OVERVIEW

The BytePipe_x9002 is a highly integrated RF System on Module (SOM). It integrates a RF transceiver, baseband processor (BBP), and all the necessary peripherals onto a single module. This application note describes how to use the BytePipe x9002’s internal Digital Pre-Distortion (DPD) function to linearize Richardson RFPD’s RadioCarbon RFFE.
# TABLE OF CONTENTS

- OVERVIEW .................................................................................................................. 0
- TABLE OF FIGURES ....................................................................................................... 2
- THEORY OF OPERATION ................................................................................................. 3
  - RF TRANSCEIVER ......................................................................................................... 3
  - RadioCarbon ................................................................................................................ 4
- Test Setup and Operation .............................................................................................. 5
  - Required Equipment .................................................................................................... 5
  - Recommended Test Setup ........................................................................................... 5
- Software Setup ................................................................................................................ 6
- Common Commands ....................................................................................................... 9
- RFLAN v9.16.22 Known Bugs ...................................................................................... 11
- Appendix A: tx_dpd_disabled.txt .................................................................................... 12
- Appendix B: tx_dpd_enabled.txt ..................................................................................... 13
# TABLE OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ADRV9002 Block Diagram</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>High Level RFPD-RC-4450-50 System Diagram</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>Test Setup Diagram</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>CLI Boot Output</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>CLI DPD Disabled</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>Spectrum Analyzer DPD Disabled</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>CLI DPD Enabled</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>Spectrum Analyzer DPD Enabled</td>
<td>8</td>
</tr>
</tbody>
</table>
THEORY OF OPERATION

The BytePipe_x9002 is a highly integrated RF System on Module (SOM). It integrates a RF transceiver, baseband processor (BBP), and all the necessary peripherals onto a single module.

RF TRANSCEIVER

The BytePipe_x9002 uses the Analog Devices Inc.’s ADRV9002 Agile Transceiver™. The ADRV9002 is a highly integrated RF transceiver with two transmitters, two receivers, two integrated synthesizers, and an integrated DPD engine. It operates from 30MHz to 6000MHz and supports both narrow band and wideband bandwidths from 12.5kHz to 40MHz. This application note will describe the operation of the ADRV9002 integrated DPD and provide instructions on how this can be demonstrated on a RadioCarbon RF Front End.

![Figure 1: ADRV9002 Block Diagram](image)
RadioCarbon

For this application note, we will use Richardson RFPD's RadioCarbon RFPD-RC-4450-50, which has been designed for easy integration with the BytePipe SOM, however other Power Amplifiers may be linearized by using the BytePipe Hardware Development Kit (HDK) and including the user's own peripherals such as a feedback coupler and PA bias controls.

![RadioCarbon Diagram](image)

Figure 2: High Level RFPD-RC-4450-50 System Diagram
Test Setup and Operation

The following describes the equipment and accessories required for this test setup. This will demonstrate the DPD capabilities of the ADRV9002 by showing a 10MHz wide signal with DPD disabled vs. enabled.

Required Equipment

- RichardsonRFPD RadioCarbon™ RFPD-RC-4450-50
- TTL to USB Serial Converter Cable
- RadioCarbon GUI Software (included on USB drive with the RadioCarbon board)
- NextGen RF Design BytePipe™ SoM
- SMPM Barrel Adapters (min qty 2)
- SoM heatsink and fan assembly
- Micro SD Card (1GB - 32GB)
- DPD Demo Software Files (see download link in later section)
- Micro-B USB Cable
- TeraTerm Software
- Power Attenuator (5W, 20dB or greater recommended)
- SMA cable and RF Adapters, as necessary
- Spectrum Analyzer (with adequate dynamic range and ability to measure accurate ACP)
- DC Power Supply capable of at least 3A @ 45V
- Additional Heatsink and/or Fan

Recommended Test Setup

At a minimum, a PC running a terminal emulator supporting serial connections is required to configure the BytePipe SOM. Ensure the BytePipe SOM is installed on the RadioCarbon board using at least two SMPM barrel adapters – one for Tx2 and one for Rx2A as indicated on the SOM. SMPM barrel adapters can be used in the other four locations as well, but are not specifically required for this demo.

