# PXIe-5830 Features

2024-04-01

# Contents

# PXIe-5830

5 GHz to 12 GHz, 1 GHz Bandwidth, Vector Signal Transceiver

- Two 5 GHz to 12 GHz bi-directional IF test ports
- 1 GHz signal bandwidth

## Instrument Components

The PXIe-5830 is a modular vector signal transceiver instrument consisting of PXI Express hardware.

The PXIe-5830 is comprised of the following modules:

- PXIe-5820 Vector Signal Transceiver
- PXIe-3621 Vector Signal Up/Down Converter

There is no single instrument labeled "PXIe-5830." To control multiple hardware modules as a single RF instrument, you must first associate the modules in MAX.

### Related information:

- Associating NI-RFSG Modules
- Associating NI-RFSA Modules

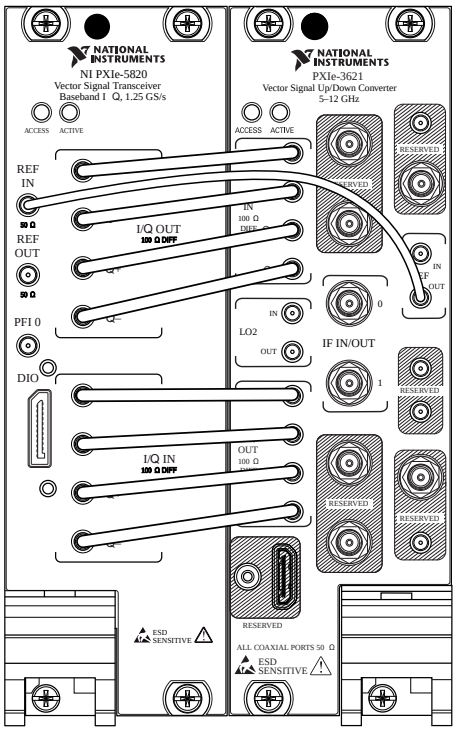


## PXIe-5830 Front Panel

The PXIe-5830 Vector Signal Transceiver instrument consists of the following hardware modules, which contain multiple connectors and LED indicators.

- PXIe-5820 Front Panel and LEDs
- PXIe-3621 Front Panel and LEDs

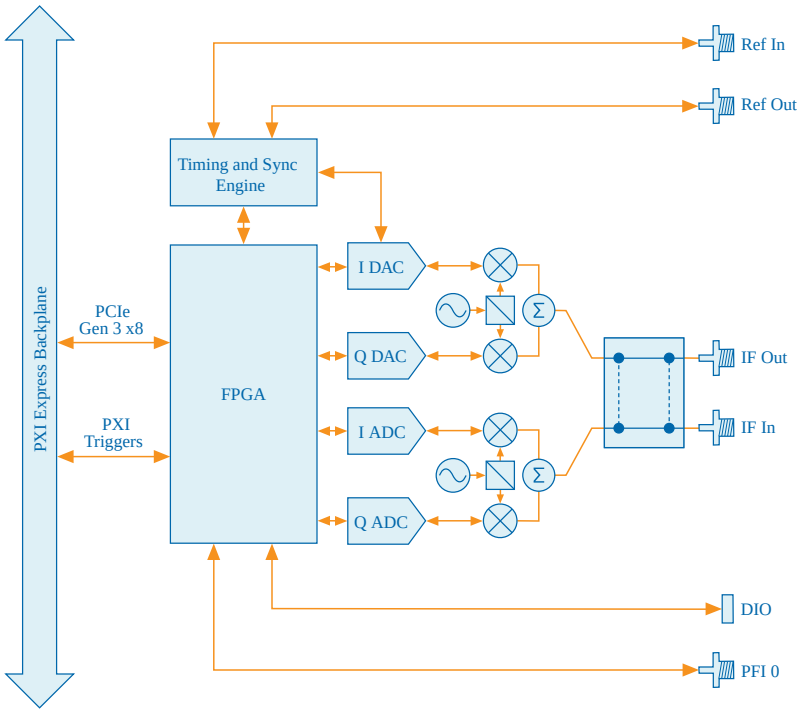
**Figure 1.** Interconnected PXIe-5830




# PXIe-5830 Block Diagram

The PXIe-5830 is an IF vector signal transceiver instrument comprised of the PXIe-5820 and PXIe-3621.

The following block diagram represents the PXIe-5830 instrument configuration.




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

The PXIe-3621 contains an I/Q modulator and I/Q demodulator with wide instantaneous bandwidth to upconvert and downconvert signals from the PXIe-5820 baseband transceiver through differential I and Q ports. The switched IF ports, IF0 and IF1, cover a tuning range of 5 GHz to 12 GHz and have variable reference levels that support a wide range of power levels.

The PXIe-3621 is a full-duplex transceiver with IF0 and IF1 ports. The signal direction on each port, IF0 and IF1, can be reversed to facilitate testing of DUTs with two bi-directional ports. The module also contains two internal synthesizers, LO2 0 and LO2 1, that can be tuned independently thus allowing transmitting and receiving at different tune frequencies. Additionally, a single synthesizer, LO2 0 or LO2 1, can be shared between the I/Q modulator and I/Q demodulator to achieve phase noise cancellation when the transmit and receive tune frequencies are the same. Finally, the PXIe-3621 contains internal loopback paths to facilitate calibration and improve measurement performance.

### Related concepts:

- LO Sharing Using NI-RFSA and NI-RFSG
- Local Oscillator

## Local Oscillator

The PXIe-5830 contains a local oscillator (LO) for analyzing and generating RF signals.

- **LO2**—Mixed within the PXIe-3621 with the I/Q signals from the PXIe-5820 to analyze or generate an signal between 5 GHz and 12 GHz. You can use this signal to analyze or generate RF signals from the IF bi-directional ports on the PXIe-3621 front panel.

### Related concepts:

- LO Sharing Using NI-RFSA and NI-RFSG

## PXIe-5820

500 MHz, 1 GHz I/Q Bandwidth, Baseband PXI Vector Signal Transceiver

- 1 GHz complex I/Q equalized bandwidth
- DC to 500 MHz frequency range
- Serves as the baseband module within the PXIe-5830 Vector Signal Transceiver instrument

Refer to the PXIe-5820 stand-alone device section for information about the functionality and architecture of the module, including the FPGA basecard subsystem, I/Q input, and I/Q output.

