SignalCore™

PRESERVING SIGNAL INTEGRITY

Hardware Manual

SC5317A & SC5318A

6 GHz to 26.5 GHz RF Downconverter

www.signalcore.com

Table of Contents

1 General Information

1.1 Warranty

This product is warranted against defects in materials and workmanship for a period of three years from the date of shipment. SignalCore will, at its option, repair or replace equipment that proves to be defective during the warranty period. This warranty includes parts and labor.

Before any equipment will be accepted for warranty repair or replacement, a Return Material Authorization (RMA) number must be obtained from a SignalCore customer service representative and clearly marked on the outside of the return package. SignalCore will pay all shipping costs relating to warranty repair or replacement.

SignalCore strives to make the information in this document as accurate as possible. The document has been carefully reviewed for technical and typographic accuracy. If technical or typographical errors exist, SignalCore reserves the right to make changes to subsequent editions of this document without prior notice to possessors of this edition. Please contact SignalCore if errors are suspected. In no event shall SignalCore be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, SIGNALCORE, INCORPORATED MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF SIGNALCORE, INCORPORATED SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. SIGNALCORE, INCORPORATED WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of SignalCore, Incorporated will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against SignalCore, Incorporated must be brought within one year after the cause of action accrues. SignalCore, Incorporated shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow SignalCore, Incorporated's installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

1.2 Copyright & Trademarks

SignalCore, Incorporated respects the intellectual property rights of others, and we ask those who use our products to do the same. Copyright and other intellectual property laws protect our

products. Use of SignalCore products is restricted to applications that do not infringe on the intellectual property rights of others.

"SignalCore", "signalcore.com", and the phrase "preserving signal integrity" are registered trademarks of SignalCore, Incorporated. Other product and company names mentioned herein are trademarks or trade names of their respective companies.

1.3 International Materials Declarations

SignalCore, Incorporated uses a fully RoHS compliant manufacturing process for our products. Therefore, SignalCore hereby declares that its products do not contain restricted materials as defined by European Union directive 2002/95/EC (EU RoHS) in any amounts higher than limits stated in the directive. This statement assumes reliable information and data provided by our component suppliers and may not have been independently verified through other means. For products sold into China, we also comply with the "Administrative Measure on the Control of Pollution Caused by Electronic Information Products" (China RoHS). In the current stage of this legislation, the content of six hazardous materials must be explicitly declared. Each of those materials, and the categorical amount present in our products, are shown below:

鉛 Lead (Pb)	汞 Mercury (Hg)	镉 Cadmium (Cd)	六价铬 Hexavalent Chromium (Cr(VI))	多溴联苯 Polybrominated biphenyls (PBB)	多溴二苯醚 Polybrominated diphenyl ethers (PBDE)
✓	✓	✓	✓	✓	✓

A ✓ indicates that the hazardous substance contained in all of the homogeneous materials for this product is below the limit requirement in SJ/T11363-2006. An X indicates that the particular hazardous substance contained in at least one of the homogeneous materials used for this product is above the limit requirement in SJ/T11363-2006.

1.4 CE European Union EMC & Safety Compliance Declaration

The European Conformity (CE) marking is affixed to products with input of 50 - 1,000 Vac or 75 - 1,500 Vdc and/or for products which may cause or be affected by electromagnetic disturbance. The CE marking symbolizes conformity of the product with the applicable requirements. CE compliance is a manufacturer's self-declaration allowing products to circulate freely within the European Union (EU). SignalCore products meet the essential requirements of Directives 2004/108/EC (EMC) and 2006/95/EC (product safety) and comply with the relevant standards. Standards for Measurement, Control and Laboratory Equipment include EN 61326 and EN 55011 for EMC, and EN 61010-1 for product safety.

1.5 Warnings Regarding Use of SignalCore Products

(1) PRODUCTS FOR SALE BY SIGNALCORE, INCORPORATED ARE NOT DESIGNED WITH COMPONENTS NOR TESTED FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.

