# AD9963-EBZ
# Quick Start Guide

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106, U.S.A. • Tel: 781.329.4700 • Fax: 781.461.3113 • www.analog.com

## Getting Started with the AD9963-EBZ Evaluation Board

### WHAT'S IN THE BOX

**AD9963-EBZ Evaluation Board**
**Evaluation Board CD**
**Mini-USB Cable**

### RECOMMENDED EQUIPMENT

**Sinusoidal Clock Source**
**Sinusoidal Signal Source (for ADC tests)**
**Spectrum Analyzer**
**DPGI/O Pattern Generator/Receiver (AD-DPGIOZ)**
**DPG2 Pattern Generator (HSC-DAC-DPG-BZ)** (optional)

### INTRODUCTION

The AD9963-EBZ is designed to work with either a DPG2 or DPGI/O for Transmit into its dual DACs, and with a DPGI/O for Receive from its dual ADCs. The board allows for quick evaluation of nearly all of the AD9963's capabilities. To evaluate both the receive and transmit functionality, an DPGI/O is required. To evaluate just the transmit capabilities, either a DPG2 or DPGI/O can be used.

### SOFTWARE

The DAC Software Suite, plus the AD9963 update, is required to drive data into the AD9963. The DAC Software Suite is included on the Evaluation Board CD, or can be downloaded from the DPG website at http://www.analog.com/dpg. This will install DPGDownloader (for loading vectors into the DPG2 and DPGI/O) and the AD9963 SPI application. In addition, Visual Analog, available at http://www.analog.com/visualanalog, is required for ADC data capture and post-processing. Note that VisualAnalog should be installed before the DAC Software Suite if the receive capability of the AD9963 is to be evaluated.

### HARDWARE SETUP

*Note that the PC software (described above) must be installed before connecting the USB cables to the computer.*

Connect a +5Vdc power supply to J9 SMA. The USB cable should be connected to P6. The main clock source (AC or DC coupled) should be connected to J13 SMA Jack (MAIN CLOCK INPUT).

For transmit evaluation, the DAC outputs are available on J8 (from IDAC) and J7 (from QDAC), for connecting to an oscilloscope or spectrum analyzer.

For Rx, the ADC analog inputs should be fed into J3 (for IADC) and J1 (for QADC). The DPGI/O should be connected to the AD9963 evaluation board for data capture.

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 www.analog.com
Fax: 781.461.3113
©2010 Analog Devices, Inc. All rights reserved.

### *Jumpers*

The setups described in this Quick-Start guide use the factory-default jumper settings. Ensure that each jumper matches these settings:

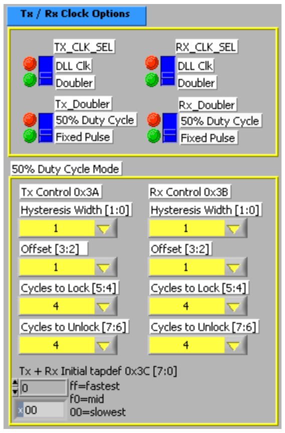
JP1 – Shunt between 1 and 2
JP2 – Shunt between 1 and 2
JP3 – Shunt between 1 and 2
JP4 – Shunt between 1 and 2
JP5 – Shunt between 1 and 2
JP6 – Shunt between 1 and 2
JP8 – Shunt between 1 and 2
JP16 - Installed
P1 – Installed
P2 – Installed
P8 – Installed
P9 – Installed
P11 – Installed
P12 – Installed
P14 – Installed

These settings enable the on-board regulators, AD9963 internal LDO regulators, and setup for 3.3V logic.

## GETTING STARTED

Four different setups are explained here, each utilizing different sections of the AD9963 and the AD9963-EBZ Evaluation Board. All four setups require a low-jitter input clock. Additional details on using a low frequency reference clock along with the internal DLL multiplier are also provided as needed. For purposes of all the set ups described here, the clock source should be set for 50MHz at 6dBm.

Open the AD9963 SPI application from the Start menu (Start > Programs > Analog Devices > AD9963-EBZ > AD9963 SPI). Click on the *DLL/Clock Doubler* tab. Change *TX_Doubler* and *Rx_Doubler* to the *50% Duty Cycle* position. Change both *Offset [3:2]* items to 1. See Figure 1. Click the Run button (⇨) to apply the changes to the part.



