed.
3. Mixer related parameters:
   - **F_c (MHz):** Center frequency in MHz, the precision is 100 Hz.
   - **NCO Gain (dB):** NCO Gain in dB, choice includes 0, -3, -6, -9.
   - **Cell ID:** Cell ID value is used to calculate the phase rotation of OFDM output data.

The Uplink Configuration window also provides some additional features to help you set the UL parameters. These features include the following:

- **Load/Save configuration menu items:** These two menu items can help load existing UL CC configuration files to the window or save the current configuration (which has been applied to hardware successfully) to files.
- **Load Custom Data Tab:** This tab includes Load signals and Load Channel filter. When the files are loaded successfully, the custom options are listed in the drop-down menus. An example of custom data/filters file can be generated by running the script `dfe_create_custom_basebandData_CF_CPulse.m`.

Chapter 7

Install and Run the Standalone Tools Based on the MATLAB Runtime

In the following package, there are two standalone applications that are based on the MATLAB® Runtime 9.14:

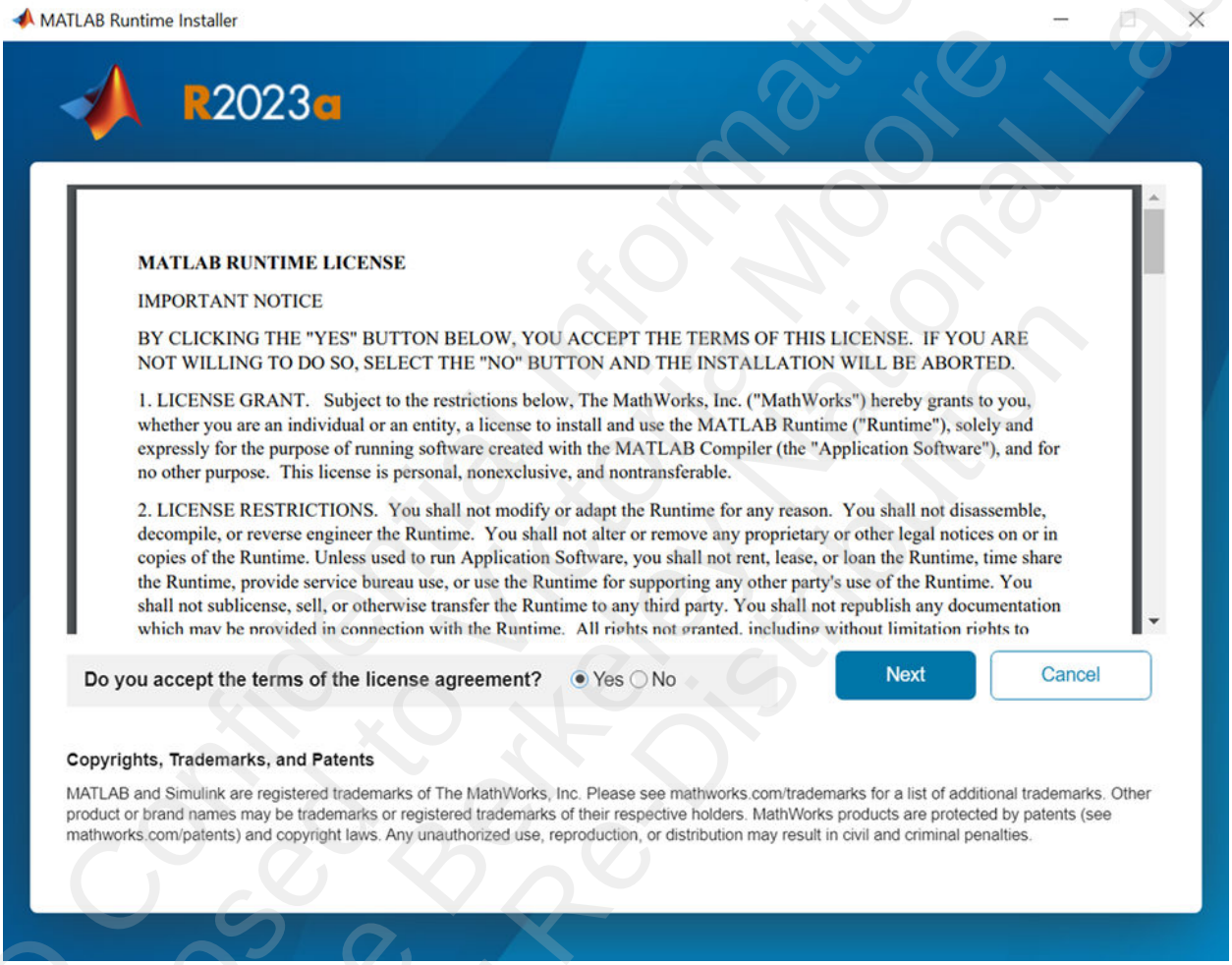
- `DFEEvalToolWizard.exe`
- `EvalToolDemo.exe`

Installing the MATLAB Runtime

See the following web page https://www.mathworks.com/products/compiler/matlab-runtime.html to download the 64-bit Windows version for R2023a (9.14).