(2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED “SYSTEM FAILURES”). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE SOLELY RELIANT UPON ANY ONE COMPONENT DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM SIGNALCORE'S TESTING PLATFORMS, AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE SIGNALCORE PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY SIGNALCORE, THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF SIGNALCORE PRODUCTS WHENEVER SIGNALCORE PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

2 Physical Description

2.1 Unpacking

All SignalCore products ship in antistatic packaging (bags) to prevent damage from electrostatic discharge (ESD). Under certain conditions, an ESD event can instantly and permanently damage several of the components found in SignalCore products. Therefore, to avoid damage when handling any SignalCore hardware, you must take the following precautions:

1. Ground yourself using a grounding strap or by touching a grounded metal object.
2. Touch the antistatic bag to a grounded metal object before removing the hardware from its packaging.
3. **NEVER touch exposed signal pins.** Due to the inherent performance degradation caused by ESD protection circuits in the RF path, the device has minimal ESD protection against direct injection of ESD into the RF signal pins.
4. When not in use, store all SignalCore products in their original antistatic bags.

Remove the product from its packaging and inspect it for loose components or any signs of damage. Notify SignalCore immediately if the product appears damaged in any way.

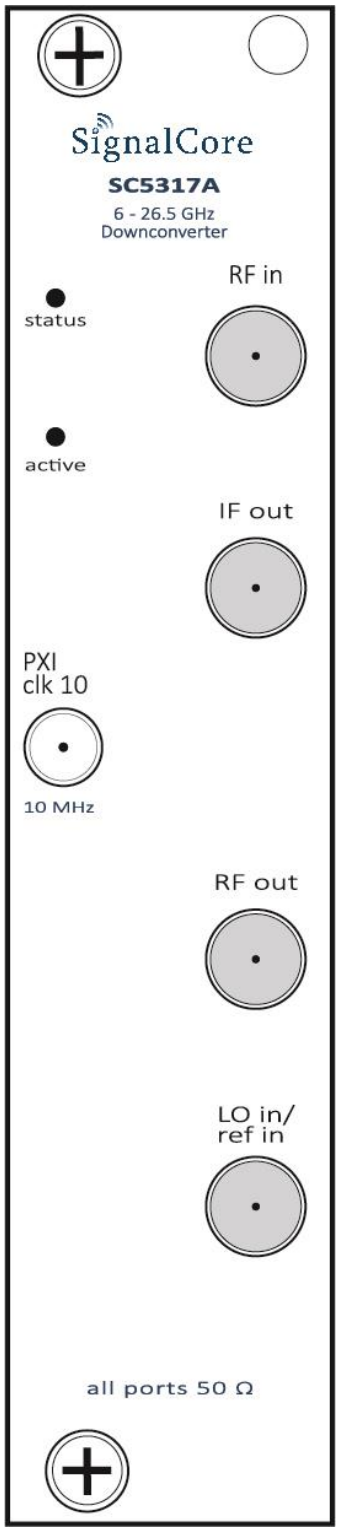
2.2 Setting Up the Device

Integration of the SC5317A and SC5318A modules requires attention to maintain effective cooling. Inadequate cooling can cause the temperature inside the RF housing to rise above the maximum for this product, leading to improper performance, reduction of product lifespan, or complete product failure. SignalCore suggests providing moderate airflow across the RF housing. If active cooling is not an option, use thermal interface materials to bond the RF housing to a larger heatsinking surface (i.e. a system enclosure). As each device's integrated system configuration is unique, detailed cooling options cannot be provided.

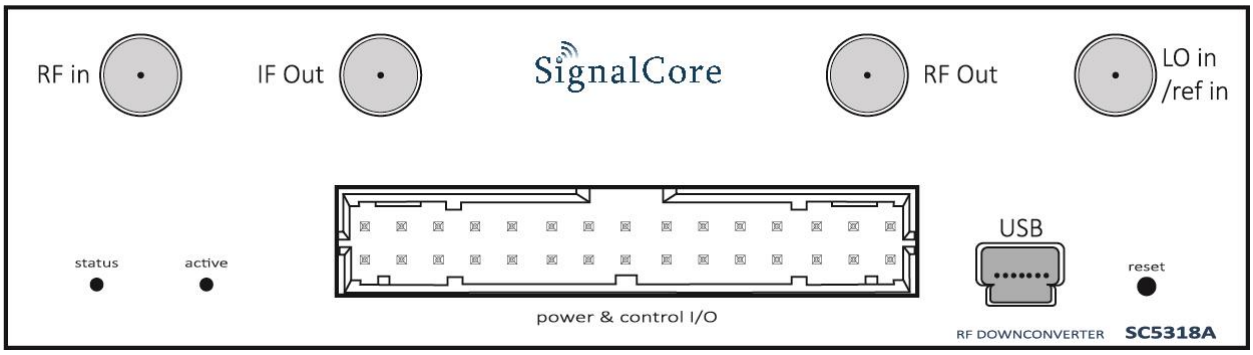
A cooling plan is sufficient when the SC5317A and SC5318A on-board temperature sensors indicate a rise of no more than 20°C above ambient temperature under normal operating conditions.