Tx Control 0x3A
Hysteresis Width [1:0]: 1
Offset [3:2]: 1
Cycles to Lock [5:4]: 4
Cycles to Unlock [7:6]: 4

Rx Control 0x3B
Hysteresis Width [1:0]: 1
Offset [3:2]: 1
Cycles to Lock [5:4]: 4
Cycles to Unlock [7:6]: 4

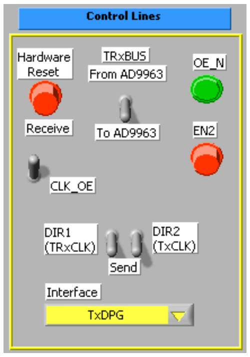
Tx + Rx Initial tapdef 0x3C [7:0]: 0, x00
ff=fastest
f0=mid
00=slowest

***Figure 1***

### *Transmit via Txport using DPG2*

In this setup, the DPG2 is used to drive data into the Txport on the AD9963-EBZ. TxDATA and TxIQ marker signals are fed into the Txport from the DPG2.

In the AD9963 SPI application, switch to the *LDO/Transceiver* tab. Under the *Control Lines* heading, set the *Interface* to TxDPG. See Figure 2.



Interface: TxDPG

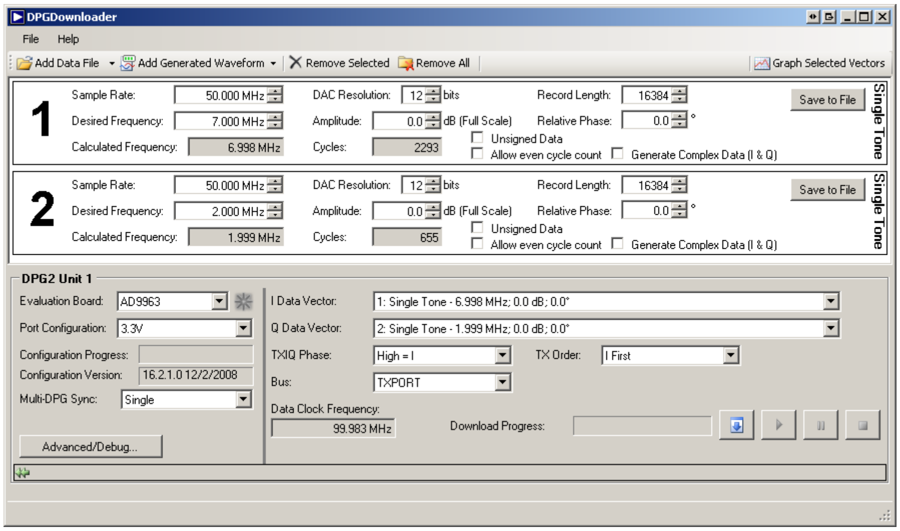
***Figure 2***

Click the Run button (⇨), and wait for the run to complete.

Open the DPGDownloader application (Start > Programs > Analog Devices > DPG > DPGDownloader). With the AD9963 Evaluation Board connected to USB, DPGDownloader will load a unique panel to correctly format the data for the AD9963. Ensure that "AD9963" is selected in the *Evaluation Board* drop-down. Then select the IO Voltage from the *Port Configuration* list (select 3.3V in this case).

Next, click on “Add Generated Waveform” along the top toolbar, and click “Single Tone”. Enter 50MHz for the clock frequency, 7MHz for the desired frequency, and select 12 bits for the DAC resolution. Uncheck the “Unsigned Data” box. Repeat this process to add a Single Tone with a desired frequency of 2MHz.

In the lower half of the screen, select “1:Single Tone” for the *I Data Vector*, and “2: Single Tone” for the *Q Data Vector*. The Data Clock Frequency readout should indicate approximately 100MHz, representative of the interleaved TXCLK. DPGDownloader should now appear as shown in Figure 3.



1 — Single Tone

Sample Rate: 50.000 MHz
Desired Frequency: 7.000 MHz
Calculated Frequency: 6.998 MHz
DAC Resolution: 12 bits
Amplitude: 0.0 dB (Full Scale)
Cycles: 2293
Record Length: 16384
Relative Phase: 0.0 °
□ Unsigned Data
□ Allow even cycle count
□ Generate Complex Data (I & Q)
Save to File