### Related information:

- PXIe-5820 (Stand-Alone) Overview
- FPGA Basecard Subsystem
- I/Q Input
- I/Q Output

## PXIe-5820

**Figure 2.** PXIe-5820 Front Panel and LEDs

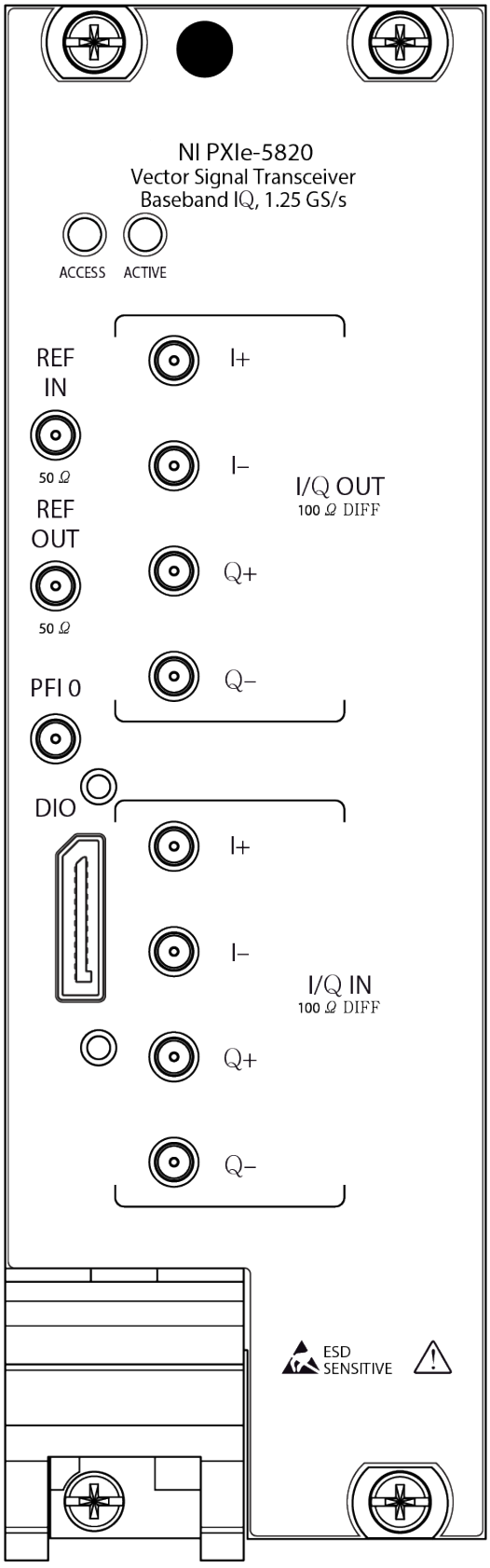


**Table 1.** General Connector Descriptions

| Connector | Description | Connector Type |
|---|---|---|
| REF IN | Input connector that allows for the use of an external 10 MHz Reference Clock. | MMPX (f) |
| REF OUT | Output connector 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 I/O |

**Table 2.** I/Q Connector Descriptions

| Connector | | Description | Connector Type |
|---|---|---|---|
| I/Q OUT | I+ | Output connector for I+ signals. | MMPX (f) |
| | I- | Output connector for I- signals. | MMPX (f) |
| | Q+ | Output connector for Q+ signals. | MMPX (f) |
| | Q- | Output connector for Q- signals. | MMPX (f) |
| I/Q IN | I+ | Input connector for I+ signals. | MMPX (f) |
| | I- | Input connector for I- signals. | MMPX (f) |
| | Q+ | Input connector for Q+ signals. | MMPX (f) |

| Connector | | Description | Connector Type |
|---|---|---|---|
| | Q- | Input connector for Q- signals. | MMPX (f) |

**Table 3.** LED Indicators

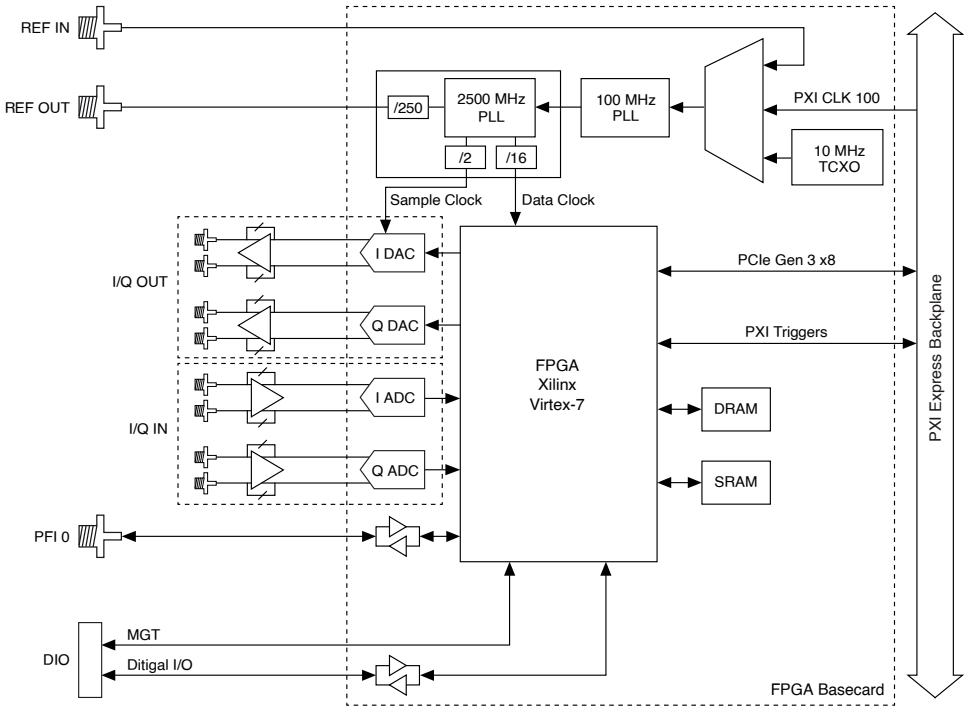
| LED | Indications |
|---|---|
| ACCESS | Indicates the basic hardware status of the device.<br>Off—The device is not yet functional or has detected a problem with a PXI Express power rail.<br>Amber—The device is being accessed. **Accessed** means that you are writing to the device setup registers to control the device, reading from the device to monitor the device status, or transferring data to/from the device.<br>Green—The device is controllable through the software. |
| ACTIVE | Off—The device is idle.<br>Solid green—The device is generating a waveform.<br>Dim amber—The device is waiting for an acquisition Reference Trigger.<br>Solid amber—The device is acquiring a waveform.<br>Solid red—The device has detected an error. The LED remains red until the error condition is removed.<br>**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. |

**Figure 3.** PXIe-5820 DIO 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 |

## PXIe-5820 Block Diagram

The PXIe-5820 has the I/Q Input, I/Q Output, and FPGA Basecard subsystems.