2.3 Front Interface Indicators and Connectors

The SC5317A is a PXIe-based RF downconverter with all user I/O located on the front face of the module.



The SC5318A is a serial controlled core module with all user connections on the front face of the module.



2.3.1 Signal Connections

All signal connections (ports) on the device are female SMA-type. Exercise caution when fastening cables to the signal connections. Over-tightening any connection can cause permanent damage to the device.

The condition of your system signal connections can significantly affect measurement accuracy and repeatability. Connections that are improperly mated, dirty, damaged, or worn can degrade measurement performance. Clean out any loose, dry debris from connectors with clean, low-pressure air (available in spray cans from office supply stores).

If deeper cleaning is necessary, use lint-free swabs and isopropyl alcohol to gently clean inside the connector barrel and the external threads. Do not mate connectors until the alcohol has completely evaporated. Excess liquid alcohol trapped inside the connector may degrade measurement performance until fully evaporated (this may take several days).

Tighten all SMA connections with 3 in-lb min to 5 in-lb max (56 N-cm max)

RF In This is the RF input port to the device with nominal impedance of 50 Ω. Its maximum input power is +27 dBm.

RF out This is the bypass output port of the RF input signal with nominal impedance of 50 Ω.

IF out This is the IF output of the device with nominal impedance of 50 Ω.

LO/Ref In This is the 10 MHz refence input to the device, enabling the device to phase lock its internal clocks to an external reference source. This nominally 50 Ω port is AC coupled. This port is also shared with the device external LO input. Maximum rated input power is +10 dBm for both Ref and LO signals.

LO rear This is the other selectable external LO input port. Only one LO port, either the front or rear, is selected as the input.

2.3.2 Device LED Indicators

These are LED indicator lights for the device, and their functions are listed in *Table 1* and *Table 2*.

Table 1. Status LED Indicator

LED Color	Description
Green	The device is functioning properly in the state that it is programmed for.
Amber	Indicates that all functions are on standby mode.
Red	Indicates that one or more local oscillators are not functioning correctly.
Off	No supply or supply error

Table 2. Active Indicator

LED Color	Description
Green	An external interface port has accessed the device.
Red	Input supply voltage exceeded
Off	No current interface access

2.3.3 Communication and Supply Connection

Figure 1. Power and Digital IO Connector

Power and communication to the SC5318A is provided through a rectangular connector from Samtec whose part number is TFM-115-01-L-D-RA. It also serves as the digital connector interface for RS232/SPI, trigger, and other digital signals. With exception for the RS232 logic levels, all other logic levels are 3.3V LVCMOS whose inputs are also 5V tolerant. A mating pig-tail cable, part number SFSD-15-28-H-10.00-SR, is provided with the product. The pin definitions are listed in *Table 3*.

Pinouts are different for different SignalCore products with the same connector type. Please ensure that mating connectors and cables are wired correctly before connection.

Table 3. Interface connector pin out description

PIN #	Description
22,24,26,28,30	12V Supply Rail
3,7,11,15,19,21,23,25,27,29	GND
13,14,16,17,18,20	Reserved, Do Not Connect
2	RX for RS232 or MISO for SPI (Host referred)

PIN #	Description
4	TX for RS232 or MOSI for SPI (Host referred)
5	SPI CLK
6	SPI CS_b
8	RS232 Baud Rate: 1(default) -> 57600, 0 -> 115200 SPI Mode Select: 1 mode 1, 0 mode 0
9	Device status
10	Device active
12	System Reset, Logic 0 to Reset Device
18	Reserved, Pull High to 3.3V or DNC
1	SRDY. Optional serial ready for SPI. DNC for RS232

