
PXle-5841 User Manual

2026-01-23



Contents

PXle-5841 User Manual	6
PXle-5841 Overview	7
PXle-5841 Driver Support	8
Components of a PXle-5841 System	10
Part Numbers for Recommended Cables and Accessories	13
PXle-5841 Theory of Operation	14
PXle-5841 Block Diagram	14
RF Input Subsystem	15
RF Input Signal Path	16
RF IN Signal Gain and Attenuation	17
RF Output Subsystem	17
RF Output Signal Path	19
Using the PXle-5841 in CW Mode	20
Average Power and Crest Factor Considerations	20
FPGA Basecard Subsystem	20
Xilinx Virtex-7 FPGA	21
Clocking	22
ADC and DAC	23
PFI 0 Connector	24
DRAM	24
Configuring DRAM with FPGA Memory Items	24
SRAM	25
SRAM Interface	25
Configuring SRAM with Socketed CLIP	25
SRAM Properties Dialog Box	26
Power and Thermal Monitoring and Shutdown Conditions	27
PCI Express Interface	28
DIGITAL I/O (DIO)	28
Cables and Accessories	30
LO Path	30
PXle-5655 Block Diagram	32
VST State Models	32

NI-RFSA State Model	33
NI-RFSG State Model	36
PXle-5841 Front Panel and LEDs	38
PXle-5841 Pinout.....	43
PXle-5655 Front Panel and LEDs	46
PXle-5841 Installation and Configuration	49
Preparing the Environment.....	49
Unpacking the Kit	50
Verifying the Kit Contents.....	51
Other Equipment	51
PXle-5841 Mounting Guidelines	52
Installing the Software	53
Installing the PXle-5841	53
Direct Connections to the PXle-5841	56
Connecting the PXle-5655 Analog Signal Generator to the PXle-5841.....	56
Configuring the Hardware in MAX	57
Configuring the PXle-5841 in Hardware Configuration Utility.....	59
PXle-5841 Examples	61
Making a First Measurement.....	67
Using the PXle-5841	68
Analyzing and Generating RF Signals with NI VSTs.....	68
Analyzing RF Signals with RFmx and NI-RFSA	69
Analyzing RF Signals with RFmx	70
Initialize a Session	71
Configure Hardware Settings.....	72
Configure the Measurement	72
Close the Session	73
Analyzing RF Signals with NI-RFSA.....	73
Initialize a Session	74
Configure the Reference Clock	74
Configure the Acquisition.....	75
Coordinate Actions with Triggers and Events	76
Read and Plot Acquired Data.....	79
Close the Session	79
Generating RF Signals with NI-RFSG	79
Initialize a Session	80

Configure the Reference Clock	81
Configure the RF Output Signal	81
Coordinate Actions with Triggers and Events	82
Initiate Waveform Generation	88
Check the Status of RF Signal Generation	88
Close the Session	89
Improving NI-TClk Synchronization of RF Instruments	89
NI-TClk Manual Calibration Using NI-RFSA	89
NI-TClk Manual Calibration Using NI-RFSG	90
Local Oscillators	90
LO Sharing Between NI-RFSA and NI-RFSG	93
Using an External LO	95
Spurious Performance	96
Timing Configurations	97
Configuring Frequency and Bandwidth for PXle-5841	98
Frequency and Bandwidth Selection to Optimize Performance	100
Automatic Frequency Offset	104
Using an External LO	107
Peer-to-Peer Data Streaming	108
Configuring a Peer-to-Peer Endpoint	109
Configuring a Peer-to-Peer Stream	110
RF Path Transitions	113
Frequency Offset	116
Using In-Band Retuning	118
Interconnecting Multiple PXle-5841 RF Channels (Homogeneous Channel Types) ..	120
Quadrature Mixing (Direct Conversion)	123
Configuring Attenuator Hold	125
De-embedding Overview	126
Unsupported Software Features	129
Self-Calibration	130
Performing a Device Self-Calibration Using the NI-RFSA or NI-RFSG SFP	130
Self-Calibrating the PXle-5841 with PXle-5655	131
Cleaning the PXle-5841 System	132
Troubleshooting	133
What Should I Do if My Hardware Doesn't Appear in Hardware Configuration Utility or MAX?	133

Why Is the ACCESS LED Off When the Chassis Is On?	134
Fundamental RF Concepts	135
Common NI RF Terminology	135
Carrier Wave	137
Message Signal	138
Modulation	138
I/Q Modulation	139
Sample Rate	142
Impedance Matching	144
Resolution Bandwidth	147
Noise Fundamentals	148
Streaming	155
Scripts	155
How Signal Routing Works	157
Triggers and Events	160
De-embedding Overview	161

PXIe-5841 User Manual

The PXIe-5841 User Manual provides detailed descriptions of the product functionality and the step by step processes for use.

Looking for Something Else?

For information not found in the User Manual for your product, such as specifications and API reference, browse ***Related Information***.

Related information:

- [NI Hardware and Software Operating System Compatibility](#)
- [PXIe-5841 Specifications](#)
- [PXIe-5841 Calibration Procedure](#)
- [RFmx User Manual](#)
- [NI RF Signal Analyzers User Manual](#)
- [NI RF Signal Generators User Manual](#)
- [Software and Driver Downloads](#)
- [License Setup and Activation](#)
- [Release Notes](#)
- [NI Learning Center](#)
- [NI Community](#)

PXIe-5841 Overview

The PXIe-5841 is a vector signal transceiver (VST) with 1 GHz of instantaneous bandwidth. It combines vector signal generator, vector signal analyzer and high-speed serial interface capabilities with FPGA-based real-time signal processing and control. You can use the VST to test a variety of cellular and wireless standards such as Wi-Fi 6 and 5G NR. In addition, you can expand the PXIe-5841 to support multiple input, multiple output (MIMO) configurations.

The PXIe-5841 name encompasses the following instruments:

Table 1. PXIe-5841 Instruments

Instrument	Part Numbers	Constituent Modules
PXIe-5841	786982-01, 788842-01, 788843-01	PXIe-5841
PXIe-5841 with Enhanced LO Synthesizer	785832-01, 788844-01, 788845-01	PXIe-5841, PXIe-5655

PXIe-5841 Key Features

The PXIe-5841 has the following features and capabilities:

Frequency range	9 kHz to 6 GHz
RF Instantaneous Bandwidth	1 GHz
Generator Tuning Time (Typical)	175 μ s

PXle-5841 Driver Support



Tip To optimize product performance, update to the most recent driver version.

Table 2. Driver Software

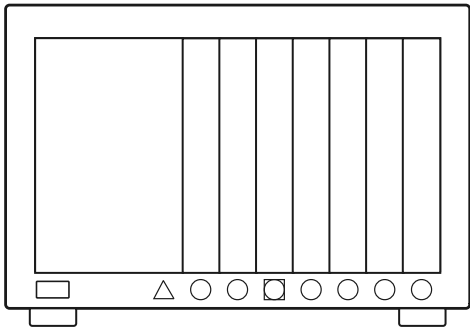
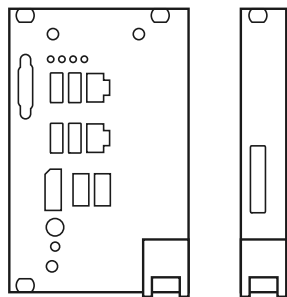
Driver Name	Description	Use Case
NI-RFSA	The NI-RFSA and NI-RFSG APIs each provide a fixed API for interacting with your hardware. NI-RFSA and NI-RFSG provide standard IVI-based functionality needed for most vector signal analyzer and vector signal generator applications.	• Use NI-RFSA to create custom measurements or applications that require I/Q data. • Use NI-RFSA and NI-RFSG with the NI Modulation Toolkit to develop software-defined radio transmitters and receivers.
NI-RFSG	You can use NI-RFSA and NI-RFSG FPGA extensions to customize the device FPGA using precompiled bitfiles.	• Use NI-RFSG to generate continuous wave (CW) or modulated waveforms. • Use NI-RFSG with the NI Modulation Toolkit to create and generate modulated signals to test receivers.
RFmx	Provides a single-handle instrument driver with built-in measurements. You can use instrument driver FPGA extensions to customize the device FPGA using precompiled bitfiles.	• Use RFmx SpecAn to perform spectral measurements. • Use RFmx Demod to perform modulation quality measurements, such as EVM. • Use RFmx personalities to perform physical layer

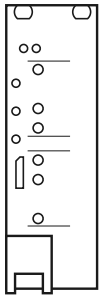
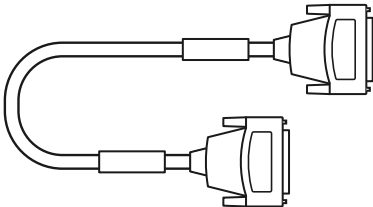
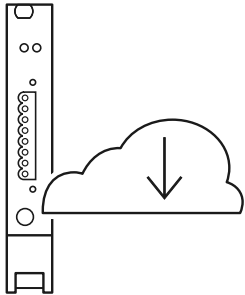
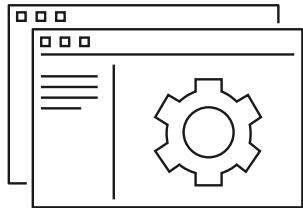
Driver Name	Description	Use Case
		analysis on supported communications and connectivity signals.

Components of a PXIe-5841 System

PXIe-5841 is designed for use in a system that may require hardware, drivers, and software to optimize PXIe-5841 for your application. Use the minimum required PXIe-5841 system components as a starting point for building your system.

Table 3. Minimum Required PXIe-5841 System Components

Component	Description and Recommendations
PXI Chassis 	A PXI chassis houses the PXIe-5841 and supplies power for PXIe-5841 functions.
PXI Controller or PXI Remote Control Module 	You can install a PXI controller or a PXI remote control (MXI) module depending on your system requirements. These components, installed in the same PXI chassis as the PXIe-5841, interface with the instrument using NI device drivers.
VST	Your VST instrument.

Component	Description and Recommendations
	
Cables and Accessories 	Cables and accessories allow connectivity to/from your instrument for measurements. Refer to <i>Cables and Accessories</i> for cable and accessory recommendations and guidance.
Driver Software 	Instrument driver software that provides functions to interact with the PXIe-5841 and execute measurements using the PXIe-5841.  Note For optimal performance, use the most current version of driver software with the PXIe-5841. You can find the driver requirements on ni.com.
NI Applications 	Instrument driver software offers support for the following applications: • InstrumentStudio • LabVIEW • LabWindows/CVI

Component	Description and Recommendations
	• C/C++ • .NET • Python

Related reference:

- [Part Numbers for Recommended Cables and Accessories](#)

Part Numbers for Recommended Cables and Accessories

Use part numbers to purchase the cables and accessories NI recommends to optimize the performance of PXIe-5841.

Table 4. Part Numbers for Recommended Cables and Accessories

Accessory/Cable	Description	Part Number
Cable	MMPX (m)-to-MMPX (m) flexible cables (x2), Length 47 mm	769732-0R047
Cable	MMCX (m)-to-MMCX (m) flexible cable, Length 56 mm	769769-0R056
Cable	OCuLink x4, 1 m cable	785486-01
Cable	OCuLink x4 to mini-SAS HD, 1 m cable	786215-01

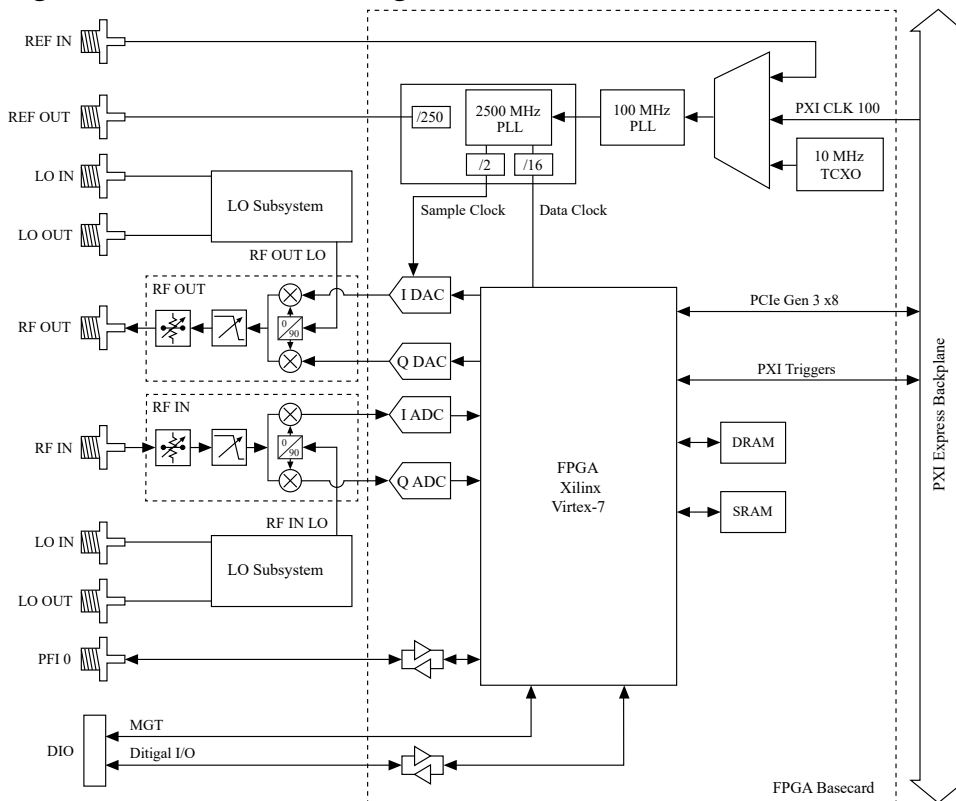
PXIe-5841 Theory of Operation

Refer to block diagrams and state models to understand the hardware and software design behind the PXIe-5841.

PXIe-5841 Block Diagram

The PXIe-5841 is a vector signal transceiver (VST) with 1 GHz of instantaneous bandwidth. It has the RF Input, RF Output, FPGA Basecard, and LO Path subsystems.

Figure 1. PXIe-5841 Block Diagram



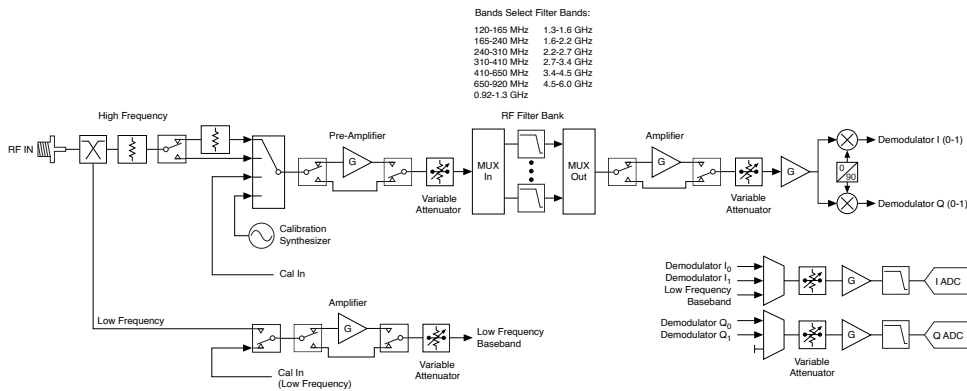
The LO IN terminals can import a higher performance external LO such as the PXIe-5655.



Note Some calibration-related routing paths and front panel connectors have been omitted from the illustration of the high-level block diagram for simplicity.

RF Input Subsystem

The PXle-5841 has two configurable RF paths comprised of an independent vector signal analyzer (RF input) path and vector signal generator (RF output) path. The following high-level block diagram represents the PXle-5841 RF input path.



The PXle-5841 RF input path is AC coupled and covers a frequency range of 9 kHz to 6,000 MHz center frequency. For frequencies below 120 MHz (low frequency), RF input signals are directly sampled via analog-to-digital conversion. Frequencies greater than 120 MHz (high frequency) are translated to baseband via a single stage direct conversion (I/Q) downconverter, allowing for wide instantaneous bandwidth.

High Frequency RF Input Path

The high frequency subsystem of the RF input path consists of a programmable and filtered front end that feeds into a direct conversion downconverter. Once at baseband frequencies, the signal is sampled with an analog-to-digital converter (ADC) capable of up to 1,000 MHz of complex equalized bandwidth.

RF attenuation greater than 70 dB can be nominally configured through the combination of switchable attenuation and programmable step attenuators. RF attenuation is automatically chosen in conjunction with amplification stages to provide optimal reference levels over the support range of the device.

A switchable pre-amplifier is also configurable in the front end of the device to further optimize the signal path for noise figure. This pre-amplifier can be configured for bypass, forced inline, or automatically configured based on application needs. The configurable range of reference levels may vary based on which option is selected for

the pre-amplifier.

Low Frequency RF Input Path

The low frequency subsystem of the RF input path is used to directly sample RF input signals below 120 MHz. The RF input signal is provided via a frequency selective diplexer to programmable gain components before being filtered and sampled with an analog-to-digital converter.

The programmable gain elements at low frequency consist of a switchable amplifier and a programmable attenuator. When disabled, the switched amplifier is bypassed by a 10 dB attenuator. The programmable attenuator is capable of 31 dB attenuation in 1 dB steps.

Refer to the ***RF Path Transitions*** topic for more information on using the low frequency RF input path.

Cal In Path

The RF input subsystem has internal calibration capability in both the high frequency and low frequency paths. Cal In is used in conjunction with Cal Out from the RF Output subsystem to facilitate steps within the following VIs or functions:

- NI-RFSA Self-Calibrate VI or niRFSA_SelfCalibrate
- NI-RFSG Self-Calibrate VI or niRFSG_SelfCal
- NI-RFSA Self-Calibrate Range VI or niRFSA_SelfCalibrateRange
- NI-RFSG Self-Calibrate Range VI or niRFSG_SelfCalibrateRange

Related concepts:

- [RF Path Transitions](#)

RF Input Signal Path

The RF source signal takes the following path from the PXIe-5841 front panel to the PXI Express controller.

1. The RF signal is connected to the RF IN front panel connector.

2. The RF signal is passed through the frequency-selective path of the front end diplexer to either the high frequency or low frequency path.
3. The signal is amplified or attenuated based on the reference level to the required mixer level, based on available gain in the selected RF path. The amplification and attenuation maintain optimized dynamic range and avoid signal saturation or ADC overflow.
4. The RF signal is frequency-translated based on the signal path used:
 - When in the high frequency path, the attenuated or amplified signal is mixed with a 0° phase and 90° phase LO signal at the tuned center frequency, translating the resulting signal into a pair of I and Q signals 90° out-of-phase from each other. These I and Q signals are then conditioned and sampled by an ADC converter.
 - When in the low frequency path, the attenuated or amplified signal is directly passed to signal conditioning and the ADC without frequency translation.
5. The translated signal is sampled by the ADC at 1,250 MS/s. The resulting signals are accessible through LabVIEW FPGA I/O nodes for further processing or transfer to the host computer.

RF IN Signal Gain and Attenuation

The RF IN signal path chooses gain and attenuation settings based on the requested reference level.

The gain definition measured in calibration is adjusted and applied according to the following equation:

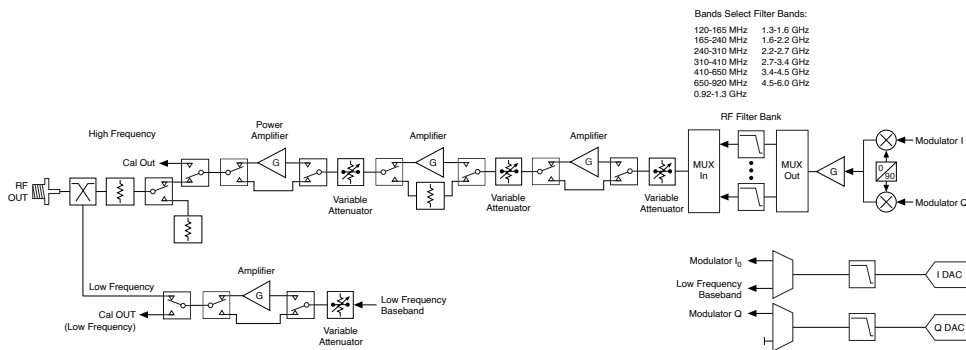
$$\text{RF Input Gain (dB)} = \text{Complex Power}_{\text{ADC}}(\text{dB}) - \text{Real Power}_{\text{RF IN Terminal}}(\text{dB})$$

The resulting gain scales the acquired data to reference back to the RF IN front panel connector. The reported gain scales both I and Q evenly. If you compute power by reading I/Q data and running a power spectrum or Fast Fourier Transform (FFT), scale the gain by a gain compensation value of 0.5 or reduce the power result by 3 dB. If you use Spectrum Mode, NI-RFSA scales the power for you.

RF Output Subsystem

The PXIe-5841 has two configurable RF paths comprised of an independent vector

signal analyzer (RF input) path and vector signal generator (RF output) path. The following high-level block diagram represents the PXIe-5841 RF output path.



The PXIe-5841 RF output path is AC coupled and covers a frequency range of 9 kHz to 6,000 MHz center frequency. For frequencies below 120 MHz (low frequency), RF output signals are directly generated via digital-to-analog (DAC) conversion. Frequencies greater than 120 MHz (high frequency) are translated from baseband I and Q signals to the specified RF frequency via a single stage direct conversion (I/Q) upconverter, allowing for a wide instantaneous bandwidth.

High Frequency RF Output Path

The high frequency subsystem of the RF output path consists of a wideband direct conversion upconverter and a front end signal path with programmable RF filtering and gain stages. I and Q baseband signals generated by the DAC are mixed and combined to create an RF signal with up to 1,000 MHz of complex equalized bandwidth centered at the specified RF center frequency. This signal can then be conditioned for both bandwidth and level.

Greater than 80 dB of RF attenuation can be configured nominally through the combination of switchable fixed attenuation and programmable step attenuators. RF attenuation is automatically chosen in conjunction with amplification to provide optimal reference levels over the support range of the device. There are two switchable amplifiers and one switchable high power amplifier.

The switchable high power amplifier is used for generating signals significantly larger than 0 dBm. This amplification stage is automatically selected based on output power level setting and extends the available peak power range to +20 dBm, calibrated. Additional power may be achieved above +20 dBm, however it is not guaranteed to meet absolute or relative accuracy specifications.

Low Frequency RF Output Path

The low frequency subsystem of the RF output path is used to directly generate the requested RF output signals below 120 MHz from the DAC. The RF signal is generated from the DAC through programmable gain stages and transmitted from the RF OUT front panel connector through a frequency selective diplexer. This allows for a seamless transition from low frequency to high frequency paths. Refer to the ***RF Path Transitions*** topic for more information about using the low frequency path.

The low frequency path programmable gain stages consist of a 31 dB step attenuator with 1 dB resolution and a switchable amplification stage. The off state of the switchable amplifier has additional fixed attenuation integrated into it. This provides a nominal gain range of +5 dB to -40 dB. Additional attenuation is possible with the DAC full scale configuration settings as well as digital waveform scaling.



Note NI recommends external attenuation for applications requiring a significant amount of analog attenuation when using the low frequency RF output path.

Related concepts:

- [RF Path Transitions](#)

RF Output Signal Path

The RF output signal takes the following path from the PXIe-5841 FPGA to the RF OUT front panel connector.

1. The I and Q digital signals are sent to the baseband digital-to-analog (DAC) converter through a LabVIEW FPGA I/O node after being written from the host.
2. The DAC converts the digital I and Q waveforms into analog I and Q signals at 1,250 MS/s. Internal to the DAC, the output is two times interpolated to 2,500 MS/s.
3. Based on the specified center frequency, the analog I and Q signals are conditioned and passed either to the direct conversion modulator high frequency path, or sent directly to the low frequency path for additional level control.
4. Depending on the selection of high frequency path or low frequency path, the converted RF frequency is further attenuated or amplified based on the specified

output power settings.

5. The conditioned RF output signal is transmitted from the PXle-5841RF OUT front panel connector through a frequency selective diplexer stage.

Using the PXle-5841 in CW Mode

You can use the PXle-5841 as a continuous waveform (CW) generator when programming the PXle-5841 using NI-RFSG. To generate a CW signal, use NI-RFSG Configure Generation Mode.

Average Power and Crest Factor Considerations

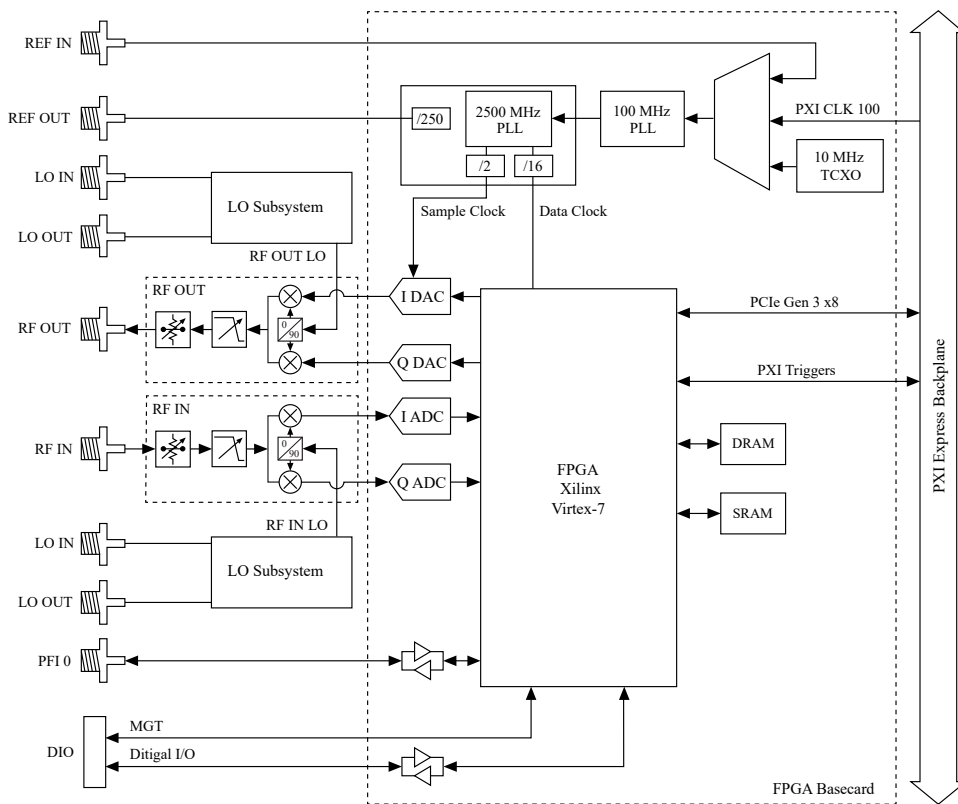
Crest factor is the difference between peak signal power and average power.

The crest factor for a sinusoidal signal, as is used in CW mode, is 3 dB. In other words, the average RMS power of the sinusoid is 3 dB less than its peak instantaneous power. For modulated signals, specifically orthogonal frequency-division multiplexing (OFDM), the crest factor can be much larger, in the order of 10 dB to 12 dB.

Consider both the average power and the crest factor of a signal when you configure the device for generation. The PXle-5841 supports a maximum specified peak output power. Generating signals beyond specification are not guaranteed to be calibrated or linear. If the peak power is set higher than specified as a result of signal crest factor added with requested average power, severe saturation might occur or the reverse power protection circuitry of the PXle-5841 may be enabled. Refer to the **PXle-5841 Specifications** for more information about output power range specifications.

FPGA Basecard Subsystem

The PXle-5841 has an FPGA basecard, represented by the following high-level block diagram.



The FPGA basecard of the PXIe-5841 consists of the following components: FPGA, Clocking, ADCs and DACs, PFI 0, Digital I/O, DRAM, SRAM, PCIe Interface, and PXI triggers.

Xilinx Virtex-7 FPGA

The PXIe-5841 contains a Xilinx Virtex-7 VX690T FPGA, which is used for system configuration, digital data movement, and digital signal processing.

The FPGA has direct connections to the ADC, DAC, PCI Express bus, DRAM, SRAM, PFI 0, digital I/O, and PXI trigger lines, allowing for custom programming using LabVIEW FPGA to meet the needs of many types of applications.

The Xilinx Virtex-7 FPGA has the following resources.

Table 5. Reconfigurable FPGA Resources

Resource Type	Number
Slice registers	866,400
Slice look-up tables (LUT)	433,200

Resource Type	Number
DSP48E1s	3600
18 Kb block RAMs	2940

Clocking

The PXIe-5841 has multiple clocks available on the device and inside the FPGA. The main device clock is the Sample Clock, which is used to clock the ADC, clock the DAC, and create the Data Clock for the related FPGA logic.

Sample Clock

The Sample Clock runs at 1.25 GHz and is exported by the phase-locked loop (PLL).

You can select one of the following resources as the reference signal for the PLL:

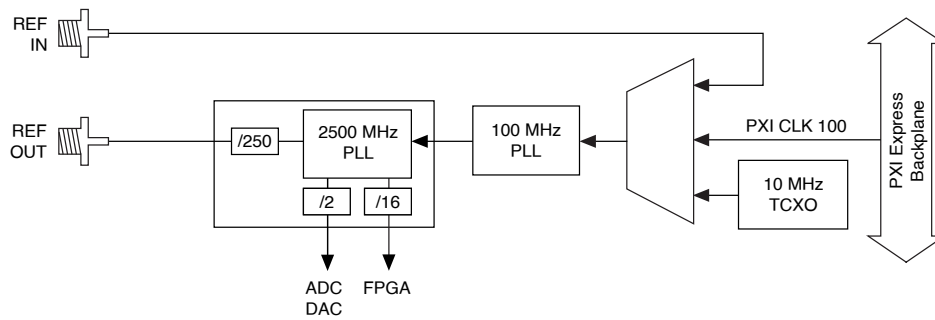
- The internal temperature compensated crystal oscillator (TCXO)
- The REF IN front panel connector
- PXI_CLOCK

While the Sample Clock frequency is fixed at 1.25 GHz, you can achieve high-resolution I/Q data rates using the Fractional Interpolator and Fractional Decimator DSP FPGA VIs.

Data Clock

The Sample Clock is divided by eight to create the Data Clock, which is sent to the FPGA.

The Data Clock runs at 156.25 MHz, and it is the main clock used for the acquisition and generation data paths inside the FPGA. Because the Data Clock is one-eighth the rate of the Sample Clock, eight samples are processed on each cycle of the Data Clock.



FPGA Clocks

The following table lists the clocks available in the FPGA. In addition to these clocks, LabVIEW FPGA allows for derived clocks at user-defined frequencies.

Name	Frequency (MHz)	Description
Data Clock	156.25	Main clock used for the acquisition and generation data paths in the FPGA.
Data Clock x2	312.5	In-phase with the Data Clock and used for DSP VIs.
Data Clock x4	625	In-phase with the Data Clock.
40 MHz Onboard Clock	40	Free-running 40 MHz oscillator.
200 MHz Onboard Clock	200	Free-running 200 MHz oscillator.
PXIe_CLK100	100	100 MHz clock from the backplane.

Front Panel Clocking Connectors

You can use the PXIe-5841 REF IN front panel connector to apply an external 10 MHz reference to the device.

ADC and DAC

The PXIe-5841 uses a dual-channel, 14-bit ADC and 16-bit DAC.

The ADC and DAC are clocked at 1.25 GS/s to provide up to 1 GHz of complex

bandwidth, and high-resolution I/Q data rates are achieved by using the Fractional Interpolator and Fractional Decimator DSP FPGA VIs. The ADC and DAC are automatically synchronized to the Data Clock domain inside the FPGA, which allows for interfacing to both the ADC and DAC in the same clock domain with full synchronization.

PFI 0 Connector

PFI 0 is a 3.3 V LVTTTL, bidirectional, general-purpose digital I/O signal.

The most common use of PFI 0 is as a trigger input or a marker/event output. However, because the PFI 0 I/O buffer is connected directly to the FPGA, you can program the PFI 0 functionality for custom applications using LabVIEW FPGA.

Refer to the **PXIe-5841 Specifications** document for information about acceptable characteristics for input and output signals for PFI 0.

DRAM

The PXIe-5841 has two banks of dynamic random-access memory (DRAM), which are independently accessible from the FPGA.

Refer to the **PXIe-5841 Specifications** document for information about the DRAM size and throughput. These DRAM banks are general purpose, but they are often used for storing waveforms to be generated or waveforms that have been acquired.



Note National Instruments recommends using LabVIEW FPGA memory items for most applications requiring the use of DRAM.

Configuring DRAM with FPGA Memory Items

Use the FPGA memory item interface to use DRAM in the same way that you use block memory and look-up tables (LUT). DRAM memory items appear in the Project Explorer window under the FPGA target. The FPGA memory item interface allows you to partition the physical DRAM banks into multiple memory items.