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

## PXIe-3621

5 GHz to 12 GHz, Vector Signal Up/Down Converter

- 5 GHz to 12 GHz frequency range
- Used only within the PXIe-5830 Vector Signal Transceiver instrument as an up and down converter between baseband I/Q and IF signals

## PXIe-3621 Front Panel and LEDs

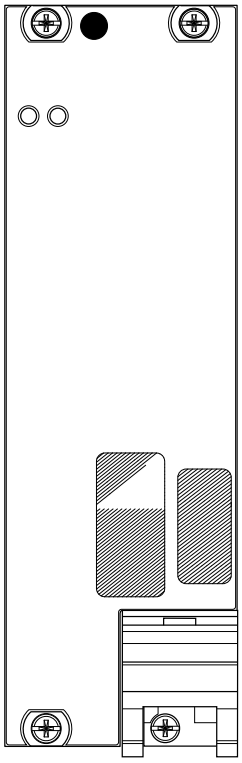


**Table 4.** I/Q Connector Descriptions

| Connector | | Description | Connector Type |
|---|---|---|---|
| I/Q IN | I+ | Input connector for I+ signals. | MMPX (f) |
| | I- | Input connector for I- signals. | MMPX (f) |
| | Q+ | Input connector for Q+ signals. | MMPX (f) |
| | Q- | Input connector for Q- signals. | MMPX (f) |
| I/Q OUT | I+ | Output connector for I+ signals. | MMPX (f) |
| | I- | Output connector for I- signals. | MMPX (f) |
| | Q+ | Output connector for Q+ signals. | MMPX (f) |
| | Q- | Output connector for Q- signals. | MMPX (f) |

**Table 5.** General Connector Descriptions

| Connector | | Description | Connector Type |
|---|---|---|---|
| LO2 | IN | Input connector for the external local oscillator signal used for the up and down conversion between baseband and IF. | MMPX (f) |
| | OUT | Output connector for the local oscillator signal used for the up and down conversion between baseband and IF. | MMPX (f) |
| IF IN/OUT | 0 | IF0 bi-directional connector for both input and output. | SMA 27 GHz (f) |

| Connector | | Description | Connector Type |
|---|---|---|---|
| | 1 | IF1 bi-directional connector for both input and output. | SMA 27 GHz (f) |
| REF | IN | Input connector for the Reference Clock. | MMPX (f) |
| | OUT | Output connector for the Reference Clock. | MMPX (f) |
| RESERVED | | Reserved connectors are not enabled for use on the PXIe-5830 and should remain capped with 50 Ω terminators to ensure specified instrument performance. | — |

**Table 6.** LED Indicators

| LED | Indications |
|---|---|
| ACCESS | Indicates the basic hardware status of the module.<br>• **OFF**—The driver software has not yet loaded, or the module exceeded approved operating temperature and thermal shutdown occurred or has detected a problem with a power rail.<br>• **AMBER**—The module is being accessed. **Accessed** means that the device setup registers are being written to in order to control the device.<br>• **GREEN**—The module is ready to be programmed by software. |
| ACTIVE | • **OFF**—The module is idle.<br>• **RED**—The module detected an error state. |

## Related reference:

- Instrument Configurations

# Instrument Terminology

Refer to the following list for definitions of common PXIe-5830 instrument terms used throughout this document.

**Table 7.** Instrument Terminology Definitions

| Term | Definition |
|---|---|
| **IF IN/OUT Ports** | Refers to the IF IN/OUT 0 and IF IN/OUT 1 connectors on the PXIe-3621 front panel for IF signals. These are the primary RF input/output ports for RF signals 5-12 GHz.<br>These ports are named as IF ports because the hardware topography is the same as that found on the PXIe-3622. |
| **LO2** | Refers to the local oscillator internal to the PXIe-3621 that executes the up or down conversion from baseband. |
| **Onboard** | Refers to the value of the LO Source property and changes purpose depending on your instrument configuration.<br>The PXIe-5830 refers to the LO2 of the PXIe-3621 module as the onboard LO. |
| **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.<br>The PXIe-5830 contains 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 enable low IF mode when the signal bandwidth is small enough to allow it. |

| Term | Definition |
|---|---|
| | Automatic is the default value. NI recommends keeping offset mode set to the default value. |
| **Offset Mode is Enabled** | Refers to the NI-RFSA Downconverter Frequency Offset Mode property or NI-RFSG Upconverter Frequency Offset Mode property set to Enabled.<br>The PXIe-5830 contains 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. |
| **Offset Mode is User-Defined** | Refers to the NI-RFSA Downconverter Frequency Offset Mode property or NI-RFSG Upconverter Frequency Offset Mode property set to User-Defined.<br>Offset Mode set to User-Defined allows the instrument to operate with maximum instantaneous bandwidth. By default, the offset is minimized to maximize the available instantaneous bandwidth. |

# Instrument Configurations

The following table lists the hardware connectors that correspond to the valid port strings used by the driver software to specify the ports available for you to select and configure based on the hardware configuration you purchased.

**Table 8.** Valid Port Strings and Corresponding Hardware Connectors

| Port String | Hardware Module | Hardware Connector |
|---|---|---|
| if0 | PXIe-3621 | IF IN/OUT 0 |
| if1 | | IF IN/OUT 1 |

For example, if you specify if0, you are specifying the IF IN/OUT 0 connector on the PXIe-3621 front panel to transmit or receive an IF signal.

**Note** While both IF IN/OUT 0 and IF IN/OUT 1 ports are configurable as the generator output or analyzer input, NI recommends you use

> IF IN/OUT 0 as the generator output using the OUT port and IF IN/OUT 1 as the analyzer input as the IN port.

**Related tasks:**

- Selecting Available Ports

**Related reference:**

- PXIe-3621 Front Panel and LEDs

## Timing Configurations

The timebases of the PXIe-5830 modules must be frequency-locked to a common reference clock. The following clock sources are available:

- **10 MHz Onboard Reference Clock of the PXIe-3621 Vector Signal Up/Down Converter**—The PXIe-3621 onboard reference clock source supplies the reference clock.
- **10 MHz External Reference Clock**—Connect the external clock signal, from your stable frequency reference, to the REF IN connector on the PXIe-3621.
- **10 MHz PXI Express Backplane Clock**—This 10 MHz Reference Clock signal is supplied on the PXI backplane.

# Configuring Onboard Reference Clock Timing

The configuration of the PXIe-5830 exports a 10 MHz signal from the PXIe-3621 to PXIe-5820.

Complete the following steps to apply this configuration.

1. Use the 200 mm MMPX (m)-to-MMPX (m) cable to connect the PXIe-5820REF IN front panel connector to the PXIe-3621REF OUT front panel connector. This completes the Reference Clock connection.
2. Using NI-RFSA or NI-RFSG, set the Ref Clock Source property or attribute to OnboardClock.

## Configuring External Reference Clock Timing

Complete the following steps to configure the PXIe-5830 to lock to an external reference source.

1. Connect the external signal to the PXIe-3621REF IN connector.
   The clocks need to remain connected in their respective default configuration, as described in the preceding section.
2. Set the NI-RFSA or NI-RFSG Ref Clock Source property or attribute to RefIn as the Reference Clock source.

## Configuring PXI 10 MHz Backplane Clock Timing

To configure the PXIe-5830 to use the PXI 10 MHz backplane clock, set the NI-RFSA or NI-RFSGRef Clock Source property or attribute to PXI_Clk.

## Selecting Available Ports

You can use the NI-RFSA and NI-RFSG APIs to query available ports on your VST and specify which ports to use. This functionality is important for VSTs with millimeter wave functionality because these VSTs use different ports depending on the frequency of the signal you are generating or analyzing.