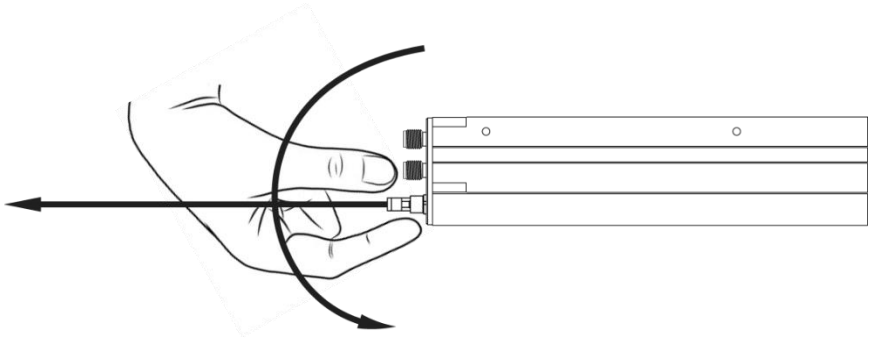
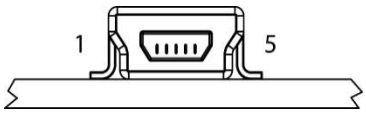


Figure 2. Proper Removal of Latching-Style Ribbon Connectors.

2.3.4 Mini-USB Connection



This is a mini-USB Type B connector for USB communication with the device using the standard USB 2.0 protocol (full speed) found on most host computers. The pinout of this connector, viewed from the front, is shown in the following table.

PIN #	USB Function	Description
1	VBUS	Vcc (+5 Volts)
2	D-	Serial Data (neg)
3	D+	Serial Data (pos)
4	ID	Not Used
5	GND	Device Ground (also tied to connector shell)

2.3.5 Reset Button (Pin Hole)

Behind this pin hole is the reset button, which is only available on the SC5318A. Using a pin to lightly depress this momentary-action push button switch will cause a hard reset to the device, putting it back to its default settings. All user settings will be lost. System reset capability can also be accessed through the communication header connector.

3 Functional Description

3.1 Overview

The SC5318A uses USB as its primary interface with an optional SPI or RS232 interface. The SC5317A is a PXIe version of the product.

The downconverter assembly consists of 2 module parts:

- The Signal Conversion Module contains the mixers, filters, signal amplifiers, and attenuators. This module is referred to as the "signal chain". This conversion module also includes a built-in LO synthesizer.
- The Power Conditioning and Digital Control Board contains the supply switchers that generate the needs rails for the RF modules and an onboard MCU that provides both the computation engine and interface between the user and the RF modules.

The figure below shows how the modules relate to each other.