2 — Single Tone

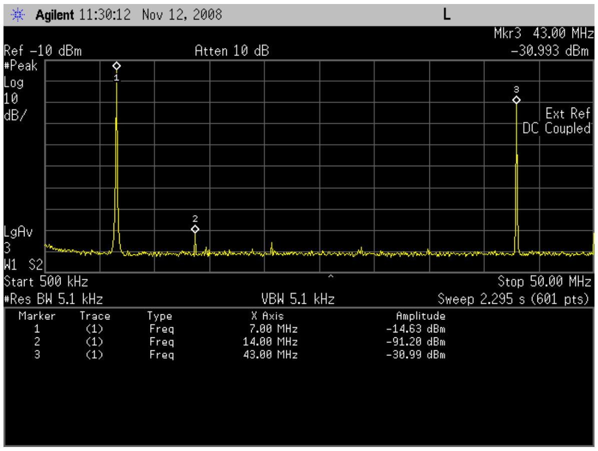
Sample Rate: 50.000 MHz
Desired Frequency: 2.000 MHz
Calculated Frequency: 1.999 MHz
DAC Resolution: 12 bits
Amplitude: 0.0 dB (Full Scale)
Cycles: 655
Record Length: 16384
Relative Phase: 0.0 °
□ Unsigned Data
□ Allow even cycle count
□ Generate Complex Data (I & Q)
Save to File

DPG2 Unit 1

Evaluation Board: AD9963
Port Configuration: 3.3V
Configuration Progress:
Configuration Version: 16.2.1.0 12/2/2008
Multi-DPG Sync: Single
Advanced/Debug...
I Data Vector: 1: Single Tone - 6.998 MHz; 0.0 dB; 0.0°
Q Data Vector: 2: Single Tone - 1.999 MHz; 0.0 dB; 0.0°
TXIQ Phase: High = I
TX Order: I First
Bus: TXPORT
Data Clock Frequency: 99.983 MHz
Download Progress:

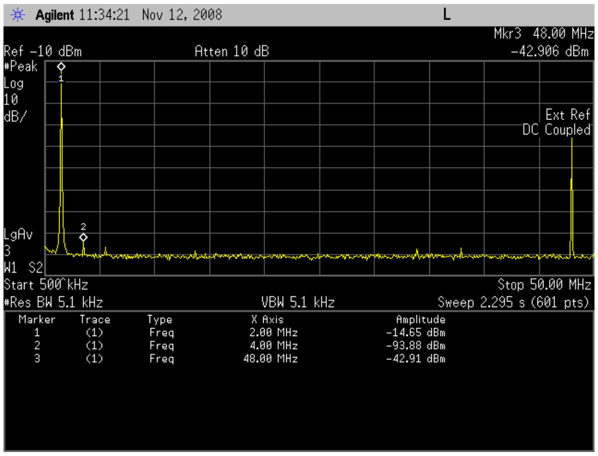
**Figure 3**

Click the Download Vector (⬇) button, and then the Play Vector (▶) button. Observe the output of each DAC via SMA jacks J7 and J8 with a spectrum analyzer. The results should closely match those provided in Figure 4 and Figure 5.



```
Agilent 11:30:12  Nov 12, 2008                      L
                                          Mkr3  43.00 MHz
Ref -10 dBm            Atten 10 dB             -30.993 dBm
#Peak
Log
10
dB/                                              Ext Ref
                                              DC Coupled
LgAv
3
W1  S2
Start 500 kHz                               Stop 50.00 MHz
#Res BW 5.1 kHz        VBW 5.1 kHz   Sweep 2.295 s (601 pts)
  Marker   Trace    Type       X Axis          Amplitude
    1       (1)     Freq      7.00 MHz        -14.63 dBm
    2       (1)     Freq     14.00 MHz        -91.20 dBm
    3       (1)     Freq     43.00 MHz        -30.99 dBm
```

*Figure 4*



```
Agilent 11:34:21  Nov 12, 2008                      L
                                          Mkr3  48.00 MHz
Ref -10 dBm            Atten 10 dB             -42.906 dBm
#Peak
Log
10
dB/                                              Ext Ref
                                              DC Coupled
LgAv
3
W1  S2
Start 500 kHz                               Stop 50.00 MHz
#Res BW 5.1 kHz        VBW 5.1 kHz   Sweep 2.295 s (601 pts)
  Marker   Trace    Type       X Axis          Amplitude
    1       (1)     Freq      2.00 MHz        -14.65 dBm
    2       (1)     Freq      4.00 MHz        -93.88 dBm
    3       (1)     Freq     48.00 MHz        -42.91 dBm
```

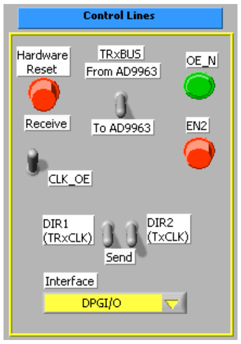
*Figure 5*

*Tx Settings: $f_{DAC}$=50MHz, $f_{out}$=7MHz and 2MHz, 2mA Full-Scale, 1X interpolation*

### *Transmit via Txport using DPGI/O*

In this setup, the DPGI/O is used to drive data into the Txport on the AD9963-EBZ. TxDATA and TxIQ marker signals are fed into the Txport from the DPGI/O.