Complete the following steps to configure DRAM with FPGA memory items.

1. To create a target-scoped memory item, right-click the FPGA target in the Project Explorer window and select **New » Memory** from the shortcut menu.

The Memory Properties dialog box appears.

2. Select **DRAM** from the **Implementation** pull-down menu.
3. Select the DRAM bank you want to use from the **DRAM bank** pull-down menu.
4. Type in the number of elements based upon your desired memory size in the **Requested number of elements** textbox.
5. Click **OK** in the Memory Properties dialog box.

The memory item is now populated in the Project Explorer window under the target.

6. Use the memory item in an FPGA VI.

SRAM

The PXle-5841 has one bank of static random-access memory (SRAM), which is accessible from the FPGA.

Refer to the **PXle-5841 Specifications** document for information about the SRAM size and throughput.

SRAM Interface

The PXle-5841 supports SRAM access through socketed component-level IP (CLIP).

Refer to the **PXle-5841 Specifications** document to determine the available amount of onboard SRAM for your device.

Configuring SRAM with Socketed CLIP

Use the socketed CLIP interface to communicate directly with the onboard SRAM. Socketed CLIP lists all memory interfaces that are compatible with the SRAM. The SRAM CLIP is present in the LabVIEW project by default but is disabled.

Complete the following steps to configure SRAM with socketed CLIP:

1. Right-click the SRAM CLIP and select **Properties** from the shortcut menu to display the SRAM Properties dialog box.
2. Select **Enable Memory** if it is not already selected to display the SRAM configuration options.
3. Select the appropriate clock for your project in the Clock Selections dialog box.

You can use the SRAM CLIP in any single clock domain. You can select only clocks already added to the project.

SRAM Properties Dialog Box

Right-click the **SRAM Bank** item in the Project Explorer window and select **Properties** from the shortcut menu to display this dialog box. Check the **Enable SRAM** checkbox if it is not already selected to display the following pages in the Category list:

- General
- Clock Selections

General Page

Use the General page to configure the type of memory interface that should be used when communicating with external SRAM.

To display this page, in the SRAM Properties dialog box, select **General** from the **Category** list.

This page contains the following components:

- **Enable SRAM**—Enables the SRAM. Unchecking this box disables access to the SRAM.
- **Memory Interface**—Lists all memory interfaces that are compatible with the SRAM. If multiple versions of a memory interface are available, the version information displays next to the memory interface name.
- **Details**—Displays general information about the SRAM memory interface.
- **Path**—Displays the file system path to the XML file for the currently selected memory interface file.
- **Reload**—Reloads the currently selected memory interface in the table. Use the

Reload button if you modify a memory interface XML file on disk after you configure it for use with your FPGA target. Reload updates the I/O in the LabVIEW project and details information, but changes may not be visible in the Memory Interface or Path dialog boxes.

The PXle-5841 ships with one memory interface support option, which provides access to the external SRAM memory.

Clock Selections Page

Use the Clock Selections page to link each clock port defined by the component-level IP (CLIP) to a clock on the FPGA target. You must add the FPGA clock to the LabVIEW project before you can link to the FPGA clock.

To display this page, in the SRAM Properties dialog box, select **Clock Selections** from the **Category** list.

This page includes the following components:

- **Component-Level IP Clock**—Lists clock(s) defined in the CLIP declaration XML file.
- **Connection**—Lists clocks available on the FPGA target.

Power and Thermal Monitoring and Shutdown Conditions

The PXle-5841 is protected against excessive temperatures and power consumption and shuts down in the presence of excessive heat or power consumption. When the module shuts down, it automatically loads a low-power, nearly empty FPGA image. Any user instrument driver session or FPGA I/O stops, and any program that tries to read or write to the module returns an error.

To recover from thermal shutdown, resolve the excessive temperature and restart the system, or reset the device from MAX. Check that the ambient temperature around the chassis is within specifications and that the device is receiving proper airflow. Make sure that chassis fans are clean and that filler panels or slot blockers cover any empty slots to maximize cooling airflow. Position the chassis so that inlet and outlet vents are not obstructed.

To recover from an over-power shutdown, restart the system or reset the device from

MAX. Avoid running the application or FPGA VI that triggered the excess power consumption. Reducing the ambient temperature and improving cooling and airflow may also reduce power consumption.

Monitor temperature and power consumption in your application to detect shutdown conditions. To monitor how far the module is from the temperature limit, use the FPGA Temperature property or the `NIRFSA_ATTR_FPGA_TEMPERATURE` attribute. The thermal shutdown limit is 99 °C. To monitor how far the module is from the power limit, use the Module Power Consumption property or the `NIRFSA_ATTR_MODULE_POWER_CONSUMPTION` attribute. The power consumption shutdown limit is 125 W¹.



Note Ensure your application has a wide enough margin to allow for temperature and power variations between your development environment and deployment environment. The deployed application may run hotter or cooler than in it does in development due to hardware differences or variations in ambient temperature, air flow, chassis positioning, adjacent modules, power consumption, and other factors.

PCI Express Interface

The PXIe-5841 has a PCI Express (up to Gen3 x8) backplane connection, which is used for programmed I/O, DMA transfers, and peer-to-peer streaming.

DIGITAL I/O (DIO)

The PXIe-5841 DIGITAL I/O (DIO) connector supports eight parallel LVCMOS lines in addition to four lanes of high speed serial multi-gigabit transceivers (MGT), using a 42-pin Molex Nano-Pitch I/O connector. The eight LVCMOS lines support 1.2 V, 1.5 V, 1.8 V, 2.5 V, and 3.3 V voltage levels and can be used in a variety of applications including serial peripheral interface bus (SPI), inter-integrated circuit (I2C) bus, and digital triggering and events. You can control the lines directly from the instrument's onboard FPGA.

The PXIe-5841 DIGITAL I/O (DIO) can use DIO signals in a variety of applications

1. Based solely on processing power used by the PXIe-5841 rather than the total amount of power the module may consume.

including serial peripheral interface bus (SPI), inter-integrated circuit (I²C) bus, and digital triggering and events.

Signal	Type	Direction	Description
MGT Tx± <3..0>	Xilinx Virtex-7 GTH	Output	Dedicated MGT transmit differential pairs (AC-coupled).
MGT Rx± <3..0>			Dedicated MGT receive differential pairs (AC-coupled).
MGT REF±	Differential	Input	User reference clock source for Virtex-7 GTH banks. This clock is multiplexed with DIO <1..0> at the connector. The reference clock function and the DIO function of these pins cannot be used at the same time.
DIO <1..0>	Single-ended	Bidirectional	Single data rate (SDR) is default. Double data rate (DDR) can be achieved using user-supplied component-level IP (CLIP) or IP integration node. DIO <1..0> are multiplexed with a user-supplied MGT reference clock. The DIO and reference clock functions cannot be used at the same time.
DIO <7..2>			
5.0 V	DC	Output	Single-ended LVCMOS digital lines. 5.0 V power source, off by default.


Note Refer to the ***PXIe-5841 Specifications***

Signal	Type	Direction	Description
			document available at ni.com/manuals for the voltage and current capabilities.
GND	Ground	—	Connector ground.

Cables and Accessories

NI offers several cables and accessories that are compatible with the DIO connector.

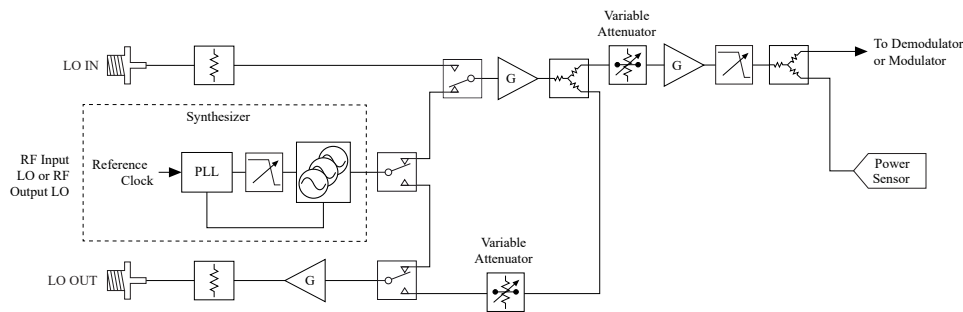


Note These cables and accessories use a custom pinout that matches the PXle-5841 DIO connector and maintains the 50 Ω transmission line environment. The use of other Molex Nano-Pitch I/O cables is not recommended.

Category	Part Number	Description
Cable	785486-01	OCuLink x4, 1 m cable
	786215-01	OCuLink x4 to mini-SAS HD, 1 m cable
Connector (cable required)	786335-01	SCB-8 Noise Rejecting, Shielded Molex Nano-Pitch I/O connector block

LO Path

The PXle-5841 has two tunable local oscillators (LO) for the RF input and RF output signal paths. The following high-level block diagram represents the common design of these two LO paths.



The LO path consists of the following individual blocks:

- Synthesizer
- Signal distribution and LO source selection
- LO out signal path

Synthesizer

The onboard synthesizer of the PXIe-5841 consists of integrated tunable voltage control oscillators (VCO), phase-locked loops (PLL), and RF dividers. The combined circuitry is capable of tuning from 9 kHz to 6,000 MHz.

Signal Distribution and LO Source Selection

The LO source is distributed to the downconverter or upconverter and the LO output signal path through the Signal Distribution. You can select the LO source as either the onboard synthesizer or the LO IN front panel connector. Refer to the External Local Oscillator section of the **Local Oscillators** topic for more information about using an external LO with the LO IN front panel connector.

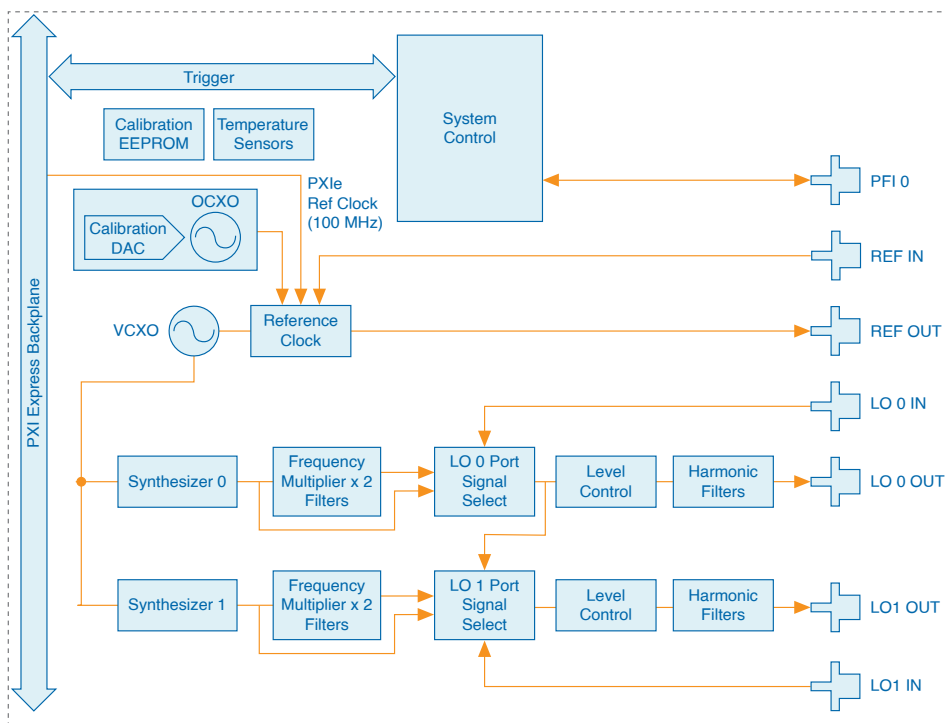
LO Out Signal Path

The PXIe-5841 supports the export of the LO signal to the LO OUT front panel connector. The LO output signal path can either export the LO directly from the synthesizer or through a programmable gain stage. Refer to the **Interconnecting Multiple PXIe-5841 RF Channels (Homogeneous Channel Types)** topic for more information on using the features of the LO output signal path.

PXIe-5655 Block Diagram

The PXIe-5655 extends the performance of the PXIe-5841 with an enhanced, dual local oscillator (LO).

The following block diagram represents a simplified view of the main functional blocks within the PXIe-5655.



VST State Models

At the lowest level, NI VSTs use two separate instrument drivers—NI-RFSA and NI-RFSG—to control the signal analysis and generation functionality, respectively, of the instrument. The underlying **state model** of each driver informs how you should construct programs to make use of your VST.

The state model determines what actions you can program the instrument to take in each state and tells you how to progress through the states when programming your instrument to build a successful program.

NI-RFSA State Model

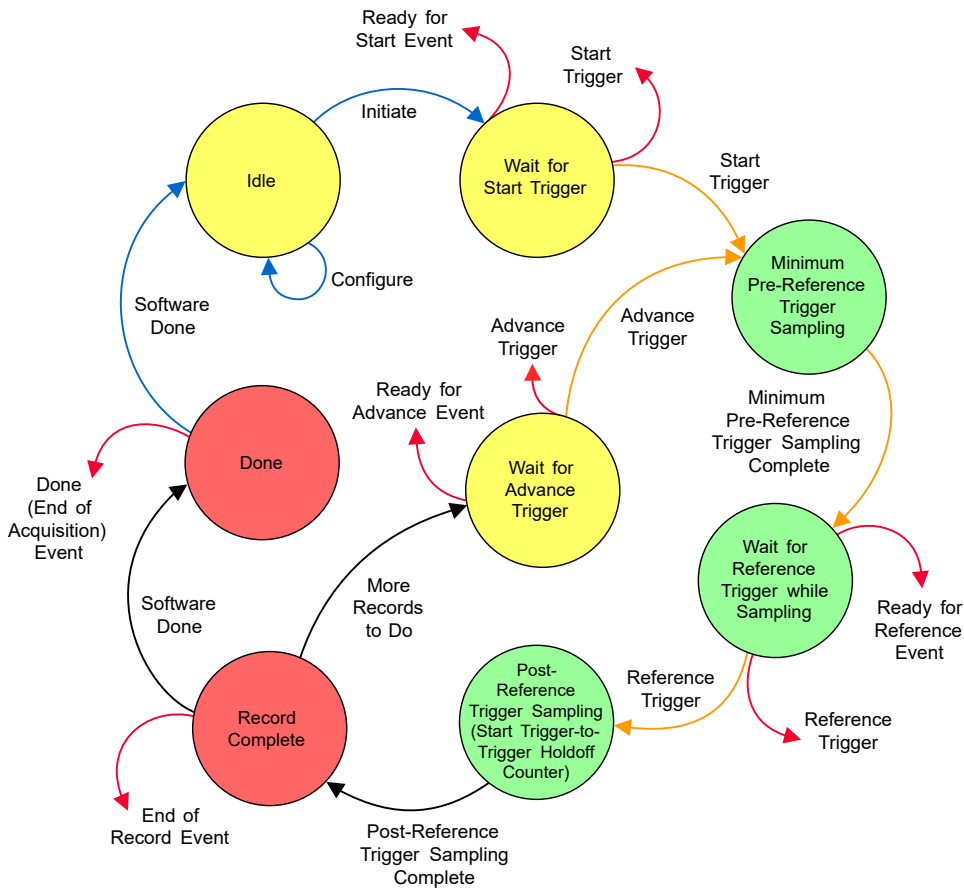
The NI-RFSA programming model for vector signal analyzers and the analysis functionality of vector signal analyzers has eight main states. The state of the instrument at a given time affects the actions you can take during that time.

These states are as follows:

- ***Idle***
- ***Wait for Start Trigger***
- ***Minimum Pre-Reference Trigger Sampling***
- ***Wait for Reference Trigger while Sampling***
- ***Post-Reference Trigger Sampling***
- ***Record Complete***
- ***Wait for Advance Trigger***
- ***Done***

The following diagram depicts the state model for the PXle-5841 when programmed with NI-RFSA software. This state diagram models the instrument when it acquires data in the I/Q mode.

Figure 2. NI-RFSA State Model



Arrow legend:

- **Blue**—State transitions always caused by software
- **Black**—State transitions caused by the internal state machine of the instrument
- **Red**—Output signals
- **Orange**—User-configurable state transitions caused by software or hardware

The major states are defined as follows.

- **Idle**—The instrument is not sampling a waveform. All the session properties are programmable in this state. In this state, the properties have not necessarily been applied to hardware yet, so the hardware configuration of the module may not match values of the properties in the session. In addition, the instrument remains configured as it was the last time a session was committed. When you call Initiate, all the properties are programmed to the hardware. If you recently reset the computer, the instrument is in the Idle state.
- **Wait for Start Trigger**—The instrument transitions to this state when you initiate

an acquisition. If the Start Trigger source is configured as None, the instrument immediately transitions from this state and generates a Start Trigger. If you configure the Start Trigger source as a software or hardware trigger from one of the available sources, the instrument remains in this state until the configured trigger occurs. When the instrument recognizes a trigger condition, it transitions from this state on the next clock cycle and generates a Start Trigger. The default Start Trigger source is None.

- **Minimum Pre-Reference Trigger Sampling**—The instrument can transition into this state two ways: receiving the Start Trigger from the Start Trigger source or receiving the Advance trigger from the Advance trigger source. The transition into this state depends on the previous state of the instrument. While in this state, the instrument samples according to the session attributes configured. The module remains in this state until three conditions are satisfied: the minimum Pre-Reference trigger sampling completes, the time-to-digital converter (TDC) is ready, and the trigger-to-trigger delay has expired. The first time through this state, the trigger-to-trigger delay does not have an effect. When the three conditions are satisfied, the instrument transitions from this state on the next clock cycle. Use the Pretrigger Samples property to specify the number of samples to be acquired before the Reference Trigger is received.
- **Wait for Reference Trigger while Sampling**—The instrument transitions into this state after the instrument finishes the Minimum Pre-Reference Trigger Sampling state. If you configure the Reference Trigger source as a software or hardware trigger from one of the available sources, the instrument remains in this state until the configured trigger occurs. When the instrument recognizes a trigger condition, the instrument transitions from this state. The default Reference Trigger source is None.
- **Post-Reference Trigger Sampling**—After the instrument receives the Reference Trigger, the instrument transitions into this state. The instrument begins sampling Post-Reference Trigger samples according to the properties configured for the session. When the Post-Reference Trigger sampling completes, the instrument transitions from this state.
- **Record Complete**—After the instrument completes Post-Reference trigger sampling state, the instrument transitions into this state. The instrument leaves this state after the current record has been stored in the onboard memory. Upon leaving this state, the instrument generates an End of Record Event.
- **Wait for Advance Trigger**—After the instrument has completed a record and determines that there are still more records to complete, the instrument

transitions into this state. If you configure the Advance Trigger source as None, the instrument transitions from this state on the next clock edge. If you configure the Advance Trigger source as a software or hardware trigger from one of the available sources, the instrument remains in this state until the configured trigger occurs. When the instrument recognizes a trigger condition, the instrument transitions from this state. The default Advance Trigger source is None.

- **Done**—After the instrument completes a record and determines that all the records are done, it transitions into this temporary state. Upon entering this state, the instrument generates the End of Acquisition Event. The software transitions the instrument from this state and back to the Idle state when you call either a Fetch function or NI-RFSA Check Acquisition Status.

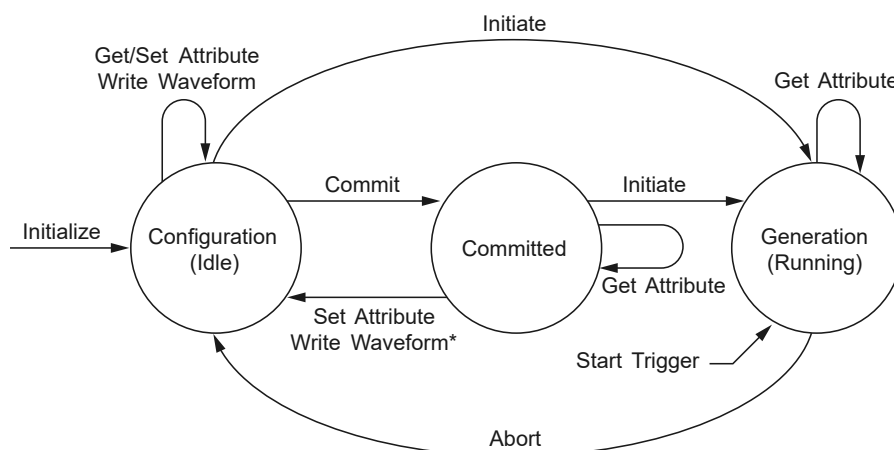
NI-RFSG State Model

The NI-RFSG programming model for vector signal generators and the generation functionality of vector signal transceivers has three main states. The state of the instrument at a given time affects the actions you can take during that time.

These states are **Configuration** (idle), **Committed**, and **Generation** (running).

The following diagram depicts the state model for the PXle-5841 when programmed with NI-RFSG software.

Figure 3. NI-RFSG State Model



The major states are defined as follows.

- **Configuration**—You can program all session properties in the Configuration state; however, properties are not yet applied to the instrument in this state. Therefore, the actual instrument configuration may not match the values of the properties in the session. The NI-RFSG instrument does not generate a signal in the Configuration state.



Note When you read a property, NI-RFSG may analyze many properties in the current configuration in order to return the coerced value for that property or attribute. In general, avoid reading any properties until you have finished changing all the properties you would like to change. Otherwise, you may encounter errors if a property is read while the configuration is in an inconsistent state.

- **Committed**—Calling NI-RFSG Commit from the Configuration state performs the following actions:
 - Verifies the settings of all properties
 - Validates the specified configuration
 - Writes all settings to the hardware modules
 - Writes the waveform to the instrument onboard memory
 - Transitions to the Committed state

If any properties are changed while in the Committed state, the session implicitly transitions back to the Configuration state and the hardware configuration reflects the previously committed properties or attributes.

- **Generation**—The instrument is either waiting on a trigger or is generating a signal. In the Generation state, session properties always reflect the current state of the instrument.



Note You can call NI-RFSG Close from any state. Calling Close stops signal generation.

PXle-5841 Front Panel and LEDs

Front Panel and LEDs

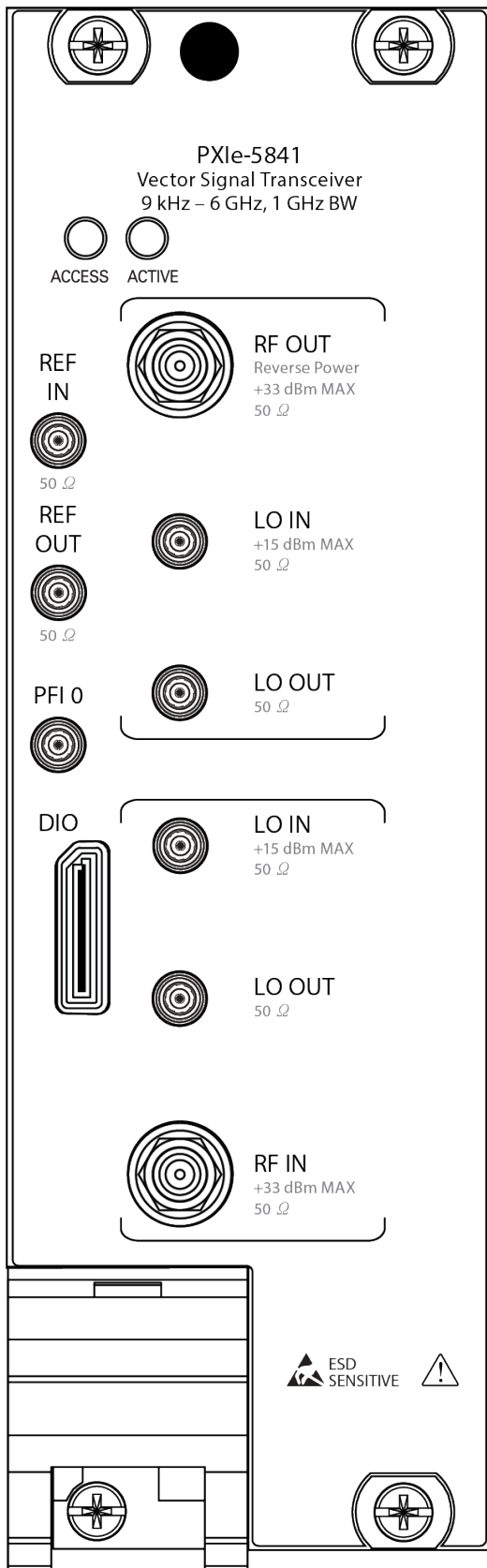


Table 6. General Connector Descriptions

Connector	Use	Connector Type
REF IN	Input terminal that allows for the use of an external 10 MHz Reference Clock.	MMPX (f)
REF OUT	Output terminal that can export a 10 MHz Reference Clock.	MMPX (f)
PFI 0	Programmable-function digital I/O (DIO) connector for use with triggers or events.	MMPX (f)
DIO	Multi-signal DIO connector that provides access to FPGA multi-gigabit transceivers (MGTS) and general purpose LVCMOS signals.	Nano-Pitch



Notice The DIO connector is not an OCUlink interface. Do not connect the DIO connector on the PXle-5841 to the OCUlink interface of another device. NI is not liable for any damage resulting from such signal connections.

Table 7. RF Connector Descriptions

Connector		Use	Connector Type
RF OUT	RF OUT	Output terminal for RF signals.	SMA (f)
	LO IN	Input terminal that allows for the use of an external local oscillator (LO) for RF OUT.	MMPX (f)
	LO OUT	Output terminal for	MMPX (f)

Connector		Use	Connector Type
		exporting the RF OUT LO source.	
RF IN	LO IN	Input terminal that allows the use of an external LO for RF IN.	MMPX (f)
	LO OUT	Output terminal for exporting the RF IN LO source.	MMPX (f)
	RF OUT	Input terminal for RF signals.	SMA (f)

Table 8. LED Indicators

LED	Indications
ACCESS	Indicates the basic hardware status of the device. Off—The device is not yet functional or has detected a problem with a PXI Express power rail. Amber—The device is being accessed. Accessed means that you are writing to the device setup

LED	Indications
	registers to control the device, reading from the device to monitor the device status, or transferring data to/from the device. Green—The device is controllable through the software.
ACTIVE	Off—The device is idle. Solid green—The device is generating a waveform. Dim amber—The device is waiting for an acquisition Reference Trigger. Solid amber—The device is acquiring a waveform. Solid red—The device has detected an error. The LED remains red until the error condition is removed.  Note The indicators are listed in increasing order of priority. For example, if you are generating a waveform using NI-RFSG and waiting on an acquisition Reference Trigger in NI-RFSA, the LED is dim amber.

PXIe-5841 Pinout

Use the pinout to connect to terminals on the PXIe-5841.

Figure 4. PXle-5841 Digital I/O Connector Pinout

Reserved	A1	B1	5.0 V
GND	A2	B2	GND
MGT Rx+ 0	A3	B3	MGT Tx+ 0
MGT Rx– 0	A4	B4	MGT Tx– 0
GND	A5	B5	GND
MGT Rx+ 1	A6	B6	MGT Tx+ 1
MGT Rx– 1	A7	B7	MGT Tx– 1
GND	A8	B8	GND
DIO 4	A9	B9	DIO 6
DIO 5	A10	B10	DIO 7
GND	A11	B11	GND
MGT REF+ / DIO 0	A12	B12	DIO 2
MGT REF– / DIO 1	A13	B13	DIO 3
GND	A14	B14	GND
MGT Rx+ 2	A15	B15	MGT Tx+ 2
MGT Rx– 2	A16	B16	MGT Tx– 2
GND	A17	B17	GND
MGT Rx+ 3	A18	B18	MGT Tx+ 3
MGT Rx– 3	A19	B19	MGT Tx– 3
GND	A20	B20	GND
5.0 V	A21	B21	Reserved

Table 9. PXle-5841 Digital I/O Signal Descriptions

Signal	Type	Direction
MGT Tx± <3..0>	Xilinx Virtex-7 GTH	Output
MGT Rx± <3..0>	Xilinx Virtex-7 GTH	Input
MGT REF±	Differential	Input
DIO <1..0>	Single-ended	Bidirectional
DIO <7..2>	Single-ended	Bidirectional
5.0 V	DC	Output
GND	Ground	—



Note DIO <1..0> pins are multiplexed with MGT REF±.

PXIe-5655 Front Panel and LEDs

Figure 5. PXIe-5655 with MMPX Front Panel

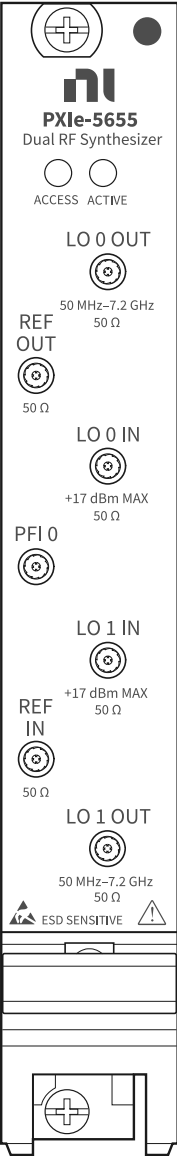


Table 10. PXIe-5655 Connector Descriptions

Connector	Use	Connector Type
REF OUT	Output terminal that can export a 10 MHz Reference Clock.	MMPX (f)
PFI 0	This is reserved and not enabled for use with the PXIe-5841.	—
REF IN	Input terminal that allows the use of an external	MMPX (f)

Connector	Use	Connector Type
	Reference Clock with one of the following frequencies: • 10 MHz • 100 MHz • 270 MHz • $3.84 \text{ MHz} \times y$, where y is 4, 8, 16, 24, 25, or 32	
LO 0 OUT	Output terminal for the LO source. This terminal has no signal at power on and reset.	MMPX (f)
LO 0 IN	Input terminal that allows the use of an external local oscillator.	MMPX (f)
LO 1 IN	Input terminal that allows the use of an external local oscillator.	MMPX (f)
LO 1 OUT	Output terminal for the LO source. This terminal has no signal at power on and reset.	MMPX (f)

Table 11. PXIe-5655 LED Indicators

LED	Indications
ACCESS	Indicates the basic hardware status of the module. • Off—The module is not yet functional or has detected a problem with a PXI Express power rail. • Amber—The module is being accessed. Accessed means that you are writing to the device setup registers to control the device, reading from the device to monitor the device status, or transferring data to or from the device. • Green—The module is controllable through the software.
ACTIVE	• Off—The module is idle or locked to the requested reference. • Solid red—The PLL is unlocked or the

LED	Indications
	module temperature is out of the specified operating range.

PXIe-5841 Installation and Configuration

Complete the following steps to install the PXIe-5841 and prepare it for use.

1. [Preparing the Environment](#)

Ensure that the environment you are using the PXIe-5841 in meets the following specifications.

2. [Unpacking the Kit](#)

3. [Verifying the Kit Contents](#)

4. [PXIe-5841 Mounting Guidelines](#)

To ensure optimal thermal performance, follow the placement, clearance, and mounting guidelines for the product platform.

5. [Installing the Software](#)

6. [Installing the PXIe-5841](#)

7. [Configuring the Hardware in MAX](#)

You can use Measurement & Automation Explorer (MAX) to configure your NI hardware. MAX is automatically installed with NI-RFSA and NI-RFSG.

8. [Configuring the PXIe-5841 in Hardware Configuration Utility](#)

NI recommends using Hardware Configuration Utility to perform initial hardware configuration.

Preparing the Environment

Ensure that the environment you are using the PXIe-5841 in meets the following specifications.

Temperature and humidity	
Operating ambient temperature	
Chassis with slot cooling capacity $\geq 58 \text{ W}^2$	0 °C to 55 °C
Chassis with slot cooling capacity $\geq 38 \text{ W}^3$	0 °C to 40 °C

2. Not all chassis with a slot cooling capacity $\geq 58 \text{ W}$ can achieve this ambient temperature range. Refer

Operating relative humidity	10% to 90%, noncondensing
Maximum altitude	2,000 m (800 mbar) (at 25 °C ambient temperature)
Pollution Degree	2

Indoor use only.



Notice Clean the hardware with a soft, nonmetallic brush or lint free cloth. Make sure that the hardware is completely dry and free from contaminants before returning it to service.



Note Refer to the *PXIe-5841 Specifications* for complete specifications.

Unpacking the Kit



Notice To prevent electrostatic discharge (ESD) from damaging the device, ground yourself using a grounding strap or by holding a grounded object, such as your computer chassis.

1. Touch the antistatic package to a metal part of the computer chassis.
2. Remove the device from the package and inspect the device for loose components or any other sign of damage.



Notice Never touch the exposed pins of connectors.



Note Do not install a device if it appears damaged in any way.