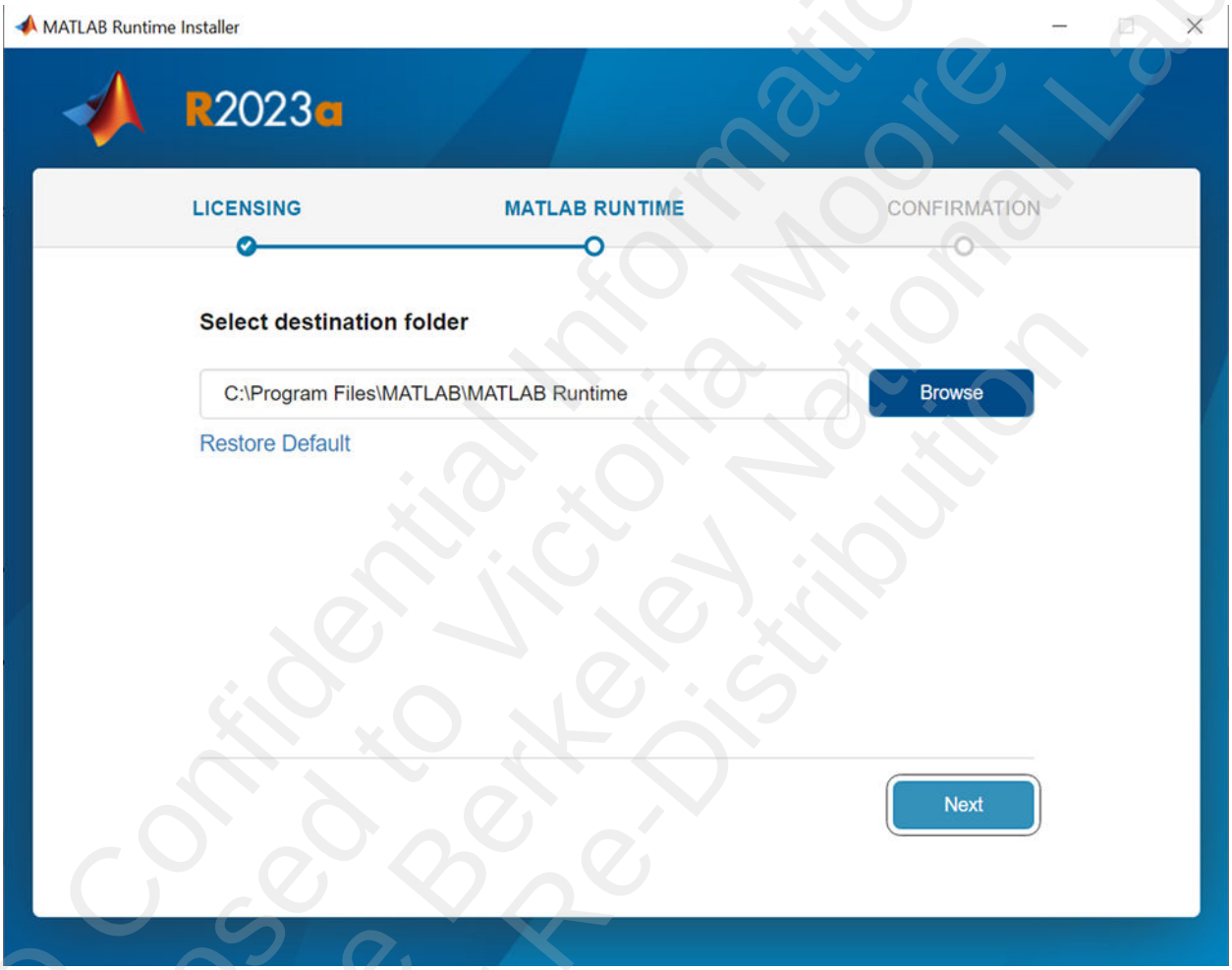
1. Unzip the ZIP file and click the application `setup.exe` to install the MATLAB Runtime.
2. Click **Yes** and **Next**.