In the AD9963 SPI application, switch to the *LDO/Transceiver* tab. Under the *Control Lines* heading, set the *Interface* to DPGI/O. See Figure 6.



Interface: DPGI/O

***Figure 6***

Click the Run button ( ), and wait for the run to complete.

Open the DPGDownloader application (Start > Programs > Analog Devices > DPG > DPGDownloader). With the AD9963 Evaluation Board connected to USB, DPGDownloader will load a special panel to correctly format the data for the AD9963. Ensure that "AD9963" is selected in the *Evaluation Board* drop-down. Then select the IO Voltage from the *Port Configuration* list (select 3.3V in this case).

Next, click on "Add Generated Waveform" along the top toolbar, and click "Single Tone". Enter 50MHz for the clock frequency, 7MHz for the desired frequency, and select 12 bits for the DAC resolution. Uncheck the "Unsigned Data" box. Repeat this process to add a Single Tone with a desired frequency of 2MHz.

In the lower half of the screen, select "1:Single Tone" for the *I Data Vector*, and "2: Single Tone" for the *Q Data Vector*. The Data Clock Frequency readout should indicate approximately 100MHz at this point.

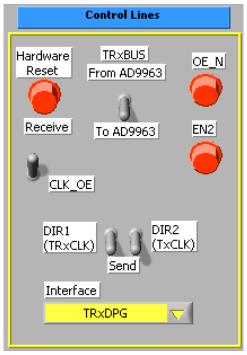
Click the Download Vector ( ) button, and then the Play Vector ( ) button. Observe the output of each DAC via SMA jacks J7 and J8 with a spectrum analyzer.

If needed, the bus timing can be adjusted to correct any errors in the output spectrum. Use the Timing Adjustment box to adjust the bus timing until any errors are fixed.

### *Transmit via TRxport using DPG2*

This setup is similar to the first setup, except that the data and framing signals are sent over the TRxport instead of the Txport. The TRxport defaults to Rx and needs to be turned around before the DPG2 attempts to transmit data into it.

In the AD9963 SPI application, switch to the *LDO/Transceiver* tab. Under the *Control Lines* heading, set the *Interface* to TRxDPG, and change *OE_N* to off. See Figure 7. Then switch to the *Dig Interface* tab. Under the *Bus Configuration 0x3F* heading, set the *Interface Mode* to *Half-Duplex* (see Figure 8).



Interface: TRxDPG

*Figure 7*

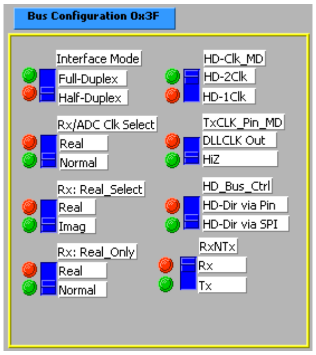


*Figure 8*

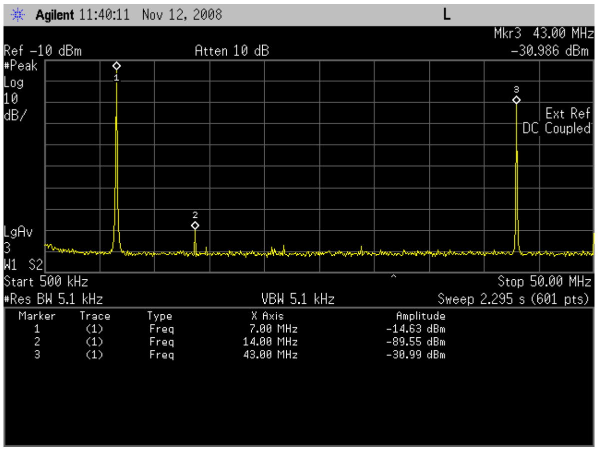
Click the Run button ( ), and wait for the run to complete.

Open the DPGDownloader application (Start > Programs > Analog Devices > DPG > DPGDownloader). With the AD9963 Evaluation Board connected to USB, DPGDownloader will load a special panel to correctly format the data for the AD9963. Ensure that "AD9963" is selected in the *Evaluation Board* drop-down. Then select the IO Voltage from the *Port Configuration* list (select 3.3V in this case). Select "TRXPORT" in the *Bus* drop-down.