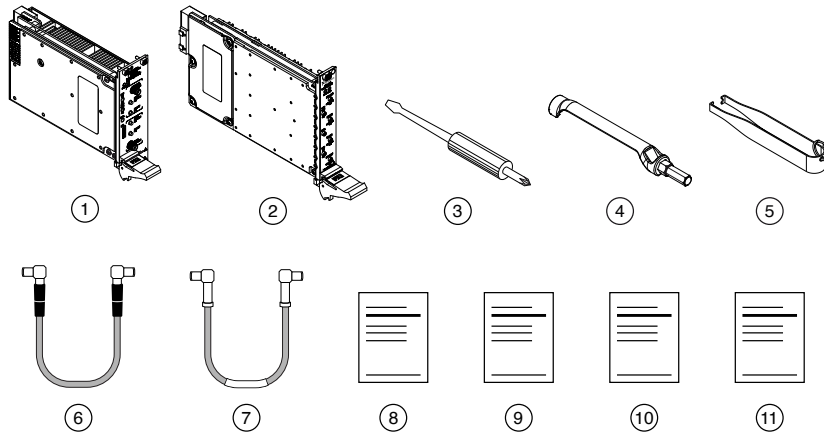
3. Unpack any other items and documentation from the kit.
to PXI chassis specifications to determine the ambient temperature ranges your chassis can achieve.
3. For chassis with a slot cooling capacity of 38 W, the fan speed must be set to HIGH for use in ambient temperature range.



Note Store the device in the antistatic package when the device is not in use.

Verifying the Kit Contents

Figure 6. PXle-5841 Kit Contents



1. PXle-5841 Module
2. PXle-5655 Module (Included in Kit Number: 785832-01)
3. Screwdriver, Part Number 772006-01
4. SMA Driver Bit, Part Number 190487B-01
5. Tweezers, Part Number 145455B-01
6. MMPX (m)-to-MMPX (m) flexible cables (x2), Length 47 mm (Included in Kit Number: 785832-01)
7. MMCX (m)-to-MMCX (m) flexible cable, Length 56 mm
8. PXle-5841 Safety, Environmental, and Regulatory Information
9. PXle-5655 Safety, Environmental, and Regulatory Information (Included in Kit Number: 785832-01)
10. Maintain Forced-Air Cooling Note to Users
11. PXle-5841 Getting Started Guide (This Document)

Other Equipment

There are several required items not included in your device kit that you need to operate the PXle-5841. Your application may require additional items not included in your kit to install or operate your PXle-5841.

Required Items

- A PXI Express chassis and chassis documentation
- A PXI Express embedded controller or PC with MXI controller system that meets the system requirements specified in this guide and chassis documentation

Optional Items

- PXI Chassis Slot Blocker kit (Part number 199198-01)
- SMA (m)-to-SMA (m) cables
- SMA (m)-to-MMPX (m) cables
- Additional MMPX (m)-to-MMPX (m) cables
- OCuLink x4, 1 m cable (Part number 785486-01)⁴
- SCB-8 Noise Rejecting, Shielded Nano-Pitch Connector Block (Part number 786335-01)

Visit ni.com for more information about these additional items.

PXIe-5841 Mounting Guidelines

To ensure optimal thermal performance, follow the placement, clearance, and mounting guidelines for the product platform.



Notice Failure to follow the mounting instructions in the product documentation can cause temperature derating.



Notice This product is intended for use in indoor applications only.

To understand the placement requirements of the chassis, such as required clearances, refer to the user manual for your PXI chassis. For environmental requirements for storage and operation, refer to the chassis specifications.



Note The environmental requirements of modules within the chassis may differ from those of the chassis. Similarly, the requirements of the installed

4. NI does not recommend using an OCuLink x4 cable supplied by another manufacturer.

modules may differ from each other, and any devices connected to the modules within the chassis may have different requirements of their own. Consult the relevant documentation for all equipment within and connected to the chassis to understand the aggregate mounting and environmental requirements of the system.

Chassis placement must also facilitate making a safety ground connection, if the user manual for your chassis indicates this is required.

To understand more specific aspects of rack mounting the PXI chassis, such as installation procedures, refer to the documentation included with the rack mount kit for your chassis or available on ni.com.

Installing the Software

You must be an Administrator to install NI software on your computer.

1. Install an ADE, such as LabVIEW or LabWindows™/CVI™.
2. Download the driver software installer from ni.com/downloads.
Package Manager downloads with the driver software to handle the installation. Refer to the [Package Manager Manual](#) for more information about installing, removing, and upgrading NI software using Package Manager.
3. Follow the instructions in the installation prompts.



Note Windows users may see access and security messages during installation. Accept the prompts to complete the installation.

4. When the installer completes, select **Restart** in the dialog box that prompts you to restart, shut down, or restart later.

Installing the PXIe-5841



Notice To prevent damage to the PXIe-5841 caused by ESD or contamination, handle the module using the edges or the metal bracket.



Note

You must install the software before installing the hardware.

Before you install the hardware, refer to ***Maintaining Forced-Air Cooling*** to ensure that the device can cool itself effectively.

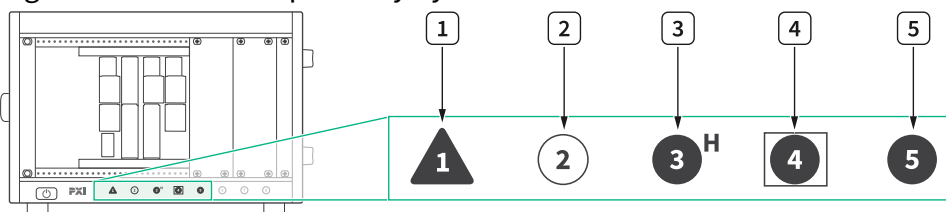
1. Ensure the AC power source is connected to the chassis before installing the module.
The AC power cord grounds the chassis and protects it from electrical damage while you install the module.
2. Power off the chassis.
3. Inspect the slot pins on the chassis backplane for any bends or damage prior to installation. Do not install a module if the backplane is damaged.
4. Ensure the chassis fan settings meet the requirements in the ***Preparing the Environment*** section.



Note Inadequate air circulation could cause the temperature inside the chassis to rise above the optimal operating temperature for the device, potentially causing thermal shutdown, shorter lifespans, or improper performance.

5. Position the chassis so that inlet and outlet vents are not obstructed.
6. Remove the black plastic covers from all the captive screws on the module front panel.
7. Identify two adjacent PXI Express-compatible slots in the chassis. The following figure shows the symbols that indicate the slot types.

Figure 7. Chassis Compatibility Symbols



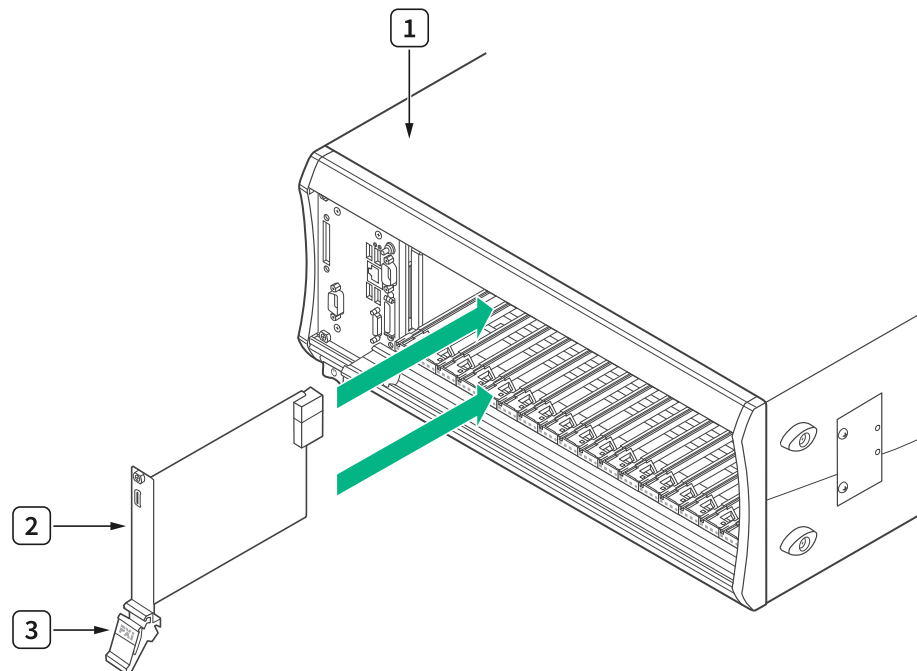
1. PXI Express System Controller Slot
2. PXI Peripheral Slot
3. PXI Express Hybrid Peripheral Slot

4. PXI Express System Timing Slot
5. PXI Express Peripheral Slot

The PXIe-5841 can be placed in PXI Express peripheral slots, PXI Express Hybrid peripheral slots, or PXI Express system timing slots.

8. Touch any metal part of the chassis to discharge static electricity.
9. Ensure that the ejector handle is in the downward (unlatched) position.
10. Hold the module by the edges and slide it into the empty compatible slots. Ensure the base engages with the guides in the chassis.

Figure 8. PXI Express Module Installation



1. PXI Express Chassis
2. PXI Express Module
3. Ejector Handle in Down Position

11. Latch the module in place by pulling up on the ejector handle.
12. Secure the module front panel to the chassis using the front-panel mounting screws.



Note Tightening the top and bottom mounting screws increases mechanical stability and also electrically connects the front panel to the chassis, which can improve the signal quality and electromagnetic performance.

13. Use slot blockers and cover all empty slots with filler panels to maximize cooling air flow.



Note Proper airflow is critical to device operation.

14. Power on the chassis.

Related information:

- [Maintaining Forced-Air Cooling](#)

Direct Connections to the PXle-5841

The PXle-5841 is a precision RF instrument that is sensitive to ESD and transients. Ensure you take the following precautions when making direct connections to the PXle-5841 to avoid damaging the device.



Notice Apply external signals only while the PXle-5841 is powered on. Applying external signals while the device is powered off may cause damage.

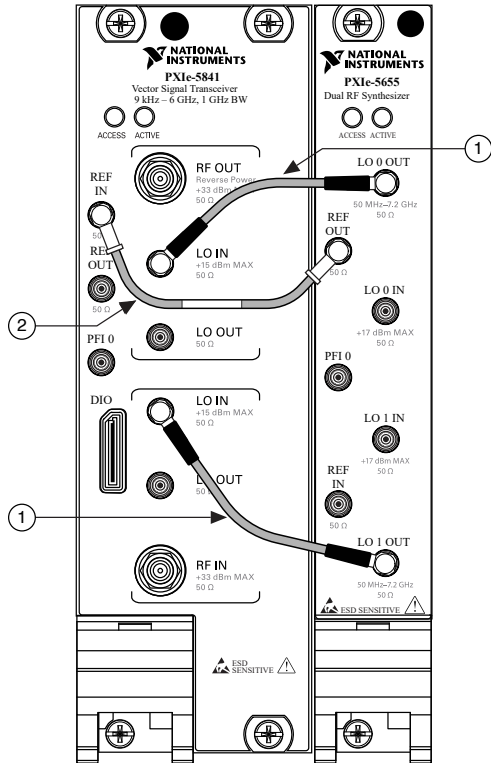
- Ensure you are properly grounded when manipulating cables or antennas connected to the PXle-5841RF IN connector.
- If you are using nonisolated devices, such as a nonisolated RF antenna, ensure the devices are maintained in a static-free environment.
- If you are using an active device, such as a preamplifier or switch routed to the PXle-5841RF IN connector, ensure that the device cannot generate signal transients greater than the RF and DC specifications of the PXle-5841RF IN connector.

Connecting the PXle-5655 Analog Signal Generator to the PXle-5841

If you are using the PXle-5655 with the PXle-5841, complete the following steps to connect the PXle-5655 as an LO source.

Skip this section if your configuration does not include the PXle-5655.

Figure 9. PXIe-5841 with PXIe-5655 Setup



1. MMPX (m)-to-MMPX (m) Cables
2. MMCX (m)-to-MMCX (m) Cable

1. Install the PXIe-5655 in the slot directly to the right of the PXIe-5841.
2. Using a flexible 47 mm MMPX (m)-to-MMPX (m) cable, connect the RF OUT LO IN connector on the PXIe-5841 to the LO 0 OUT connector on the PXIe-5655.
3. Using a flexible 56 mm MMCX (m)-to-MMCX (m) cable, connect the REF IN connector on the PXIe-5841 to the REF OUT connector on the PXIe-5655.
4. Using a flexible 47 mm MMPX (m)-to-MMPX (m) cable, connect the RF IN LO IN connector on the PXIe-5841 to the LO 1 OUT connector on the PXIe-5655.

Configuring the Hardware in MAX

You can use Measurement & Automation Explorer (MAX) to configure your NI hardware. MAX is automatically installed with NI-RFSA and NI-RFSG.

1. Launch MAX.
2. In the configuration tree, expand **Devices and Interfaces** to see the list of installed NI hardware.

MAX lists all installed modules under the name of their associated chassis.

3. Expand your **Chassis** tree item.

PXIe-5841 devices appear as NI-RIO modules in the list. The default names of your device may vary.



Note If your module is not listed, refer to the **Troubleshooting** section of this document.

4. Record the identifier MAX assigns to the hardware. Use this identifier when programming the PXIe-5841.



Notice When you install, uninstall, or move an NI-RIO module in your system, resource identification of your NI-RIO devices may change. Whenever any of these changes occur, verify resource identification of all your NI-RIO modules in MAX, and, if necessary, make changes to your software and documentation.

5. (Optional) Associate the hardware modules that comprise your instrument.

- Select the PXIe-5841 you want to associate.
- If you are using the PXIe-5655 as an LO source, open the Associated Devices section, and select the PXIe-5655 from both the **RF In LO** list for use with RF IN and the **RF Out LO** list for use with RF OUT. The values you select from the **RF In LO** and **RF Out LO** lists should be the same.

6. Click **Save** in the MAX toolbar.



Note For detailed information about MAX, refer to the Measurement & Automation Explorer Help by selecting **Help » MAX Help**.



Note Module associations may be lost when you move the modules to different chassis slots.

Related tasks:

- [What Should I Do if My Hardware Doesn't Appear in Hardware Configuration Utility or MAX?](#)

Configuring the PXle-5841 in Hardware Configuration Utility

NI recommends using Hardware Configuration Utility to perform initial hardware configuration.

1. Open Hardware Configuration Utility.
The PXle-5841 appears in the system pane automatically.
If the PXle-5841 does not appear automatically, do the following:
 - a. Click the + button or, in the menu, select **Edit » Add hardware....**
 - b. Find the modules of your instrument in the Discovered pane and click **Add**.
2. Record the name Hardware Configuration Utility assigns to the PXle-5841 or, if desired, provide a custom name to the PXle-5841.
Use this name when programming the PXle-5841.
3. **Optional:** Associate the hardware modules that compose your instrument.



Tip You can expand **Hardware associations** in the configuration pane and click **Available associations** to see illustrations of the valid configurations for PXle-5841 instruments.

- a. In the system pane, select the PXle-5841 module to associate.
 - b. In the **Hardware associations** section of the configuration pane, click either **RF In LO** or **RF Out LO** and choose the PXle-5655 to associate with the PXle-5841 module.
4. Validate that your instrument is installed and connected correctly by selecting the PXle-5841 module of your instrument in the system pane and choosing either of the following:

Option	Description
Verify associations	Expand the Hardware associations area of the configuration pane and click Verify associations .



Note Verify associations performs an intermediate self-test.

Option	Description
Self-test	- Expand the Troubleshooting area of the configuration pane and click Self-test. - Select Intermediate and then click Ok to run the test.

Hardware Configuration Utility reports when the hardware setup is validated.

Related tasks:

- [What Should I Do if My Hardware Doesn't Appear in Hardware Configuration Utility or MAX?](#)

PXIe-5841 Examples

NI installs example code with your software or driver that demonstrates the functionality of PXIe-5841, helps you advance your knowledge, and provides building blocks for your own applications.

While the experience can differ slightly across products and versions, examples generally install with your NI product or driver and can be accessed from within NI software.

RF API Locations

Access the API to program your instrument, depending on your application development environment and RF driver.

Table 12. API Location for RFmx

ADE	API Location
LabVIEW	Available on the LabVIEW Functions palette at Measurement I/O » NI-RFmx .
LabWindows/CVI	Available from the LabWindows/CVI Library menu at Library » RFmx SpecAn Library and Library » RFmx Demod Library .
Microsoft Visual C/C++	Use the header files located in the <NIDir>\Shared\ExternalCompilerSupport\C\include directory and import library files located in one of the following directories: • 32-bit installation—<NIDir>\Shared\ExternalCompilerSupport\C\lib32\msvc • 64-bit installation—<NIDir>\Shared\

ADE	API Location
	ExternalCompilerSupport\C\lib64\msvc where <NIDir> is the following location: Program Files (x86)\National Instruments.
Microsoft .NET	For the location of .NET class libraries, refer to the installed RFmx readme.

Table 13. API Location for NI-RFSA

ADE	API Location
LabVIEW	Available on the LabVIEW Functions palette at Measurement I/O » NI-RFSA .
LabWindows/CVI	Depending on bitness, available in the <IVIROOTDIR32>\Drivers\niRFSA or <IVIROOTDIR64>\Drivers\niRFSA directory, where <IVIROOTDIR32> and <IVIROOTDIR64> are the following: • <IVIROOTDIR32>—Program Files (x86)\IVI Foundation\IVI • <IVIROOTDIR64>—Program Files\IVI Foundation\IVI
Microsoft Visual C/C++	Depending on bitness, use the header files located in the <IVIROOTDIR32>\Include or <IVIROOTDIR64>\Include directory and import library files located in the <IVIROOTDIR32>\Lib or <IVIROOTDIR64>\Lib directory, where <IVIROOTDIR32> and <IVIROOTDIR32> are the following: • <IVIROOTDIR32>—Program Files (x86)\IVI Foundation\IVI • <IVIROOTDIR64>—Program Files\IVI Foundation\IVI

ADE	API Location
Microsoft .NET	Use the NI-RFSA .NET class library by adding a reference to <code>NationalInstruments.ModularInstruments.NIRfsa.Fx40</code> or <code>NationalInstruments.ModularInstruments.NIRfsa.Fx45</code> and any dependent class libraries from within the Solution Explorer in Visual Studio.

Table 14. API Location for NI-RFSG

ADE	API Location
LabVIEW	Available on the LabVIEW Functions palette at Measurement I/O» NI-RFSG .
LabWindows/CVI	Depending on bitness, available in the <code><IVIROOTDIR32>\Drivers\niRFSG</code> or <code><IVIROOTDIR64>\Drivers\niRFSG</code> directory, where <code><IVIROOTDIR32></code> and <code><IVIROOTDIR64></code> are the following: • <IVIROOTDIR32>—Program Files (x86)\IVI Foundation\IVI • <IVIROOTDIR64>—Program Files\IVI Foundation\IVI
Microsoft Visual C/C++	Depending on bitness, use the header files located in the <code><IVIROOTDIR32>\Include</code> or <code><IVIROOTDIR64>\Include</code> directory and import library files located in the <code><IVIROOTDIR32>\Lib</code> or <code><IVIROOTDIR64>\Lib</code> directory, where <code><IVIROOTDIR32></code> and <code><IVIROOTDIR64></code> are the following: • <IVIROOTDIR32>—Program Files (x86)\IVI Foundation\IVI • <IVIROOTDIR64>—Program Files\IVI Foundation\IVI

ADE	API Location
Microsoft .NET	Use the NI-RFSG .NET class library by adding a reference to <code>NationalInstruments.ModularInstruments.NIRfsg.Fx40</code> or <code>NationalInstruments.ModularInstruments.NIRfsg.Fx45</code> and any dependent class libraries from within the Solution Explorer in Visual Studio.

Browsing and Searching for Examples in NI Example Finder

Use NI Example Finder to browse and to search for examples.

You can use NI Example Finder to find examples for the following products.

- LabVIEW
- LabWindows/CVI
- NI drivers accessible from LabVIEW
- NI drivers accessible from LabWindows/CVI

1. Launch LabVIEW or LabWindows/CVI.
2. Open NI Example Finder.

Option	Description
LabVIEW	Select Help » Find Examples. from the menu bar.
LabWindows/CVI	Click Find Examples... from the Examples section of the Welcome Page.

NI Example Finder launches.

3. **Optional:** Configure NI Example Finder for LabWindows/CVI.
 - a. Click **Setup**. Configure Example Finder opens.
 - b. In Configure Example Finder, click **Software**, then select LabWindows/CVI, and click **OK**.

NI Example Finder updates with all the examples for LabWindows/CVI.

4. Search the example VIs for your product.

Option	Description
Click the Browse tab.	Choose Browse when you want to drill down through folders to find examples organized by task category.  Tip Examples installed with NI drivers or third-party drivers are often found within the Hardware Input and Output folder. Examples installed with toolkits or modules are often found within the Toolkits and Modules folder.
Click the Search tab.	Choose Search when you want to find examples by searching for topics, products, or modules relevant to your application.

5. To open an example, double-click the folder or the example.

 **Tip** You can modify an example VI to fit your application. You can also copy and paste from one or more examples into a VI that you create.

Finding Microsoft Visual Studio Examples

If you are developing VST applications with Microsoft Visual Studio, locate examples in the following directories.

Table 15. Location of Microsoft Visual Studio Programming Examples

Software Option	Examples Location
RFmx	<NIDocDir>\<Personality>\Examples where <NIDocDir> is the Users\Public\Public Documents\National Instruments directory and <Personality> is the directory of your RFmx

Software Option	Examples Location
	personality.
NI-RFSA	<NIDocDir>\NI-RFSA\examples where <NIDocDir> is the Users\Public\Public Documents\National Instruments directory.
NI-RFSG	<NIDocDir>\NI-RFSG\examples where <NIDocDir> is the Users\Public\Public Documents\National Instruments directory.

Making a First Measurement

You can verify proper installation and configuration of your device by making a measurement using the NI-RFSG and NI-RFSA SFPs.

1. Connect the RF IN port to the RF OUT port using an SMA (m)-to-SMA (m) cable.
2. Launch the NI-RFSG Soft Front Panel at **Start » National Instruments » NI-RFSG Soft Front Panel**.
3. Select the device identifier assigned to the PXIe-5841 in MAX in the **Device** drop-down menu.
4. Click the **RF On/Off** button to generate a 1 GHz CW tone RF signal at the RF output terminal of your device.
5. Launch the NI-RFSA Soft Front Panel with **Start » National Instruments » NI-RFSA Soft Front Panel**.
6. Select the device identifier assigned to the PXIe-5841 in MAX in the **Device** drop-down menu.
7. Verify that the NI-RFSA Soft Front Panel is receiving the CW tone.
8. Close the NI-RFSA Soft Front Panel and NI-RFSG Soft Front Panel.

Using the PXIe-5841

Understand how to use the PXIe-5841, including taking basic measurements, programming the PXIe-5841, and optimizing the performance of complex applications.

Analyzing and Generating RF Signals with NI VSTs

In broad terms, NI VSTs are designed to receive (analyze) and output (generate) RF signals. To do this, NI VSTs use separate instrument drivers—NI-RFSA, NI-RFSG—as well as RFmx and other purpose-built applications that enable specific measurement types.

The documentation in this manual focuses on programming your hardware using the NI-RFSA, NI-RFSG, and RFmx APIs.

You can use the following software to control your NI VST:

- **NI-RFSA**—Instrument driver that provides API-level control over RF signal analysis functionality
- **NI-RFSG**—Instrument driver that provides API-level control over RF signal generation functionality
- **RFmx**—Advanced measurement software for RF test systems that includes personalities and purpose-built applications. Each personality enables a specific family of RF measurements via a soft front panel and a programmatic API; the API allows you to access signal analysis functionality of the personality in test programs.

Each API has its own programming flow you need to understand to successfully control your VST and meet the needs of your RF application.

In addition to NI-RFSA, NI-RFSG, and RFmx, NI provides high-level applications to control RF instruments:

- **InstrumentStudio**—Interactive soft front panel application that, for NI RF instruments, surfaces the functionality of NI-RFSG and RFmx; also supports other instrument types

- **Application-specific RF software**—Purpose-built application environments for specific RF testing workflows, such as RFIC test

Refer to the documentation for these applications for information on using them with NI RF hardware.



Note Your instrument must have the appropriate hardware option—for generation, analysis, or both—to access associated functionality in your programs.

Analyzing RF Signals with RFmx and NI-RFSA

RFmx and NI-RFSA both allow you to develop programs to control the signal analysis functionality of NI VSAs and VSTs. NI recommends using the RFmx API for most signal analysis applications, but there are specific cases where using the NI-RFSA API is necessary to access specific functionality.

Use the following guidelines to determine if you should use NI-RFSA instead of RFmx.

Software Option	When to Use
RFmx	Recommended for all signal analysis applications unless you need specific functionality available only through NI-RFSA. Simplifies program development relative to NI-RFSA.
NI-RFSA	• Supporting existing NI-RFSA applications • If RFmx does not have the flexibility for triggering and events that your application requires • If your application makes use of NI-TClk synchronization • If your application makes use of peer-to-peer streaming • If you are writing custom demodulation algorithms • If your application requires the fastest

Software Option	When to Use
	access to raw I/Q data

Refer to the documentation for the signal analysis software you chose to develop a program with that software.

Analyzing RF Signals with RFmx

RFmx is a set of RF measurement personalities, each designed to enable a specific type of RF measurements. Each RFmx personality includes a soft front panel and the accompanying low-level APIs that give you access to the functionality of that personality in programs you develop.

This documentation describes using the RFmx API to develop signal analysis programs for your RF instrument.



Note While RFmx panels may make use of both signal analysis and signal generation functionality, the low-level RFmx APIs provide access to signal analysis functionality only. Use the NI-RFSG API to include signal generation functionality in programs you develop.

There are two general parts of an RFmx program:

- **RFmxInstr**—Set of entry points that create and end the RFmx session and that focus on configuring your hardware to the settings you want for your measurement; common to all RFmx personalities.
- **Measurement entry points**—API entry points specific to and named for each RFmx measurement personality; define the parameters of the measurement itself.

Each RFmx personality includes both RFmxInstr entry points and the measurement-specific entry points.



Note Each RFmx personality includes a wide array of entry points to control

your instrument and optimize its behavior for your application. This process describes only the basic programming flow to create an RF signal analysis program successfully.

1. Initialize a Session

Use RFmxInstr Initialize NIRFSA to create a session with the PXIe-5841. All RFmx programs must begin with an RFmxInstr function.

2. Configure Hardware Settings

Use RFmxInstr functions and properties to configure your hardware to use the correct settings for the measurement.

3. Configure the Measurement

Use the functions and properties specific to your RFmx personality to configure that measurement.

4. Close the Session

Every RFmx session must be closed in order to deallocate memory, destroy threads, and free operating system resources.

Initialize a Session

Use RFmxInstr Initialize NIRFSA to create a session with the PXIe-5841. All RFmx programs must begin with an RFmxInstr function.

Initializing returns an instrument handle with the session configured to a known state. Initialization can take a significant amount of time compared to other functions, so you should not include it in loops. Ideally, your program should call RFmxInstr Initialize NIRFSA one time.

1. Call RFmxInstr Initialize NIRFSA.



Tip For most RFmx personalities, you can include measurement functions from any number of personalities in a single session initialized with RFmxInstr Initialize NIRFSA; you do not need to create unique sessions for each personality. The exception is RFmx VNA, which cannot be used in a session that includes any other RFmx personalities.

2. **Optional:** Define an `option string`.

Setting `option string` to `AnalysisOnly` allows your program to perform

analysis of preexisting I/Q data or spectrum. Use this option to analyze RF data without using hardware.

Configure Hardware Settings

Use RFmxInstr functions and properties to configure your hardware to use the correct settings for the measurement.

1. Call RFmxInstr Configure Frequency Reference and select the reference clock source that corresponds to the physical configuration your instrument.
 - Onboard Clock: Uses the clock within the PXIe-5841 instrument that is provided by the PXIe-5655 module(s). The default option.
 - RefIn: Uses an external reference clock source. For PXIe-5841 instruments, this clock is from external equipment that supplies a compatible clock, which you connect to the free REF IN port of an associated PXIe-5655 module.
 - PXI_Clock: Uses the PXI_CLK signal from the PXI backplane.
2. If you selected an external reference clock source (RefIn), provide a valid `Frequency`.
3. **Optional:** Make additional configuration selections using configuration and utility functions and session properties as desired.
For example, you can use RFmxInstr configuration functions to account for attenuation due to RF components in the signal path.

Configure the Measurement

Use the functions and properties specific to your RFmx personality to configure that measurement.

Each RFmx personality has its own measurement API to enable the related measurement type.



Tip For most RFmx personalities, you can include measurement functions from any number of personalities in a single session initialized with RFmxInstr Initialize NIRFSA; you do not need to create unique sessions for each personality. The exception is RFmx VNA, which cannot be used in a session with any other RFmx personalities.

If you are programming in LabVIEW, you can refer to the VI Tree function on the palette for your RFmx personality for an inventory of all the functions included in the API placed in the order in which you need to call them to create a successful program. View the block diagram of the VI Tree to see the VI programming flow.

Close the Session

Every RFmx session must be closed in order to deallocate memory, destroy threads, and free operating system resources.

Required: Use RFmxInstr Close to conclude the session and free system resources.

Analyzing RF Signals with NI-RFSA

The primary function of vector signal analyzers and a primary function of vector signal transceivers is to analyze RF signals. To perform this operation while gaining access to the lowest level functionality of NI VSAs and VSTs, use the NI-RFSA instrument driver.



Note NI-RFSA includes a wide array of entry points to control your instrument and optimize its behavior for your application. This process describes only the basic programming flow to create an RF signal analysis program successfully.

1. [Initialize a Session](#)
Use Initialize to create a session with the PXIe-5841. All programs must begin with Initialize.
2. [Configure the Reference Clock](#)
Configure the correct settings for the reference clock, which ensures that instruments in the session are operating from a common timebase.
3. [Configure the Acquisition](#)
Use NI-RFSA to configure the acquisition parameters for the RF signal you want to analyze. There are two categories of acquisitions you can perform: I/Q acquisitions and power spectrum acquisitions.
4. [Coordinate Actions with Triggers and Events](#)
Use triggers and events to implement conditional behavior on your instrument and to coordinate the action of multiple instruments.
5. [Read and Plot Acquired Data](#)

Once you have defined the parameters for the RF signal acquisition, you can perform the acquisition and visualize the RF data you acquire.

6. Close the Session

Every NI-RFSA session must be closed in order to deallocate memory, destroy threads, and free operating system resources.

Initialize a Session

Use Initialize to create a session with the PXIe-5841. All programs must begin with Initialize.

Initializing returns an instrument handle with the session configured to a known state. Initialization can take a significant amount of time compared to other functions, so you should not include it in loops. Ideally, your program should call Initialize one time.

Configure the Reference Clock

Configure the correct settings for the reference clock, which ensures that instruments in the session are operating from a common timebase.

1. Call Configure Ref Clock and select the reference clock source that corresponds to the physical configuration your instrument.
 - Onboard Clock: Uses the clock within the PXIe-5841 instrument that is provided by the PXIe-5655 module(s). The default option.
 - RefIn: Uses an external reference clock source. For PXIe-5841 instruments, this clock is from external equipment that supplies a compatible clock, which you connect to the free REF IN port of an associated PXIe-5655 module.
 - PXI_Clk: Uses the PXI_CLK signal from the PXI backplane.
2. If you selected an external reference clock source (RefIn), provide a valid `clock rate`.

Valid clock rates for the PXIe-5655 LO within your PXIe-5841 instrument are as follows:

- 10 MHz
- 100 MHz
- 270 MHz
- $3.84 \text{ MHz} \times y$, where y is 4, 8, 16, 24, 25, or 32

Configure the Acquisition

Use NI-RFSA to configure the acquisition parameters for the RF signal you want to analyze. There are two categories of acquisitions you can perform: I/Q acquisitions and power spectrum acquisitions.

1. Call `Configure Acquisition Type` to designate the type of acquisition to perform
 - `IQ`: Acquire I/Q data
 - `Spectrum`: Compute a power spectrum over a given frequency range
2. Call `Configure Reference Level` and provide the maximum expected power of the RF signal you are acquiring.
3. Depending on the type of acquisition you want to make, set other parameters for the acquisition:

Acquisition Type	Procedure
I/Q	a. Call <code>Configure IQ Carrier Frequency</code> and provide the carrier frequency—the center frequency of the I/Q acquisition. b. Call <code>Configure IQ Rate</code> to set the I/Q rate, in samples per second, for the acquisition. c. Use <code>Configure Number of Samples</code> to choose whether you want to acquire continuously or to acquire a finite number of samples.
Spectrum	a. Define the spectrum to analyze. ▪ Use <code>Configure Spectrum Frequency Start Stop</code> to set lower and upper bounds on the spectrum. ▪ Use <code>Configure Spectrum Frequency Center Span</code> to define a center frequency and an associated span surrounding that frequency. b. Use <code>Configure Resolution Bandwidth</code> to set the size of the frequency bins in the power spectrum; higher resolution yields wider bins.