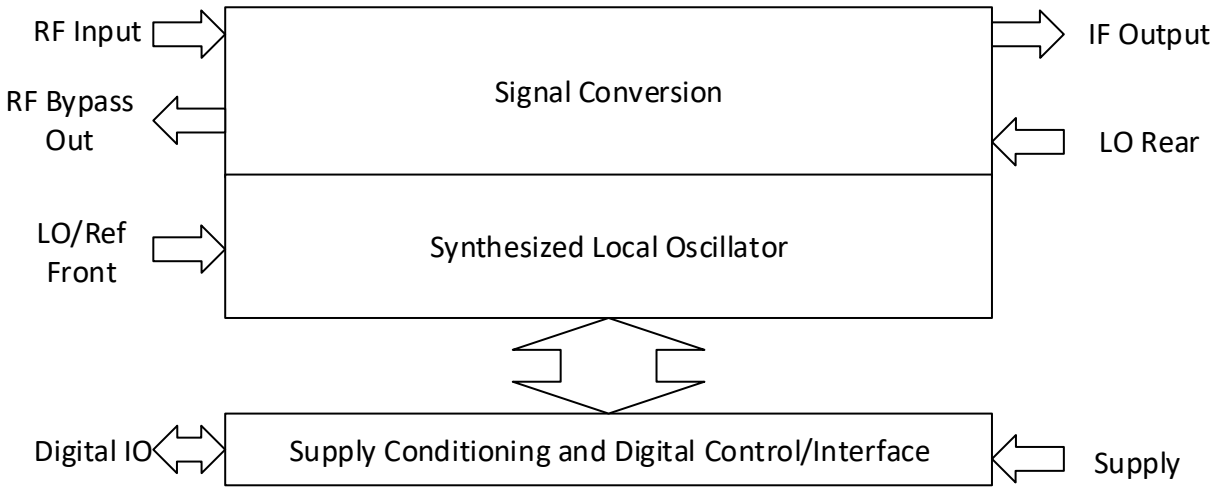


Figure 3. Simplified Block Representation of the Downconverter Assembly Module

The downconverter has only a single conversion stage that converts the RF signal to an intermediate frequency (IF) stage. The input range of this downconverter is from 6 GHz to 26.5 GHz, while the output IF range is from 50 MHz to 3000 MHz. The LO signal can be selected from the internal synthesizer or an external signal generator. The frequency accuracy and stability of the internal LO is derived from an onboard Temperature Controlled Crystal Oscillator (TCXO) with accuracy better than 500 ppb. For better accuracy and stability, the synthesizer can lock to an external reference of higher precision.

3.2 The Signal Chain

The conversion module contains the mixers, filters, amplifiers, and attenuators used to convert, purify, and maintain the amplitude of the signal, all the way from the input of the device to its output.

This cascade of signal conversion and conditioning stages is often referred to as the signal chain, which is shown on the system block drawing of the next figure.

3.2.1 The RF Input

All RF connectors are SMA types rated to 26.5 GHz of operation. The RF port is AC coupled with a 0.1 uF broadband capacitor as indicated in the next figure. The input frequency range at this port is slightly less than 6 GHz to slightly greater than 26.5 GHz. The upper limit to which the device tunes to above 26.5 GHz varies slightly from unit to unit depending on the limits of the oscillator circuitry. The usability of the upper out-of-bounds region depends on the roll-off response of the mixer.

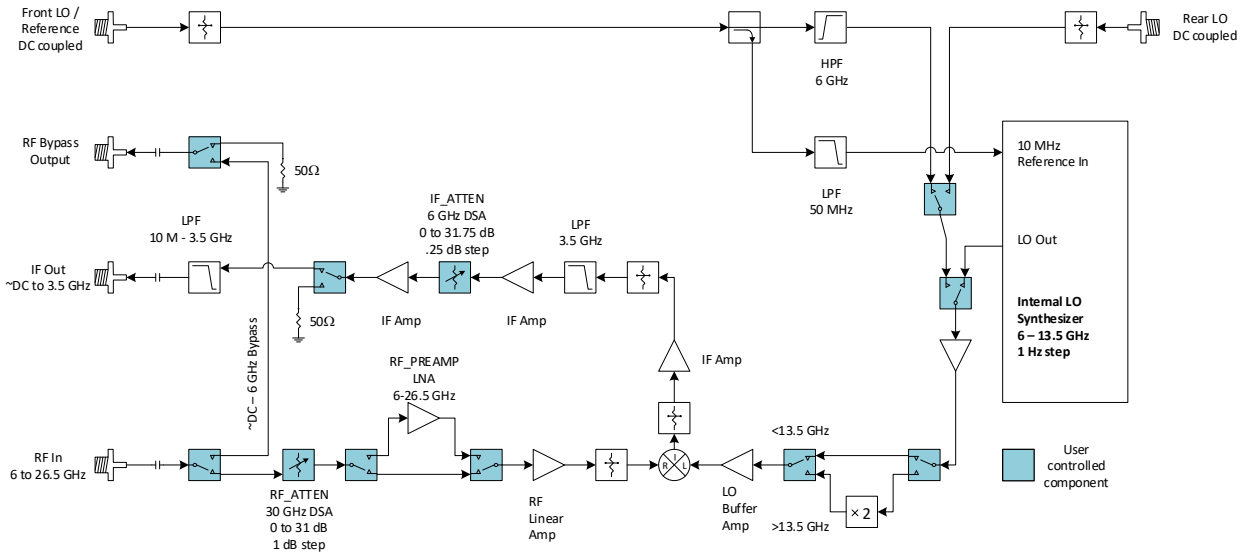


Figure 4. Block diagram of the downconverter assembly

3.2.2 The Input Path Switch

Immediately following the capacitor is an RF switch to selectively direct the RF signal down two paths:

1. The default is the conversion path where the RF signal is converted to an IF.
2. The alternative path is directly to the RF output port, bypassing all means of frequency conversion. The upper frequency limit of this path is about 7 GHz, although path loss calibration is only up to 6 GHz.