Next, click on "Add Generated Waveform" along the top toolbar, and click "Single Tone". Enter 50MHz for the clock frequency, 7MHz for the desired frequency, and select 12 bits for the DAC resolution. Uncheck the "Unsigned Data" box. Repeat this process to add a Single Tone with a desired frequency of 2MHz.

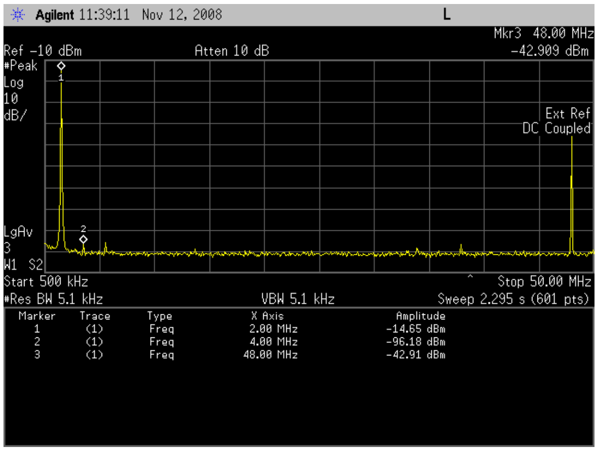
In the lower half of the screen, select "1:Single Tone" for the *I Data Vector*, and "2: Single Tone" for the *Q Data Vector*. The Data Clock Frequency readout should indicate approximately 100MHz at this point.

Click the Download Vector ( ) button, and then the Play Vector ( ) button. Observe the output of each DAC via SMA jacks J7 and J8 with a spectrum analyzer. The results should closely match those provided in Figure 9 and Figure 10.



```
Agilent 11:40:11  Nov 12, 2008                L
                                      Mkr3  43.00 MHz
Ref -10 dBm          Atten 10 dB          -30.986 dBm
Start 500 kHz                          Stop 50.00 MHz
#Res BW 5.1 kHz      VBW 5.1 kHz   Sweep 2.295 s (601 pts)
  Marker   Trace    Type        X Axis       Amplitude
    1       (1)     Freq       7.00 MHz     -14.63 dBm
    2       (1)     Freq      14.00 MHz     -89.55 dBm
    3       (1)     Freq      43.00 MHz     -30.99 dBm
```

*Figure 9*



```
Agilent 11:39:11  Nov 12, 2008                L
                                      Mkr3  48.00 MHz
Ref -10 dBm          Atten 10 dB          -42.909 dBm
Start 500 kHz                          Stop 50.00 MHz
#Res BW 5.1 kHz      VBW 5.1 kHz   Sweep 2.295 s (601 pts)
  Marker   Trace    Type        X Axis       Amplitude
    1       (1)     Freq       2.00 MHz     -14.65 dBm
    2       (1)     Freq       4.00 MHz     -96.18 dBm
    3       (1)     Freq      48.00 MHz     -42.91 dBm
```

*Figure 10*

*Tx Settings: $f_{DAC}$=50MHz, $f_{out}$=7MHz and 2MHz, 2mA Full-Scale, 1X interpolation*

### Receive via TRxport using DPGI/O

This setup will use the DPGI/O to receive data from the AD9963-EBZ.

In the AD9963 SPI application, switch to the *LDO/Transceiver* tab. Under the *Control Lines* heading, set the *Interface* to DPGI/O. Set the *TRxBUS* switch to the *From AD9963* position, and turn off the *OE_N* button. See Figure 11. Switch to the *DLL/Clock Doubler* tab, and turn off Duty Cycle Correction by enabling the DCS_DIS button (see Figure 12).