Coordinate Actions with Triggers and Events

Use triggers and events to implement conditional behavior on your instrument and to coordinate the action of multiple instruments.

- Call **Configure Trigger** and choose the instance that supports your application needs.
Configure Trigger defines the named trigger, as well as the type of signal that represents the named trigger, that the instrument should expect before taking the action the rest of your program defines.
- If you chose a software instance of **Configure Trigger**, call **Send Software Edge Trigger** later in your program to have the program provide that trigger, regardless of whether an actual trigger has occurred within your hardware setup.
- Use **Export Signal** to route a signal of your choosing to a terminal of your choosing.

For example, you can use **Export Signal** to route a signal from a first instrument to a second instrument that is awaiting that signal, as defined by the **Configure Trigger** instance for the second instrument.

NI-RFSA Named Trigger Types

Named trigger types in NI-RFSA define the action you want an instrument to take upon detecting a specific signal condition.

- **Start**—Transitions an instrument into a state where the instrument can respond to Sample Clocks. This trigger transitions the instrument from a nonsampling state into a sampling state—the instrument starts sampling and storing data.

Supported signal conditions: None, Digital Edge, Software

- **Reference**—Establishes the reference point that separates pretrigger and posttrigger samples; transitions an instrument from the Wait for Reference Trigger sampling state into the Post Reference Trigger sampling state. In the Post Reference Trigger sampling state, a counter begins counting Sample Clock cycles. When the configured number of samples is acquired, the device transitions into a Done state.

Supported signal conditions: None, Digital Edge, IQ Power Edge, Software

- **Advance**—Initiates the acquisition of the additional records in a multirecord acquisition.

Supported signal conditions: None, Digital Edge, Software

NI-RFSA Trigger Signal Conditions

NI-RFSA includes four possible signal conditions that can serve as the stimulus for an action an instrument or channel can take: digital edge, I/Q power edge, software edge, and none.



Note Not all named trigger types in NI-RFSA support all signal conditions.

None

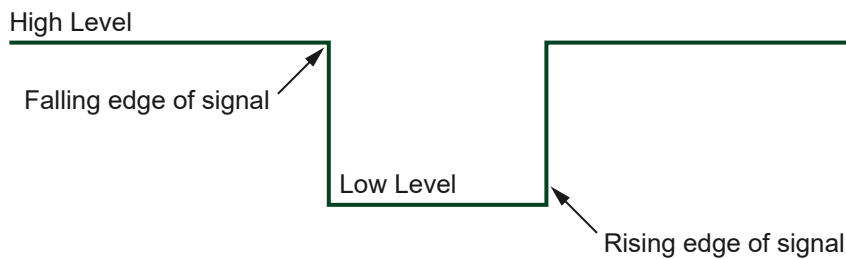
Signal acquisition starts immediately upon execution of the NI-RFSA Initiate function; no Start or Script trigger is necessary. This configuration is the default for instruments programmed with NI-RFSA.

Digital Edge

A channel performs an operation when the signal transitions from high to low or from low to high.

A digital signal has two discrete levels: a high level and a low level. When the signal transitions from high to low or from low to high, a **digital edge** is created. There are two types of edges: **rising**, which occurs when the signal transitions from low level to high level, and **falling**, which occurs with a transition from high level to low level. Triggers configured to act on a rising or falling edge of a digital signal are called **edge triggers**. Digital triggering is possible on the RTSI lines, PFI lines, and the PXI Star Trigger line.

As the figure shows, an edge trigger can be configured to occur either at the place labeled Falling Edge of Signal or Rising Edge of Signal.

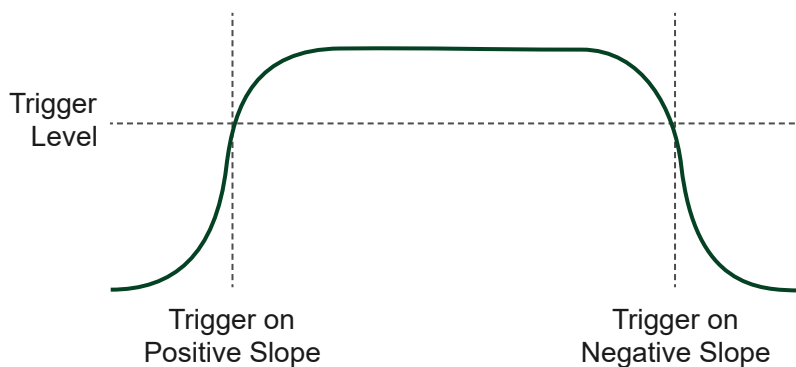


You can configure each named trigger to operate based on a digital edge.

I/Q Power Edge Trigger

After data has been downconverted and decimated, NI-RFSA calculates the instantaneous power of the I/Q data ($I^2 + Q^2$) in the digital downconverter and uses this value as a source for the Reference Trigger for the IF digitizer.

As shown in the figure, an I/Q power edge trigger can be configured to occur either at the positive slope or the negative slope.



Only Reference triggers may be configured to trigger based on the I/Q power edge.

Software Trigger

A software trigger is generated internally by a programmatic call to the NI-RFSA Send Software Edge Trigger function and can occur at any time, based upon the conditions specified in the program.

You can configure each named trigger in NI-RFSA to operate based on a software edge.

NI-RFSA Event Types

NI-RFSA includes specific events you can use in tandem with triggers to coordinate actions across instruments.

- **Ready for Start**—Indicates the instrument is configured and ready to receive a Start trigger.
- **Ready for Advance**—Indicates that the acquisition of a previous record is complete and that the instrument has entered its Wait for Advance Trigger state.
- **Ready for Ref**—Indicates when the instrument enters its Wait for Reference Trigger while Sampling state.
- **End of Record Event**—Indicates that the current record has been acquired and that the instrument has entered its Record Complete state.
- **Done**—Indicates the device has completed its acquisition.

Read and Plot Acquired Data

Once you have defined the parameters for the RF signal acquisition, you can perform the acquisition and visualize the RF data you acquire.

1. Based on whether you chose an I/Q acquisition or a spectrum acquisition, read the data in order to plot it.
 - I/Q: Call Read IQ and choose the data type to return.
 - Spectrum: Call Read Power Spectrum and choose the data type to return.
2. If you acquired I/Q data, call Get IQ Components to separate the data into separate I and Q components you can visualize.

Close the Session

Every NI-RFSA session must be closed in order to deallocate memory, destroy threads, and free operating system resources.

Required: Use Close to conclude the session and free system resources.

Generating RF Signals with NI-RFSG

The primary function of vector signal generators and a primary function of vector signal transceivers is to generate RF signals. To perform this operation with NI VSGs

and VSTs, use the NI-RFSG instrument driver.



Note NI-RFSG includes a wide array of entry points to control your instrument and optimize its behavior for your application. This process describes only the basic programming flow to create an RF signal generation program successfully.

1. [Initialize a Session](#)

Use Initialize to create a session with the PXIe-5841. All programs must begin with Initialize.

2. [Configure the Reference Clock](#)

Configure the correct settings for the reference clock, which ensures that instruments in the session are operating from a common timebase.

3. [Configure the RF Output Signal](#)

Use NI-RFSG to configure the parameters of the RF signal you want to generate.

4. [Coordinate Actions with Triggers and Events](#)

Use triggers and events to implement conditional behavior on your instrument and to coordinate the action of multiple instruments.

5. [Initiate Waveform Generation](#)

Initiate the instrument to apply any configured settings and begin generating the RF signal. All NI-RFSG waveform generation programs must call this function.

6. [Check the Status of RF Signal Generation](#)

All NI-RFSG programs must call this function, which provides an opportunity to check for errors in the program that occur during generation and indicates whether generation has concluded.

7. [Close the Session](#)

Every NI-RFSG session must be closed in order to deallocate memory, destroy threads, and free operating system resources. Closing a session also stops waveform generation, if it hasn't been stopped already.

Initialize a Session

Use Initialize to create a session with the PXIe-5841. All programs must begin with Initialize.

Initializing returns an instrument handle with the session configured to a known state.

Initialization can take a significant amount of time compared to other functions, so you should not include it in loops. Ideally, your program should call Initialize one time.

Configure the Reference Clock

Configure the correct settings for the reference clock, which ensures that instruments in the session are operating from a common timebase.

1. Call Configure Ref Clock and select the reference clock source that corresponds to the physical configuration your instrument.
 - Onboard Clock: Uses the clock within the PXIe-5841 instrument that is provided by the PXIe-5655 module(s). The default option.
 - RefIn: Uses an external reference clock source. For PXIe-5841 instruments, this clock is from external equipment that supplies a compatible clock, which you connect to the free REF IN port of an associated PXIe-5655 module.
 - PXI Clock: Uses the PXI_CLK signal from the PXI backplane.
2. If you selected an external reference clock source (RefIn), provide a valid `ref clock rate`.

Valid clock rates for the PXIe-5655 LO within your PXIe-5841 instrument are as follows:

- 10 MHz
- 100 MHz
- 270 MHz
- $3.84 \text{ MHz} \times y$, where y is 4, 8, 16, 24, 25, or 32

Configure the RF Output Signal

Use NI-RFSG to configure the parameters of the RF signal you want to generate.

Complete the following steps to configure the center frequency, output power level, generation mode, and, for arbitrary waveforms, the signal bandwidth.

1. **Required:** Use Configure RF to define the center frequency and power level of the RF signal to generate.



Note All NI-RFSG programs must include at least Configure RF.

2. Use Configure Generation Mode to define the type of signal to generate: CW, an arbitrary waveform, or a waveform dictated by scripts you create.
3. If generating an arbitrary waveform, with or without scripting, use Configure Signal Bandwidth to define the bandwidth for the signal.
4. If generating an arbitrary waveform, with or without scripting, use Write Arb Waveform to construct the waveform to generate.
5. If using scripting to generate an arbitrary waveform, use Write Script to incorporate your script into the program.

Coordinate Actions with Triggers and Events

Use triggers and events to implement conditional behavior on your instrument and to coordinate the action of multiple instruments.

- Call Configure Trigger and choose the instance that supports your application needs.
Configure Trigger defines the named trigger, as well as the type of signal that represents the named trigger, that the instrument should expect before taking the action the rest of your program defines.
- If you chose a software instance of Configure Trigger, call Send Software Edge Trigger later in your program to have the program provide that trigger, regardless of whether an actual trigger has occurred within your hardware setup.
- Use Export Signal to route a signal of your choosing to a terminal of your choosing.

For example, you can use Export Signal to route a signal from a first instrument to a second instrument that is awaiting that signal, as defined by the Configure Trigger instance for the second instrument.

NI-RFSG Named Trigger Types

Named trigger types in NI-RFSG define the action you want an instrument or instrument channel to take upon detecting a specific signal condition.

Trigger stimuli include digital edges, software functions, and analog levels. The trigger can be derived from attributes of the signal to be acquired, such as the level and slope of the signal.

The following named trigger types are available in NI-RFSG:

- **Start**—Initiates signal generation.

Supported signal conditions: None, Digital Edge, Software

- **Script**—General-purpose trigger with a role that is determined entirely by the context of the generation script. A script allows you to create sophisticated generation operations. For example, the script could configure the device to generate waveform A, then wait for the Script Trigger, then generate waveform B. You can configure up to four Script Triggers for use in your application. Once a digital edge Script Trigger has been received, that trigger remains true for all subsequent instructions until the clear instruction is called or the trigger is reset after being used in the wait, repeat/end repeat, or if instructions.

Supported signal conditions: None, Digital Edge, Digital Level, Software

- **Configuration List Step**—Transitions an instrument to the next step in the list.

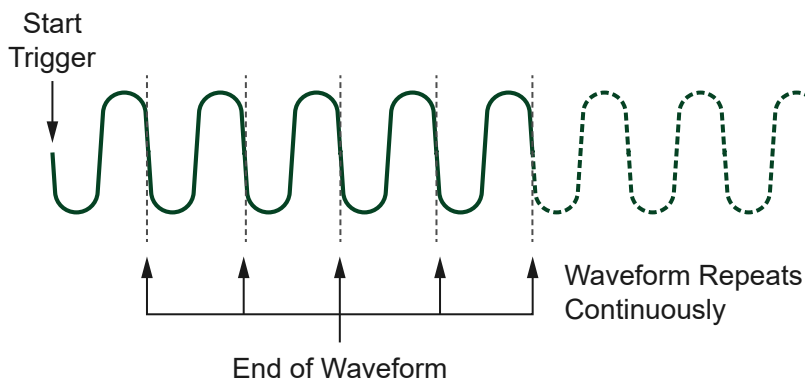
Supported signal conditions: None, Digital Edge

NI-RFSG Start Trigger Details

For waveform generators and VSTs, a Start trigger initiates signal generation.

When the instrument receives a Start trigger, the RF output signal begins to generate as shown in the following figure. The instrument ignores a Start trigger received after signal generation has started.

Figure 10. NI-RFSG Start Trigger Model





Note All devices that reside between the waveform generator and the output signal introduce a delay from the Start trigger to waveform generation.

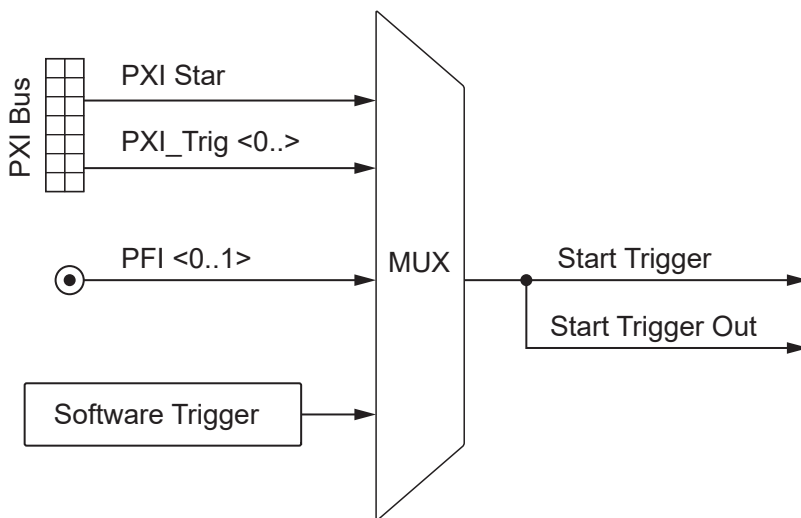
Configure the Start trigger using the Start Trigger Type property. The Configure Trigger function also provides access to this property.



Tip Sending a software trigger when you set Start Trigger Type to Digital Edge also starts signal generation. You can use this approach to create a software timeout to avoid leaving the instrument in a waiting state if the digital edge is never received.

The possible sources for a Start trigger are as follows.

Figure 11. NI-RFSG Start Trigger Sources

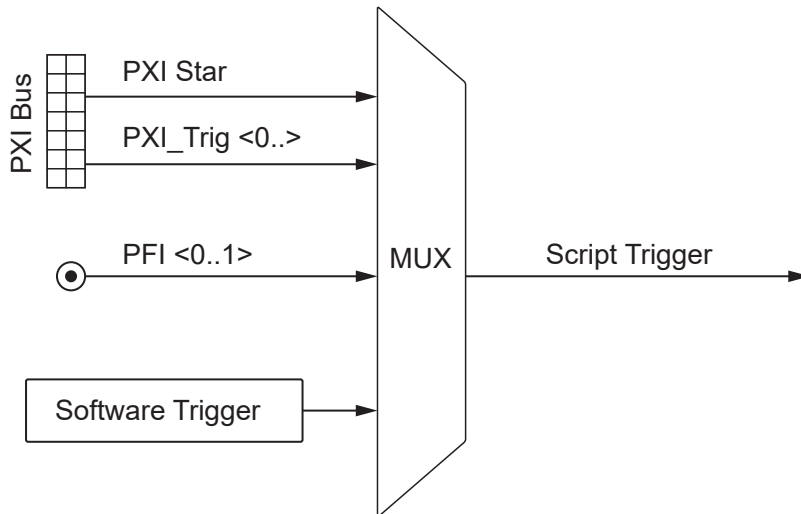


NI-RFSG Script Trigger Details

A Script trigger is a general-purpose trigger with a role determined entirely by the context of the dynamic generation script.

A script allows you to create sophisticated dynamic generation operations. For example, the script can configure the instrument to generate waveform A, then wait for a Script trigger, then generate waveform B. You can configure up to four Script triggers for use in your application.

Figure 12. NI-RFSG Script Trigger Sources



Note All devices that reside between the waveform generator and the output signal introduce a delay between the Script trigger and generation of the waveform.

Configure the Script trigger using the Script Trigger Type property. The Configure Trigger function also provides access to this property.



Tip Sending a software trigger when you set Script Trigger Type to Digital Edge also starts signal generation. You can use this to create a software timeout to avoid leaving the instrument in a waiting state if the digital edge is never received.

NI-RFSG Trigger Signal Conditions

NI-RFSG includes four possible signal conditions that can serve as the stimulus for an action an instrument or channel can take: digital edge, digital level, software edge, and none.



Note Not all named trigger types in NI-RFSG support all signal conditions.

None

Signal generation starts immediately upon execution of the NI-RFSG Initiate function;

no Start or Script trigger is necessary to begin generation.

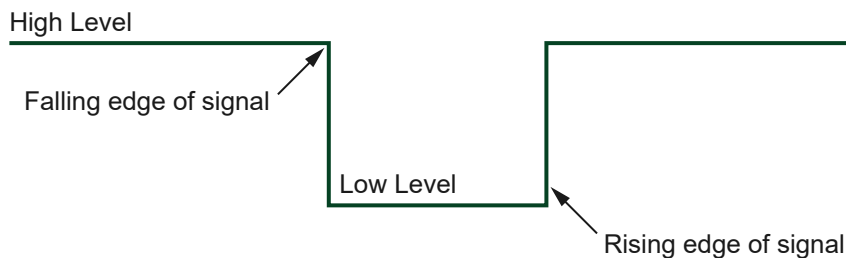
This configuration is the default behavior for instruments programmed with NI-RFSG.

Digital Edge

A channel performs an operation when the signal transitions from high to low or from low to high.

A digital signal has two discrete levels: a high level and a low level. When the signal transitions from high to low or from low to high, a **digital edge** is created. There are two types of edges: **rising**, which occurs when the signal transitions from low level to high level, and **falling**, which occurs with a transition from high level to low level. Triggers configured to act on a rising or falling edge of a digital signal are called **edge triggers**. Digital triggering is possible on the RTSI lines, PFI lines, and the PXI Star Trigger line.

As the figure shows, an edge trigger can be configured to occur either at the place labeled Falling Edge of Signal or Rising Edge of Signal.



You can configure each named trigger to operate based on a digital edge.

Digital Level

A channel performs an operation when a signal goes below a defined low level or above a defined high level.

Triggers configured to act in this way are known as **level triggers**.

Only Script triggers may be configured to trigger based on a digital level.

Software Edge

After initiation, a software trigger is generated internally by a programmatic call using NI-RFSG and can occur at any time, based upon the conditions specified in the program.

You can configure Start and Script triggers in NI-RFSG to operate based on a software edge.

Use NI-RFSG Send Software Edge Trigger and set the `trigger_type` input to the type of trigger you want to send.

NI-RFSG Event Types

NI-RFSG includes specific events you can use in tandem with triggers to coordinate actions across instruments.

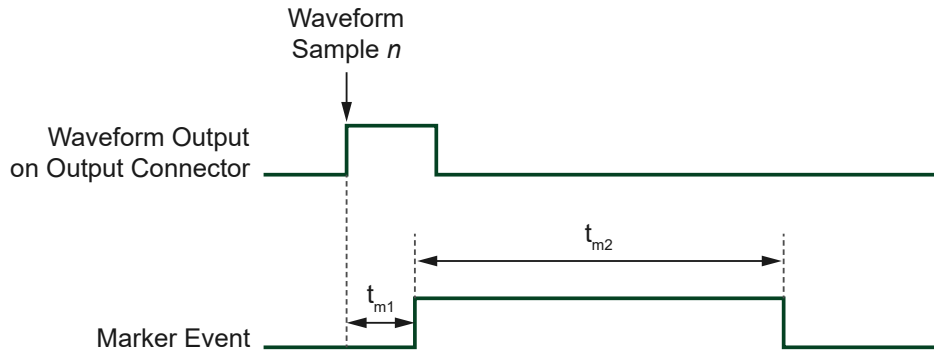
- **Marker**—Occurs at the time that a specific location or sample ***n*** in the waveform generates on the output connector. If the waveform loops multiple times in a segment, the marker generates each time the waveform loops.
- **Started**—Emitted when the RF signal generator has received a Start trigger and is generating a waveform.
- **Done**—Indicates that the generation of the previous waveform is complete.
- **Configuration Settled**—Indicates when the signal has settled for a given configuration list step. The definition of ***settled*** is determined based on the values of the Frequency Settling and Frequency Settling Units properties.

NI-RFSG Marker Event Details

A ***marker*** is an event that an NI RF signal generator produces in relation to a waveform that is generated. The marker event is delayed slightly relative to the leading edge of the waveform that it marks.

By default, the marker event is a pulse of fixed width. The following figure shows a waveform generated on the output of a waveform generator. The waveform contains a pulse that represents a waveform sample ***n*** that is one Sample Clock in width. The second pulse, the marker event, represents the pulse that generates when the corresponding waveform sample ***n*** is output at the output connector.

Figure 13. Waveform Output and Associated Marker Event



- t_{m1} —Duration of delay between the configured waveform sample n and generation of the associated marker event. For most NI VSTs, configurable with NI-RFSG Events Delay and Relative Delay.
- t_{m2} —Marker event pulse duration. For most NI VSTs, configurable with NI-RFSG Width Units and Width Value.

You can use NI-RFSG to configure the characteristics of the pulse generated as the Marker event.

In addition, you can use scripts to create markers to develop complex behaviors around generated waveforms.

For example, you can configure the Marker event as a toggled digital pulse signal that changes state between a low level and high level. While the basic pulse option for Marker events generates event pulses of fixed width, using the toggle option allows you to use scripts to change the duration of the Marker event pulse. You can control the initial state of the toggled pulse with `MarkerEvent.Toggle.InitialState`.

Initiate Waveform Generation

Initiate the instrument to apply any configured settings and begin generating the RF signal. All NI-RFSG waveform generation programs must call this function.

Call `Initiate` to apply all configured settings, if they have not already been applied, and start RF generation.

Check the Status of RF Signal Generation

All NI-RFSG programs must call this function, which provides an opportunity to check

for errors in the program that occur during generation and indicates whether generation has concluded.

Call Check Generation Status.



Tip NI recommends placing this function in a while loop to continually check the generation status. Use the Wait function to limit how frequently the loop runs to conserve CPU usage.

Close the Session

Every NI-RFSG session must be closed in order to deallocate memory, destroy threads, and free operating system resources. Closing a session also stops waveform generation, if it hasn't been stopped already.

Required: Use Close to conclude the session and free system resources.

Improving NI-TClk Synchronization of RF Instruments

The PXIe-5841 supports NI-TClk, which allows for multi-instrument synchronization with fine alignment. You can use NI-RFSA and NI-RFSG to further optimize NI-TClk synchronization.

Refer to the ***NI-TClk Synchronization Help*** for more information about instrument synchronization.

NI-TClk Manual Calibration Using NI-RFSA

Complete the following steps to improve the synchronization accuracy by lowering skew and minimizing jitter of NI-TClk synchronized devices using manual calibration.

1. Connect the external signal to the VSTs. For best performance, minimize unnecessary skew by using matched length cables.
2. Configure the VST for acquisition using NI-TClk. If using LabVIEW, use the RFSA Synchronization (TClk, Shared LO and Reference Clock) (5840) .vi example.
3. Start the acquisition and measure the skew between devices, noting the amount of

jitter.

4. Read the value from the Absolute Delay property. Stop the acquisition, modify the value, and write it back to apply skew correction.
5. Repeat steps 3 and 4 until you achieve the appropriate synchronization results.
6. Read the value from the Absolute Delay property and store the value for future synchronization sessions.

To apply the manual calibration synchronization results, write the stored value to the Absolute Delay property before calling the NI-TClk Synchronize.

NI-TClk Manual Calibration Using NI-RFSG

Complete the following steps to improve the synchronization accuracy by lowering skew and minimizing jitter of NI-TClk synchronized devices using manual calibration.

1. Connect the RF output connector on the vector signal transceivers (VSTs) to a multi-channel signal analyzer. For best performance, minimize unnecessary skew by using matched length cables.
2. Configure the VST for generation using NI-TClk. If using LabVIEW, use the `RFSG Synchronization (TClk, Shared LO).vi` example.
3. Start the generation and measure the skew between devices, noting the amount of jitter.
4. Read the value from the Absolute Delay property. Stop the generation, modify the value, and write it back to apply skew correction.
5. Repeat steps 3 and 4 until you achieve the appropriate synchronization results.
6. Read the value from the Absolute Delay property and store the value for future synchronization sessions.

To apply the manual calibration synchronization results, write the stored value to the Absolute Delay property before calling NI-TClk Synchronize.

Local Oscillators

There are separate local oscillators (LOs) on the PXIe-5841 for each RF signal path. The separate LOs allow for different LO frequencies between RF input and RF output channels.

The individual LOs of RF input and RF output channels are referenced to a common

clock source in their phase-locked loop circuitry. This configuration ensures that the relative frequency error between the individual LOs is a net of zero and allows for precision tuning of channels in the frequency domain. If you specify an external Reference Clock source, both RF input and RF output LOs reference this source.

You can configure each oscillator in both integer and fractional tuning modes.

The following table describes each mode and its capabilities. The configured tuning mode is unique for each LO.

Tuning Mode	Supported Frequency Step Resolution
Integer	1 MHz
	5 MHz
	10 MHz
	25 MHz
	50 MHz
	100 MHz
Fractional	≥50 kHz, configurable 400 kHz default

Running the device in fractional tuning mode can create close-in spurs related to the intermodulation of the VCO and the phase detect frequency harmonics. Refer to the **PXIe-5841 Specifications** document for more information about spurious performance.

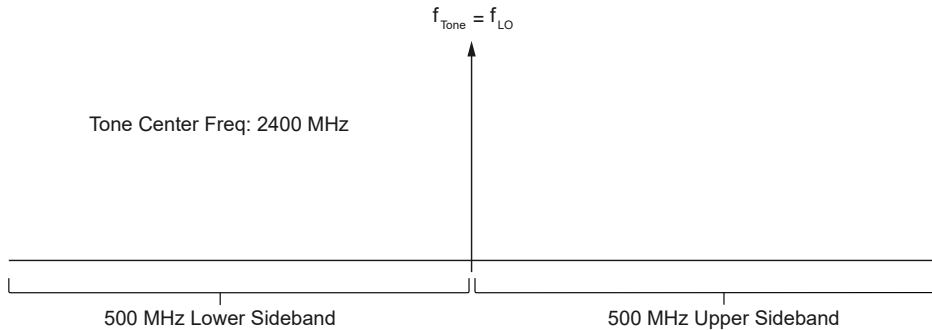
Frequency Correction (LO Coercion)

In both integer and fractional tuning modes, you can use digital frequency correction in the FPGA to compensate for any difference between the requested frequency and the coerced LO frequency. Refer to the **PXIe-5841 Specifications** for more information about integer and fractional tuning modes.

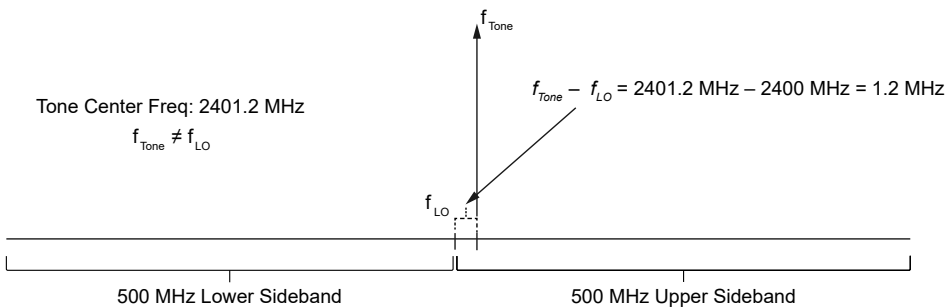


Note Digital frequency correction is implemented automatically in the NI-RFSA and NI-RFSG APIs.

The following figure shows a tone generated at a tuned frequency of 2,400 MHz. The tuned frequency is an even multiple of the 5 MHz integer step size and as such is also the LO frequency. This example does not require digital shift, so 1 GHz of equalized bandwidth is available.



The following figure shows a tone generated at a tuned frequency of 2,401.2 MHz. The closest multiple of the 5 MHz integer step size is 2,400 MHz, which creates a frequency error of 1.2 MHz between the requested frequency and the LO frequency. Digital frequency shifting on the FPGA corrects for the 1.2 MHz offset.



External Local Oscillator

The PXIe-5841 uses an internal measurement circuit that allows flexibility in external LO power. This flexibility ensures that the external LO arrives at the mixing stage with the same power level as the internal LO, which subsequently minimizes effects on sensitive specifications such as amplitude accuracy, residual LO power, and quadrature impairments.

By default, the PXIe-5841 is calibrated to assume a nominal external LO present at the LO IN (RF IN 0 and RF OUT 0) front panel connectors at 0 dBm. NI recommends external LO calibration if the external LO source varies from this nominal value by more than ± 0.25 dB. Calibrating the external LO aligns the external LO path gain of the PXIe-5841

over the full frequency range for the specific external LO source.

LO Sharing Between NI-RFSA and NI-RFSG

To reduce phase noise for some measurements, the PXIe-5841 allows for sharing the same internal LO or using the PXIe-5655 during NI-RFSA and NI-RFSG sessions.

To share the same LO between an NI-RFSA and NI-RFSG session, you must initiate both sessions and ensure both sessions use the same shared setting.

PXIe-5841 without Using the PXIe-5655

To share an LO between NI-RFSA and NI-RFSG without using the PXIe-5655, complete the following steps:

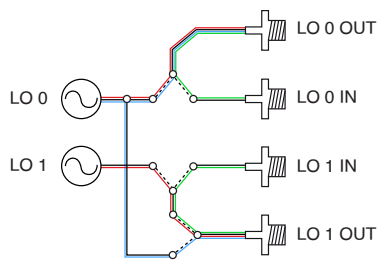
1. Connect RF OUT LO OUT to RF IN LO IN on the PXIe-5841.
2. In NI-RFSG, set the LO Out Enabled property. In NI-RFSA, set the LO Source property to LO In.

You can also share an LO in the opposite direction by completing the following steps:

1. Connect RF IN LO OUT to RF OUT LO IN on the PXIe-5841.
2. In NI-RFSA, set the LO Out Enabled property. In NI-RFSG, set the LO Source property to LO In.

PXIe-5841 with the PXIe-5655

The following high-level block diagrams show the difference between an independent and shared LO configuration within the PXIe-5655 LO0 and LO1 paths.



— LO Source is Set to Onboard	— LO Source is Set to SG SA Shared	— LO Source is Set to LO In
-------------------------------	------------------------------------	-----------------------------

In the shared configuration, the shared LO must use the same frequency or the second driver session will return an error when committing the settings to hardware.

To share the same LO signal between NI-RFSA and NI-RFSG sessions, configure the NI-RFSA or NI-RFSG LO Source property as appropriate for your hardware configuration, as shown in the following table.

Table 16. LO Source Configurations

PXIe-5841 with PXIe-5655 Hardware Configuration	NI-RFSA and NI-RFSG LO Source Values	
	Property Value	Attribute Value
PXIe-5655 (independent configuration)	Onboard	- NIRFSA_VAL_ONBOARD_STR - NIRFSG_VAL_LO_SOURCE_ONBOARD_STR
External device LO connected to PXIe-5655 LO 0 IN or LO 1 IN (independent configuration)	LO In	- NIRFSA_VAL_LO_IN_STR - NIRFSG_VAL_LO_SOURCE_LO_IN_STR
PXIe-5655 (shared configuration using LO0)	SG SA Shared	- NIRFSA_VAL_LO_SOURCE_SG_SA_SHARED_STR - NIRFSG_VAL_LO_SOURCE_SG_SA_SHARED_STR

When using the PXIe-5841 with the PXIe-5655, setting the LO Source property to SG SA Shared in an NI-RFSG session enables NI-RFSA to share the internal LO0 synthesizer if requested by NI-RFSA. If an NI-RFSG session has already set the LO Source property to something other than SG SA Shared property value in an NI-RFSG session, you cannot

request sharing using NI-RFSA.