Complete the following steps to check which ports are available and specify the ports to use:

1. Using NI-RFSA or NI-RFSG, read the Available Ports property to see the ports available for use based on the instrument configuration you have installed and connected.
   The ports are returned as a comma-separated string and refer to corresponding hardware connectors.
2. Set the Selected Ports property to specify the port you want to use to generate or acquire a signal.

### Related reference:

- Instrument Configurations

## Peer-to-Peer Data Streaming

The PXIe-5830 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 Understanding the Peer-to-Peer Data Streaming Architecture topic in the **NI Peer To Peer Streaming Help**.

- **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 "FIF0endpointN" is used to tell NI-RFSG which RF vector signal transceiver enpoint is being specified, where **N** is an integer starting with 0. If the RF vector signal transceiver supports multiple endpoints, the first is "FIF0endpoint0", the second is "FIF0Endpoint1," 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. For a continuous 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 "FIF0endpointN" is used to tell NI-RFSA which RF vector signal transceiver enpoint is being specified, where **N** is an integer starting with 0. If the RF vector signal transceiver supports multiple endpoints, the first is "FIF0endpoint0", the second is "FIF0Endpoint1," 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. For a continuous generation, set the Number of Records is Finite property to False
5. To acquire a finite number of samples, 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 the Get Stream Endpoint Handle VI or the GetStreamEndpointHandle function for your driver to get a reader endpoint handle.

### Configuring Flow Control

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

1. Use the Get Stream Endpoint Handle VI or the GetStreamEndpointHandle function for your driver to get the RF vector signal transceiver reader handle and the appropriate handle from the writer peer API.
2. Configure NI-P2P with information regarding each endpoint to link the endpoints into a stream, using the niP2P Create Peer to Peer Stream VI or the nip2pCreateStream function. 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:

- niRFSG StartTrig.P2PEndpointFullness.Level property or NIRFSG_ATTR_P2P_ENDPOINT_FULLNESS_START_TRIGGER_LEVEL attribute: 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 or attribute is used when the StartTrig.Type property is set to P2P Endpoint Fullness or the NIRFSG_ATTR_START_TRIGGER_TYPE attribute is set to NI_RFSG_VAL_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 by completing the following steps:

1. Configure both peers for peer-to-peer streaming.
2. Link both peers using the niP2P Create Peer to Peer Stream VI or the nip2pCreateAndLinkStream function.
3. Enable the stream either by setting enable stream to True using the niP2P Create Peer to Peer Stream VI or the nip2pCreateAndLinkStream function or by calling the niP2P Enable Peer to Peer Stream VI or nip2pEnableStream function.
4. Start the writer peer first so that it begins sending data to the RF vector signal transceiver prior to initiation.
5. Query the NI-RFSG Space Available in Endpoint property or the NIRFSG_ATTR_P2P_ENDPOINT_FULLNESS_START_TRIGGER_LEVEL attribute or NIRFSG_ATTR_P2P_SPACE_AVAILABLE_IN_ENDPOINT attribute after

starting the writer peer to ensure that sufficient data has been transferred to the RF vector signal transceiver.

6. Call the niRFSG Initiate VI or the niRFSG_Initiate function.

## 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:

- 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 the Number of P2P Samples to Generate property to specify the sample number.
- Continuous Generation Using User-Requested Abort: Using the niRFSG Abort VI or the niRFSG_CheckGenerationStatus returns True for Done. 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 the niRFSG Check Generation Status VI or theniRFSG_CheckGenerationStatus function to determine whether generation is complete. After generation is done, you can abort the generation using the niRFSG Abort VI or the niRFSG_Abort function. 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 the niP2P Flush and Disable Peer to Peer Stream VI or nip2pFlushAndDisableStream function or niP2P Disable Peer to Peer Stream VI or nip2pDisablePeertoPeerStream function and wait for the done? parameter from the niRFSG Check Generation Status VI or niRFSG_CheckGenerationStatus function to assert before aborting the RF vector signal transceiver. If data is still in the endpoint when the niRFSG Abort VI or niRFSG_Abort function 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 the niP2P Flush and Disable Peer to Peer Stream VI or nip2pFlushAndDisableStream function before starting a new acquisition. The niP2P Flush and Disable Peer to Peer Stream VI automatically clears the remaining data from the stream.
- Configure the reader peer to read the remaining samples in the stream, then call the niP2P Disable Peer to Peer Stream VI or nip2pDisablePeertoPeerStream function before starting a new acquisition.

## Device Warm-Up

NI recommends warming up the PXIe-5830 hardware for 30 minutes before operation.
The unit is fully functional prior to this time, but frequency, amplitude accuracy, and other specifications are not at warranted levels until the device has fully completed warming up.

**Note** Warm up begins when the PXI Express chassis has been powered on and the operating system has completely loaded.

## Power On, Reset, and Download Conditions

The PXIe-5830 sets some hardware circuitry to certain states at power on and at device reset. The PXIe-5830 applies certain conditions to the device state upon FPGA reset.

## Power On Conditions

Power on conditions are present after powering on or restarting the system and until an FPGA VI has been uploaded to the device.

- DIO lines are configured as input terminals.
- PFI 0 line is configured as an input terminal.

- PXI trigger lines are not configured and appear to the bus as high impedance.
- ADCs are reset to the component default state.
- DACs are reset to the component default state.

## Reset Conditions

The following conditions apply to the device state upon FPGA reset.

- DIO lines are configured as input terminals.
- PFI 0 line is configured as an input terminal.
- PXI trigger lines are not configured and appear to the bus as high impedance; however, reservations are maintained.
- ADC states are maintained.
- DAC states are maintained.

**Note** Reset conditions apply only when using the instrument design libraries.

## Download Conditions

The following conditions apply to the device state upon downloading a new FPGA VI to the PXIe-5830.

- DIO lines are configured as input terminals.
- PFI 0 line is configured as an input terminal.
- PXI trigger lines are not configured (appear to the bus as high impedance); however, reservations are maintained.
- ADC states are maintained.
- DAC states are maintained.

**Note** The Data Clock is disabled immediately after downloading a new FPGA VI to the PXIe-5830.

## Synchronization Using NI-RFSA and NI-RFSG

The PXIe-5830 supports NI-TClk, which allows for multidevice synchronization with fine alignment.
Refer to the **NI-TClk Synchronization Help** for more information about device synchronization.

## Unsupported Software Features

The following NI-RFSA and NI-RFSG driver software features are not supported on the PXIe-5830.

| Unsupported Feature | NI-RFSA | NI-RFSG |
|---|---|---|
| RF list mode | ✓ | ✓ |
| RF blanking | — | ✓ |
| 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.

## LO Sharing Using NI-RFSA and NI-RFSG

To reduce phase noise for some measurements, the PXIe-5830 allows for sharing the same internal LO during NI-RFSA and NI-RFSG sessions.
To share the same LO between an NI-RFSA and an NI-RFSG session, you must initiate both sessions and ensure both sessions use the same shared setting.