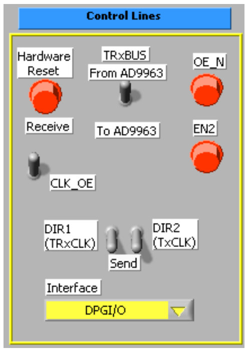


**Figure 11**

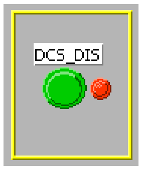
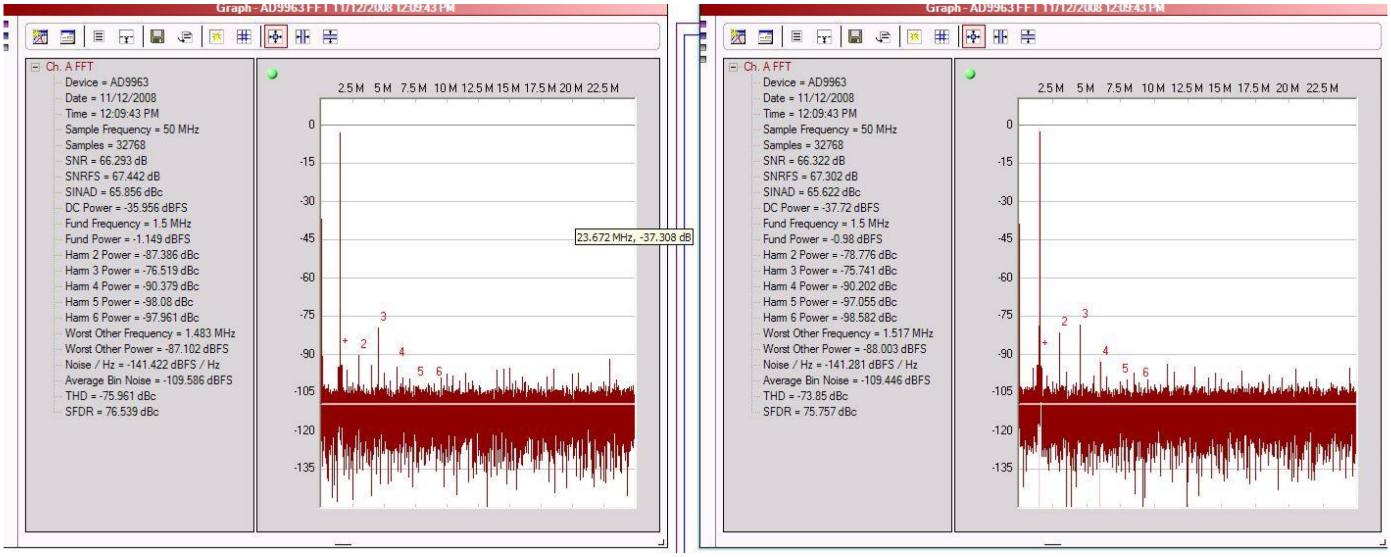


**Figure 12**

Connect the DPGI/O to the PC, and then open VisualAnalog (Start > Programs > Analog Devices > VisualAnalog > VisualAnalog). Click on the "Existing" tab, and navigate to C:\Program Files\Analog Devices\HSDAC\AD9963\. Open the AD9963_FFT_Canvas1.vac file.

Select the 3.3V for the IO Voltage in the *DPGI/O Data Capture* block. Connect a spectrally clean sinusoidal signal generator to the ADC input SMA jacks (through a splitter). Set the generator for a 1.5MHz tone at 8.4dBm. Run the canvas by clicking on the Run ( ▶ ) button in the upper left. The two FFT displays should show data similar to that shown in Figure 13.

For best results, use a band pass filter on the ADC inputs to limit spectral power outside your region of interest.



Ch. A FFT
- Device = AD9963
- Date = 11/12/2008
- Time = 12:09:43 PM
- Sample Frequency = 50 MHz
- Samples = 32768
- SNR = 66.293 dB
- SNRFS = 67.442 dB
- SINAD = 65.856 dBc
- DC Power = -35.956 dBFS
- Fund Frequency = 1.5 MHz
- Fund Power = -1.149 dBFS
- Harm 2 Power = -87.386 dBc
- Harm 3 Power = -76.519 dBc
- Harm 4 Power = -90.379 dBc
- Harm 5 Power = -98.08 dBc
- Harm 6 Power = -97.961 dBc
- Worst Other Frequency = 1.483 MHz
- Worst Other Power = -87.102 dBFS
- Noise / Hz = -141.422 dBFS / Hz
- Average Bin Noise = -109.586 dBFS
- THD = -75.961 dBc
- SFDR = 76.539 dBc

Ch. A FFT
- Device = AD9963
- Date = 11/12/2008
- Time = 12:09:43 PM
- Sample Frequency = 50 MHz
- Samples = 32768
- SNR = 66.322 dB
- SNRFS = 67.302 dB
- SINAD = 65.622 dBc
- DC Power = -37.72 dBFS
- Fund Frequency = 1.5 MHz
- Fund Power = -0.98 dBFS
- Harm 2 Power = -78.776 dBc
- Harm 3 Power = -75.741 dBc
- Harm 4 Power = -90.202 dBc
- Harm 5 Power = -97.055 dBc
- Harm 6 Power = -98.582 dBc
- Worst Other Frequency = 1.517 MHz
- Worst Other Power = -88.003 dBFS
- Noise / Hz = -141.281 dBFS / Hz
- Average Bin Noise = -109.446 dBFS
- THD = -73.85 dBc
- SFDR = 75.757 dBc

*Figure 13*

*Rx Settings: $f_{in}$=1.5MHz, $f_{ADC}$=50MHz, No decimation*

Depending on the sample rate, the clock delay may need to be adjusted to sample the ADC outputs properly. If the data does not appear to be correct, adjust the *Clock Delay* on the *DPGI/O Data Capture* block in the canvas.