Figure 44: **Install MATLAB Runtime, Step 1**



3. Select the desired destination folder and click **Next**.

Figure 45: **Install MATLAB Runtime, Step 2**



4. Confirm the selections and click **Begin Install**.

Figure 46: **Install MATLAB Runtime, Step 3**

Running the Applications

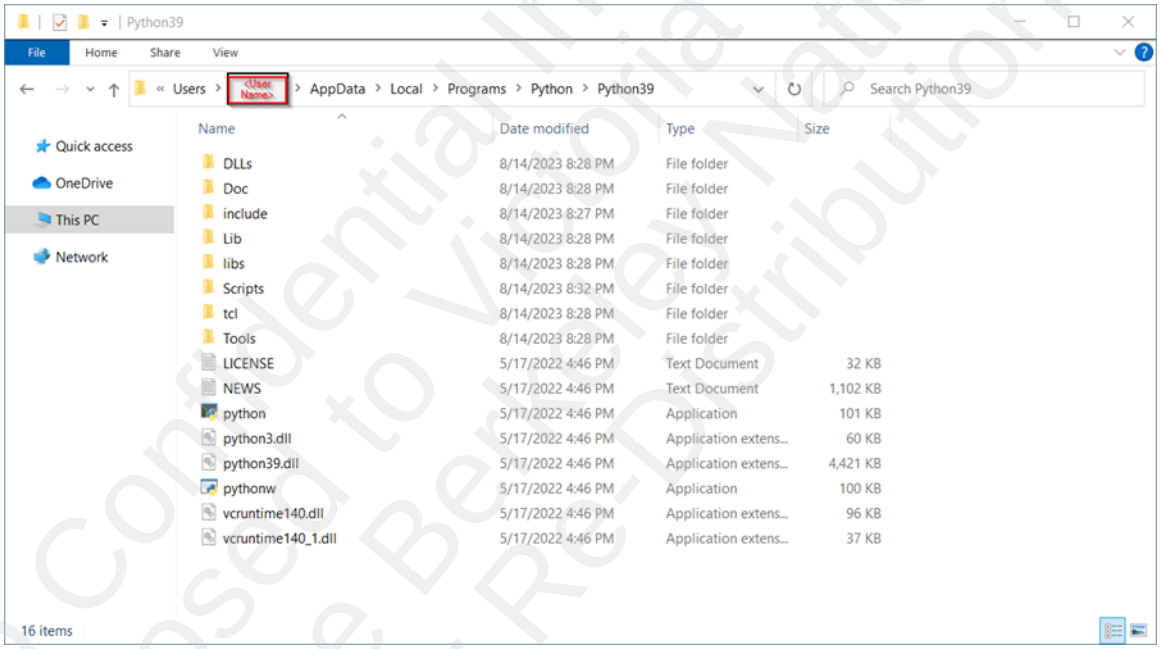
- **DFEEvalToolWizard.exe:** This application is a tool to help you install essential software required by the DFE Evaluation Tool Demo. For more information, see Chapter 4: Using the DFE Evaluation Tool Setup Wizard. The following highlights some of the differences:

  1. `python_path.mat` is used to store the python path used in the MATLAB Runtime environment. At the initial start-up, click the **Set Pyenv** button to receive the python path set properly in the MATLAB Runtime environment. After setting it successfully, `python_path.mat` file is saved for future usage.
  2. All information displayed in the Windows Command Shell is saved in the `DFEEvalToolWizard_log.txt` file.

- **EvalToolDemo.exe:** This application is a tool to run the DFE Evaluation Tool Demo. For more information, see Chapter 5: Using the DFE Evaluation Tool for CFR + DPD Only Designs and Chapter 6: Using the ZCU670 DFE Evaluation Tool with Full DFE Design. The following highlights some of the differences:

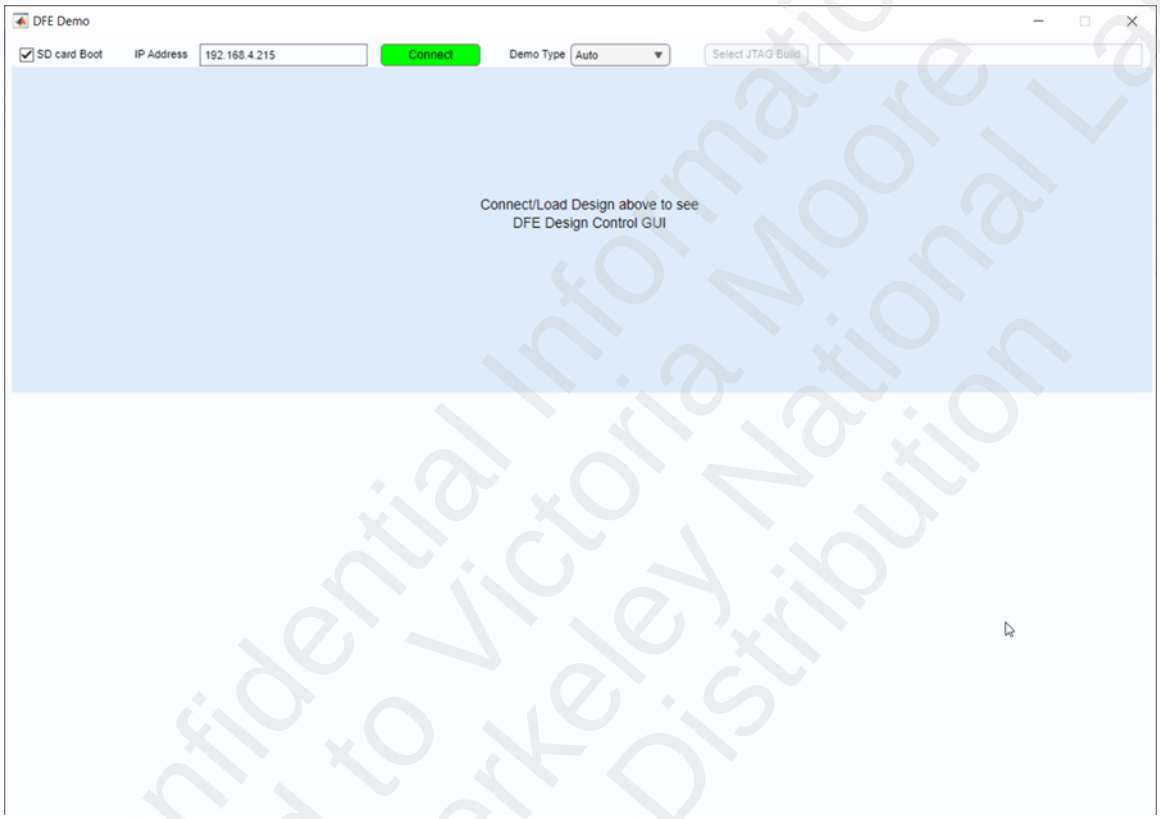
  1. `python_path.mat` file is used to store the python path used in the MATLAB Runtime environment. Without this file, running the EvalToolDemo results to the following screen capture which asks you to provide the proper Python `.exe` file path. After setting it successfully, `python_path.mat` file is saved for future usage.

Figure 47: **Example of python.exe File Path**



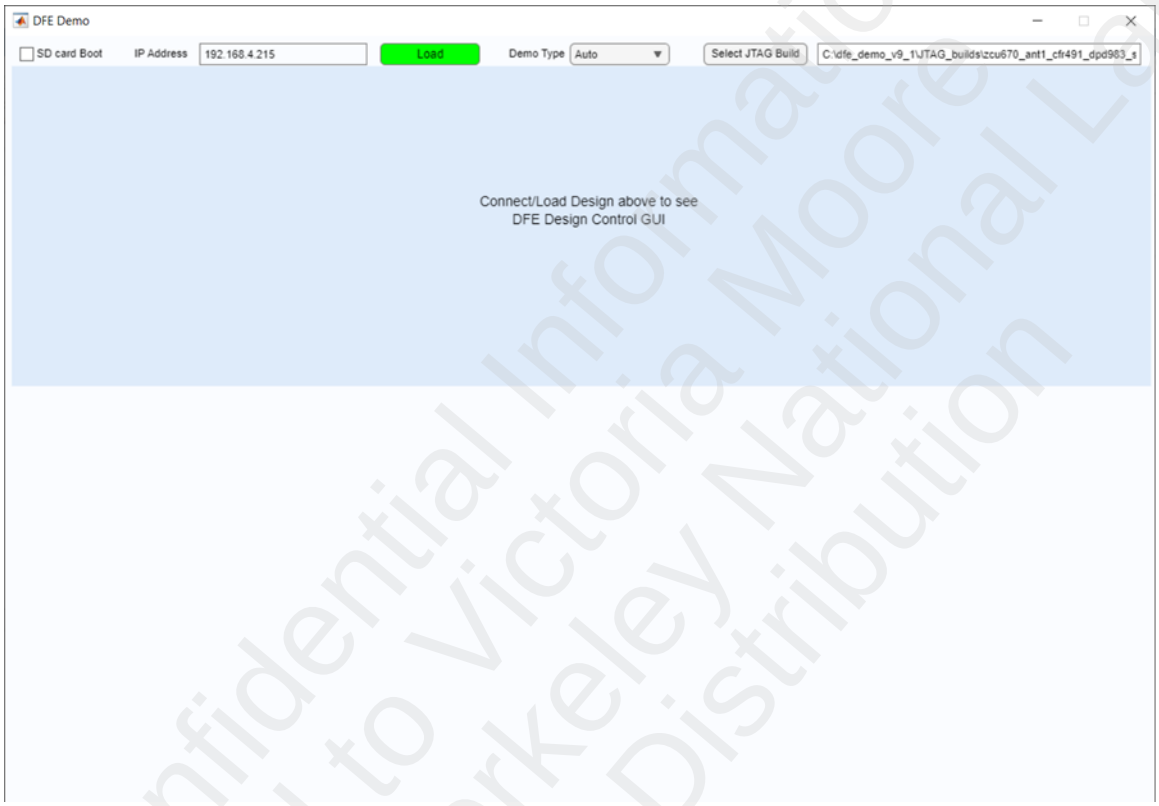
  2. After the DFE Demo starts, enter the IP address of the board and click the **Connect** button to communicate to the running builds booted by the SD card.