Note: Ensure adequate heatsinking and/or fans are used to prevent the SOM and RFFE from overheating.

Figure 3: Test Setup Diagram
Software Setup

RadioCarbon GUI

The RadioCarbon RFFE must be set up using a USB to TTL/UART cable to set bias level configuration and enable the PA prior to running the DPD demo. Documentation for the RFFE can be found at the RichardsonRFPD shop website: https://shop.richardsonrfpd.com/Products/Product/RFPD-RC-4450-50#


BytePipe CLI

The command line interface (CLI) provides an external user interface accessible through a serial port or Ethernet connection. Both interfaces are operational simultaneously allowing users to use one or both interfaces. The serial port is accessible through the USB interface using a micro-B USB cable connected to the RadioCarbon USB connector. For this demo it is recommended to use the USB interface and a Tera Term serial terminal operating at 115200 baud as the COM port will enumerate even with the RadioCarbon board unpowered. This allows for the boot sequence to be shown immediately when the board is powered. If using the Ethernet connection, the board must be powered in order for the TCP/IP connection to be established. If this is desired, the rflan_setup.txt file on the SD card will need to be modified to remove the line disabling the IP interface. To do so, open the file in a text editor such as Notepad++ and delete or comment out the line ‘RflanSetParam LwipEnable 0’. More information is available at: https://github.com/NextGenRF-Design-Inc/bytepipe_sdk/blob/main/src/rflan/README.md#cli-interface

BytePipe SD Card

The SD card files required to run this DPD demo can be downloaded from the NextGen RF Design GitHub. To accurately replicate the demo detailed in this document, be sure to use the branch specific to this demo which can be found at https://github.com/NextGenRF-Design-Inc/bytepipe_sdk/tree/dpd_branch/src/rflan_dpd.sd_card and use the version matching your specific SOM architecture (either rflan_dpd_xcru3cg_scard.zip or rflan_dpd_xcru2cg_scard.zip). This DPD-specific build includes an IQ stream file and two script files, tx_dpd_enabled.txt and tx_dpd_disabled.txt, which can be called easily from the CLI using the ExecuteScript <filename> command.

The micro-SD card must be formatted as FAT or FAT32, so a 1GB to 32GB card is recommended. Ensure the .zip file is extracted to the root folder of the SD card. The card should contain the following files, at minimum:

- adv9001_setup.txt
- BOOT.BIN
- DPD_Demo.csv
- rflan_setup.txt
- tx_dpd_disabled.txt
- tx_dpd_enabled.txt
BytePipe Operation

Begin the BytePipe DPD Demo by inserting the micro-SD card in the RFFE micro-SD card slot, opening a TeraTerm window and connecting to the CLI, then apply power to the RFFE. If configured properly, the CLI should show a brief boot sequence output from the SOM.

As noted in the boot sequence, the `help` command will give a list of available commands and the appropriate context for use.

Once booted, execute the DPD disabled script by entering `ExecuteScript tx_dpd_disabled.txt`. This output can be observed on the spectrum analyzer at a center frequency of 4.80GHz, as shown below. Note the adjacent channel power caused by non-linearities in the gain stages.

Figure 4: CLI Boot Output

Figure 5: CLI DPD Disabled

Figure 6: Spectrum Analyzer DPD Disabled
The DPD enabled script can be executed by entering `ExecuteScript tx_dpd_enabled.txt`. Note: The DPD disabled trace is stored in the spectrum analyzer (blue trace) screenshot for comparison.

The Tx output can be stopped by entering `RflanStreamStop Tx2`.

![Figure 7: CLI DPD Enabled](image1)

![Figure 8: Spectrum Analyzer DPD Enabled](image2)
Common Commands

The demo scripts are optimized for use with Richardson RFPD's RadioCarbon RFPD-RC-4450-50 and default to a center frequency of 4.80GHz. These scripts can be modified with a text editor such as Notepad++, or commands can be manually entered in order to make adjustments. The commands used to change the most common parameters are detailed below.