If other clock rates are used, the DCS_DIS option should be enabled when the clock is between 20MHz and 70MHz, and disabled otherwise.

## SPI APPLICATION

This section discusses several parts of the SPI controller interface, as they affect this evaluation system. For complete descriptions of each SPI register, see the AD9963 datasheet. In the interest of continuous quality improvements, the images below may not exactly match your version of the software.

### *LDO / Transceiver Tab*

This section allows the user to perform a hardware reset, power down specific on-chip LDOs and configure the other ICs on the evaluation board. This tab will always need to be adjusted in accordance with the interface.

### *Digital Interface Tab*

This section groups all the registers needed to select the data interface mode, digital clocking, and set up signal processing controls on the digital datapaths. The SPI format has been programmed to "MSB First" in the software and in the AD9963.

**Bus Configuration**

This selects between Full Duplex and Half-Duplex. When using Half-Duplex mode, there are additional clocking and direction controls that need to be set up. Refer to AD9963 datasheet for more details.

**Tx Datapath**

This comprises of controls for:

a) *TxFIFO*: The initial phase between the FIFO's read and write pointers can be set, and the current positions of the pointers can be read. During normal operation, the phase read back should statically display four consecutive 1s and four consecutive 0s, wrap-around cases also included.

b) *Modes to sample TxDATA*: One can choose between SDR and DDR data latching. The CLKPhase and IQphase switches invert the sampling clocks and are very useful if there is a timing violation in latching the incoming data. Make sure the Tx Datapath Clock Enable is turned ON.

c) *Interpolation Filters*: There are three 2x interpolation filters, but the 2x SRRC is intended for 8x interpolation only. If it is enabled, the other two filters need to be enabled as well.

d) *Post-filter scaling*: Setting the scaling bits incorrectly can result in digital saturation leading to excessive spurs in the DAC output spectrum.

e) *TxCLKIO pin clock modes*: If the TxCLK pin is set up as the transmit sampling clock, the clock presented on the pin is used to generate the internal TxDATA sampling clock. If the DLL clock is selected on the pin, it has to be at the interleaved data rate and meet timing to properly latch the TxDATA.

**Rx Datapath**

This comprises of controls for:

a) *Modes and formats to clock out RxDATA*: If I-Only is selected, the interleaved data becomes I-I as opposed to the default I-Q, however the clock rate on TRxCLK remains unchanged.

b) *Decimation Filter:* If the decimation filter is used, the TRxCLK, TRxIQ, and interleaved data rate reduces by a factor of 2.

c) *Post-Filter scaling*: Setting the scaling bits incorrectly can result in digital saturation leading to excessive spurs in the ADC spectrum.

d) *TRxCLK pin clock modes*: During receive, regardless of the clock present at the pin, the RxDATA and RxIQ signals are driven out at the correct rate.

### *DLL / Clock Doubler*

This section configures the analog clocking for the AD9963 DACs and ADCs. The AD9963 may be clocked with a clock doubler or the configurable DLL. These sections are described below, with a recommended settings table following.

a) Clock Doubler – Clocks the ADCs and DACs with the clock doubler, configured by Tx_Doubler and Rx_Doubler. For each clock the offset must be set to 1 and the inital tapdef must be set to 0.

b) DLL - The DLL subsection can configure an alternate DAC or ADC clock. The user sets the multiplier and subsequent divide ratios for the delay lock loop using the main clock input as a reference clock.

| FClk | Recommended Settings |
|---|---|
| 0.5MHz – 15MHz | Fixed Pulse mode Clock Doubler |
| 13MHz - 70MHz | 50% Duty Cycle mode with Duty Cycle Correction |
| 70MHz – 100MHz | 50% Duty Cycle mode |

### *TxDACs/RxADCs Tab*

This section sets up various gain and power down controls for the TxDACs and gain, offset and power down controls for the RxADCs. To program the offset on a specific RxADC, first select the desired ADC (by clicking buttons labeled I-ADC or Q-ADC or both at the top) and then set the desired offset trim.

### Aux Converters Tab

This section comprises of three subsections:

#### Aux10DACs

These do not need a clock to effect an output change. The values for AuxVDD, Range and Top of Range and the 10-bit code determine the voltage readout. Updates are hard-coded in the SPI controller to happen on writing to the code MSB.