Setting the LO Source property to SG SA Shared in an NI-RFSA session results in using the PXIe-5655 LO0 synthesizer. You can also choose not to use the LO0 shared configuration in the NI-RFSA session by instead setting the LO Source property to Onboard or LO In.



Note Output power levels at LO 0 OUT may vary slightly (<1 dB, typical) when set to SG SA Shared as NI-RFSA connects or disconnects the shared signal.

Using an External LO

Complete the following steps when using an external LO.

Cabling

When using an external LO, first ensure that the reference clocks are shared either by using the PXI backplane clock or by exporting the reference clock from the LO by completing the following steps.

1. Connect the external LO's REF OUT connector to the PXIe-5841 REF IN connector.
2. Specify REF IN as the Reference Clock source using the NI-RFSA or NI-RFSG Configure Ref Clock VI or function.
3. Connect the output of the external LO to the PXIe-5841 LO IN connector.

Specifying LO Frequency With Automatic Frequency Offset

To use automatic offset with an external LO, you should provide a frequency as requested by the NI-RFSA or NI-RFSG driver.

When writing your application, complete the following steps to specify the LO frequency in case the driver applies frequency offset as described in the ***Automatic Frequency Offset*** section.

1. Set all the properties for the acquisition or generation, including the NI-RFSA I/Q

Carrier Frequency property or the NI-RFSG Frequency property.



Note Do not set the LO Frequency property. If you previously set the LO Frequency property, reset it.

2. Set the LO Source property to LO IN
3. (Optional) Set the Bandwidth Selection property if you want to apply automatic frequency offset.
 - a. (Optional) Set the NI-RFSA Downconverter Frequency Offset Mode property or the NI-RFSG Upconverter Frequency Offset Mode property to Enabled, or query that they are already set to Enabled.



Note For a given RF frequency, NI-RFSA and NI-RFSG use the same automatic offsets for the PXIe-5841. Therefore, if the LO IN source for NI-RFSA is the LO OUT from NI-RFSG on the same device, you can skip steps 3 through 5.

4. Query the LO Frequency to get the expected frequency.
5. Set the frequency of your LO source to the expected frequency.
6. Set the LO Frequency property to the actual frequency of your LO source in case the source does any rounding.

Specifying LO Power When Using an External LO

To achieve better image and LO leakage performance with an external LO, set the LO IN power level of the PXIe-5841 to the actual output power level of the external LO.

1. Query the actual output power level of the external LO.
2. Set the NI-RFSA or NI-RFSG LO IN Power Level property to the value returned in the previous step.

Spurious Performance

Direct conversion topologies do not allow for multiple stages of filtering common in multistage IF systems. Any signals present within the instantaneous bandwidth of the device are converted down to baseband.

Timing Configurations

The timebases of the PXIe-5841 and PXIe-5655 modules must be frequency-locked to a common Reference Clock. The following clock sources are available when using the PXIe-5655 with the PXIe-5841:

- **Onboard Reference Clock of the PXIe-5655 RF Analog Signal Generator**—The PXIe-5655 onboard Reference Clock source supplies the Reference Clock.
- **External Reference Clock**—Connect the external clock signal from your stable frequency reference to the REF IN connector on the PXIe-5655.
- **PXI Express Backplane Clock**—This PXI backplane supplies the Reference Clock signal.

Configuring Onboard Reference Clock Timing

This configuration of the PXIe-5841 exports a 10 MHz signal from the PXIe-5655.

Complete the following steps to configure the PXIe-5841 to lock to an onboard reference clock source.

1. Associate the PXIe-5841 and PXIe-5655 by performing the steps in the ***Connecting the PXIe-5655 Analog Signal Generator*** section. This completes the Reference Clock connection.
2. Using NI-RFSA or NI-RFSG, set the Reference Clock Source property or attribute to Onboard Clock.

Configuring External Reference Clock Timing

Complete the following steps to configure the PXIe-5841 to lock to an external clock reference source.

1. Connect the external signal to the PXIe-5655 REF IN connector.
2. Using NI-RFSA or NI-RFSG, set the Ref Clock Source property or attribute to RefIn.

Configuring PXI Backplane Clock Timing

The PXI backplane can supply a 100 MHz reference clock signal.

To configure the PXIe-5841 to use the PXI backplane clock, set the NI-RFSA or NI-RFSG Reference Clock Source property or attribute to PXI Clock.

Configuring Frequency and Bandwidth for PXIe-5841

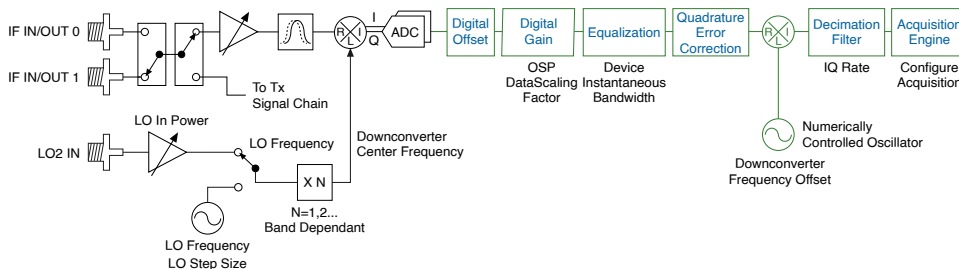
Use the NI-RFSA and NI-RFSG drivers to configure the PXIe-5841 instrument for optimal performance by specifying frequency and bandwidth constraints.

The following table shows the equivalencies between NI-RFSA and NI-RFSG properties mentioned throughout the section.

Table 17. NI-RFSA and NI-RFSG Frequency and Bandwidth Configuration Properties

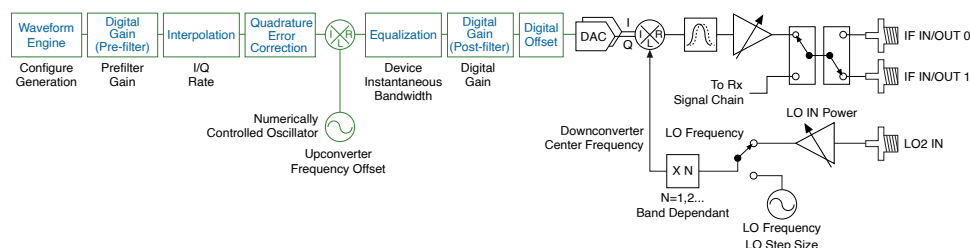
NI-RFSA Property	NI-RFSG Property
IQ Carrier Frequency	Frequency
Downconverter Center Frequency	Upconverter Center Frequency
Downconverter Frequency Offset	Upconverter Frequency Offset
LO Frequency	LO Frequency
Device Instantaneous Bandwidth	Device Instantaneous Bandwidth
Signal Bandwidth	Signal Bandwidth
IQ Rate	IQ Rate
Downconverter Frequency Offset Mode	Upconverter Frequency Offset Mode

The following block diagram represents the PXIe-5841 RF input connectors and the associated NI-RFSA properties.



Note The names of the input connectors change depending on your instrument configuration.

The following block diagram represents the PXIe-5841 RF output connectors and the associated NI-RFSG properties.



Note The names of the output connectors change depending on your instrument configuration.

Frequency and Bandwidth Configuration Terminology

Refer to the following list of terms when configuring frequency and bandwidth.

- **I/Q Rate**—The effective sampling rate of the baseband signal chain. Setting a value lower than the maximum will limit the available passband.
- **I/Q Carrier Frequency**—The center frequency of the waveform data acquired or generated.
- **Downconverter Center Frequency**— The frequency present at the mixer during downconversion or upconversion. This frequency is derived from the LO frequency through a series of multipliers. The multiplication factor may be 1, 2, 4, or 8, depending on the target frequency. In a direct conversion architecture, the downconverter center frequency contains LO leakage in the I/Q data.
- **LO Frequency**— The frequency of the local oscillator as present at the LO IN or LO OUT port. This frequency may or may not be equal to the downconverter center frequency depending on the conversion architecture and any frequency multipliers in the LO path. The signal from the onboard LO or the LO IN port passes through signal conditioning circuits and frequency multipliers or dividers to hit the mixer at the required frequency and power.
- **Device Instantaneous Bandwidth**— The total calibrated bandwidth available through the instrument signal path, centered at the downconverter center frequency. For example, if the downconverter center frequency is 6 GHz and the device instantaneous bandwidth is 1 GHz, an acquisition can contain calibrated data from 5.5 GHz to 6.5 GHz.
- **Signal Bandwidth**—When performing a signal acquisition, **signal bandwidth** is

the bandwidth of the signal present at the input port, centered at the I/Q carrier frequency. When performing a signal generation, **signal bandwidth** is the bandwidth of the waveform to be generated, centered at the I/Q carrier frequency.

- **Passband**—The bandwidth able to be acquired based on the I/Q rate, centered at the I/Q carrier frequency. This is usually defined at 80% of the I/Q rate. For example, an I/Q rate of 100 MS/s results in a passband of 80 MHz. Note that the passband edge may fall outside of the device instantaneous bandwidth edge and will, therefore, have aliased or uncalibrated data.
- **LO Step Size**—The quantum at which the LO frequency can be tuned. You can set the LO Step Size property or, on supported instruments, the VCO Step Size property to control the step size at the mixer or on the onboard LO (before multipliers), respectively, to achieve different hardware performance.

For example, assume the LO step size for an instrument is 2 MHz and the downconverter center frequency is set to 5,001.5 MHz. The downconverter center frequency will be coerced to 5,002 MHz, and the remaining 500 kHz shift will be applied digitally.

- **LO Power**—When using an external LO, the power of the LO is important for maintaining good image and LO leakage performance because the impairments correction is calibrated for a specific LO power at the mixer. Based on the LO IN power, NI-RFSA configures the LO signal path to achieve a mixer power close to the mixer power used during impairment calibration. You can also set the LO OUT power to apply gain to the output signal to avoid power loss when in an LO daisy-chain configuration.

Frequency and Bandwidth Selection to Optimize Performance

You can configure frequency and bandwidth settings of an NI VST to optimize measurement performance for different types of applications.



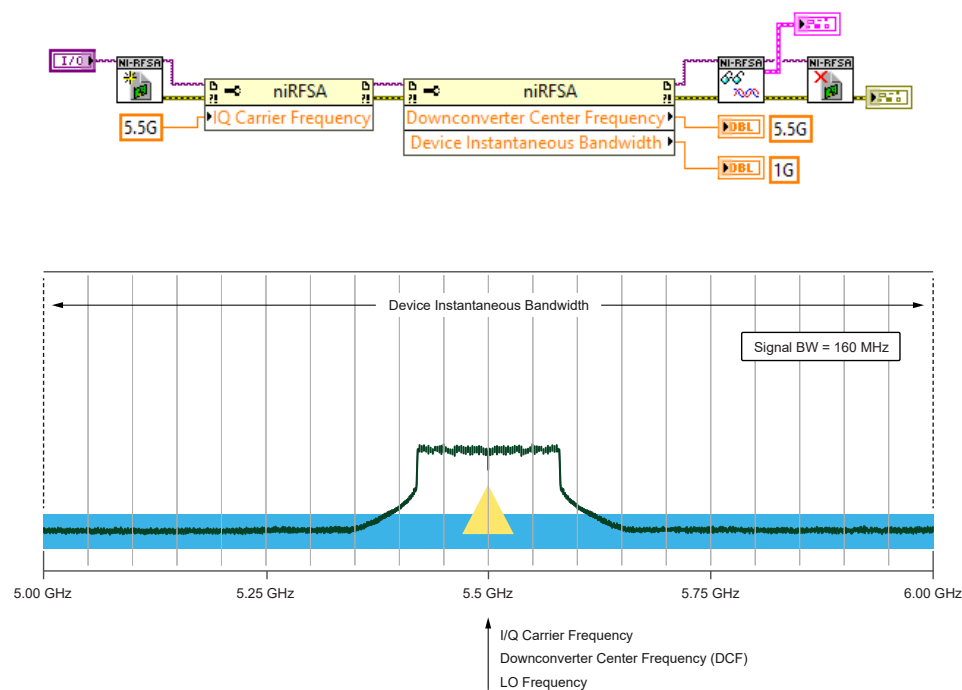
Note This information illustrates frequency and bandwidth concepts by portraying a generic example instrument of 1 GHz bandwidth; values and figures may not necessarily reflect the frequencies, bandwidth, or LO step size of your instrument.

Basic Behavior

For a simple acquisition, you can set the I/Q carrier frequency and the I/Q rate. When you do not specify a bandwidth constraint, NI-RFSA selects the widest filter for the given frequency region. You can obtain the bandwidth of the instrument for this configuration by querying the Device Instantaneous Bandwidth property. To determine the center of the device instantaneous bandwidth, you can query the Downconverter Center Frequency property.

You can specify the I/Q rate, which configures the decimation filter and, as a result, configures the available passband. When oversampling, the passband is often greater than the actual signal bandwidth present at the port, so specifying the Signal Bandwidth property gives the driver additional information about the requested bandwidth.

The following example illustrates the default behavior.



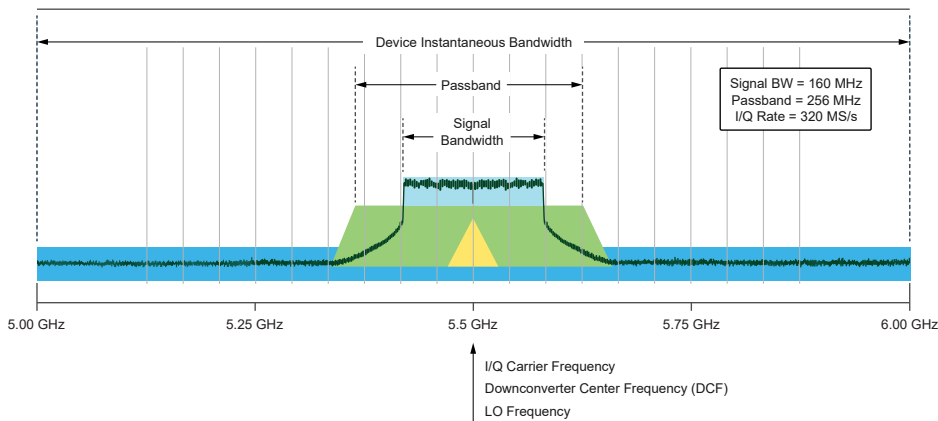
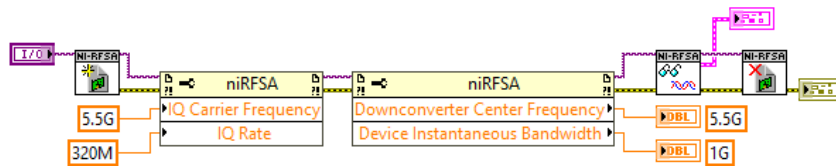
Specifying Device Instantaneous Bandwidth

In some cases, the default behavior may not be optimal. You can improve the default behavior by providing NI-RFSA with more information about the system. By setting the

Device Instantaneous Bandwidth property, you are specifying how much instrument bandwidth is required. The instrument bandwidth is used in filter selection and is only coerced up, so the filters chosen will always yield a bandwidth as wide or wider than requested. You can always read back the coerced or actual device instantaneous bandwidth to find out how much bandwidth is available in the given configuration. For example, if only 160 MHz of bandwidth is needed from an instrument and the next largest filter for that instrument is 1 GHz, NI-RFSA chooses the 1 GHz filter.

Oversampling

In the following example, the I/Q rate is set to 320 MS/s, even though the waveform signal bandwidth is only 160 MS/s. This is referred to as **oversampling**. The passband is 80% of the I/Q rate, or 256 MHz. In this example, because the passband extends outside the device instantaneous bandwidth, frequency content outside the device instantaneous bandwidth will be present in the time-domain data. This frequency content is usually filtered out by demodulation algorithms but may be problematic if you are analyzing the time-domain data manually, and there is a strong interfering signal in that region.



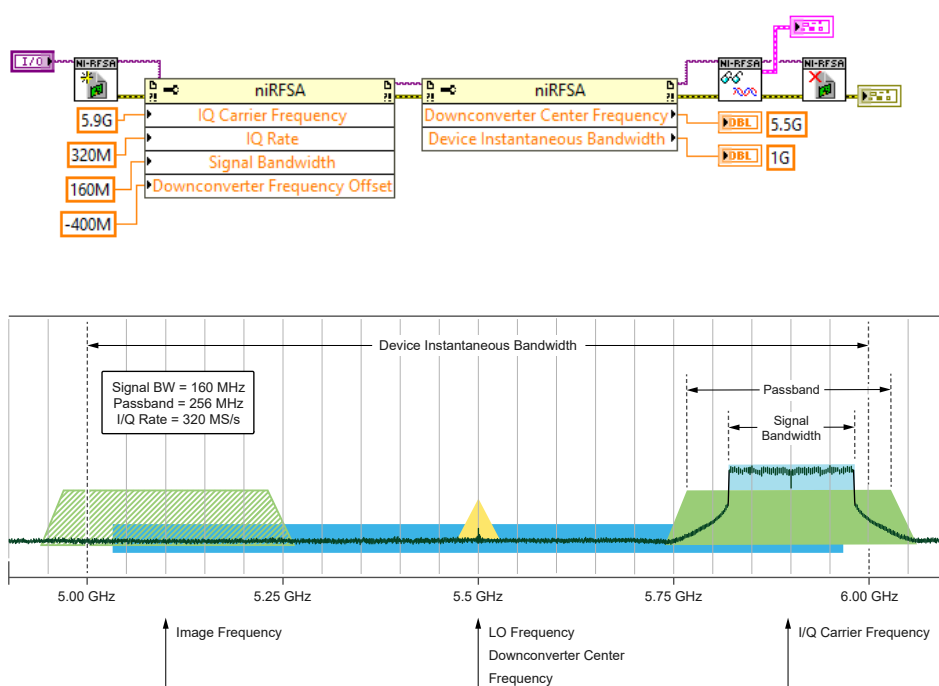
Specifying a Frequency Offset

To avoid LO leakage present in the passband, you can apply a frequency offset by setting the NI-RFSA Downconverter Frequency Offset property or the NI-RFSG Upconverter Frequency Offset property. This property maintains a constant offset if you vary the I/Q center frequency. You can always read back the actual device instantaneous bandwidth and downconverter center frequency.

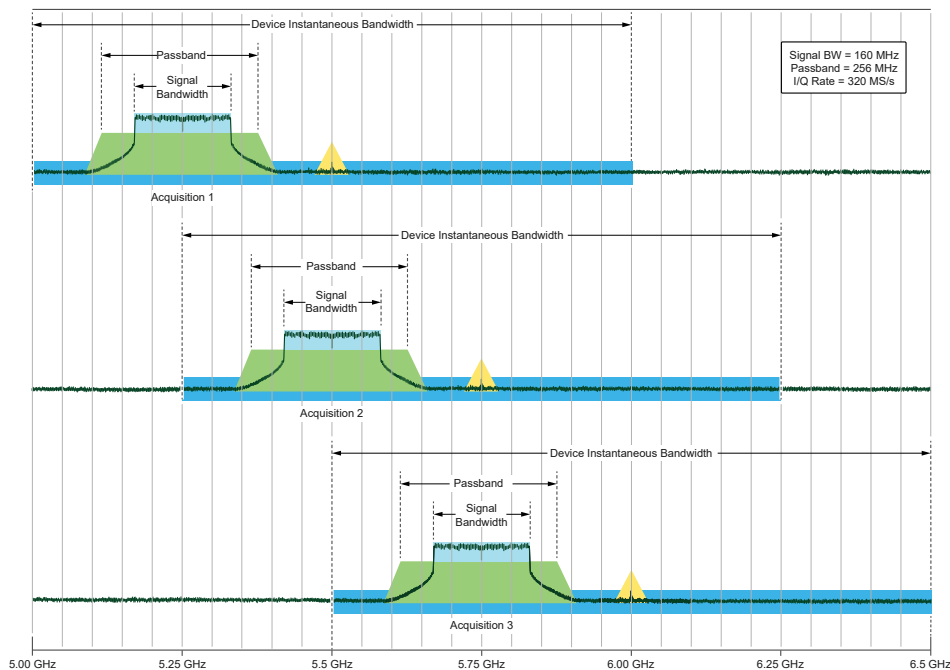
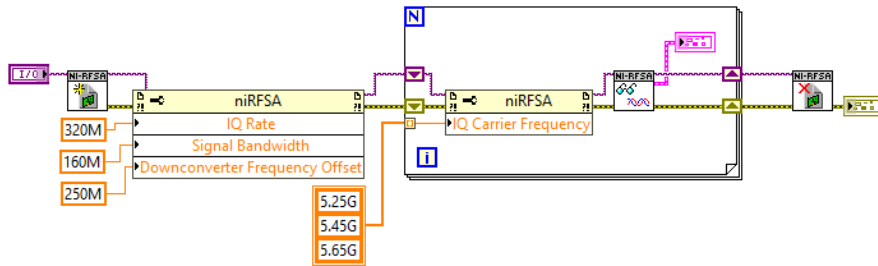
Specifying Signal Bandwidth

Another way to provide bandwidth information to the instrument driver is to specify the signal bandwidth. Previously, it was noted that the device instantaneous bandwidth is centered at the downconverter center frequency. **Signal bandwidth**, however, is centered at the I/Q carrier frequency, which may be more useful for some applications as it follows the signal of interest. Similarly, you can specify the downconverter frequency offset instead of the downconverter center frequency if it is more intuitive for the application.

The following example shows how to programmatically specify the signal bandwidth.



Note that setting the signal bandwidth does not necessarily avoid hardware reconfigurations during in-band retuning, but the settings stay relative to the I/Q carrier frequency, so it may be easier to manage in your application. The following example shows three acquisitions with the same I/Q rate, signal bandwidth, and downconverter frequency and only changes the I/Q carrier frequency for each acquisition.



Automatic Frequency Offset

The PXIe-5841 can automatically offset the downconverter center frequency from the carrier frequency when sufficient instrument bandwidth is available. For example, the signal must fit within half (the device instantaneous bandwidth minus the LO guard band). A table of the maximum available bandwidth for a given I/Q frequency when

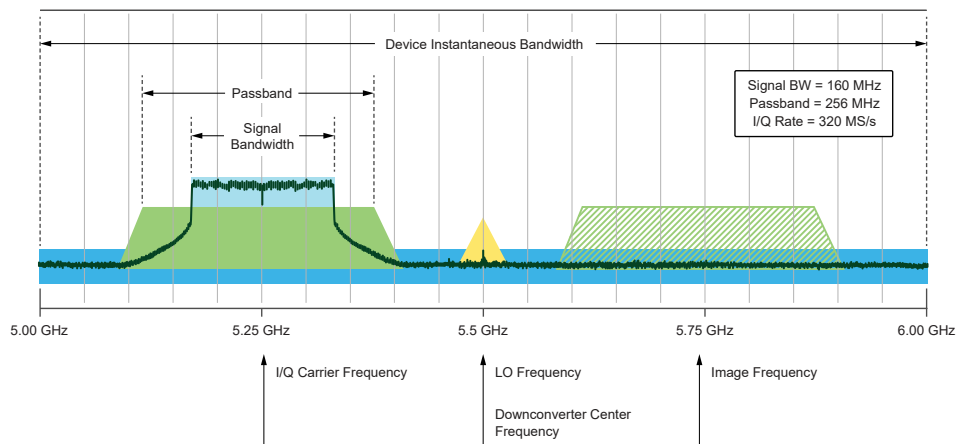
using automatic frequency offset is available in the device specifications. In auto-offset mode, the NI-RFSG filters are scaled so that a CW tone will not overflow if the frequency is within the maximum available bandwidth or ± 50 MHz, whichever is less. To prevent an overflow at any offset, you can set Prefilter Gain to less than or equal to -2.0. Outside of auto-offset mode, the filters are scaled so that no CW tone will overflow, even with Prefilter Gain equal to 0.0.

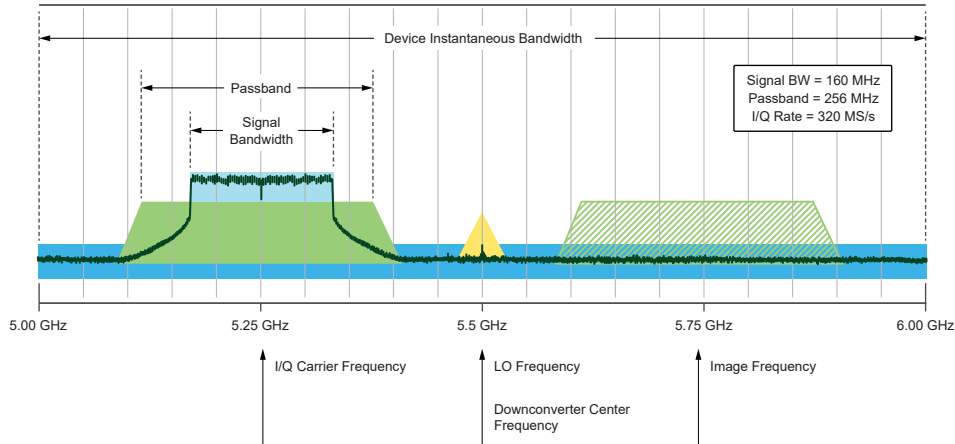
The actual frequency offset applied can be determined by reading the coerced downconverter center frequency. For Automatic Offset mode to function, you must specify the signal bandwidth. Additionally, when the downconverter frequency offset mode is enabled and the signal bandwidth is known, NI-RFSA can further optimize the dynamic range of the instrument. Using the instrument when downconverter frequency offset mode is enabled also ensures the gain accuracy is maintained, as noted in the device specifications. Otherwise, using the user-defined offset mode, the gain accuracy at an offset is limited to the amplitude accuracy specification $\pm$ the frequency response specification.



Note To use automatic frequency offset with an external LO, including sharing an LO between acquisition and generation, refer to the ***Using an External LO*** section.

The following example shows how to programmatically set the PXIe-5841 to automatically offset the downconverter center frequency from the carrier frequency, if possible. Refer to the Downconverter Frequency Offset Mode and Upconverter Frequency Offset Mode properties for more information.



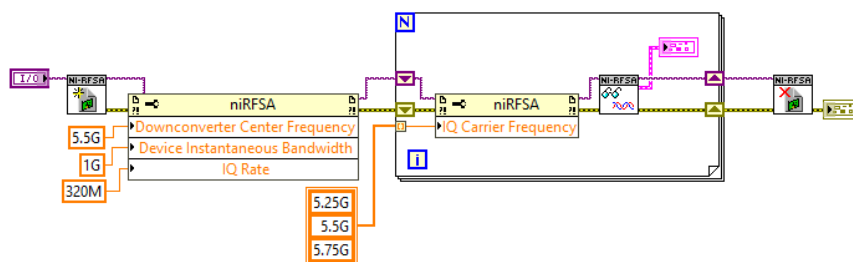


In-Band Retuning

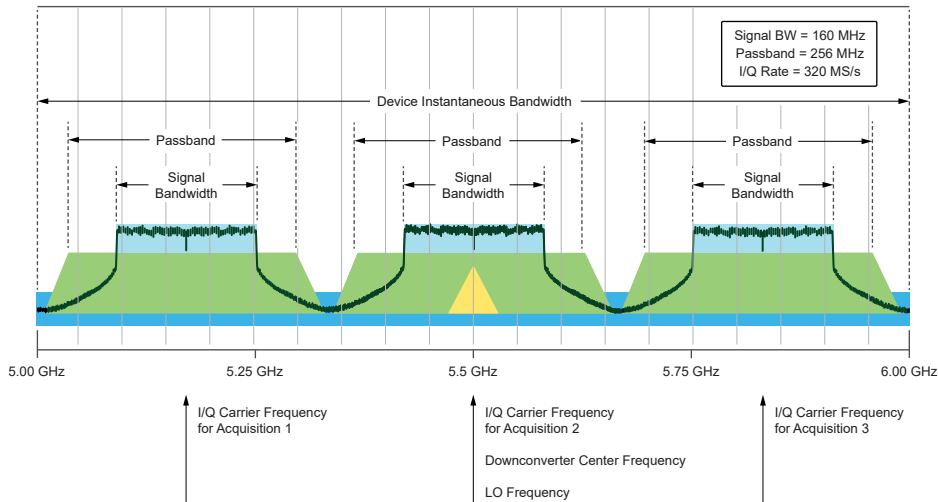
In-band retuning allows the instrument to quickly tune between signals of interest or generation within a given device instantaneous bandwidth using digital upconversion or downconversion to digitally tune the I/Q carrier frequency. The advantages of in-band retuning include reduced settling time for both frequency and amplitude. However, in-band retuning can only occur within the supported instantaneous bandwidth of the instrument. You can use digital tuning in combination with LO tuning to more efficiently sweep over a spectrum for supported applications.

You can use in-band retuning by setting the device instantaneous bandwidth and leaving the downconverter center frequency fixed, which significantly reduces reconfiguration and settling times.

The following example shows how to programmatically set the PXIe-5841 for in-band retuning.



The following figure shows three acquisitions with only digital frequency shift reconfigurations.



Using an External LO

Complete the following steps when using an external LO.

Cabling

When using an external LO, first ensure that the reference clocks are shared either by using the PXI backplane clock or by exporting the reference clock from the LO by completing the following steps.

1. Connect the external LO's REF OUT connector to the PXIe-5841 REF IN connector.
2. Specify REF IN as the Reference Clock source using the NI-RFSA or NI-RFSG Configure Ref Clock VI or function.
3. Connect the output of the external LO to the PXIe-5841 LO IN connector.

Specifying LO Frequency With Automatic Frequency Offset

To use automatic offset with an external LO, you should provide a frequency as requested by the NI-RFSA or NI-RFSG driver.

When writing your application, complete the following steps to specify the LO frequency in case the driver applies frequency offset as described in the **Automatic Frequency Offset** section.

1. Set all the properties for the acquisition or generation, including the NI-RFSA I/Q Carrier Frequency property or the NI-RFSG Frequency property.



Note Do not set the LO Frequency property. If you previously set the LO Frequency property, reset it.

2. Set the LO Source property to LO IN
3. (Optional) Set the Bandwidth Selection property if you want to apply automatic frequency offset.
 - a. (Optional) Set the NI-RFSA Downconverter Frequency Offset Mode property or the NI-RFSG Upconverter Frequency Offset Mode property to Enabled, or query that they are already set to Enabled.



Note For a given RF frequency, NI-RFSA and NI-RFSG use the same automatic offsets for the PXIe-5841. Therefore, if the LO IN source for NI-RFSA is the LO OUT from NI-RFSG on the same device, you can skip steps 3 through 5.

4. Query the LO Frequency to get the expected frequency.
5. Set the frequency of your LO source to the expected frequency.
6. Set the LO Frequency property to the actual frequency of your LO source in case the source does any rounding.

Specifying LO Power When Using an External LO

To achieve better image and LO leakage performance with an external LO, set the LO IN power level of the PXIe-5841 to the actual output power level of the external LO.

1. Query the actual output power level of the external LO.
2. Set the NI-RFSA or NI-RFSG LO IN Power Level property to the value returned in the previous step.

Peer-to-Peer Data Streaming

The PXIe-5841 supports peer-to-peer (P2P) data streaming using the NI-P2P API. **Peer-to-peer streaming** exchanges data directly between supported devices, bypassing the host computer memory and making applications that require real-time data

transfer between devices possible.

For more information about the following terminology, refer to the NI-P2P peer-to-peer streaming driver documentation.

- **Stream**—The data path connection between two peer-to-peer endpoints. A peer-to-peer stream is independent of the data generation and consumption of the two peers.
- **Endpoint**—The collection of hardware resources needed to support one end of a peer-to-peer stream. Multiple peer-to-peer endpoints may exist in a single device.
- **Writer Peer**—The peer sending the data over the bus to the reader peer.
- **Reader Peer**—The peer receiving the data over the bus from the writer peer.

Vector signal transceivers support simultaneous streaming between both peer-to-peer endpoints.

Configuring a Peer-to-Peer Endpoint

Any driver property that is associated with an instance of an endpoint is an endpoint-based property.