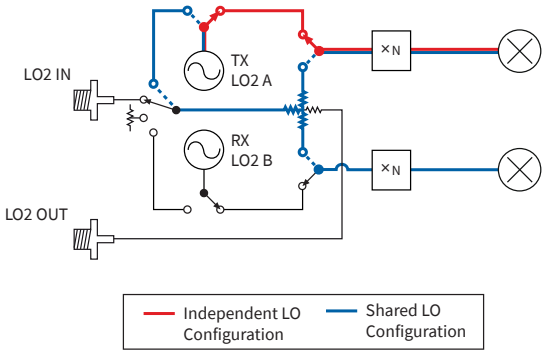
## Internal and External LOs

When LO sharing between NI-RFSA and NI-RFSG sessions, you can use an internal LO or external LO.

- **Onboard LO**—Refers to the onboard LO of the PXIe-3621.
- **External LO**—Refers to the LO of an external (non-NI) device.

## Shared and Independent Configurations

The following high-level block diagram shows the difference between an independent and shared LO configuration within the PXIe-3621 LO2 path.




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

### Related concepts:

- Local Oscillator

### Related reference:

- LO Sharing with IF Ports

## LO Sharing with IF Ports

Only the LO2 stage is used when connecting the PXIe-5830 with a DUT using one of the IF ports.

To share the same LO signal between NI-RFSA and NI-RFSG sessions, set the NI-RFSA or NI-RFSGLO Source property for LO2 as appropriate for your hardware configuration, as shown in the following table.

**Note** Only one LO can be used in a shared configuration at a time.

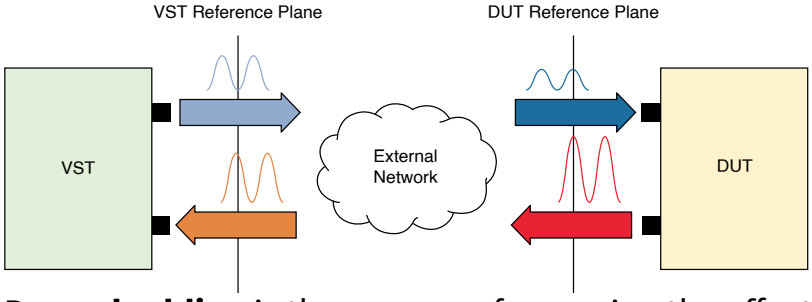
**Table 9.** LO2 Source Configurations

| PXIe-5830 Hardware Configuration | NI-RFSA and NI-RFSG LO Source Values | |
|---|---|---|
| | Property Value | Attribute Value |
| PXIe-3621 LO2 (independent configuration) | Onboard | ▪ NIRFSA_VAL_ONBOARD_STR<br>▪ NIRFSG_VAL_LO_SOURCE_ONBOARD_STR |
| External device LO connected to PXIe-3621LO2 IN (shared configuration) | LO In | ▪ NIRFSA_VAL_LO_IN_STR<br>▪ NIRFSG_VAL_LO_SOURCE_LO_IN_STR |
| PXIe-3621 LO2 (shared configuration) | SG SA Shared | ▪ NIRFSA_VAL_LO_SOURCE_SG_SA_SHARED_STR<br>▪ NIRFSG_VAL_LO_SOURCE_SG_SA_SHARED_STR |

## 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 4.** 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. 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.

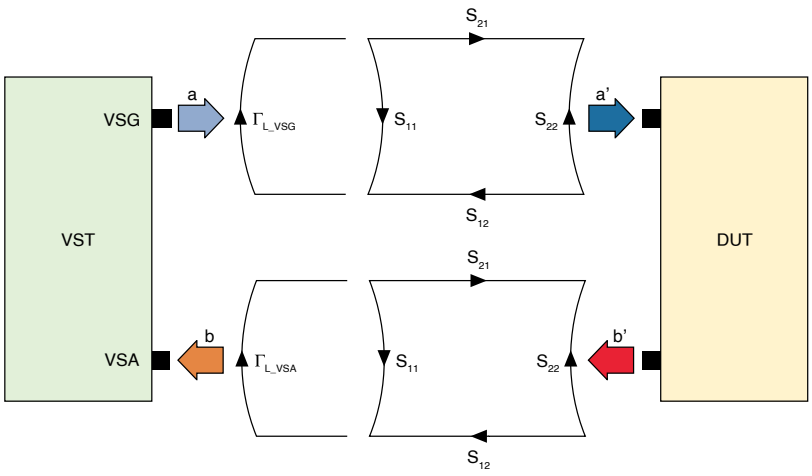
**Table 10.** S-parameters

| 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:

- $\Gamma_{L\_VSA}$
- $\Gamma_{L\_VSG}$

**Figure 5.** 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.

## Frequency and Bandwidth Configuration

This section describes how to use the NI-RFSA and NI-RFSG drivers to configure the PXIe-5830 instrument for optimal performance by specifying frequency and bandwidth constraints.

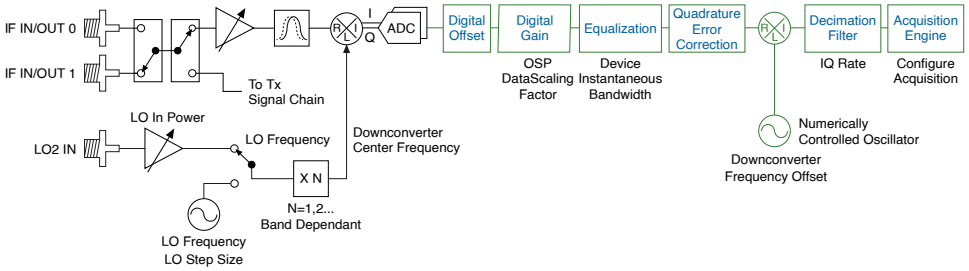
The following table shows the equivalencies between NI-RFSA and NI-RFSG properties mentioned throughout the section.

**Table 11.** 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 |

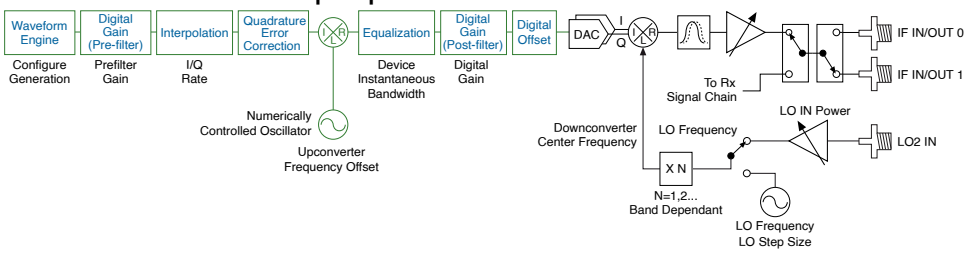
| NI-RFSA Property | NI-RFSG Property |
|---|---|
| Downconverter Frequency Offset Mode | Upconverter Frequency Offset Mode |

The following block diagram represents the PXIe-5830 RF input connectors and the associated NI-RFSA properties.




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

The following block diagram represents the PXIe-5830 RF output connectors and the associated NI-RFSG properties.




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

## Related information:

- NI-RFSA Programming Reference
- NI-RFSG Programming Reference

# 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

This section describes how to configure the PXIe-5830 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 and bandwidth of your instrument.