#### Aux12DACs

These do not need a clock to effect an output change. The values for AuxVDD, Range and AuxREF determine the voltage readout. Updates are hard-coded in the SPI controller to happen on writing to the code MSB.

#### Aux12ADC

This derives its sampling clock from the main AD9963 clock based on the divide ratio. The CLK Enable button has to be ON to initiate a conversion. The pre-sampling delay can be set in cycles of the divided down Aux12ADC clock. This is most useful to allow the input to settle after a new channel is selected and before a sample is taken. To read back the DUT temperature, select the Temp Sensor input on Input Ch Select and make sure the Temp Sensor Enable button is ON. For more information, refer to the AD9963 datasheet.

### *Saving SPI Register Values to a File*

The SPI Controller application is able to save all current register values to a file, for later reference. To utilize this feature, click the *Save* button to "ON" in the lower right side of the screen. The next time the Run button (⇨) is pressed, a prompt will ask where to save the file.

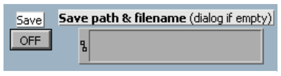


*Figure 14*

## SETUP SEQUENCES

The following are example SPI register sequences for setting up various features of the part. Some of the sequence is often hidden in the SPI application that comes with the evaluation board, so these sequences are provided to alleviate any confusion.

### *DLL Setup Sequence*

The below series of register writes will configure the DLL to drive the DACs with a multiplication of 10 and a division of 3 from the main clock input. From reset, this would take a 20MHz main clock, multiply by ten to an internal frequency of 200MHz, then divide the clock down by 3 to produce 66.67MHz. Then a write to register 0x71 configures the DAC clock to be the DLL. Since the default SDR mode for the DACs was not changed, each DAC is being clocked at 66.67MHz and the TxCLK pin will output 133.33MHz (Interleaved bus SDR clock). Fast SPI clocks will require delays after the DLL registers have been set before and after the reset bit has been pulled high.

| Register | Data | Comments |
|---|---|---|
| 0x60 | 0x80 | Enable DLL |
| 0x71 | 0x53 | Set DAC clock to DLL / enable DLL reference / N=3 |
| 0x72 | 0x09 | M=9, Effective multiplication is M+1=10 |
| *Delay 100pS* | | |
| 0x75 | 0x08 | Hold DLL reset high |
| *Delay 100pS* | | |
| 0x75 | 0x00 | Hold DLL reset low |
| 0x72 | *Read* | Check Bit 7 to verify the DLL has locked |

### *Clock Doubler Setup Sequence*

The 50% Duty Cycle clock doubler is recommended for clocking the ADCs and DACs above 15MHz. When operating below 75MHz, the duty cycle correction capability should be bypassed and care should be taken to ensure the main clock input has a duty cycle of 45-55%. The below series of writes configures the clock doubler to clock the ADCs and output on the RxCLK pin from reset. These writes are for an ADC clock of < 75MHz.

The same sequence could be used for creating a clock >75MHz by removing the write to register 0x66.

| Register | Data | Comments |
|---|---|---|
| 0x3C | 0x00 | The recommended tap delay is '0' |
| 0x39 | 0x02 | Configure RxCLK as clock doubler |
| 0x66 | 0x04 | Bypass duty cycle correction (for main clock < 75MHz) |
| 0x3B | 0x55 | The recommended offset is 1 (changing bit 3 from default) |
| *Delay 100pS* | | |
| 0x39 | 0x82 | Reset Rx clock doubler |
| *Delay 100pS* | | |
| 0x39 | 0x02 | Pull Rx clock doubler out of reset |
| 0x63 | 0x08 | Set drive strength to 3 for the RxClk |
| 0x84 | *Read* | Bit 1 flags Rx DLL is locked |

### *Temperature Sensor Setup Sequence*

The following sequence should be performed to setup the internal auxiliary ADC to read the internal temperature sensor.

| Register | Data | Comments |
|---|---|---|
| 0x77 | 0x03 | Channel Temp Sensor |
| 0x7A | 0x80 | Aux ADC Enable |
| 0x7B | 0x80 | Temp Sensor Enable |

### *Temperature Sensor Read Sequence*

The following sequence should be performed to read the temperature sensor.

| Register | Data | Comments |
|---|---|---|
| 0x77 | 0x83 | Choose Channel to sample with AUX ADC |
| *Delay at least 20 AUXADCCLK cycles* | | |
| 0x78 | *Read* | MSB 7:0 = Temp[11:4] |
| 0x79 | *Read* | LSB 7:4 = Temp[3:0] |

## NOTES:

©2010 Analog Devices, Inc. All rights reserved. Trademarks and registered trademarks are the property of their respective owners.