Configuring a Peer-to-Peer Endpoint with NI-RFSG

1. Set the Active Channel property using an appropriate string when configuring endpoint-based properties.
The syntax `"FIFOendpointN"` is used to tell NI-RFSG which RF vector signal transceiver endpoint is being specified, where **N** is an integer starting with 0. If the RF vector signal transceiver supports multiple endpoints, the first is `"FIFOendpoint0"`, the second is `"FIFOendpoint1"`, and so on.
2. Set the P2P Enabled property to True.
3. To determine how many endpoints your RF vector signal transceiver supports, query the Endpoint Count property.
4. Perform either a continuous generation or a finite generation.
 - Continuous generation: Set the Number of Records is Finite property to False
 - Finite generation: Set the Is Finite P2P Generation property to True, and set the Number of P2P Samples to Generate property to the number of samples that should be generated

Configuring a Peer-to-Peer Endpoint with NI-RFSA

1. Set the Active Channel property using an appropriate string when configuring endpoint-based properties.
The syntax "FIFOendpointN" is used to tell NI-RFSA which RF vector signal transceiver endpoint is being specified, where **N** is an integer starting with 0. If the RF vector signal transceiver supports multiple endpoints, the first is "FIFOendpoint0", the second is "FIFOendpoint1, " and so on.
2. Set the P2P Enabled property to True.
3. To determine how many endpoints your RF vector signal transceiver supports, query the P2P FIFO Endpoint Count property.
4. Perform either a continuous acquisition or a finite acquisition.
 - Continuous acquisition: Set the Number of Records is Finite property to False
 - Finite acquisition: Set the Number of Samples is Finite property to True and set the Number of Samples property to the number of samples that should be acquired

Configuring a Peer-to-Peer Stream

To configure a peer-to-peer stream using the NI-P2P API, a writer and reader handle for each endpoint are required.

Use Get Stream Endpoint Handle to generate a reader endpoint handle.

Related information:

- [niRFSA Get Stream Endpoint Handle VI](#)
- [niRFSG Get Stream Endpoint Handle VI](#)

Configuring Flow Control

To configure flow control in NI-RFSA and NI-RFSG, complete the following steps:

1. Call Get Stream Endpoint Handle for your driver to get the RF vector signal transceiver reader handle and the appropriate handle from the writer peer API.
2. Call NI-P2P Create Peer to Peer Stream to configure NI-P2P with information regarding each endpoint to link the endpoints into a stream.
You must specify both a reader and writer endpoint handle.

Starting Peer-to-Peer Generation

If the RF vector signal transceiver begins generating data from a peer-to-peer stream immediately after the first sample is received, the device may not have enough data to continue the generation and can underflow at startup. This problem is due to latency across the bus, and it is especially likely when heavy, possibly unrelated, traffic is on the bus. To avoid underflow at startup, prime the endpoint with data before starting generation so the device has a backlog of data to insulate the generation from the bursty nature of data flowing across the bus.

The following two methods are available for preparing the RF vector signal transceiver endpoint for startup:

- **NI-RFSG StartTrig.P2PEndpointFullness.Level**—Starts generation after the endpoint receives the specified number of samples. If no number of samples is specified, the driver automatically selects an appropriate value. This method should work for most applications and requires the least amount of configuration. This property is used when StartTrig.Type is set to P2P Endpoint Fullness.
- **Manual preparation**—Configure the application so that the writer peer sends data to the RF vector signal transceiver endpoint prior to the RF vector signal transceiver initiating generation.

To prepare for generation manually, complete the following steps:

1. Configure both peers for peer-to-peer streaming.
2. Link both peers with NI-P2P Create Peer to Peer Stream.
3. Enable the stream in either of the following ways:
 - Set `enable_stream` to True with NI-P2P Create Peer to Peer Stream
 - Call NI-P2P Enable Peer to Peer Stream
4. Start the writer peer first so that it begins sending data to the RF vector signal transceiver prior to initiation.
5. Depending on the language you are using, query the following after starting the writer peer to ensure that sufficient data has been transferred to the RF vector signal transceiver.
 - LabVIEW: NI-RFSG Space Available in Endpoint
 - C: NIRFSG_ATTR_P2P_ENDPOINT_FULLNESS_START_TRIGGER_LEVEL or NIRFSG_ATTR_P2P_SPACE_AVAILABLE_IN_ENDPOINT
6. Call NI-RFSG Initiate.

Stopping Peer-to-Peer Generation

Avoid receiving an underflow error by using a supported technique to stop peer-to-peer generation.

Use either of the following techniques to stop peer-to-peer generation without generating an underflow error:

Option	Description
Is Finite P2P Generation	If the RF vector signal transceiver receives the specified number of samples from the writer peer, the NI-RFSG session completes without an underflow error. Use Number of P2P Samples to Generate to specify the sample number
Continuous Generation Using User-Requested Abort	Using the NI-RFSG Abort or NI-RFSG Check Generation Status returns True for <code>Done</code> . However, if the writer peer is not configured carefully for this event, it may overflow because the RF vector signal transceiver is no longer allowing it to send data through the stream.

In either preceding technique, use NI-RFSG Check Generation Status function to determine whether generation is complete. After generation is done, you can abort the generation using NI-RFSG Abort. When generation aborts, all data in the endpoint is cleared to prepare the device for subsequent generations.



Note When using the NI-P2P API to configure the stream, call NI-P2P Flush and Disable Peer to Peer Stream or NI-P2P Disable Peer to Peer Stream and wait for the `done?` parameter from NI-RFSG Check Generation Status to assert before aborting the RF vector signal transceiver. If data is still in the endpoint when NI-RFSG Abort is called, it is cleared to prepare the device for subsequent generations.

Stopping Peer-to-Peer Acquisition

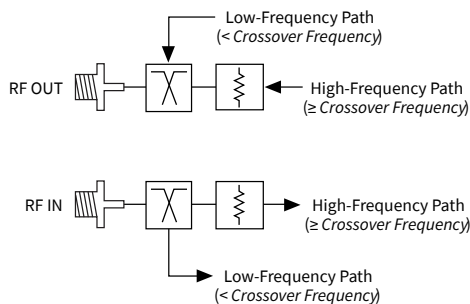
When stopping peer-to-peer acquisition, you must clear the data from the stream before starting a new acquisition.

Use either of the following techniques to clear the stream:

- Call NI-P2P Flush and Disable Peer to Peer Stream before starting a new acquisition. This function automatically clears the remaining data from the stream.
- Configure the reader peer to read the remaining samples in the stream, then call NI-P2P Disable Peer to Peer Stream before starting a new acquisition.

RF Path Transitions

The PXIe-5841 supports acquisition and generation of RF signals from 9 kHz to 6 GHz. For both the RF Input and RF Output channels, the transition from low frequency to high frequency is implemented with a passive diplexer scheme. This passes RF energy below 120 MHz to the low frequency path and RF energy above 120 MHz to the high frequency path. Because the diplexer has a frequency response of its own, it is possible to extend RF frequency coverage above or below 120 MHz in both paths.



While the automatic selection of the signal path using the NI-RFSA or NI-RFSG driver is optimal for most use cases, some applications may require you to select a particular mode of operation. Example situations where this may be useful include measurements or applications sensitive to speed, phase noise, relative gain, or dynamic range.

For example, the high frequency path is implemented with a wide range of amplification and attenuation as a result of more stages within the signal chain. This is useful for applications that may require a significant amount of analog attenuation or amplification. The high frequency path also mixes the RF signal to and from baseband, so phase noise may be degraded compared to a directly sampled signal. Some performance parameters may have more or less impact depending on the nature of the application. Refer to **RF Input Subsystem** and **RF Output Subsystem** for more information about the PXIe-5841 components and frequency paths.

RF Input Signal Path Selection

The primary method of controlling which path is used on the PXIe-5841 is based on center frequency. For RF Input, this frequency is tied to the NI-RFSA IQ Carrier Frequency property in I/Q mode. Refer to the ***NI RF Vector Signal Analyzers Help*** for more information about using the PXIe-5841 in I/Q mode.

The NI-RFSA Downconverter Center Frequency property, NI-RFSA LO Frequency property, or NI-RFSA Downconverter Frequency Offset property can be used to further specify which path the device uses in addition to the NI-RFSA IQ Carrier Frequency property. You can use the NI-RFSA Downconverter Center Frequency property, or, for an external LO, the NI-RFSA LO Frequency property to lock the device into the low-frequency or high-frequency path despite the value specified for I/Q carrier frequency. When you set the NI-RFSA Signal Bandwidth property, NI-RFSA may choose a downconverter center frequency for you. Refer to the ***Automatic Frequency Offset*** section for information on how to automatically offset the downconverter frequency.



Note The downconverter center frequency is coerced to 0 Hz when it is set to any value less than 120 MHz. Refer to the ***Programming Reference*** section in the ***NI RF Vector Signal Analyzers Help*** for more information about the NI-RFSA Downconverter Center Frequency property.

Figure 14. I/Q Carrier Frequency or Center Frequency Set to 100 MHz

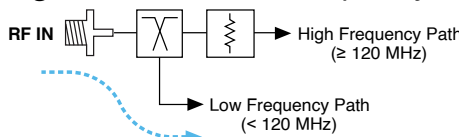


Figure 15. I/Q Carrier Frequency or Center Frequency Set to 125 MHz

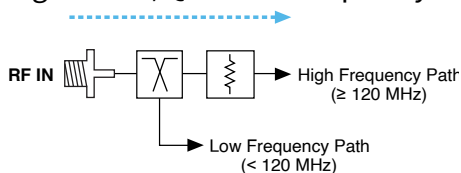


Figure 16. I/Q Carrier Frequency or Center Frequency Set to 100 MHz and Downconverter Center Frequency Set to 125 MHz

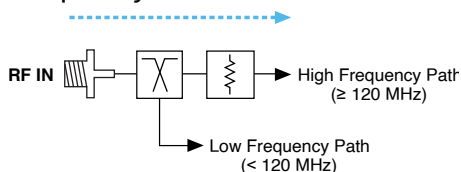
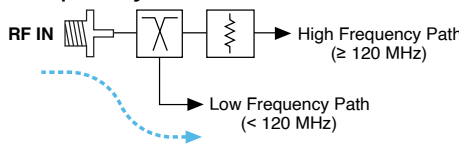


Figure 17. I/Q Carrier Frequency or Center Frequency Set to 150 MHz and Downconverter Center Frequency Set to Less Than 120 MHz



Note Calibrated performance is bound by the equalization bandwidth at a given frequency. As such, the ability to digitally tune to a frequency with the NI-RFSA IQ Carrier Frequency property or NI-RFSA Center Frequency property is restricted to within $\pm 0.5 \times$ Equalized Bandwidth of the downconverter center frequency. You can exceed this limitation by using the NI-RFSA Allow Out of Specification User Settings property.

RF Output Signal Path Selection

Similar to RF Input, the RF Output channel is divided into a low frequency signal path and high frequency signal path. RF energy is diverted to either path through a diplexer with a crossover frequency of 120 MHz. The signal path is automatically configured when the NI-RFSG Frequency property is modified. If you set the NI-RFSG Frequency property value below 120 MHz, the low frequency signal path is used, while frequency values greater than or equal to 120 MHz are generated with the high frequency signal path.

Similar to RF input signal path selection, in I/Q mode you can override the signal path used for RF output with the NI-RFSG Upconverter Center Frequency property. Using the NI-RFSG Upconverter Center Frequency property locks the device into the low-frequency or high-frequency signal path despite the specified frequency value. If you set the NI-RFSG Signal Bandwidth property, NI-RFSG may select an upconverter center frequency for you. Refer to the **Automatic Frequency Offset** section for information on how to automatically offset the upconverter frequency.



Note The NI-RFSG Upconverter Center Frequency property is coerced to 0 Hz when set to any value less than 120 MHz. Refer to the **Programming Reference** section in the *NI RF Signal Generators Help* for more information about the NI-RFSG Upconverter Center Frequency property.

The following figures illustrate examples of configuring the signal path selection based on the NI-RFSG Frequency and NI-RFSG Upconverter Center Frequency properties.

Figure 18. Frequency (Hz) Set to 100 MHz

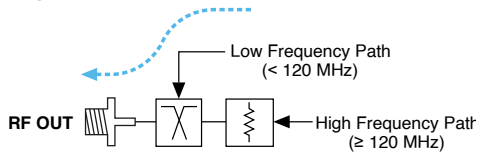


Figure 19. Frequency (Hz) Set to 125 MHz

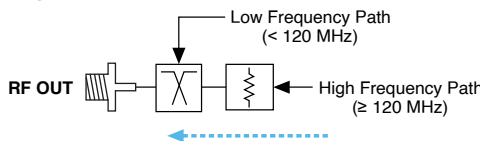


Figure 20. Frequency (Hz) Set to 100 MHz and Upconverter Center Frequency Set to 125 MHz

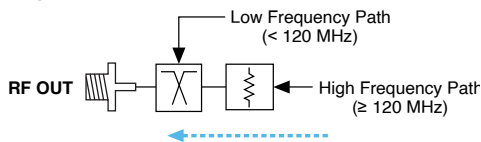
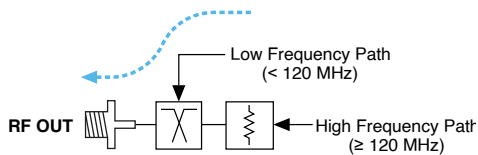


Figure 21. Frequency (Hz) Set to 150 MHz and Upconverter Center Frequency Set to Less Than 120 MHz



Note Calibrated performance is bound by the equalization bandwidth at a given frequency. As such, the ability to digitally tune to a frequency with the Frequency property is restricted to within $\pm 0.5 \times \text{Equalized Bandwidth}$ of the upconverter center frequency. It is possible to exceed this limitation by using the NI-RFSG Allow Out of Specification User Settings property.

Related concepts:

- [RF Input Subsystem](#)
- [RF Output Subsystem](#)

Frequency Offset

You can combine the capabilities of LO tuning and digital frequency shift function through additional digital processing on the FPGA to configure the PXIe-5841 to operate with a downconverter or upconverter frequency offset. Frequency offset tunes

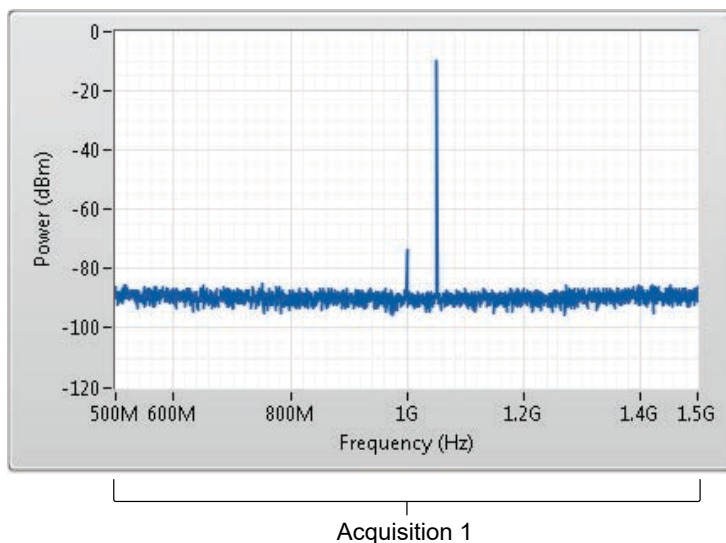
the LO away from the signal. Use the NI-RFSA Downconverter Frequency Offset Mode property or the NI-RFSG Upconverter Frequency Offset Mode property to ask the driver to select an offset, or set the **NI-RFSA Downconverter Frequency Offset** property or the **NI-RFSG Upconverter Frequency Offset** property to configure a user-defined offset.

The use of frequency offset through digital frequency shift is at the expense of restricted instantaneous bandwidth. For example, for an automatic offset, the driver places the LO outside the signal bandwidth which restricts the signal bandwidth to 50% of the device instantaneous bandwidth after subtracting an instrument-dependent LO guard band.

The following figures illustrate a 1,000 MHz direct conversion acquisition versus a 1,000 MHz frequency offset acquisition. Notice the different constraints and distortions in each figure.

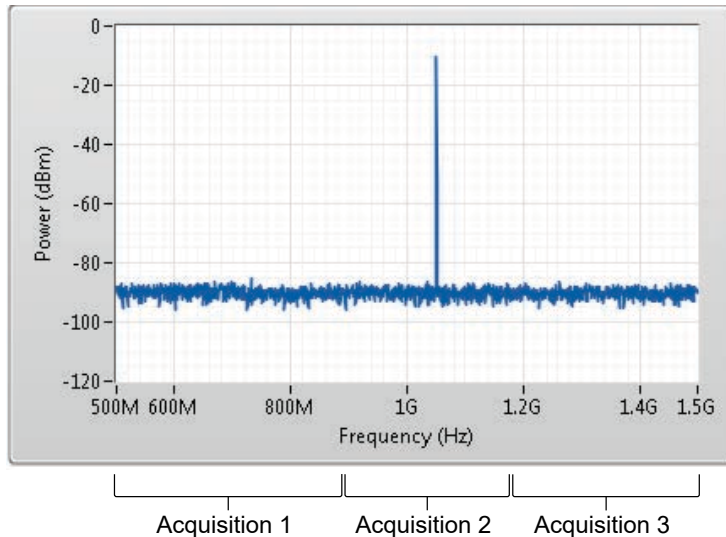
The direct conversion acquisition illustrated in the following figure involves a single acquisition to cover 1,000 MHz using the PXIe-5841. There are distortions present that effectively constrain the dynamic range of the spectrum, however. A primary example of this is the residual LO power, also known as carrier or LO leakage. Residual LO power is an artifact of the quadrature architecture used in direct conversion devices. Use the equation $|x| > (BW/2)$ to calculate the minimum frequency shift needed to avoid LO leakage, where x is the minimum digital frequency shift and **BW** is the required bandwidth.

Figure 22. Spectrum Acquisition with Direct Conversion (Single Span)



The following figure illustrates that you can acquire the same 1,000 MHz using frequency offset with minimal distortion present from direct conversion architectures. However, when you use frequency offset, the effective bandwidth of the device is restricted. The 1,000 MHz acquisition that took a single acquisition using direct conversion takes three acquisitions using frequency offset configuration.

Figure 23. Spectrum Acquisition with Frequency Offset (Multi-Span)



Images can occur when using the PXIe-5841 with frequency offset because of imperfections of the gain imbalance and skew between the I and Q channels. Residual sideband images are suppressed by calibration, but images can still be a complication when using frequency offset.



Note If you use the LO in integer mode, the usable bandwidth may be further reduced by frequency error correction.

Using In-Band Retuning

In-band retuning allows an NI VST to quickly tune between signals of interest or generation within a given instantaneous bandwidth by keeping the LO constant and using digital frequency shifts in the digital data path. The advantages of in-band retuning include reduced settling time for both frequency and amplitude.

In-band retuning can occur only within the supported instantaneous bandwidth of the instrument. You can use digital tuning in combination with LO tuning to more

efficiently sweep over a spectrum for supported applications.

You can use in-band retuning by setting the instrument instantaneous bandwidth and leaving the downconverter center frequency fixed, which significantly reduces reconfiguration and settling times.

To perform in-band retuning, set the relevant property, based on your RF driver as follows, to a value within the instantaneous bandwidth of the VST.

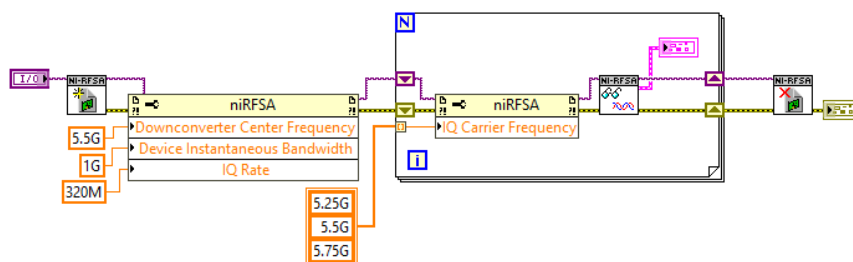
- NI-RFSA: Downconverter Center Frequency
- NI-RFSG: Upconverter Center Frequency

Programming for In-Band Returning

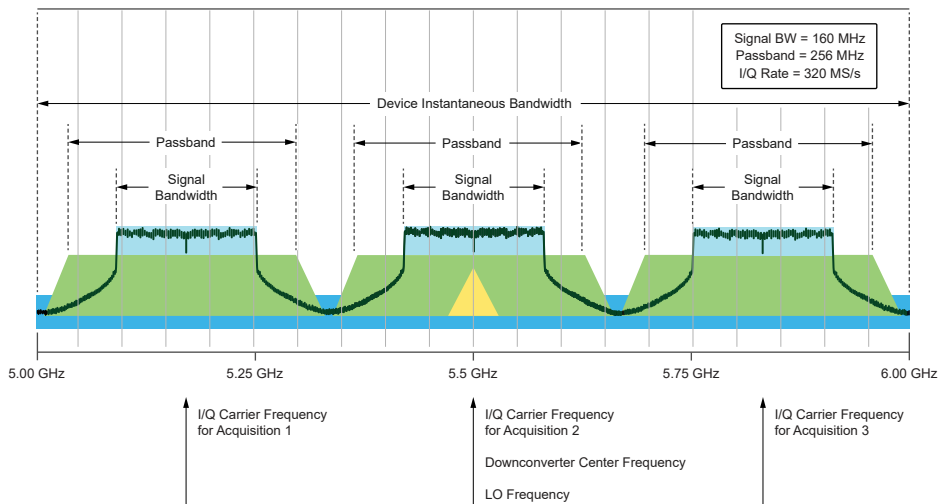
This example shows how to programmatically set an NI VST for in-band retuning during acquisition using NI-RFSA.



Note This example illustrates frequency and bandwidth concepts by portraying a generic example instrument of 1 GHz bandwidth; values and figures may not necessarily reflect the frequencies, bandwidth, or LO step size of your instrument.



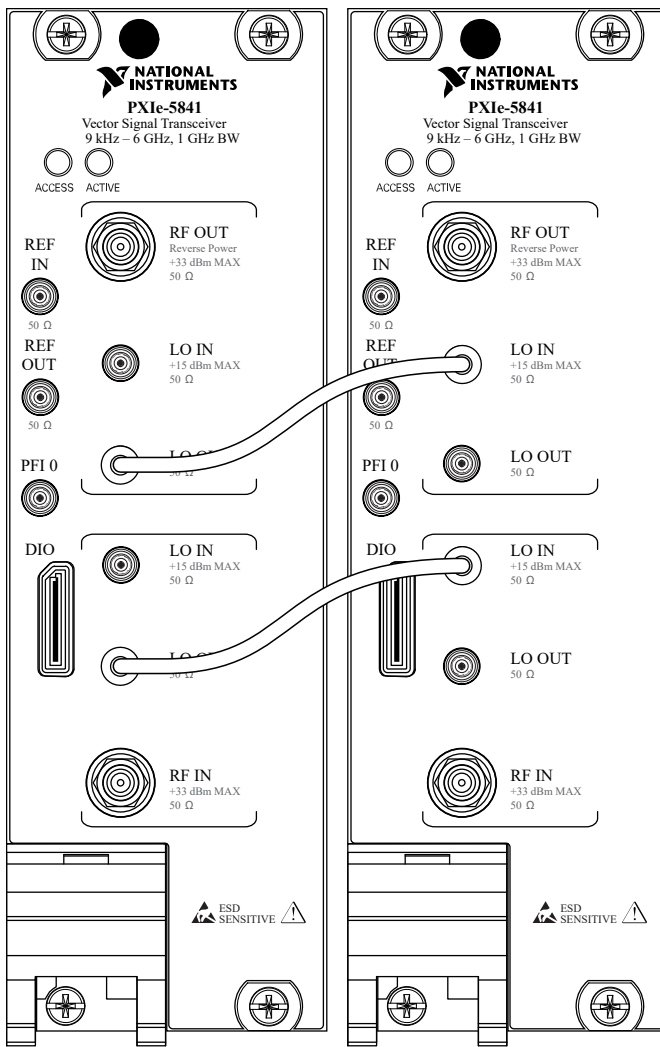
The following figure shows three acquisitions with only digital frequency shift reconfigurations.



Interconnecting Multiple PXIe-5841 RF Channels (Homogeneous Channel Types)

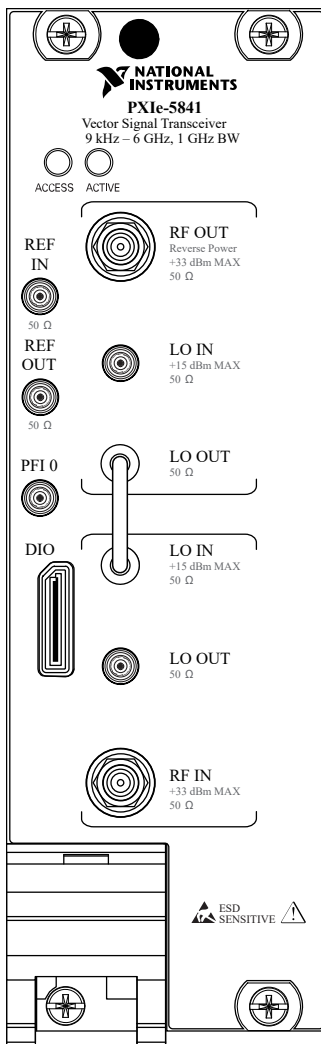
There are certain RF applications that benefit from phase coherency or phase synchronization between multiple RF input and/or RF output channels.

One key aspect of this synchronization is sharing local oscillators (LO) to tune the RF channel. The PXIe-5841 allows for import and export of the individual LOs on each RF input and RF output channel. The following figure illustrates recommended cabling for interconnecting LOs between multiple modules. The connections shown are between channels of similar type (for example, RF input to RF input or RF output to RF output).

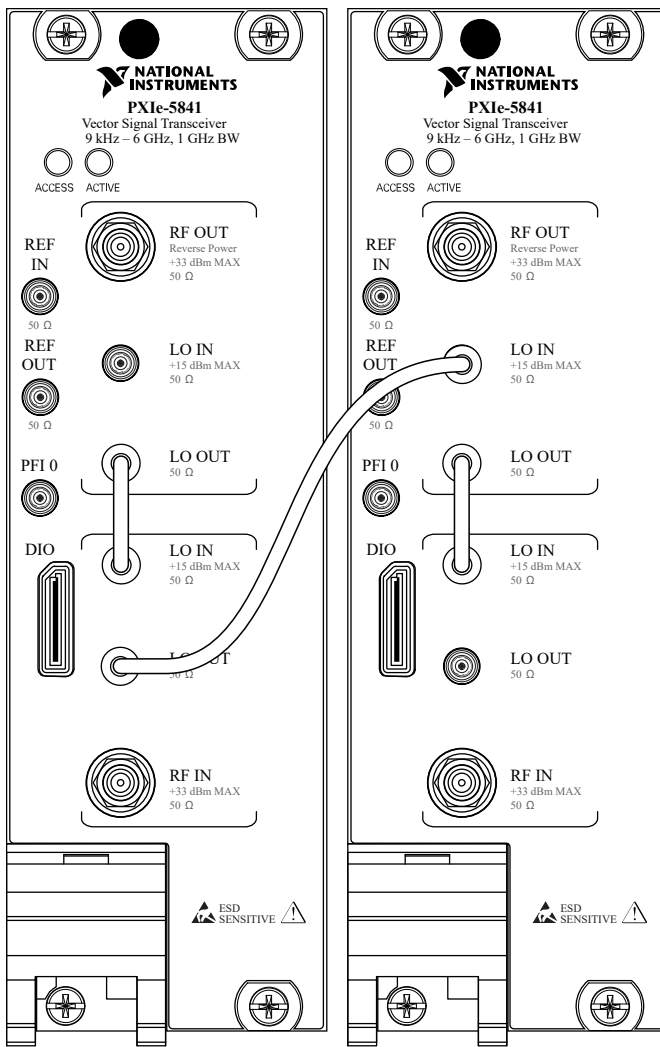


Interconnecting Receive and Transmit Channels

A more specific application may require the RF input and RF output channels to share an LO. The following figure illustrates how to share an LO between the RF input and RF output channels on a single module.



The following figure illustrates how to share an LO between the RF input and RF output channels on multiple modules.



Note The PXIe-5841 does not support synchronization using daisy-chained Reference Clocks, for example, REF OUT to REF IN. For multidevice synchronization, lock all modules to phase-aligned Reference Clocks.



Note The RF output channel expects a 3 dBm LO power level above 3.6 GHz, and the RF input channel expects a 0 dBm LO power level. This expectation can affect performance for frequencies above 3.6 GHz and should be considered when implementing applications.

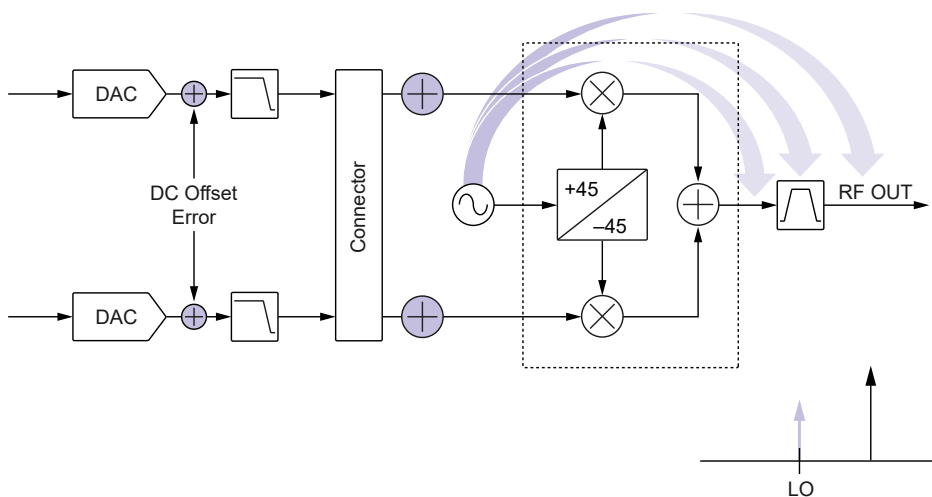
Quadrature Mixing (Direct Conversion)

Quadrature mixing (direct conversion) allows for the complex modulation or demodulation of signals as represented by two orthogonal signals, I and Q.

Quadrature mixing has the added benefit of an increased complex bandwidth that allows for net bandwidths up to the sample rate of the DAC.

Analog quadrature mixing exposes the signal to impairments because of the imperfections of the circuitry involved. These imperfections can create distortion products or unwanted signals, which may degrade modulation and demodulation quality. The following section illustrates key impairments involved in quadrature mixing and how they affect the respective signal.

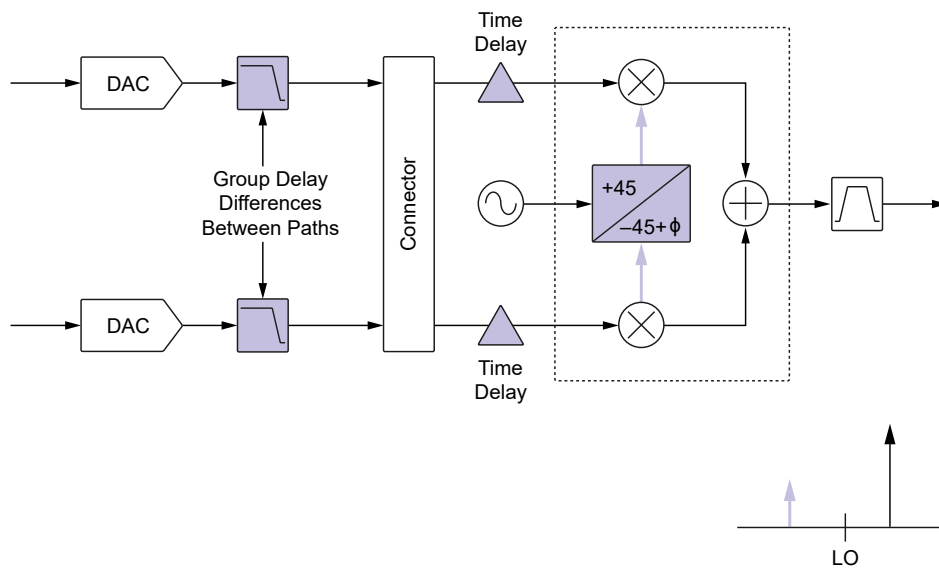
The following figure shows a typical quadrature transmission device. The shaded sections represent the presence of several impairments, including DC offsets within the various baseband sections and LO signal power leaking into the RF path because of insufficient isolation.



Offsets and the lack of LO isolation can cause unwanted distortion, defined as residual LO power, which is always centered on the tuned LO frequency. Digital equalization schemes can help suppress residual LO power. This suppression improves the performance of the system for many modulation schemes.

For an I/Q waveform to be represented perfectly, I and Q should have a gain ratio of 1 and be 90 degrees out of phase (orthogonal). Several components in an analog quadrature mixing system can corrupt the gain or the phase offset. These impairments are known as gain imbalance and skew.

The shaded sections in the following figure represent several components that affect the phase offset and create skew.