When the conversion path is bypassed, the conversion path and the internal LO are disabled by powering down.

3.2.3 The RF Path

The first device in the RF conversion path is a step attenuator with 1 dB step resolution. This attenuator (RF_ATTEN) is used to control the signal level at the mixer or at the RF preamplifier. If the RF level at the input of these devices is relatively large with respect to their input compression points, the signal would experience greater nonlinear effects, distorting its waveform, and producing large spurious 3rd intermodulation distortion effects.

Following RF_ATTEN is a switchable RF amplifier that can be switched into the signal path to improve the device sensitivity, effectively lowering the device noise figure. In other words, the effective input noise level of the device is lowered when this amplifier is enabled. When signals with levels lower than -30 dBm are expected, turning on this amplifier is recommended to improve the signal-to-noise dynamic range (DR_{SNR}). The typical maximum gain of the downconverter without the amplifier enabled is +20 dB, and when the amplifier is enabled, its additional +20 dB will boost the typical maximum gain to +45 dB.

There are no band selective filters in the RF path so in open environments the device is unable to differentiate a RF signal from its image. A band select filter at the RF input port is recommended to properly filter out the signal of interest before down conversion.

3.2.4 The First Mixer and IF Path

The mixer (RF mixer) of the downconverter is extremely critical as it sets the dynamic ranges of the device, both the signal-to-noise and third order IMD (DR_{IMD}) dynamic ranges. The DR_{SNR} and DR_{IMD} are directly related to the mixer input compression point (IP1dB) and input third-order intercept point (IIP3). The IIP3 of the mixer is typically better than +18 dBm, and its P1dB is better than 5 dBm. In applications that require better signal linearity, it is recommended that the level at the mixer input is kept at -20 dBm or less. Typically, the IMD is better than -75 dBc for two -20 dBm tones at the mixer. Increasing the level at the mixer improves SNR of the IF output signal at the expense of higher nonlinear byproducts.

Another important characteristic of the mixer is the LO-IF isolation and LO-RF isolation. The higher the isolation, the lower the leakage LO signal is at the RF and IF ports. The RF mixer is driven with a ~+17 dBm LO signal and the LO-IF isolation is about 45 dB, resulting in a -28 dBm LO leakage in the IF path. The low-pass IF filters suppress this leakage to levels well below -60 dBm. The LO-RF isolation is also about 45 dB so the LO leakage at the RF port of the mixer is on the order of -28 dBm. The high reverse isolation of the RF linear power amplifier suppresses this LO leakage to < 60 dBm. When the low noise pre-amplifier is enabled, it will provide another 30 dB of reverse isolation, lowering the LO leakage at the RF input port to < 90 dBm.

The frequency relationship between the three ports of the mixer is given as:

$$IF = \begin{cases} LO - RF \\ RF - LO \end{cases}$$

From the above equation, the IF output spectrum is inverted with respect to the RF in the first relationship, whereas in the second it is noninverted. When the LO > RF, the IF signal is inverted and it is not inverted when LO < RF.

3.2.5 Signal Chain Configurations

Recommended signal chain configurations are provided below for various application scenarios. These are strictly recommendations; they are not indicative of the downconverter function limitations; users will need to adapt the device to their applications at hand.

3.2.5.1 *Signals Bypassing the Conversion Stage*

For frequencies less than 6 GHz, the RF input signal can be directed to the RF output port, bypassing the conversion process. This RF output signal may be used by another downconverter to bring it down to baseband IF, for example, the SC5308A 6 GHz downconverter mates well with this device to form an integrated converter with continuous frequency coverage from 100kHz to 26.5 GHz.

3.2.5.2 *Dynamic Range Setting*

There are 2 digital step attenuators to control the conversion gain of the downconverter. The first set consists of RF_ATTEN in the RF stage and the second set consists of IF_ATTEN in the IF stage. Both attenuators have 30 dB of settable range, however the IF attenuator has fine resolution of 0.25 dB while the RF attenuator has resolution of 1 dB.