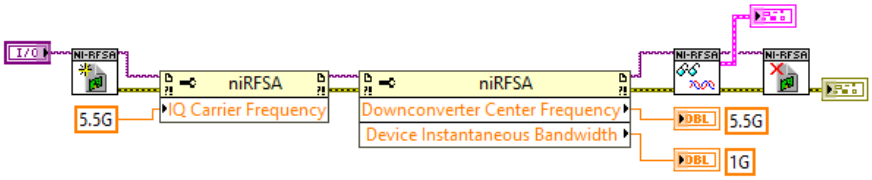
## Basic Behavior

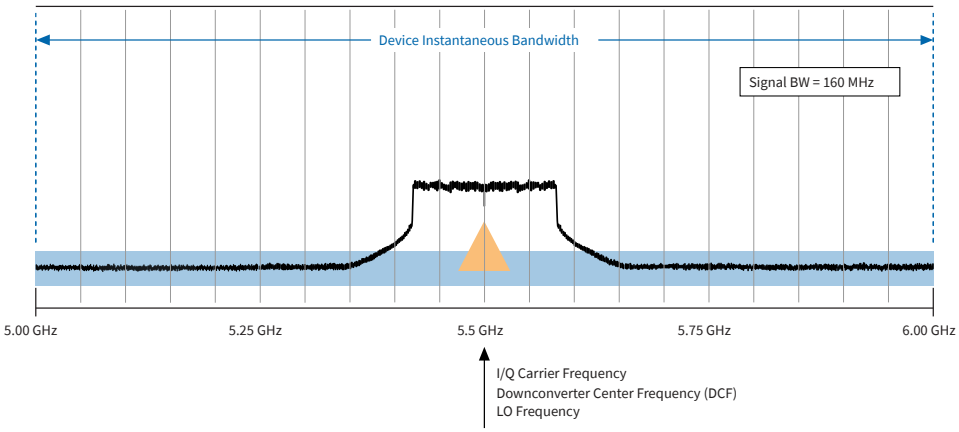
For a simple acquisition, you can set the I/Q carrier frequency and the I/Q rate. When you do not specify a bandwidth constraint, NI-RFSA selects the widest filter for the given frequency region. You can obtain the bandwidth of the instrument for this configuration by querying the Device Instantaneous 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 will configure the decimation filter and, as a result, configure 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.

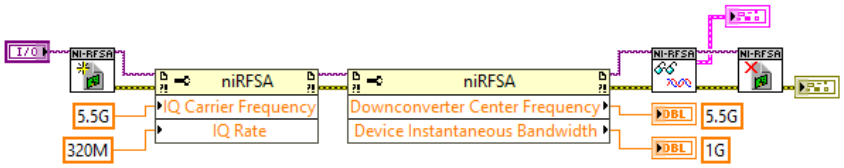
The following example illustrates the default behavior.

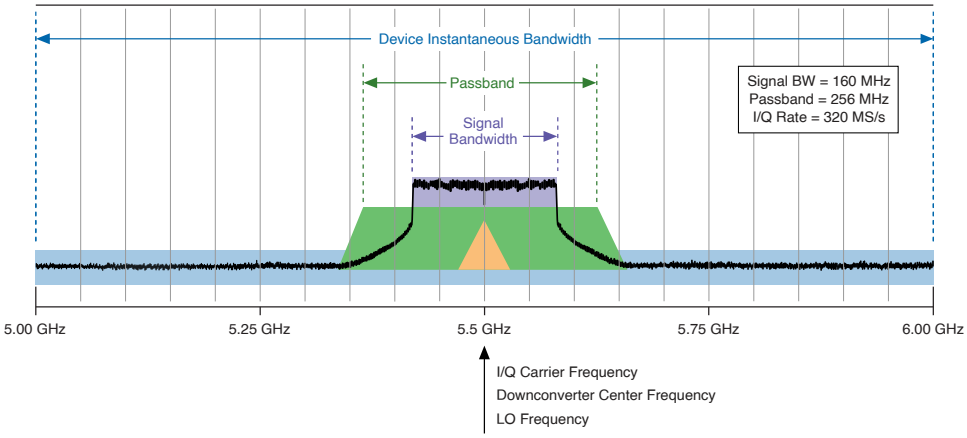



## Specifying Device Instantaneous Bandwidth

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. In the following example, only 160 MHz of bandwidth is needed from the instrument, so NI-RFSA chooses the next largest filter, which is 1 GHz in this example.




## 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

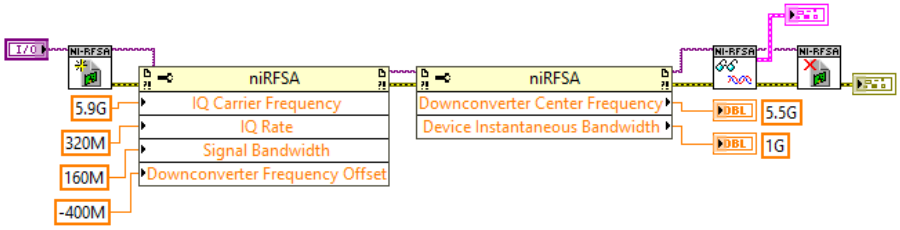
To avoid LO leakage present in the passband, you can apply a frequency offset by setting the NI-RFSA Downconverter Frequency Offset property or the NI-RFSG Upconverter Frequency Offset property. This property maintains a constant offset if

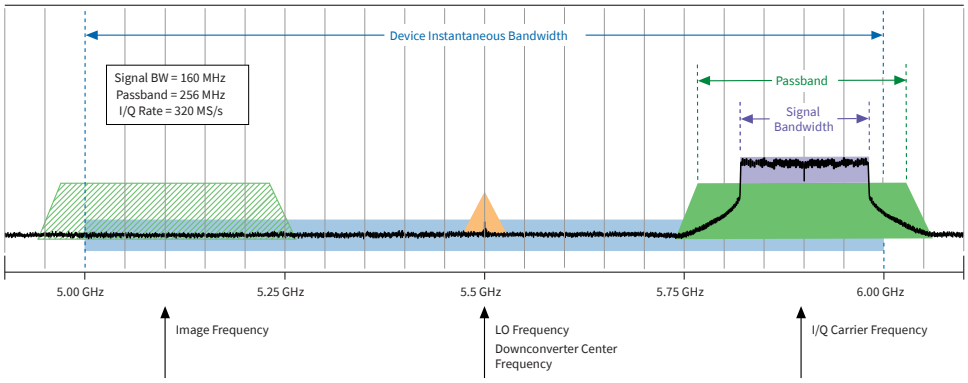
you vary the I/Q center frequency. You can always read back the actual device instantaneous bandwidth and downconverter center frequency.