Tx Attenuation

This parameter can be adjusted from 0 to 40dB in 0.001dB increments using the command `Adrv9001SetParam Tx2Attn <value>`. The value should be entered by multiplying the desired attenuation in dB by 1000. For example:

- Desired attenuation of 0dB: `Adrv9001SetParam Tx2Attn 0`
- Desired attenuation of 0.5dB: `Adrv9001SetParam Tx2Attn 500`
- Desired attenuation of 3.25dB: `Adrv9001SetParam Tx2Attn 3250`
- Desired attenuation of 17.125dB: `Adrv9001SetParam Tx2Attn 17125`

Center Frequency

To adjust the center frequency, the ADRV must be in the RfCalibrated state. Enter this state by using `Adrv9001ToRfCalibrated Tx2`. Change the center frequency by entering `Adrv9001SetParam Tx2CarrierFrequency <value>` where value is the desired frequency in Hz. Then return the ADRV to the RfPrimed state by entering `Adrv9001ToRfPrimed Tx2`. For example, to set the center frequency to 4.65GHz, enter the following commands prior to starting the stream:

`Adrv9001ToRfCalibrated Tx2`
`Adrv9001SetParam Tx2CarrierFrequency 4650000000`
`Adrv9001ToRfPrimed Tx2`

**Note:** These state commands can only be used if the ADRV is in SPI mode. If it is in PIN mode, the state is controlled by a digital IO pin. To set it to SPI mode, use the command `Adrv9001SetParam Tx2EnableMode 0`. 
TDD Operation

The DPD profile is set for TDD operation, and the DPD coefficient calculations may require iterations in order to minimize adjacent channel power. However, in following this application note you are only beginning the transmission one time, so the coefficients are only calculated once. It is recommended to manually start and stop the transmit waveform for best results. This can be done by simply restarting the stream multiple times with some delay. The stream should also be restarted if the Tx attenuation is changed while transmitting. For example, to adjust the Tx attenuation and restart the stream, include the following commands in the script file:

```
Adrv9001SetParam Tx2Attn 3250
RflanStreamStart Tx2 1 38400
DelayMs 1000
RflanStreamStart Tx2 1 38400
DelayMs 1000
RflanStreamStart Tx2 1 38400
```

**Note:** If entering these commands manually, the ‘DelayMs’ command is not required. Just allow some time between each entry.

Expected Results

The following results are typical measured values on an RFPD-RC-4450-50. The objective is to measure the maximum output power achievable while meeting a -50 dBm adjacent channel (ACP) requirement. The results are as follows:

<table>
<thead>
<tr>
<th>Freq</th>
<th>TX Attenu</th>
<th>Power (dBm)</th>
<th>IMD upper (dBm)</th>
<th>IMD lower (dBm)</th>
<th>Power (dBm)</th>
<th>IMD upper (dBm)</th>
<th>IMD lower (dBm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4400</td>
<td>0</td>
<td>35.89</td>
<td>-31.06</td>
<td>-31.42</td>
<td>35.96</td>
<td>-55.42</td>
<td>-55.85</td>
</tr>
<tr>
<td>4500</td>
<td>2</td>
<td>36.71</td>
<td>-30.20</td>
<td>-30.34</td>
<td>36.81</td>
<td>-53.67</td>
<td>-52.56</td>
</tr>
<tr>
<td>4600</td>
<td>5.5</td>
<td>35.91</td>
<td>-30.39</td>
<td>-30.41</td>
<td>36.02</td>
<td>-52.19</td>
<td>-52.19</td>
</tr>
<tr>
<td>4700</td>
<td>6</td>
<td>36.30</td>
<td>-31.62</td>
<td>-31.38</td>
<td>36.46</td>
<td>-52.12</td>
<td>-51.73</td>
</tr>
<tr>
<td>4800</td>
<td>6</td>
<td>36.11</td>
<td>-33.25</td>
<td>-33.14</td>
<td>36.30</td>
<td>-52.33</td>
<td>-53.15</td>
</tr>
<tr>
<td>4900</td>
<td>4.5</td>
<td>35.39</td>
<td>-34.54</td>
<td>-34.27</td>
<td>35.59</td>
<td>-51.72</td>
<td>-51.92</td>
</tr>
<tr>
<td>5000</td>
<td>2.25</td>
<td>34.52</td>
<td>-34.81</td>
<td>-34.48</td>
<td>34.83</td>
<td>-51.67</td>
<td>-51.25</td>
</tr>
</tbody>
</table>
RFLAN v9.16.22 Known Bugs