The presence of gain imbalance or skew-related impairments creates a distortion at the negative complex frequency of the tone of interest. This distortion is referred to as the residual sideband image or the image. Wideband applications drive the necessity to suppress this distortion to avoid unwanted signals that can corrupt transmitted or received data. On the PXIe-5841 calibrated digital equalization suppresses most of the image to improve the performance of the system.

Configuring Attenuator Hold

Attenuator hold—holding the power level fixed while adjusting the digital gain—ensures the analog path is not altered while the instrument is in the Generation state.

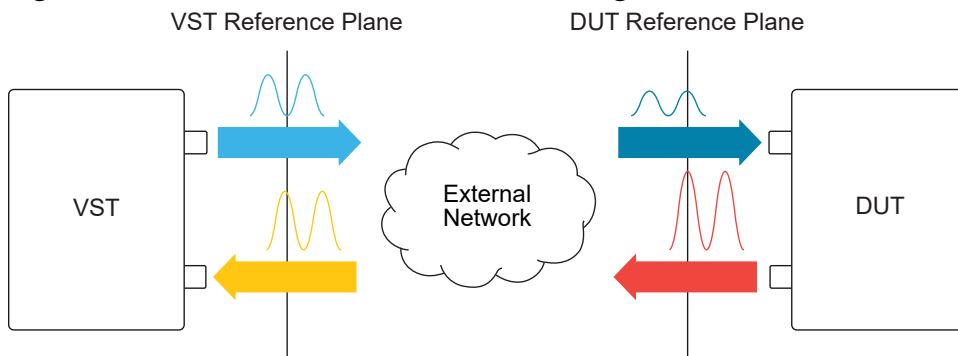
To configure attenuator hold, adjust the digital gain by completing the following steps:

1. Configure the NI-RFSG Power Level (dB) property to the max desired output power level.
2. Adjust the digital gain using the NI-RFSG Digital Gain (dB) property to reach the final desired output power level.
3. Initiate generation.
4. Adjust the digital gain to adjust the output power as desired.

De-embedding Overview

NI RF instrument performance specifications are for signals at the instrument's ports. However, you must also account for external cabling and components between the instrument and DUT, which also affects input and output signals.

Figure 24. Effect of the External Network on Signals



De-embedding is the process of removing the effects of these components on an input or output signal, so the accuracy of the instrument is extended to the DUT.

The NI-RFSA and NI-RFSG instrument drivers can de-embed measurements and generations using your characterization data. To use this feature, first characterize the external components with an S-parameter table. Once the tables are passed to the driver, they can be applied to move the reference plane from the instrument port to the DUT port.

The NI driver compensates for the effects of the networks by modifying the analog path and adjusting the data with digital signal processing (DSP).

When using NI-RFSG, the signal seen at the DUT port matches the requested signal. When using NI-RFSA, the data returned from the driver reflects the signal at the DUT port rather than the signal seen at the instrument's input port.

S-parameters

S-parameters characterize the effects of a linear network on a signal when it passes from one port of the network to another port. For purposes of de-embedding for VST operation, the instrument driver compensates for the effects of the networks by modifying the analog path and adjusting the data with DSP.

S-parameters of the de-embedding network can be obtained by either extracting them using a vector network analyzer (VNA) or by simulating the network in an RF computer assisted design (CAD) tool.

NI-RFSA and NI-RFSG support de-embedding of two port networks. Two port networks are characterized by four s-parameters. S-parameters are written in the format ***Snm***, where ***n***, designates the receiving port, and ***m***, designates the driving port. Each S-parameter is referred to as a gain or reflection term.

The following table illustrates which terms characterize gain or reflection of the network.

Table 18. S-parameters

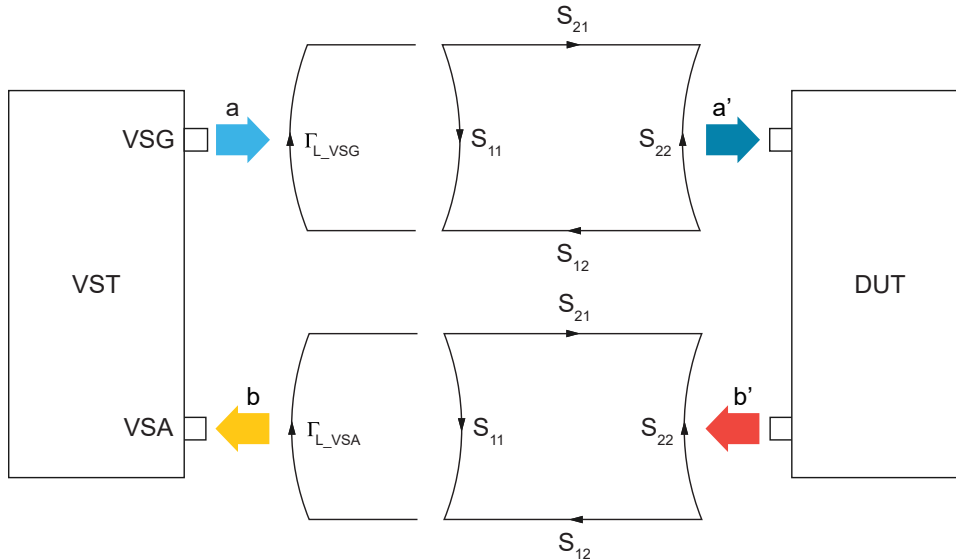
S-parameter	Description
S_{11}	reflection (input reflection)
S_{12}	gain (forward transmission)
S_{21}	gain (reverse isolation)
S_{22}	reflection (output reflection)

Gain terms, S_{12} and S_{21} , model the gain an input signal experiences as it passes through the network to the receiving port. The reflection terms, S_{11} and S_{22} , are used when computing the effects of signal reflection due to impedance mismatch between the network's ports and driving ports.

The following figure provides a visualization of the external network as an S-parameter model. Signal ***b'***, output from the DUT, passes through the external network and arrives at the VST as signal ***b***. Signal ***a***, generated by the VST, passes through the network and arrives at the DUT as signal ***a'***. The following are reflection terms for the instrument ports:

- Γ_{L_VSA}
- Γ_{L_VSG}

Figure 25. S-parameter modeling of an external network for NI-RFSA and NI-RFSG



Scalar De-embedding

Scalar de-embedding compensates only for the gain term of the S-parameters.

In the previous figure, the DUT output is connected to port2 of the external network and port1 is connected to the NI device port. By convention, the S12 parameter specifies the gain due to the network, and applied to the signal **b'** output from the DUT toward the VSA port. NI-RFSA computes the signal at the DUT port using the following formula:

$$b' = \frac{b}{S_{12}}$$

To acquire the output signal of the DUT and not the signal seen at the NI instrument's port, NI-RFSA compensates for the network by amplifying the signal. The driver makes DSP adjustments to compensate for any remaining effects that analog path adjustments do not compensate for. NI-RFSA returns data reflecting the amplitude of the signal at the DUT.

Similarly, when using NI-RFSG scalar de-embedding, the driver computes the output signal at the DUT port using the following formula:

$$a' = S_{21} \times a$$

The receiving port in this case is port2 at the DUT and the input port is port1 at the NI instrument's port. NI-RFSG compensates for the network via analog path adjustments and DSP so that the signal you request arrives at the DUT with the correct amplitude.

Unsupported Software Features

The following NI-RFSA and NI-RFSG driver software features are not supported on the PXIe-5841.

Unsupported Feature	NI-RFSA	NI-RFSG
Pulse shaping	—	✓

Restrictions on Scripts

- `if else` is not supported.
- `break` instruction is not supported.
- `stream` instruction is not supported.
- The maximum compiled script size is 6,100 instructions. Each script language instruction requires roughly one compiled instruction, plus one instruction for each marker.
- Nested repeats are only allowed if the outer `repeat` is a `repeat forever` instruction, and there are no other instructions preceding it or past the end of the `repeat` block. You can only use a single nesting level.
- Markers are allowed in `generate` and `finite wait` instructions. They are not allowed in `conditional wait` instructions.

Self-Calibration

Self-calibration adjusts the PXIe-5841 for variations in the module environment using an onboard high-precision calibration tone.

PXIe-5841 modules are externally calibrated at the factory; however, you should perform a self-calibration in any of the following situations:

- After first installing the PXIe-5841 into your chassis
- After any module in the chassis is installed, uninstalled, or moved
- When the system is in an environment where the ambient temperature varies or the module temperature has drifted more than ± 5 °C from the temperature at the last self-calibration
- To periodically adjust for small performance drifts that occur with product aging



Note Allow the module to warm up for 30 minutes after the PXI Express chassis has powered on and the operating system has completely loaded before you use the module or perform self-calibration.



Note Disconnect external RF and LO (RF IN/RF OUT and LO IN/LO OUT) port cables prior to self calibration for best performance. If this is not possible then make sure external devices are turned off or in idle state and not transmitting. Performing self-calibration while connected in direct loopback configuration from RF OUT to RF IN is not recommended.

Performing a Device Self-Calibration Using the NI-RFSA or NI-RFSG SFP

Perform the self-calibration using the NI-RFSA or NI-RFSG Soft Front Panel (SFP). The NI-RFSA and NI-RFSG SFPs perform the same self-calibration, so you only need to use one SFP when you self-calibrate the module.

1. Launch the NI-RFSA or NI-RFSG Soft Front Panel by navigating to **Start » All Programs » National Instruments » <Driver> » <Driver> Soft Front Panel**.

2. Click **Device/System**.
3. Click **Calibration**.
4. Click **Self Calibration**.
5. Click **OK** to run the self-calibration.

Self-Calibrating the PXIe-5841 with PXIe-5655

Perform the following steps to self-calibrate the modules:

1. Associate the PXIe-5841 and PXIe-5655 in MAX.
2. Complete the steps described in the preceding ***Performing a Device Self-Calibration Using the NI-RFSA or NI-RFSG SFP*** section.

The composite self-calibration data of the PXIe-5841 with the PXIe-5655 does not replace the standalone PXIe-5841 calibration data. These two sets of calibration data can exist simultaneously. If you use the PXIe-5841 only, the PXIe-5841 uses its last unassociated self-calibration. If you use the PXIe-5841 associated with a PXIe-5655, the PXIe-5841 uses its last associated self-calibration.



Note When performing a standalone self-calibration, disconnect the PXIe-5655 from the PXIe-5841 or reset the PXIe-5655 in MAX to make sure the module is not generating.

Cleaning the PXIe-5841 System

NI recommends the following to clean and maintain your instrument's system:

- Clean the fan filters on the chassis regularly to prevent fan blockage and to ensure efficient air circulation. Cleaning frequency depends on the amount of use and the operating environment. For specific information about cleaning procedures and other recommended maintenance, refer to the chassis user documentation.
- Clean the hardware with a soft, nonmetallic brush. Make sure that the hardware is completely dry and free from contaminants before returning it to service.

Troubleshooting

If an issue persists after you complete a troubleshooting procedure, search our KnowledgeBase for additional information our technical support engineers create as they answer common user questions and resolve unexpected issues.

What Should I Do if My Hardware Doesn't Appear in Hardware Configuration Utility or MAX?

1. Check that the ACCESS/ACC LED on the hardware is solid green.
2. Check if the connection between the hardware and software needs to be refreshed.

Software	Description
Hardware Configuration Utility	Click the refresh button ().
MAX	a. In the MAX configuration tree, expand Devices and Interfaces. b. Expand the Chassis tree to see the list of installed hardware and press F5 to refresh the list.

3. If the instrument is still not listed, power off the system, ensure that all hardware is correctly installed, and restart the system.
4. Navigate to the Device Manager by right-clicking the **Start** button and selecting **Device Manager**.
5. Depending on your controller type, verify Device Manager settings.

Controller Type	Description
PXI controller	a. Verify that a National Instruments entry appears in the system device list b. If error conditions appear in the list, right-click the module you are using in the Device Manager and select Update Driver.

Controller Type	Description
MXI controller	Right-click PCI-to-PCI Bridge and select Properties from the shortcut menu to verify that the bridge is enabled.

- Restart your computer.

Why Is the ACCESS LED Off When the Chassis Is On?

The LEDs may not light until the module has been configured in Hardware Configuration Utility or MAX. Before proceeding, verify that the module appears in Hardware Configuration Utility or MAX.

If the module appears in Hardware Configuration Utility or MAX but the ACCESS LED fails to light after you power on the chassis, a problem may exist with the chassis power rails, the hardware module, or the LED.



Notice Apply external signals only while the hardware is powered on. Applying external signals while the hardware is powered off may cause damage.

- Disconnect any signals from the PXI Express module front panel.
- Power off the PXI Express chassis.
- Remove the module from the PXI Express chassis and inspect it for damage.



Notice Do not reinstall a damaged device.

- Install the module in a different PXI Express chassis slot from which you removed it.
- Power on the PXI Express chassis.
- Restart your computer.
- Verify that the module appears in Hardware Configuration Utility or MAX.

Fundamental RF Concepts

Understanding RF concepts and the concepts that underpin NI's RF hardware will help you be successful with your instrument.

Common NI RF Terminology

Refer to the following list for definitions of common NI terms related to RF hardware and software-configured settings for the PXIe-5841 and used throughout documentation.

Table 19. Common Terminology Definitions

Term	Definition
Center Frequency	Refers to the IQ Carrier Frequency property in NI-RFSA, the Frequency property in NI-RFSG, and the Center Frequency property in RFmx.
Frequency Offset Mode is Automatic	Refers to the NI-RFSA Downconverter Frequency Offset Mode property or NI-RFSG Upconverter Frequency Offset Mode property set to Automatic. Equivalent to the term Offset Mode is Automatic. The PXIe-5841 uses a direct conversion architecture. Offset Mode allows the instrument to operate in low IF mode, which increases the separation between the signal of interest and the residual sideband image and residual LO leakage power. However, low IF mode limits the available instantaneous bandwidth. A setting of Automatic allows the driver to set Offset Mode to Enabled when the signal bandwidth is configured as small enough to allow it. You can read back the Offset Mode to determine if the driver selected Enabled or User-Defined. Automatic is the default value. NI recommends keeping Offset

Term	Definition
	Mode set to the default value.
<i>Frequency Offset Mode is Enabled</i>	Refers to the NI-RFSA Downconverter Frequency Offset Mode property or NI-RFSG Upconverter Frequency Offset Mode property set to Enabled. Equivalent to the following terms or conditions: • <i>Offset Mode is Enabled</i> • <i>Signal Bandwidth ≤ Maximum Offset Bandwidth</i> The PXIe-5841 uses a direct conversion architecture. Offset Mode allows the instrument to operate in low IF mode, which increases the separation between the signal of interest and the residual sideband image and residual LO leakage power.
<i>Frequency Offset Mode is User-Defined</i>	Refers to the NI-RFSA Downconverter Frequency Offset Mode property or NI-RFSG Upconverter Frequency Offset Mode property set to User-Defined. Equivalent to the following terms or conditions: • <i>Offset Mode is User-Defined</i> • <i>Signal Bandwidth > Maximum Offset Bandwidth</i> The PXIe-5841 uses a direct conversion architecture. Offset Mode set to User-Defined allows the instrument to operate with maximum instantaneous bandwidth.
<i>Onboard</i>	With respect to local oscillators, refers to the value of the LO Source property. A value of Onboard configures the hardware to use the integral local oscillator of the instrument, that of the associated PXIe-5655 module(s). With respect to reference clocks, refers to the value of the NI-RFSG Reference Clock Source or NI-RFSA Ref Clock Source

Term	Definition
	properties. A value of Onboard Clock configures the hardware to use the integral reference clock of the instrument, that of the associated PXIe-5655 module(s).
dBr	• For input—Power of an acquired signal with respect to the instrument's configured reference level. For example, if the reference level is set to -10 dBm and the acquired tone is -17 dBm, the signal is said to be at -7 dBr. • For output—Generated power of a CW with respect to the instrument's peak power setting. For example, with a peak power level setting of +5 dBm and a -3 dBr setting, the power of the generated CW is +2 dBm.

Carrier Wave

The **carrier wave** is a sinusoidal wave that is modulated by a **message signal**, $m(t)$, prior to transmission.

The message signal modifies the carrier wave amplitude, frequency, or phase. The modulation process may vary these characteristics individually or in combination. The modified carrier signal, also referred to as the **modulated wave**, is transmitted to a receiver.

The process of **demodulation** recovers a replica of the original message signal from the carrier wave. In advanced communication systems, the carrier may be a moving signal, also known as a **spread spectrum signal**. When the characteristics of the carrier signal are deterministic and known by the receiver, virtually any type of carrier signal can be used.

The nominal frequency of the carrier wave is the **carrier frequency**. The carrier frequency is the center frequency of the spread spectrum signal.

Message Signal

The **message signal** contains the data for transmission.

The message signal is used to modulate the carrier wave to create the modulated wave for transmission. The message signal data is recovered from the modulated wave by a process of demodulation.

The message signal is often referred to as the **baseband signal**.

Modulation

Modulation is a process that alters the characteristics of a carrier wave according to information in the message signal to generate and transmit a modulated wave. There are several different types of modulation.

The unmodulated carrier is represented by the following equation:

$$v(t) = A_c \cos(2\pi f t + \phi)$$

The amplitude-modulated carrier signal is represented by the following equation:

$$v(t) = (m(t) + A_c) \cos(2\pi f t + \phi)$$

The frequency-modulated carrier signal is represented by the following equation:

$$v(t) = A_c \cos\left(\left(m(t) + 2\pi f\right)t + \phi\right)$$

This yields the following equation:

$$A_c \cos\left(2\pi f t + \phi + \int m(t) dt\right)$$

The phase-modulated carrier signal is represented by the following equation:

$$v(t) = A_c \cos(2\pi ft + \phi + m(t))$$

where

- $m(t)$ is the time-varying modulation
- A_c is the amplitude of the carrier wave
- f is the frequency of the carrier wave

I/Q Modulation

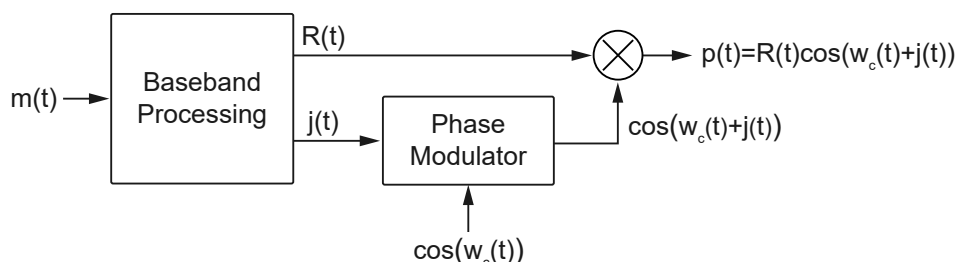
I/Q is a common way to represent message signals in modern communication systems.

There are two fundamental ways (degrees of freedom) to produce a modulated wave from a carrier wave: you can perturb either its amplitude or its phase. I/Q signaling allows you to simultaneously change both amplitude and phase. Using both degrees of freedom when modulating a carrier allows greater spectral efficiency with regard to bits per hertz of channel bandwidth (for a given bits per second transmission rate) compared to using only one degree of freedom.

To simultaneously perturb both the amplitude and phase of a carrier, two canonical forms of frequency-translating transmitters use the two degrees of freedom in modulation simultaneously. These canonical forms are referred to as **canonical polar modulation** and **canonical rectangular modulation**.

Canonical Polar Modulation

Figure 26. Canonical Polar Modulation



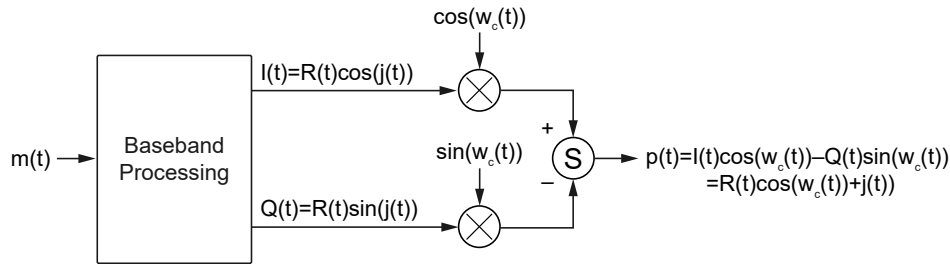
In polar modulation, the message signal is split into two components.

- $R(t)$ controls the carrier wave amplitude changes

- $\phi(t)$ drives the carrier wave phase changes

Canonical Rectangular Modulation

Figure 27. Canonical Rectangular Modulation



In rectangular modulation, the message signal is split into two components.

- $I(t)$ controls the in-phase carrier wave changes
- $Q(t)$ controls the quadrature-phase carrier wave changes

Mathematical Relationship between Canonical Modulation Forms

The modulated outputs of the two canonical forms are mathematically and physically equivalent. To show their equality, you can compare and translate between the polar and rectangular representations.

Begin with the following equation:

$$R(t)\cos(\omega_c(t) + \phi(t))$$

Plug in the following trigonometric identity:

$$\cos(\alpha + \beta) = \cos \alpha \cos \beta - \sin \alpha \sin \beta$$

to yield the following equation:

$$R(t)[\cos\omega_c(t)\cos\phi(t) - \sin\omega_c(t)\sin\phi(t)]$$

Simplify the preceding equation to arrive at the following relationship:

$$I(t)\cos(\omega_c(t)) - Q(t)\sin(\omega_c(t))$$

where

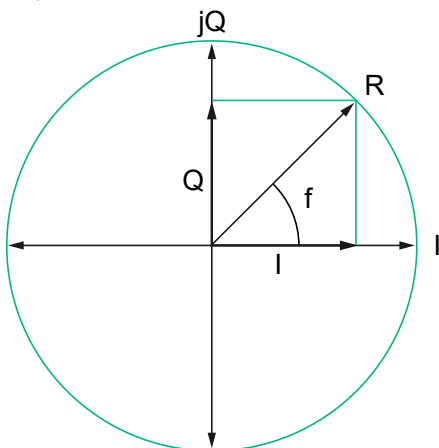
- $I(t) = R(t)\cos(\phi(t))$
- $Q(t) = R(t)\sin(\phi(t))$
- $\omega_c = 2\pi f_c$
- f_c is the carrier frequency in Hz

The rectangular form of modulation, often called ***I/Q modulation***, has become popular for certain technical reasons.

In the rectangular modulation figure, the real baseband signals of $I(t)$ and $Q(t)$ are created (in some way) to contain all the information of message input $m(t)$.⁵ Because the carriers $\cos(\omega_c(t))$ and $\sin(\omega_c(t))$ are orthogonal functions, we use the terminology of ***quadrature modulation***, where the signal applied to the cosine mixer is called the In-phase component (I), and the signal applied to the sine mixer is called the Quadrature-phase component (Q). The I and Q designations are useful because the $I(t)$ baseband signal is applied to the cosine mixer, and the $Q(t)$ baseband signal is applied to the sine mixer.

The analytical relationship between the polar form ($R(t)$ and $\phi(t)$) of the baseband signal and Cartesian form ($I(t)$ and $Q(t)$) of the baseband signal is shown in the following figure.

Figure 28. Relationship Between $R(t)$ and $\phi(t)$



The diagram is a "snapshot in time" of the complex (analytical) baseband envelope $g(t)$. Because the diagram is for an arbitrary instant in time, the independent time

5. The polar $R(t)$ and $\phi(t)$ also contain the information of $m(t)$.

variable t is dropped. The amplitude of Q is projected onto the imaginary (j) axis and the amplitude of I is projected onto the real axis.

I/Q Receiver

An I/Q receiver recovers a signal containing both amplitude and phase modulation.

For the incoming signal, $p(t)$, the normalized outputs of the cosine and sine mixers are respectively:

$$I(t) + I(t)\cos(2\omega_c t) + Q(t)\sin(2\omega_c t)$$

$$Q(t) + Q(t)\sin(2\omega_c t) + I(t)\cos(2\omega_c t)$$

After the mixers, the lowpass filters remove all the high-frequency $2\omega_c t$ terms, leaving the recovered $I(t)$ and $Q(t)$ signals.

The diagrams and discussion in this topic are ideal. They imply that the original message signal $m(t)$ is recovered along the transmitter-path-receiver chain. This implication is useful when focusing purely upon the signaling method. In reality, however, the receiver returns a version of $m(t)$ corrupted by noise and distortion impairments.

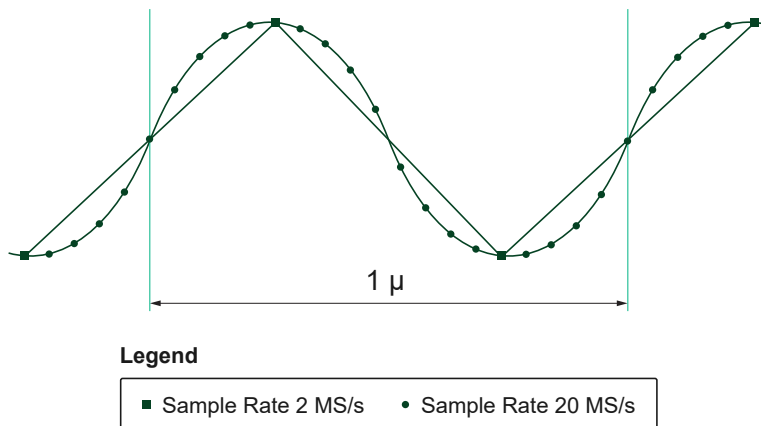
The vector signal analyzer measures the unit under test (UUT) noise and distortion impairments and determines if the performance of the UUT is sufficient for the desired transmission scheme. Because the I/Q signal representation is relatively straightforward (and also has practical and physical application in communication systems), it is conceptually used to evaluate the noise and distortion performance of these communication systems.

Sample Rate

Sample rate is the rate at which digital data is transferred from the memory to the digital-to-analog converter (DAC).

According to Shannon's Sampling theorem, a digital waveform must be updated at least twice as fast as the bandwidth of the signal to be accurately generated. Ideally, a

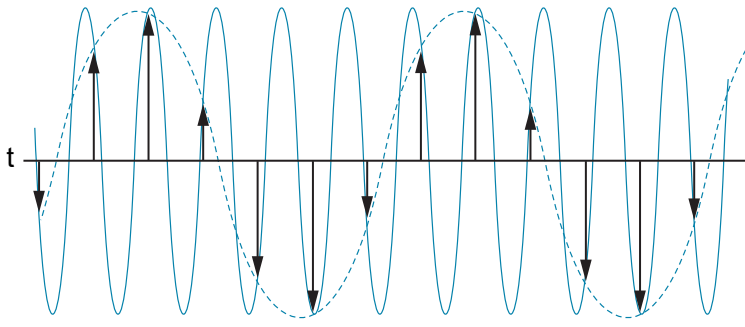
sample rate many times greater than the frequency of the signal produces accurate waveforms. A higher sample rate also captures more waveform details. The following figure illustrates a 1 MHz sine wave generated by a sampled 2 MS/s DAC and a 20 MS/s DAC. The faster DAC generates 20 points per cycle of the expected signal compared with 2 points per cycle with the slower DAC. In this example, the higher sample rate more accurately defines the waveform shape.



Nyquist and Shannon's Sampling Theorems

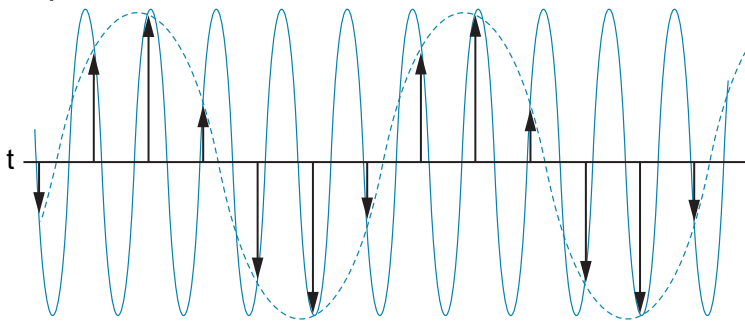
The Nyquist theorem concerns digital sampling of a continuous time analog waveform, while Shannon's Sampling theorem concerns the creation of a continuous time analog waveform from digital, discrete samples.

- Nyquist Theorem**—The Nyquist theorem states that an analog signal must be sampled at least twice as fast as the bandwidth of the signal to accurately reconstruct the waveform; otherwise, the high-frequency content creates an alias at a frequency inside the spectrum of interest (passband). An alias is a false lower frequency component that appears in sampled data acquired at too low a sampling rate. The following figure shows a 5 MHz sine wave digitized by a 6 MS/s analog-to-digital converter (ADC). In this figure, the solid line represents the sine wave being digitized, while the dotted line represents the aliased signal recorded by the ADC at that sample rate.



The 5 MHz frequency aliases back in the passband, falsely appearing as a 1 MHz sine wave.

- Shannon's Sampling Theorem**—Shannon's Sampling theorem states that a digital waveform must be updated at least twice as fast as the bandwidth of the signal to be accurately generated. The same image that was used for the Nyquist example can be used to demonstrate Shannon's Sampling theorem. The following figure shows a desired 5 MHz sine wave generated by a 6 MS/s DAC. The solid line represents the desired waveform, and the arrows represent the digitized samples that are available to recreate the continuous time 5 MHz sine wave. The dotted line indicates the signal that would be seen, for example, with an oscilloscope at the output of a DAC.



In this case, the high-frequency sine wave is the desired signal, but was severely undersampled by only being generated by a 6 MS/s DAC; the actual resulting waveform is a 1 MHz signal.

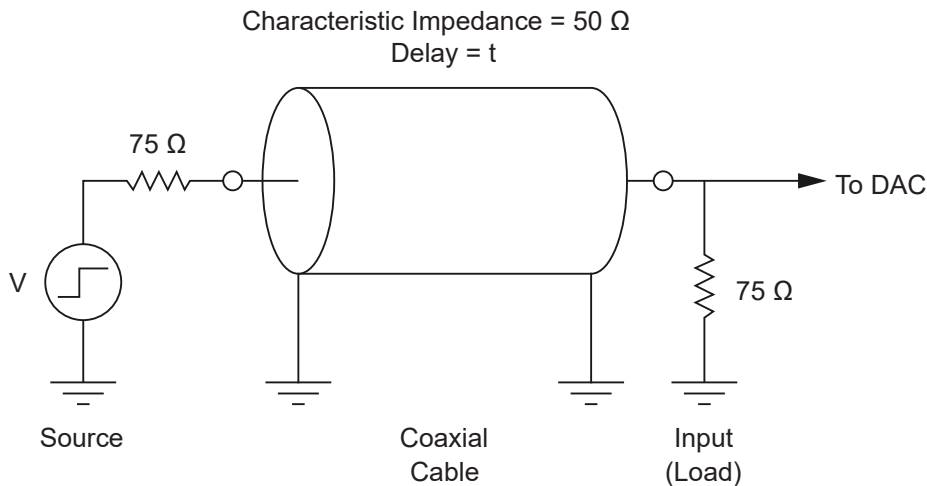
In systems where you want to generate accurate signals using sampled data, the sampling rate must be set high enough to prevent aliasing.

Impedance Matching

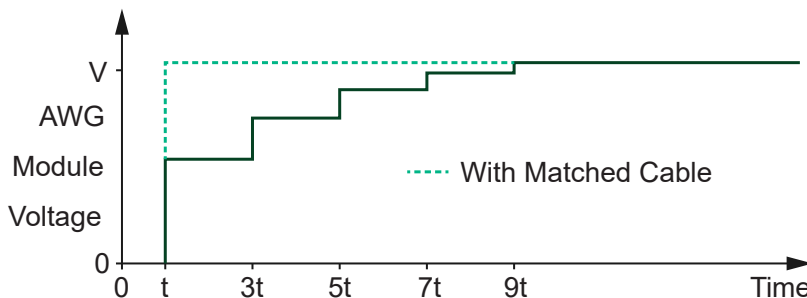
When broadband signals are carried on transmission lines of any significant length, the

presence of impedance discontinuities or mismatches degrade the amplitude and phase accuracy, as well as the temporal fidelity, of waveforms generated with a signal generator. To minimize signal reflections, the source and load impedances should equal the characteristic impedance of the transmission line.

One of the most common mismatch errors encountered in such measurements is shown in the following figure:



In this example, selectable source impedances are provided at the signal generator outputs to accommodate the most popular coaxial cable characteristic impedances: $50\ \Omega$ and $75\ \Omega$. The following figure shows what happens when, as in this example, a coaxial cable of the wrong characteristic impedance ($50\ \Omega$) is used with $75\ \Omega$ source and load impedances:



The pulse encounters impedance mismatches at each end of the cable, causing the pulse to be partially reflected. The reflected pulse traverses the cable back and forth

numerous times, diminishing at each end by the reflection coefficient, Γ .

$$\Gamma = \frac{V_r}{V_i} = \frac{Z_t - Z_0}{Z_t + Z_0}$$

where

- V_r = reflected voltage
- V_i = incident voltage
- Z_t = terminating impedance
- Z_0 = characteristic impedance

The resulting voltage waveform is distorted by the asymptotic decay of the reflected pulse as shown, exaggerated for visual effect. Impedance discontinuities of smaller magnitude and/or duration have correspondingly smaller effects. Also displayed is the waveform that results when a cable of matched impedance (75 Ω) is used.