## 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.

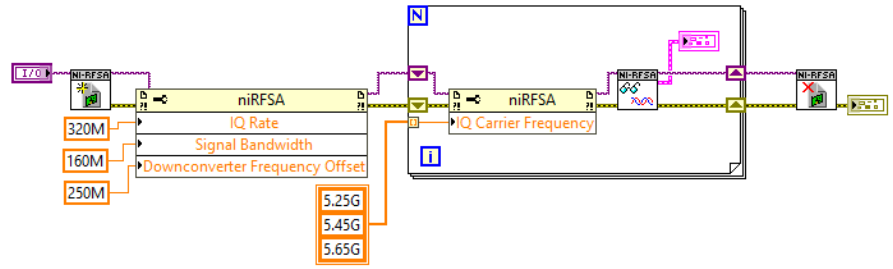
The following example shows how to programmatically specify the signal bandwidth.

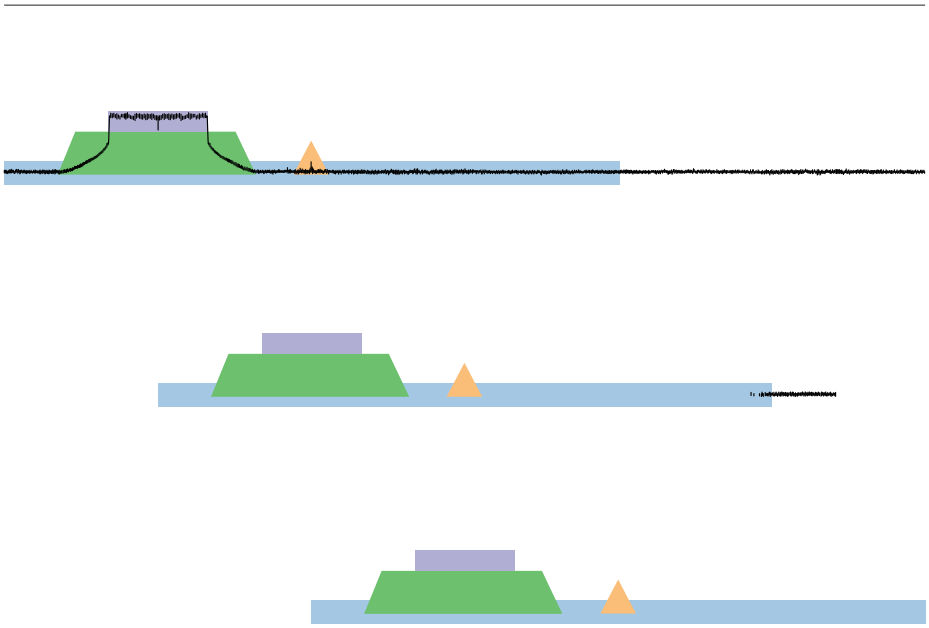



Note that setting the signal bandwidth does not necessarily avoid hardware reconfigurations during in-band retuning, but the settings stay relative to the I/Q carrier frequency, so it may be easier to manage in your application. The following example shows three acquisitions with the same I/Q rate, signal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

downconverter frequency and only changes the I/Q carrier frequency for each acquisition.



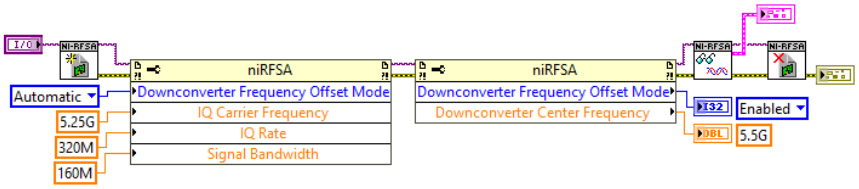
## Automatic Frequency Offset

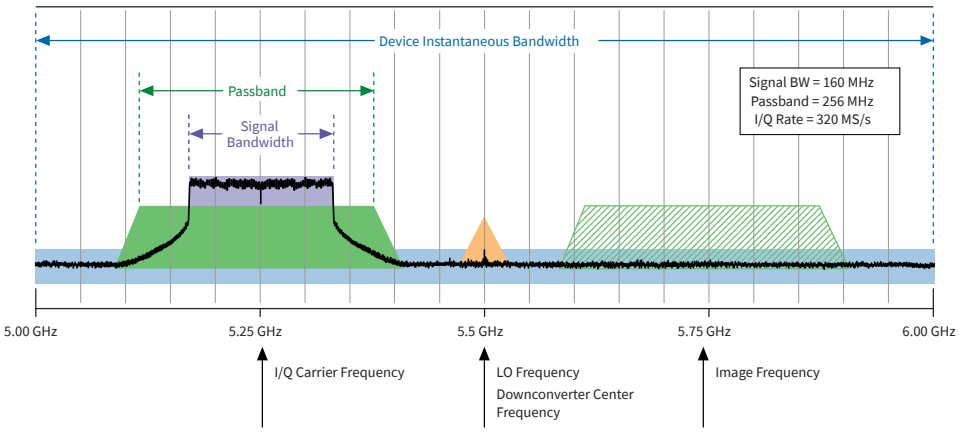
The PXIe-5830 can automatically offset the downconverter center frequency from the carrier frequency when sufficient instrument bandwidth is available. For example, the signal must fit within half the device instantaneous bandwidth minus an LO guard band. The actual frequency offset applied can be determined by reading the coerced downconverter center frequency. For this 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 specifications document for your instrument. Otherwise, using the user-defined offset mode, the gain accuracy with an offset is limited to the accuracy specification +/- the flatness specification.

**Note** When using a shared LO source across NI-RFSA and NI-RFSG, you must ensure the frequency configuration, including any frequency offset, matches.

The following example shows how to programmatically set the PXIe-5830 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.



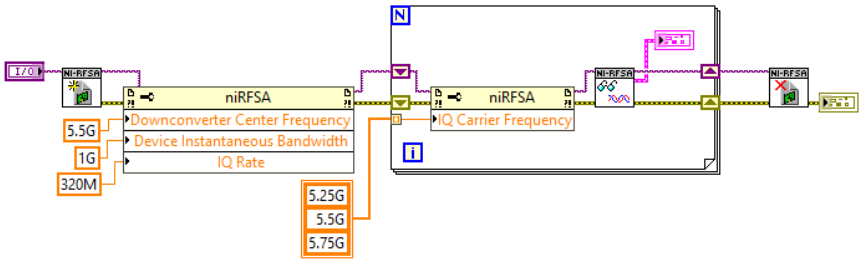

## In-Band Retuning

In-band retuning allows the instrument to quickly tune between signals of interest or generation within a given device instantaneous bandwidth. The advantages of in-