Figure 48: **DFE Demo Start Screen with SD Card Boot Mode**



3. Or point to the path of the JTAG builds and click the **Load** button to load builds using JTAG to bring up the Demo GUI.

Figure 49: **DFE Demo Start Screen with JTAG Load Mode**



4. All information displayed in the Windows Command Shell is saved in the `EvalToolDemo_log.txt` file.

Appendix A

Equivalent DPD Architectures for Different Filter Memory Depth and Filter Structure Builds

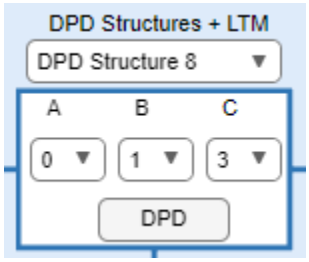
Most of the supplied DFE demo bit-files have large Filter Memory Depths (typically 4, 5, or 6) and support a super-set of the hardware required to evaluate all filter structures on that device. These large filter builds facilitate the evaluation of selected Filter Structures and all smaller filters with a single bit-file using the ARCH_SEL parameter. The DFE demo also uses a RESERVED parameter field in the ARCH_PARAMETER space to control the emulation of the Filter Structure. RFSoC DFE based demo designs are supplied with Filter Structure 10 hardware support. Filter Structure 10 hardware can be configured to emulate Filter Structure 8, 9, and 11 hardware.

Note: For Structure 10, it is designed to support optimized Macro PA performance (only supported for RFSoC DFE), usually B can go up to six for Memory Depths with and without Long Term Memory (LTM).

The amount of FPGA hardware resources allocated for DPD coefficients increases with the size of the Filter Memory Depth that is selected at build time. Initial evaluation of DPD performance is often performed using a build with a larger Filter Memory Depth. Smaller DPD architectures can be emulated with appropriate selection of the `0xABC` architecture parameter. If a smaller Filter Memory Depth is determined to achieve acceptable DPD performance, the smaller Filter Memory Depth can be used at build time to reduce the FPGA cost of the final DPD solution for non-RFSoC DFE solutions.

When switching between different Filter Memory Depths, the 'A' value might need to be adjusted to get the exact same architecture. In the DFE demo, the yellow 'A' control has a tool-tip that indicates how to adjust the 'A' parameter for the currently selected 'B' and 'C' parameters.

Figure 50: **'A' Control for DPD Architectural Equivalence**



Appendix B

Debugging

This appendix shows you the steps to perform to clear status errors that might occur while using the demo.

Hardware Communication (HWC) Debug

Initialization of the DFE Demonstration uses JTAG connection through Xilinx System Debugger (XSDB) hardware server to download a bitstream and software to the board. Once software is running, the DFE Demonstration relies on communications from MATLAB® to a TCP Server running on the board for board configuration, control, and vector download. This differs from use of the standalone DPD Debug Interface (*Digital Pre-Distortion Debug Interface User Guide* (UG989) (registration required)) which only communicates to DPD IP Core through the JTAG Interface and does not need services provided by the TCP Server.