Mismatch Uncertainty

Impedance matching is also important for preserving the absolute delivered power from a device. The accuracy with which power can be delivered is limited by mismatch error. The mismatch error in a z_0 system can be shown to be bounded by:

$$\frac{(1 - |\Gamma_L|^2)}{(1 + |\Gamma_L| + |\Gamma_G|)^2} \leq \text{mismatch error} \leq \frac{(1 - |\Gamma_L|^2)}{(1 - |\Gamma_L| + |\Gamma_G|)^2}$$

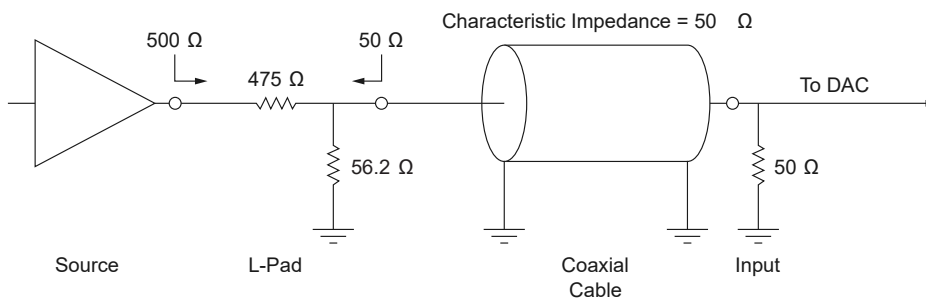
where

- Γ_L = load reflection coefficient
- Γ_G = generator reflection coefficient

The denominator term represents mismatch uncertainty, which is a fundamental limit to the power transfer accuracy that can be achieved across a mismatched junction.

Resistive Matching

RF signal generators with low or high-source impedance can be matched with a resistor placed in series (shunt) such that the total source impedance (admittance) is matched to the cable characteristic impedance (admittance). RF signal generators that are not capable of driving the cable impedance directly can be coupled through a matching L-pad. In this case, the RF signal generator sees an approximately $500\ \Omega$ load, while the source impedance presented to the cable is $50\ \Omega$, as shown in the following figure:

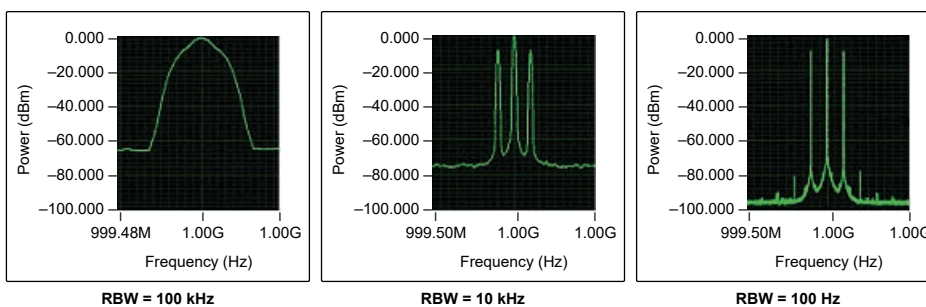


High-frequency components and layout techniques should be used throughout to minimize parasitic effects.

Resolution Bandwidth

For NI VSAs and VSTs, the **resolution bandwidth (RBW)** determines the fast-Fourier transform (FFT) bin size, or the smallest frequency that can be resolved.

The following graphs represent the same signal with varying RBW.



The smaller RBW, on the right, has much finer resolution which allows the sidebands to be visible. Finer resolution requires a longer acquisition time. When acquisition time is a factor and the display needs to be updated rapidly or when the modulation bandwidth is wide, you can use a larger RBW. RBW and acquisition time are inversely proportional.

The following table shows the advantages and disadvantages of both larger and smaller RBWs:

Characteristic	Larger RBW	Smaller RBW
FFT size	Smaller	Larger
Number of samples	Fewer	More
Measurement speed	Faster	Slower
Ability to resolve tones	Often unable to resolve two closely spaced tones in a spectrum	Tones are easily resolved

In FFT-based (digital) spectrum analyzers and vector signal analyzers, RBW is inversely proportional to the number of samples acquired. By taking more samples in the time domain, or making the acquisition time longer while keeping the sampling rate the same, the RBW is lowered. The result is more bins in the same span and improved frequency resolution.

The FFT process is equivalent to passing a time-domain signal through a bank of bandpass filters with center frequencies corresponding to the frequencies of the FFT bins. For wide sweeps, a wide RBW is required to shorten acquisition times. For narrow sweeps, a narrow filter improves frequency resolution.

Carefully consider which FFT window type to use. As an example, a Flat Top window minimizes amplitude measurement error and is recommended for amplitude measurements even though it has non-optimal selectivity.

Noise Fundamentals

There are multiple types of noise to consider when analyzing RF signals.

Noise Figure

All devices have inherent noise. When a noise is quantified, it is usually referred to the device input. In other words, all the noise power a unit under test (UUT) inherits is assumed to come from its input. The noise figure of a UUT is the ratio, in dB, of its noise power to the noise power that a matched resistive load would deliver at room temperature.

If you terminate a UUT input with a matched resistive load (typically 50 Ω) and measure the noise power density, in watts/Hz, at its output (N_0), the noise figure (NF) is given by the following equation:

$$NF = 10\log\left(\frac{N_0 / G}{kT}\right) \quad NF = 10\log\left(\frac{N_0 / G}{kT}\right)$$

where

- G is the power gain (in linear units) of the UUT
- k is Boltzmann's constant ($\approx 1.38 \times 10^{-23}$ J/K)
- T is the ambient temperature ($\approx 290^\circ$ K)

If you use the vector signal analyzer to measure the output noise of a UUT, the resulting measurement contains both UUT noise and noise intrinsic to the vector signal analyzer. If the UUT gain (G_{UUT} in dB) is known, compute the noise figure of the UUT with the following equation:

$$NF = 10\log\left[\frac{N_m - N_{rfsa}}{kT} + 1\right] - G_{UUT} \quad NF = 10\log\left[\frac{N_m - N_{rfsa}}{kT} + 1\right] - G_{UUT}$$

where

- N_{rfsa} is the noise, in watts/Hz, measured by the vector signal analyzer when its input is terminated with a matched resistive load
- N_m is the measured noise, in watts/Hz, with the UUT attached
- G is the linear power gain
- NF is expressed in dB

Noise Floor

Noise floor is the measure of the noise density, in dBm/Hz, or the noise power in a 1 Hz bandwidth.

Noise can be classified into several types, including the following:

- Shot noise
- Thermal noise
- Flicker noise
- Burst noise
- Quantization noise
- Avalanche noise

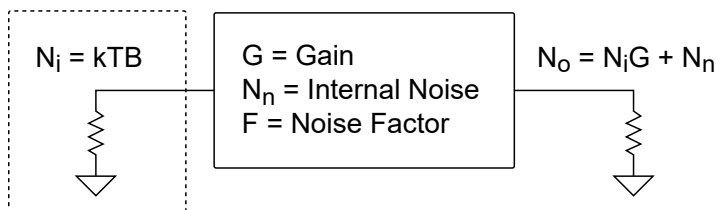
The noise due to a matched resistive load (N_i) can be expressed as the following relation:

$$(N_i) = kTB \text{ watts}$$

where

- k is Boltzmann's Constant ($k = 1.38 \times 10^{-23} \text{ J/K}$)
- T is the resistor temperature in Kelvin
- B is the bandwidth in Hz

If B is set to 1 Hz, then N_i is equal to the output noise density in watts/Hz. For the system shown in the following figure, the output noise floor (N_o) is the combination of the input noise multiplied by the gain or loss of the system plus the internal noise of the system (N_n).



The internal noise of a system is therefore represented by the following equation:

$$N_n = N_i G - N_0$$

Harmonic Distortion

Harmonic distortion is a measure of the amount of power contained in the harmonics of a fundamental signal compared to the power of the fundamental signal.

Harmonic distortion is inherent to devices and systems that possess nonlinear characteristics. The more nonlinear the device, the greater its harmonic distortion.

Harmonic distortion can be expressed as a power ratio or as a percentage. Use the following formula to express it as a power ratio:

$$P_{HD} = P_{fund} - P_{harm}(\text{dBc})$$

where

- P_{HD} is the power of the harmonic distortion in dBc
- P_{fund} is the fundamental signal power in dB or dBm
- P_{harm} is the power of the harmonic of interest in dB or dBm

Convert power to voltage and use the following equation to express harmonic distortion as a percentage ratio:

$$\text{Percentage of Distortion} = \frac{V_{harm}}{V_{fund}} \times 100 \% \quad \text{Percentage of Distortion} = \frac{V_{harm}}{V_{fund}} \times 100 \%$$

Total Harmonic Distortion

Total harmonic distortion (THD) is a representation of the effect that multiple harmonic spurs throughout a spectrum band have on the fundamental tone.

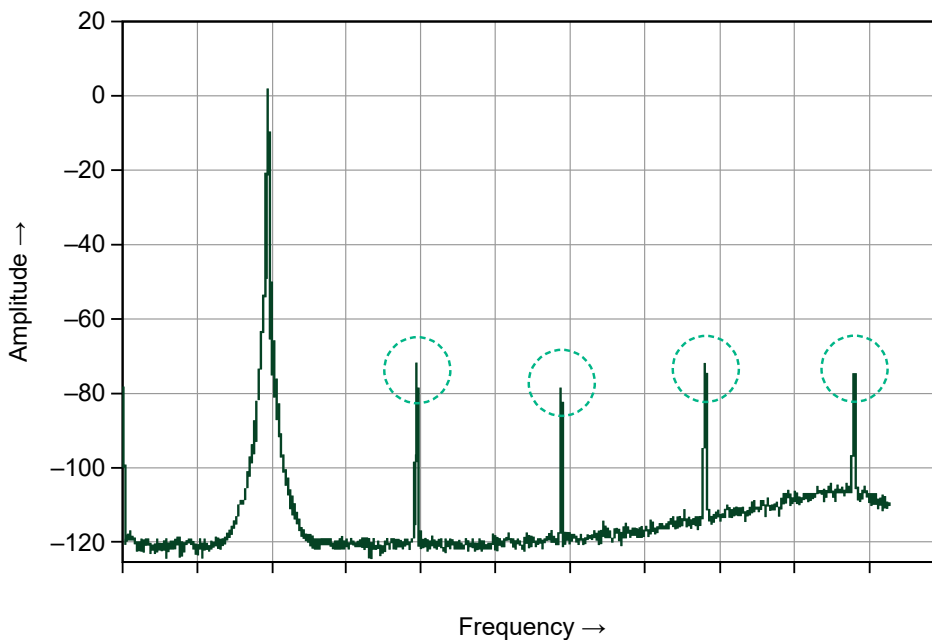
You can calculate THD by summing the power in each of the harmonics and dividing by the total power of the fundamental. As a general principle, a signal becomes visibly distorted when the THD approaches -30 dB.

The equation for THD is:

$$THD = \frac{\sqrt{V_{h2}^2 + V_{h3}^2 + V_{h4}^2 + \dots + V_{hN}^2}}{V_{fund}} \times 100 \% \quad THD = \frac{\sqrt{V_{h2}^2 + V_{h3}^2 + V_{h4}^2 + \dots + V_{hN}^2}}{V_{fund}} \times 100 \%$$

As this equation suggests, the THD specification evaluates the power in harmonic spurs from the second through the n^{th} harmonic. In practice, typical signal generators feature THD specifications for harmonics two through six. The following figure shows an example plot of a generated signal and illustrates the concept of THD specification in signal generators.

Figure 29. Total Harmonic Distortion Example



The previous figure shows an example plot of a 20 kHz sinusoid generated with an arbitrary waveform generator. Notice the power levels of the harmonic spurs, indicated with circles. This generator features -77 dBc or better of THD for the second through sixth harmonics.

THD generally deteriorates as the generated signal increases in frequency. When working with a signal generator, consider the THD throughout the bandwidth of the generator.

Two-Tone Third-Order Intermodulation Distortion

Two-tone third-order intermodulation distortion (IMD₃) is the measure of the third-order distortion products produced by a nonlinear device when two tones that are close in frequency are received at its input.

This distortion product is usually so close to the carrier that it is almost impossible to filter out and can cause interference in multichannel communications equipment.

If F_1 and F_2 are the frequencies of the two tones, the third-order distortion products occur on both sides of these tones at $2F_2 - F_1$ and $2F_1 - F_2$. Assuming that the power levels of the two tones are equal, IMD₃ is the difference between the power of the fundamental signals and the third-order products, as defined in the following equation:

$$\text{IMD}_3 = P_o - P_{o3}$$

where P_{o3} is the power level of one of the output third-order products, and P_o is the power level of one of the fundamental tones. The math becomes more involved if the powers of the two tones are different.

After you calculate the IMD₃ using the preceding formula, you can calculate the unit under test (UUT) output third-order intercept point (OIP₃) using the following equation:

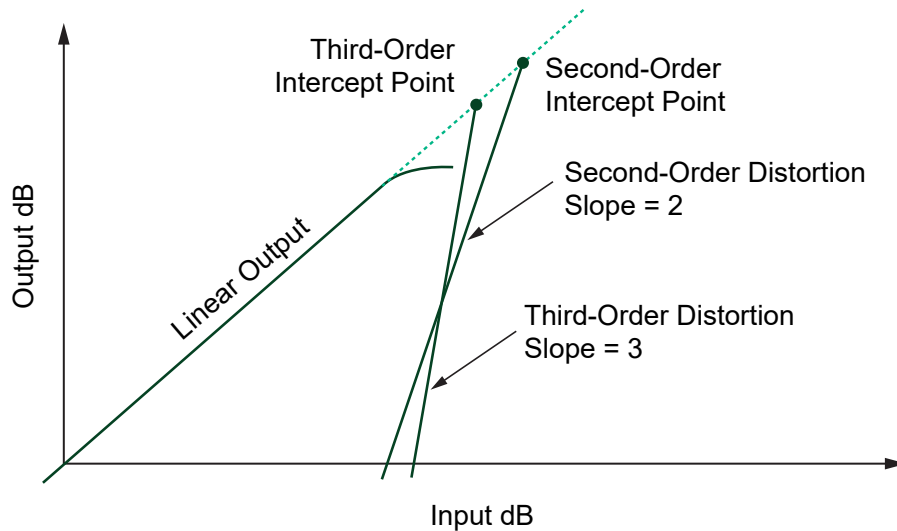
$$\text{OIP}_3 = \frac{\text{IMD}_3}{2} + P_o = \frac{1}{2} \left(3P_o - P_{o3} \right) \text{ (dB)} \quad \text{OIP}_3 = \frac{\text{IMD}_3}{2} + P_o = \frac{1}{2} \left(3P_o - P_{o3} \right) \text{ (dB)}$$

The input third-order intercept point (IIP₃) is defined as:

$$\text{IIP}_3 = (\text{OIP}_3 - G)$$

Where G is the gain of the device. The IIP₃ number quantifies the third-order linearity input referred of a device.

The following figure shows the relationship between the second- and third-order distortions and the linear output of a device.



The two tones injected into the UUT must be free from any third-order products. These two tones are combined, or summed, at or before the UUT input. If the two tones are not well isolated, they intermodulate with each other and cause distortion. A signal combiner with good input-to-input isolation is recommended to minimize distortion of the input tones.

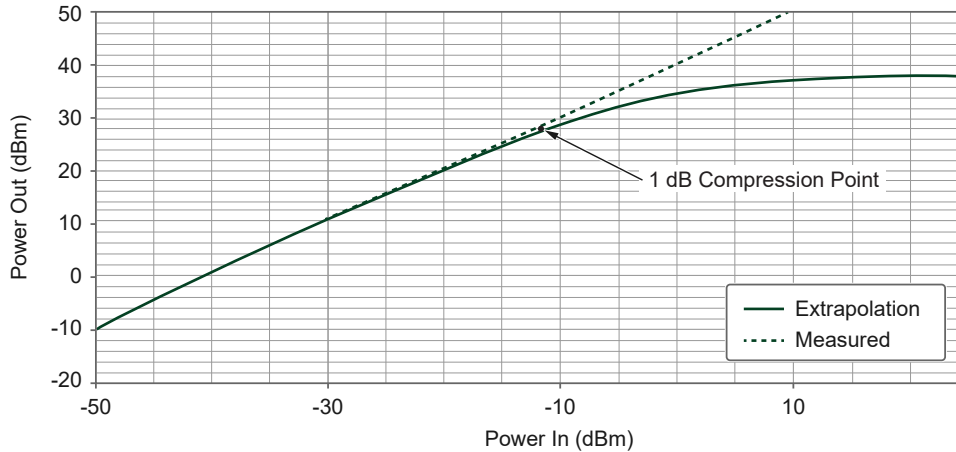
1 dB Gain Compression Point

The 1 dB gain compression point describes amplifier saturation.

An amplifier maintains a constant gain for low-level input signals. However, at higher input levels, the amplifier saturates and its gain decreases. The 1 dB gain compression point ($P_{1\text{ dB}}$) indicates the power level that causes the gain to drop by 1 dB from its small signal value.

The 1 dB gain compression point is derived from the gain relationship between output power and input power. Output power is plotted against input power in the following figure.

Figure 30. 1 dB Gain Compression Point



The straight line on this graph is an extrapolation of the small signal gain of the unit under test (UUT). The input 1 dB compression point is the input power that causes the UUT gain to drop by 1 dB from this small signal value. In this figure, the gain drop occurs at approximately -12 dBm.

Streaming

Streaming is a way to generate waveforms that are too large to fit in the onboard memory of an RF signal generator.

Streaming can be used in Arbitrary Waveform or Script generation modes. You can stream waveforms only when the power level is defined as a peak power, not an average power.

To stream waveform data, allocate and identify all or a portion of the RF signal generator onboard memory to act as an onboard waveform for streaming. Before initiating waveform generation, fill that onboard memory with the first part of your waveform. As the waveform is generated, continuously write the remaining waveform data until generation is complete. The RF signal generator uses the instrument to stream the waveform.

Scripts

You can link and loop multiple waveforms together in a generation operation using a script. A **script** is a series of instructions that indicates how waveforms saved in the

onboard memory should be sent to the DUT.

The script can specify the order in which the waveforms are generated, the number of times they are generated, and the triggers and markers associated with the generation.



Note If you are using a vector signal transceiver (VST) instrument, some script instructions may not be supported using NI-RFSG.

Basic Scripting Example

You can create a script to manage waveform generation based on multiple waveforms and triggers. For example, you could download waveforms A, B, C, and D into instrument memory and script it to do the following:

1. Wait for a trigger to initiate generation
2. Upon receiving this trigger, generate waveform A three times with a marker at position 16 each time
3. Generate waveforms B, C, and D twice (BCDBCD)

The following is the script of this example:

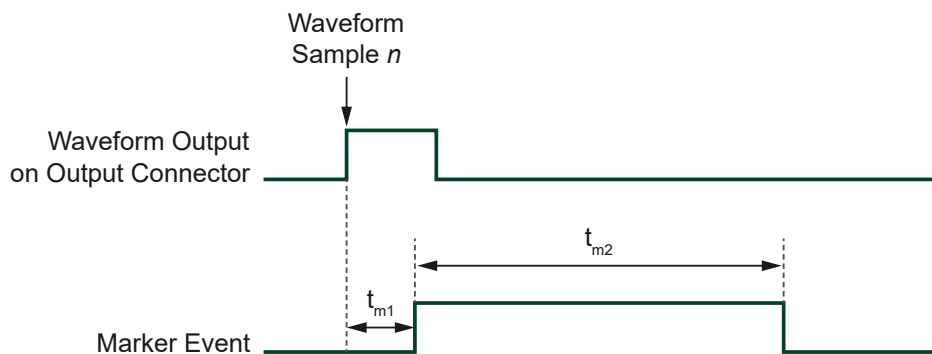
```
script myFirstScript
  wait until scriptTrigger0
  repeat 3
    generate waveformA marker0(16)
  end repeat
  repeat 2
    generate waveformB
    generate waveformC
    generate waveformD
  end repeat
end script
```

NI-RFSG Marker Event Details

A **marker** is an event that an NI RF signal generator produces in relation to a waveform that is generated. The marker event is delayed slightly relative to the leading edge of the waveform that it marks.

By default, the marker event is a pulse of fixed width. The following figure shows a waveform generated on the output of a waveform generator. The waveform contains a pulse that represents a waveform sample n that is one Sample Clock in width. The second pulse, the marker event, represents the pulse that generates when the corresponding waveform sample n is output at the output connector.

Figure 31. Waveform Output and Associated Marker Event



- t_{m1} —Duration of delay between the configured waveform sample n and generation of the associated marker event. For most NI VSTs, configurable with NI-RFSG Events Delay and Relative Delay.
- t_{m2} —Marker event pulse duration. For most NI VSTs, configurable with NI-RFSG Width Units and Width Value.

You can use NI-RFSG to configure the characteristics of the pulse generated as the Marker event.

In addition, you can use scripts to create markers to develop complex behaviors around generated waveforms.

For example, you can configure the Marker event as a toggled digital pulse signal that changes state between a low level and high level. While the basic pulse option for Marker events generates event pulses of fixed width, using the toggle option allows you to use scripts to change the duration of the Marker event pulse. You can control the initial state of the toggled pulse with `MarkerEvent.Toggle.InitialState`.

How Signal Routing Works

NI VSTs are capable of sending and receiving signals through the front panel connectors and the PXI trigger bus. Signals are routed between a source and

destination via a path through the PXI trigger bus; there are two broad categories of routes.

The front panel connectors provide connectivity for input and output signals as well as for the control lines that send and receive clocks, triggers, and events. You can use the PXI trigger bus to send and receive events, triggers, and Sample and Reference Clocks.

Signals can be routed to and from the supported front panel connectors of your instrument. All signal routing operations can be characterized by a source and a destination. The possible signal routes for your instrument depend on the instrument, the PXI chassis, and the occupied PXI chassis slot.

When a signal is exported to a PXI trigger bus line from an instrument, only instruments in the same PXI chassis segment recognize the signal from that PXI trigger bus line. To route the signal to instruments in a different PXI chassis segment, you must establish a **dynamic route** or **static route**.

- **Dynamic route**—Signal route that is specified by only the route endpoints. The actual route is determined automatically in software and is reserved and unreserved based on the driver session using that route. Closing the driver session using a dynamic route releases that route.

NI recommends using dynamic reservations and routing whenever possible.

- **Static route**—Signal route that you create manually. You can implement static reservation of trigger lines in MAX through the **Triggers** tab.

Refer to your chassis documentation to determine the PXI trigger bus segments of your chassis.

Terminal Names and Terminal Name Syntax

A **terminal name** is a unique identifier that refers to a physical terminal in your system; a terminal name is **fully qualified** if it is an unambiguous reference to a specific terminal.

To guarantee the uniqueness of a terminal name across multiple instruments, terminal names begin with a forward slash, followed by the name of the instrument as

configured in NI Measurement & Automation Explorer (MAX), such as `PXI1Slot2`. A forward slash and the name of the terminal, such as `PFI1`, follow the instrument identifier.

For example, the fully qualified terminal name for `PFI1` on `PXI1Slot2` is `/PXI1Slot2/PFI1`.

Creating Dynamic Routes

With PXI chassis that support them, dynamic routes simplify signal routing through a system by automatically choosing, allocating, and deallocating the route.

To create a dynamic route, specify a fully qualified terminal name for the source or destination signal in your programming environment.

Creating Static Routes in MAX

Static routes are a way to manually designate a route on the PXI trigger bus between a specific signal source and a specific signal destination.



Note NI recommends using dynamic reservations and routing whenever possible. If your application contains too many static routes between PXI trigger buses of your chassis, you may not be able to establish dynamic routes across chassis segments. Refer to your chassis documentation to determine the allowed number of static routes between the PXI trigger buses of your chassis.

Complete the following steps to create static routes in MAX.

1. Launch MAX by navigating to **Start » All Programs » National Instruments » NI MAX** or by clicking the NI MAX desktop icon.
2. Expand **Devices and Interfaces**.



Note If you are using a remote real-time target, expand **Remote Systems**, find and expand your remote real-time target, and then expand **Devices and Interfaces**.

3. Select your chassis.
The settings of your chassis are displayed on the right of the MAX window.
4. Click the **Trigger** tab below the settings view.
A table in the **Triggers** view shows the PXI trigger bus segments of your chassis.
5. Configure the static route that you wish to make.
6. Save the changes.

Triggers and Events

You can use triggers and events to coordinate the operation of multiple channels and instruments.

Triggers

A **trigger** is an input signal received by an instrument or instrument channel that causes the instrument or channel to perform an action. Triggers are routed to **input terminals** to coordinate actions.

An **input terminal** is a physical trigger line, such as a PXI trigger line, or an output terminal on another instrument or channel, where an instrument or channel awaits a digital edge trigger signal.

For purposes of programming instruments with NI APIs, triggers comprise two parts:

- The action, represented with the name of the trigger, that you want the instrument or channel to take.
- The signal condition you want to serve as the stimulus for that action (for example, a rising or falling digital edge on a signal, or a software-generated edge you configure).

Triggers can be internal (software-generated) or external. You can export external triggers and use them with events to synchronize hardware operation with external circuitry or other instruments.

NI RF instruments accept external triggers routed between the instruments using PXI trigger lines. Events assigned to a PXI trigger line can coordinate actions across channels and across instruments.

Events

An **event** is a signal generated by an instrument or instrument channel that signifies a state change of the instrument, the arrival of a certain kind of sample, the production of a certain amount of samples, or the passage of time. Events can be routed to **output terminals** to coordinate the action of multiple channels or multiple instruments.

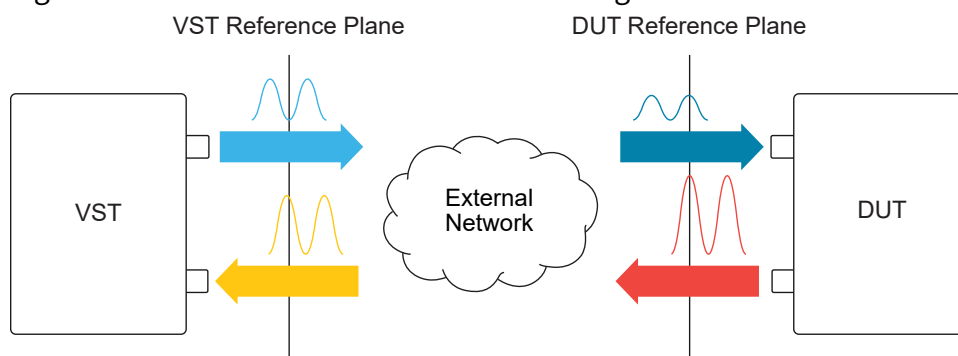
Triggers and clocks are generally considered input signals. Exportable triggers and clocks, such as the Sample Clock, can also be considered output signals. Output signals that do not have a trigger or clock counterpart are called events.

You typically configure events for a specific hardware condition and then export those events for use in the test program or export them to a PXI trigger line to cause an action in another instrument configured to wait for a trigger on the same PXI trigger line.

De-embedding Overview

NI RF instrument performance specifications are for signals at the instrument's ports. However, you must also account for external cabling and components between the instrument and DUT, which also affects input and output signals.

Figure 32. Effect of the External Network on Signals



De-embedding is the process of removing the effects of these components on an input or output signal, so the accuracy of the instrument is extended to the DUT.

The NI-RFSA and NI-RFSG instrument drivers can de-embed measurements and

generations using your characterization data. To use this feature, first characterize the external components with an S-parameter table. Once the tables are passed to the driver, they can be applied to move the reference plane from the instrument port to the DUT port.

The NI driver compensates for the effects of the networks by modifying the analog path and adjusting the data with digital signal processing (DSP).

When using NI-RFSG, the signal seen at the DUT port matches the requested signal. When using NI-RFSA, the data returned from the driver reflects the signal at the DUT port rather than the signal seen at the instrument's input port.

S-parameters

S-parameters characterize the effects of a linear network on a signal when it passes from one port of the network to another port. For purposes of de-embedding for VST operation, the instrument driver compensates for the effects of the networks by modifying the analog path and adjusting the data with DSP.

S-parameters of the de-embedding network can be obtained by either extracting them using a vector network analyzer (VNA) or by simulating the network in an RF computer assisted design (CAD) tool.

NI-RFSA and NI-RFSG support de-embedding of two port networks. Two port networks are characterized by four s-parameters. S-parameters are written in the format ***Snm***, where ***n***, designates the receiving port, and ***m***, designates the driving port. Each S-parameter is referred to as a gain or reflection term.

The following table illustrates which terms characterize gain or reflection of the network.

Table 20. S-parameters

S-parameter	Description
S_{11}	reflection (input reflection)
S_{12}	gain (forward transmission)
S_{21}	gain (reverse isolation)

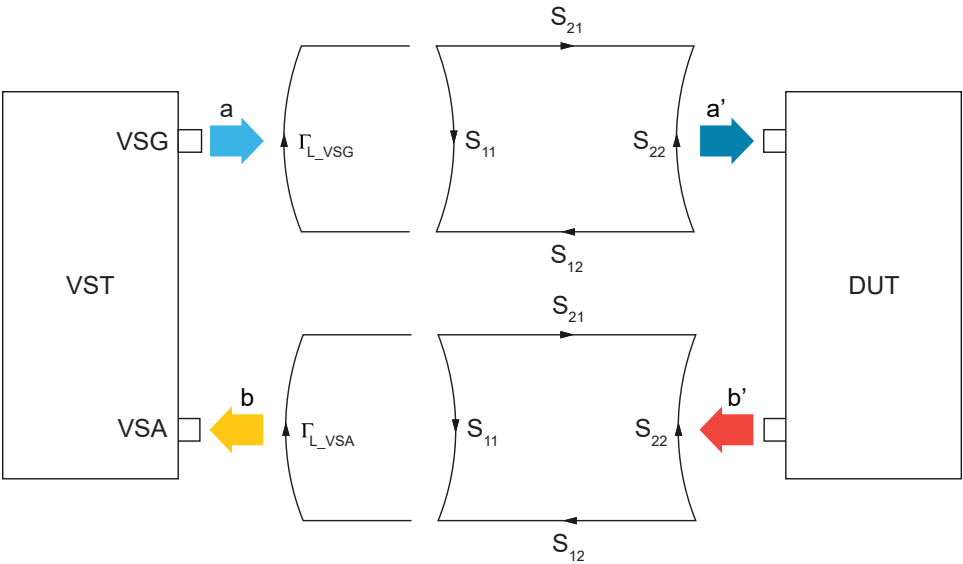
S-parameter	Description
S_{22}	reflection (output reflection)

Gain terms, S_{12} and S_{21} , model the gain an input signal experiences as it passes through the network to the receiving port. The reflection terms, S_{11} and S_{22} , are used when computing the effects of signal reflection due to impedance mismatch between the network's ports and driving ports.

The following figure provides a visualization of the external network as an S-parameter model. Signal b' , output from the DUT, passes through the external network and arrives at the VST as signal b . Signal a , generated by the VST, passes through the network and arrives at the DUT as signal a' . The following are reflection terms for the instrument ports:

- Γ_{L_VSA}
- Γ_{L_VSG}

Figure 33. S-parameter modeling of an external network for NI-RFSA and NI-RFSG



Scalar De-embedding

Scalar de-embedding compensates only for the gain term of the S-parameters.

In the previous figure, the DUT output is connected to port2 of the external network

and port1 is connected to the NI device port. By convention, the S12 parameter specifies the gain due to the network, and applied to the signal ***b'*** output from the DUT toward the VSA port. NI-RFSA computes the signal at the DUT port using the following formula:

$$b' = \frac{b}{S_{12}}$$

To acquire the output signal of the DUT and not the signal seen at the NI instrument's port, NI-RFSA compensates for the network by amplifying the signal. The driver makes DSP adjustments to compensate for any remaining effects that analog path adjustments do not compensate for. NI-RFSA returns data reflecting the amplitude of the signal at the DUT.

Similarly, when using NI-RFSG scalar de-embedding, the driver computes the output signal at the DUT port using the following formula:

$$a' = S_{21} \times a$$

The receiving port in this case is port2 at the DUT and the input port is port1 at the NI instrument's port. NI-RFSG compensates for the network via analog path adjustments and DSP so that the signal you request arrives at the DUT with the correct amplitude.