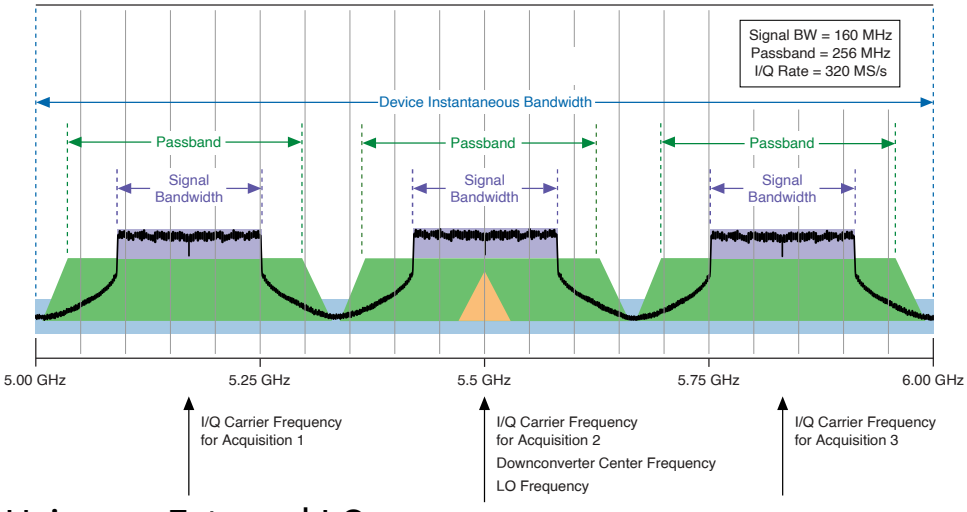
band retuning include reduced settling time for both frequency and amplitude. However, in-band retuning can only occur within the supported instantaneous bandwidth of the instrument. You can use digital tuning in combination with LO tuning to more efficiently sweep over a spectrum for supported applications.

By setting device instantaneous bandwidth and leaving the downconverter center frequency fixed, you can significantly reduce reconfiguration and settling times. When you set the I/Q carrier frequency away from the downconverter center frequency but still within the device instantaneous bandwidth, the instrument digitally shifts the waveform of interest. This is referred to as **in-band retuning**.

The following example shows how to programmatically set the PXIe-5830 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-3621 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-3621 LO IN connector.

### Specifying LO Frequency

Your external LO 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.

1. Set all the properties for the acquisition or generation.
2. Query the LO frequency.
3. Set the frequency of your LO source to the value returned in step 2.

This is important because the LO frequency may be offset or a divided down value of the desired I/Q carrier frequency.

### Specifying LO Power

To achieve better image and LO leakage performance, complete the following steps.

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 step 1.

### Related information:

- NI-RFSA Programming Reference

- NI-RFSG Programming Reference

## Calibration

The PXIe-5830 supports external adjustment of the following parameters:

- Frequency reference
- Internal LO2 path gain
- External LO2 path gain
- IF input gain
- IF output gain
- IF input frequency response
- IF output frequency response

## Self-Calibration

Self-calibration adjusts the PXIe-5830 for variations in the module environment using onboard power detectors.
PXIe-5830 modules are externally calibrated at the factory; however, you should perform a self-calibration in any of the following situations:

- After first installing and interconnecting your PXIe-5830 instrument
- After any module in the chassis is installed, uninstalled, or moved
- After inter-module cabling has been changed, reconnected, or repositioned
- After changing controllers, computers, or reinstalling the driver software
- 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** To ensure specifications before you use the instrument or perform self-calibration, power on the PXI Express chassis, set the chassis fan speeds to AUTO or HIGH, depending on the chassis; wait for the operating system to completely load; and then allow the instrument to warm up for 30 minutes.

**Note** Ensure that any connected external hardware is in an idle state and not transmitting during self-calibration. It is not recommended to perform self-calibration while connected in direct loopback configurations.

### Related concepts:

- Partial Self-Calibration

### Related tasks:

- Performing a Self-Calibration Using NI-RFSA

## Self-Calibration Adjustment Parameters

Self-calibration adjusts the following parameters of the PXIe-5830 to facilitate temperature correction:

- LO2 internal path gain
- IF input gain
- IF output gain
- IF input frequency response
- IF output frequency response
- IF input residual LO power
- IF output residual LO power
- IF input residual sideband image
- IF output residual sideband image

**Note** RF residual LO power and RF residual sideband images are only adjustable in self-calibration and cannot be corrected for during external adjustment. Self-calibration must be successfully executed after an external adjustment to ensure warranted performance.

**Note** NI recommends that no external signals are transmitted to the RF input port during the self-calibration.

## Performing a Self-Calibration Using NI-RFSA

NI recommends you perform the self-calibration using the NI-RFSA example when you self-calibrate the module.

1. Select **Start** » **All Programs** » **National Instruments** » **LabVIEW** to launch LabVIEW.
2. Launch the Example Finder and navigate to the example.
    a. Select **Hardware Input and Output** » **Modular Instruments** » **NI-RFSA** » **Utilities** » **RFSA Self Calibration**.
    b. Open the example.
3. Complete the following steps to configure the example:
    a. Select the device identifier assigned to the PXIe-3621 in MAX in the [**Resource Name**] drop-down menu.
    b. Set Clock Source to **OnboardClock**.
    c. Set Self Calibration Step Operations to **Perform All Self Calibration Steps**.
4. Run the VI.

## Partial Self-Calibration

The partial self-calibration feature self-calibrates all configurations within the specified frequency and reference level limits of the PXIe-5830.
For best results, NI recommends that you perform partial self-calibration without omitting any steps. However, if certain aspects of performance are less important for your application, you can omit the following steps for faster execution:

- LO Self Cal—Omits the LO path gain adjustment.
- Amplitude Accuracy—Omits the RF absolute amplitude accuracy adjustment.
- Residual LO Power—Omits the RF residual LO power adjustment.
- Image Suppression—Omits the RF residual sideband suppression adjustment.

**Note** The default partial self-calibration does not omit any steps.

## Performing Partial Self-Calibration Using LabVIEW

To perform a partial self-calibration when using LabVIEW, use the niRFSA Self Calibrate Range VI or the niRFSG Self Calibrate Range VI.

**Note** The niRFSA Self Calibrate Range VI does not perform any RF output calibration steps and the niRFSG Self Calibrate Range VI does not perform any RF input calibration steps.

**Note** Do not access the RF input, RF output, or basecard clock circuitry while either of these VIs are executing.

**Notice** Resetting the driver results in the loss of any stored partial self-calibration data. Restarting the computer typically resets the driver.

**Note** NI recommends that no external signals are transmitted to the RF input port during the partial self-calibration.

For more information about calibration, contact NI or visit ni.com/calibration. You can obtain the calibration certificate for your product at ni.com/calibration.

## 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).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 or NIRFSA_ATTR_ABSOLUTE_DELAY attribute. 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 or NIRFSA_ATTR_ABSOLUTE_DELAY attribute, and store the value for future synchronization sessions.

To apply the manual calibration synchronization results, write the stored value to the Absolute Delay property or NIRFSA_ATTR_ABSOLUTE_DELAY attribute before calling the niTClk Synchronize VI or niTClk_Synchronize function.

## 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 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 or NIRFSG_ATTR_ABSOLUTE_DELAY attribute. 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 or NIRFSG_ATTR_ABSOLUTE_DELAY attribute, and store the value for future synchronization sessions.

To apply the manual calibration synchronization results, write the stored value to the Absolute Delay property or NIRFSG_ATTR_ABSOLUTE_DELAY attribute before calling the niTClk Synchronize VI or niTClk_Synchronize function.