The TCP Server allows MATLAB to call C-code drivers that execute on the AMD Zynq™ PS. These drivers include access of AXI peripheral in the demo, vector download and communications on I2C and SPI Interfaces. To build the connection between TCP Server (running in PS) and TCP Client (MATLAB), use the DHDP server (like a Physical Router Device embedded with DHCP server or a soft DHCP server running at PC) or use the static IP between MATLAB and ZCU board (as long as they are in same subnet). For detailed information on DHCP server or static IP settings, see the *DFE Demo with ZCU and RF Board User Guide* `DFE_Demo_ZCU_and_RF_Boards_user_guide_vx_y.pdf` (https://www.xilinx.com/member/rfsoc-dfe-eval-platform.html).

General Error Messages is the output to the MATLAB Command Window. The general class of error is indicated by prefix to the error messages as shown in the following table. Exclamation marks prefixing message indicate warnings and asterisks indicate errors in the log window.

Table 3: **Hardware Communication Messages**

Log Prefix	Description
HWC:	General Hardware Communications Message
HWC: (UBOOT):	Message concerning UBoot Configuration
HWC: (XSDB):	XSDB Hardware Communication Message
HWC: (TCP):	TCP Server Hardware Communications Message

Table 3: **Hardware Communication Messages** *(cont'd)*

Log Prefix	Description
ERROR:jtag_init:	JTAG Initialization Error
ERROR:dpd_socket_mex	Communications Error in MEX Function used for TCP Comms

If issues occur in initialization or communication with the Demonstration it is also advisable to review more detailed messages that might appear within the DPD Debug Server Window that is created by the MATLAB demo which logs messages from XSDB. This window shows details of the XSDB initialization of the board.

It is also advised to open a terminal console on the UART Connection attached to the Zynq PS. The UART Connection uses the following settings (115200 baud, 8 data bits, no parity, one stop bit). Further details can be found in the relevant user guide for the AMD Evaluation Board in question. The UART console contains messages from First Stage Boot Loader (FSBL), U-Boot, and Linux Log Messages.

DPD Core Debugging

The table contains suggested steps to take if DPD status errors occur during the operation of the DFE Radio Demo.

Table 4: **Debugging Suggestions**

Issue	Suggestion
Pre-distortion correction less than expected or bad	Perform the *Essential Checks* using the Advanced Debug Interface. Explore different architecture. A smaller B value can also help. Check that all connectors are tightened with a torque wrench and that cables are not damaged. Switch between different feedback receiver connections and see if there is any difference in performance.
Error Reported	Check the *Digital Pre-Distortion Debug Interface User Guide* (UG989) (registration required) and the *Digital Pre-Distortion LogiCORE IP Product Guide* (PG076) (registration required) for information about the error code. If unknown error code, contact AMD Support.
Raised noise floor after pre-distortion	It is most likely that the feedback receiver power is too low. Make the appropriate adjustments using the RX RF gain.
Spectrum gradually degrades after achieving best performance	Check that the TX LO and I/Q imbalance has been removed using the DAC settings on the RF board, if using XRF2. For RFSoC boards (ZCU208/670), check that the RF filtering at the TX path removes DAC Nyquist images and RF filtering before Feedback RX ADC (does not have HD2 signals). Also check if different DPD settings allow better stability.

Contacting Support

It is highly recommended that when contacting technical support you:

- Take GUI screen captures that help demonstrate the issue.
- Log the MATLAB command window to show the commands/steps taken that caused your issue.

AMD

Appendix C

Additional Resources and Legal Notices

Support Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Support.

Finding Additional Documentation

Documentation Portal

The AMD Adaptive Computing Documentation Portal is an online tool that provides robust search and navigation for documentation using your web browser. To access the Documentation Portal, go to https://docs.xilinx.com.

Documentation Navigator

Documentation Navigator (DocNav) is an installed tool that provides access to AMD Adaptive Computing documents, videos, and support resources, which you can filter and search to find information. To open DocNav:

- From the AMD Vivado™ IDE, select **Help → Documentation and Tutorials**.
- On Windows, click the **Start** button and select **Xilinx Design Tools → DocNav**.
- At the Linux command prompt, enter `docnav`.

***Note*:** For more information on DocNav, refer to the *Documentation Navigator User Guide* (UG968).

Design Hubs

AMD Design Hubs provide links to documentation organized by design tasks and other topics, which you can use to learn key concepts and address frequently asked questions. To access the Design Hubs:

- In DocNav, click the **Design Hubs View** tab.
- Go to the Design Hubs web page.