When executing the command `RflanStreamStop Tx2`, the device incorrectly responds with ‘Stream Start Success’. The response should be ‘Stream Stop Success’, however the stream has been stopped.
Appendix A: tx_dpd_disabled.txt

// This script enables the transmitter with DPD disabled

// Set Center Frequency
Adrv9001ToRfCalibrated Tx2
Adrv9001SetParam Tx2CarrierFrequency 4800000000
Adrv9001ToRfPrimed Tx2

// Set Attenuation to 6dB
Adrv9001SetParam Tx2Attn 6000

// Enable Tx Boost
Adrv9001SetParam Tx2Boost 1

// Disable DPD
Adrv9001SetParam Tx2DpdEnable 0

// Set SSI enable delay
Adrv9001SetParam Tx2SsiEnableDly 500

// Set SSI disable delay
Adrv9001SetParam Tx2DisableDly 200

// Load DMA buffer with local file
RflanStreamBufLoad Tx2 DPD_Demo.csv

// Disable Stream
RflanStreamStop Tx2

// Enable Stream
RflanStreamStart Tx2 1 38400

// END OF SCRIPT
Appendix B: tx_dpd_enabled.txt

// This script enables the transmitter with DPD enabled

//Set Center Frequency
Adrv9001ToRfCalibrated Tx2
Adrv9001SetParam Tx2CarrierFrequency 4800000000
Adrv9001ToRfPrimed Tx2

// Set Attenuation to 6dB
Adrv9001SetParam Tx2Attn 6000

// Enable Tx Boost
Adrv9001SetParam Tx2Boost 1

// Set SSI enable delay
Adrv9001SetParam Tx2SsiEnableDly 500

// Set SSI disable delay
Adrv9001SetParam Tx2DisableDly 200

// Enable DPD
Adrv9001SetParam Tx2DpdEnable 1

// Set External Path Delay in picoseconds
Adrv9001SetParam Tx2ExternalPathDelay 3800

// Set External Loopback power to -18dBm
Adrv9001SetParam Tx2ExternalLoopbackPower -180

// Set number of samples to calculate DPD coefficients
Adrv9001SetParam Tx2DpdNumberOfSamples 4096

// Set Normalization Lower threshold to -11dBm
Adrv9001SetParam Tx2DpdRxTxNormalizationLowerThreshold 85290344

// Set Normalization upper threshold to -6dBm
Adrv9001SetParam Tx2DpdRxTxNormalizationUpperThreshold 269711751

// Set Detection Power Threshold to -16dBm
Adrv9001SetParam Tx2DpdDetectionPowerThreshold 53942350

// Set Detection Peak Threshold to -10dBm
Adrv9001SetParam Tx2DpdDetectionPeakThreshold 214748364

// Load DMA buffer with local file
RflanStreamBufLoad Tx2 DPD_Demo.csv

// Disable Stream
RflanStreamStop Tx2

// Enable stream allowing for coefficients to be calculated
RflanStreamStart Tx2 1 38400

// Delay 100ms
DelayMs 100

// Stop Stream
RflanStreamStop Tx2

// Enable stream after coefficients have been calculated
RflanStreamStart Tx2 1 38400

// Restart stream with delays to allow PA and calculations to stabilize
DelayMs 1000
RflanStreamStart Tx2 1 38400
DelayMs 1000
RflanStreamStart Tx2 1 38400
DelayMs 1000
RflanStreamStart Tx2 1 38400

// END OF SCRIPT