References

These documents provide supplemental material useful with this guide:

1. *ZCU670 Evaluation Board User Guide* (UG1532)
2. *Digital Pre-Distortion LogiCORE IP Product Guide* (PG076) (registration required)
3. *Digital Pre-Distortion Debug Interface User Guide* (UG989) (registration required)
4. *Peak Cancellation Crest Factor Reduction LogiCORE IP Product Guide* (PG097) (registration required)
5. *RFSoC DFE Channel Filter LogiCORE IP Product Guide* (PG395) (registration required)
6. *RFSoC DFE DUC-DDC LogiCORE IP Product Guide* (PG393) (registration required)
7. *RFSoC DFE Resampler LogiCORE IP Product Guide* (PG392) (registration required)
8. *RFSoC DFE PRACH LogiCORE IP Product Guide* (PG391) (registration required)
9. *RFSoC DFE Fast Fourier Transform LogiCORE IP Product Guide* (PG390)
10. *Peak Cancellation Crest Factor Reduction in a Multi-Standard Transmit System* (XAPP1174) (registration required)
11. *ZCU111 Evaluation Board User Guide* (UG1271)
12. *Xilinx RF Board v2.0 (XRF2) User Guide* (https://www.xilinx.com/member/dfe_demo.html#rfboards)
13. *DFE Demo with ZCU and RF Board User Guide* `DFE_Demo_ZCU_and_RF_Boards_user_guide_vx_y.pdf` (https://www.xilinx.com/member/rfsoc-dfe-eval-platform.html)
14. *Xilinx RF Board v4.0 (XRF4) User Guide* (https://www.xilinx.com/member/dfe_demo.html#rfboards)
15. *ZCU208 Evaluation Board User Guide* (UG1410)
16. *Zynq UltraScale+ RFSoC ZCU208 Evaluation Kit Product Brief* (https://www.xilinx.com/publications/product-briefs/xilinx-zcu208-product-brief.pdf)
17. *Zynq UltraScale+ RFSoC ZCU111 Evaluation Kit Product Brief* (https://www.xilinx.com/publications/product-briefs/zcu111-product-brief.pdf)
18. *Zynq RFSoC DFE ZCU670 Evaluation Kit Product Brief* (https://www.xilinx.com/publications/product-briefs/xilinx-zcu670-product-brief.pdf)
19. *XRF4 RF Accessory Kit Data Sheet* (DS1012)

20. *XRF4 RF Accessory Kit User Guide* (UG1614)

Revision History

The following table shows the revision history for this document.

Section	Revision Summary
10/18/2023 Version 9.1	
Whole document	• Updated to use CFR v8.0 and DPD v14.1 IPs. • CFR + DPD only demo has been migrated to appdesigner with a common look and feel of the full DFE demo.
Chapter 1: Introduction	Added MATLAB Runtime description.
Hardware Requirements	Added ZCU and RF Board Selection for DFE Demo table.
Software Requirements	Updated MATLAB description.
AMD RF Board v3.0 Documentation	Updated file name.
ZIP File Description	Updated section.
Chapter 4: Using the DFE Evaluation Tool Setup Wizard	Updated figure.
Switch Build on SD Card	Updated section.
Digital Loopback Mode	Updated figures.
RF Mode for CFR + DPD Only Designs	Updated document list.
DFE Radio Demo Parameters	Added demo support for the new DPD magnitude gain feature.
Using the Demo GUI	Updated figure.
Digital Loopback Mode	Removed Launch GUI instruction.
Spectrum Plots	Updated figure.
DPD Control and Status	Updated figure.
Glitch Protection	Updated figure.
RF Mode with AMD RF Board v4.0 or XM755	Updated figure.
Controls and Monitor Points in the Main GUI	Updated figure.
Data Loading and Downlink Subsystem Configuration	Updated figure.
Uplink Vector Loading Window	Updated figure.
Chapter 7: Install and Run the Standalone Tools Based on the MATLAB Runtime	Added chapter.

Section	Revision Summary
05/10/2023 Version 9.0	
	• Updated core versions throughout document. • Updated description and added DFE Evaluation Tool Setup Wizard in Introduction. • Removed SmartLynq and updated description in Hardware Requirements. • Added SD card boot mode in Software Requirements. • Added Switch Build on SD Card. • Updated Connecting the ZCU Board and Host PC. • Added Using the DFE Evaluation Tool Setup Wizard. • Updated description in RF Mode with AMD RF Board v4.0 or XM755.
10/21/2022 Version 8.1	
	• Updated core versions throughout document. • Added XRF4 support. • Updated description in Chapter 1, Introduction. • Added AMD DFE IP Product Documentation and AMD RF Board v4.0 Documentation. • Updated Figure 2-2 and Figure 2-3. • Added AMD RF Board Version 4.0 in Overview. • Updated GUI description for new enhanced features of CFR, DPD, and RF programming. • Updated Connecting the ZCU Board and Host PC. • Updated Connecting the ZCU670 with XRF3 or XRF4 Boards. • Updated title for Chapter 5, Using the DFE Evaluation Tool for CFR + DPD Only Designs. • Updated #4 and #10 in Digital Loopback Mode. • Updated RF Mode for CFR + DPD Only Designs. • Updated Figure 5-6, Figure 5-7, and Figure 5-9. • Added Chapter 6, Using the ZCU670 DFE Evaluation Tool with Full DFE Design. • Updated to structure 8, 9, and 11 in Appendix A, Equivalent DPD Architectures for Different Filter Memory Depth and Filter Structure Builds. • Updated Hardware Communication (HWC) Debug. • Updated Spectrum description in Table B-2.
04/20/2022 Version 8.0	
	• Updated core versions throughout document. • Added note in the beginning of Digital Loopback Mode and Using the Demo GUI. • Added dfe_signal_capture script description in Command Line Operation.

Section	Revision Summary
12/15/2021 Version 7.0	
	• Added support for RFSoC DFE chip and ZCU670 board. • Updated all GUI screen captures to match DPD v12.0 features. • Updated Table 3-1. • Updated descriptions in PC-CFR Configuration and Performance. • Added description for Filter Structure 10 in Equivalent DPD Architectures for Different Filter Memory Depth and Filter Structure Builds.
04/30/2021 Version 6.0	
	• Requirement changed to Vivado 2020.2 tools. • Added support for testing with GaN Amplifiers. • Updated all GUI screen captures to match DPD v11.0 features.
08/21/2020 Version 5.1	
	• Requirement changed to Vivado 2020.1 tools. • Added Connecting the ZCU208 and XRF3 Boards section. • Updated Figures 4-1 and 4-2.
03/16/2020 Version 5.0	
	• Requirement changed to Vivado 2019.2 tools. • Demo updated to DPD v10.0 and CFR v6.3. • Gray out CFR config button until a signal has been loaded. • Add mute_all / un_mute_all command to multi-antenna DLB demo GUI. • Update signal name wrapping in GUI. • Fix XRF3 configuration loading. • Fix small D.C. bias caused when enable_PA_enable_generation is enabled. • Update default DFE demo METERLENGTH setting to value suggested by PG076. • All demos only support SMP software configuration. • Bit-File Configurations section updated. • Updated Figures 4-1, 4-3, 4-4, 4-5, 4-9, 4-17, A-1. • Appendix A updated.
03/20/2019 Version 4.0	
	• Requirement changed to Vivado 2018.3 tools. • Demo updated to DPD v9.0 and CFR v6.2.
04/21/2017 Version 3.1	
	• Requirement changed to Vivado 2016.4 tools. • Demo updated to DPD v8.1.
12/09/2016 Version 3.0	
	• Requirement changed to Vivado 2016.3 tools. • Demo updated to DPD v8.0 and PC CFR v6.1.

Section	Revision Summary
06/30/2016 Version 3.0	
	EA: Xilinx Confidential Draft. • Requirement changed to Vivado 2016.2 tools. • Demo updated to DPD v8.0EA1 and PC CFR v6.0 rev5. • Support restricted to Xilinx RF board v2.0 (Tektelic RF board support removed).
12/18/2015 Version 2.0	
	• Added support for 2015.4 Vivado tools. • Demo updated to support DPD v7.1 rev1 and PC-CFR v6.0 rev3. • Updated support for Xilinx RF board v2.0 front end.
06/12/2015 Version 1.1	
	• Updated demos to use PC-CFR v6.0. • Updated Fig. 2-1 and 2-2. • Added Bit-File Configurations. • Updated Table 3-1: Zip File Descriptions. • Added note in Xilinx DFE Radio Demo Software Installation section. • Updated GUIs in Using the DFE Radio Demo chapter. • Updated Fig. 4-14: TEKTELIC RF Board Modulator and Transmit Attenuation • Updated in PC-CFR Configuration and Performance section.
02/11/2015 Version 1.0	
Initial release.	N/A

Please Read: Important Legal Notices

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product releases, product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. Any computer system has risks of security vulnerabilities that cannot be completely prevented or mitigated. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes. THIS INFORMATION IS PROVIDED "AS IS." AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS, OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION. AMD SPECIFICALLY

DISCLAIMS ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY RELIANCE, DIRECT, INDIRECT, SPECIAL, OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

AUTOMOTIVE APPLICATIONS DISCLAIMER

AUTOMOTIVE PRODUCTS (IDENTIFIED AS "XA" IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE ("SAFETY APPLICATION") UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD ("SAFETY DESIGN"). CUSTOMER SHALL, PRIOR TO USING OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.

Copyright