To set the downconverter for better sensitivity or better SNR, the gain should be shifted to the RF input path of the device before the mixer. The RF pre-amplifier should be enabled if necessary and/or RF attenuation reduced. The IF attenuator is then used to adjust the final IF output level. The drawback is that the signal level starts off higher as it enters the first mixer as well as the subsequent components. As a result, the apparent linearity of the device is lower.

To set the device for better linearity, the gain should be shifted to the output IF path (after the mixer) and reduced in the RF path. The signal power level at the first mixer should be lower than -20 dB for improved linearity. Since the input signal is low, the relative SNR will be lower. But, as the first mixer and subsequent components experience lower power levels, the apparent linearity of the device is improved.

When the device gain is balanced well, the device could achieve SNR better than 130 dBc/Hz while maintaining IMD3 levels close to 75 dBc. These numbers are representative of converters used in large box high end spectrum analyzers. When the device is optimized for best SNR, typical values greater than -150 dBc/Hz can be achieved, and when the device is optimized for sensitivity, the input spectral noise floor is typically lower than -165 dBm/Hz. The flexible use of these attenuators and pre-amplifier allows the downconverter to achieve better than 190 dB of measurement dynamic range.

3.3 The LO Synthesizer and External Ports

The internal LO synthesizer is a hybrid between integer-N PLL and DDS, enabling it to tune at 1 Hz steps while maintaining low phase noise. The reference signal for the generation of the LO signal comes from an internal Temperature Controlled Crystal Oscillator (TCXO).

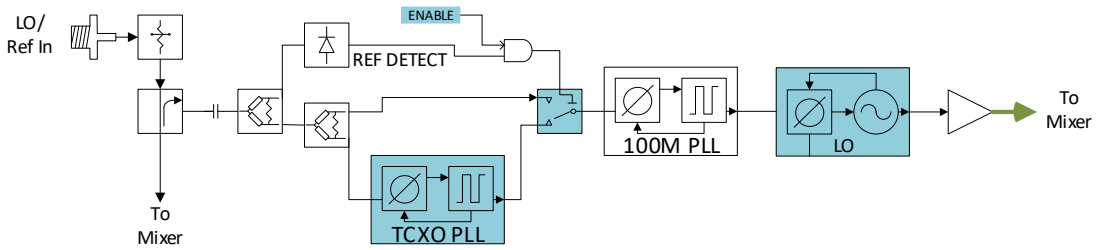


Figure 5. Block diagram of the local oscillator

3.3.1 The Reference Clocks

The base clock of the downconverter is a 10 MHz Voltage Controlled Temperature Controlled Crystal Oscillator (VCTCXO) with initial accuracy better than 500 ppb once the device has reached a stable temperature. Its initial accuracy is set at the factory via an on-board 14-bit voltage reference DAC. This DAC is accessible for dynamic accuracy calibration. The other reference is a 100 MHz Voltage Controlled Crystal Oscillator (VCXO), which is phase locked to the base reference whenever an external reference source is not used.

When an external reference is selected as the base clock by enabling the device to phase lock to it, the effect only occurs when the presence of this reference is detected. In other words, although the device is programmed to lock externally it will not attempt to do so until the reference signal is detected at the input reference port. Notice, both the reference clocks (TCXO and VCXO) will attempt to lock to the external source. Having the VCXO lock directly to the external source has the advantage of utilizing the close-in phase noise of the source; it is best to assume that the external source is superior to the internal base. Although the internal VCTCXO is not used when an external reference is selected, it is important to have it remain powered on to maintain its temperature stability. Like the VCXO, it is also phase locked to the external source.

3.3.2 LO Ports

There are 2 selectable LO ports, one located in the front of the device and the other on the opposite end (rear). The port in the front is shared between the external LO and reference clock. A coupler exists internally to direct the 10 MHz reference signal to the internal LO circuitry for phase locking. The frequency range of the LO is between 6 GHz and 18 GHz and its signal can be doubled with the internal doubler to 26.5 GHz. The input frequency range of the doubler is from 13.5 GHz to 27 GHz.

3.4 The Interface Module

The following figure shows the power and control interface block for the SC5318A. All supply rails are produced on board; their voltages are regulated and actively filtered to keep noise to a minimum. Thus, these downconverters are tolerant to “dirty” external power supplies.

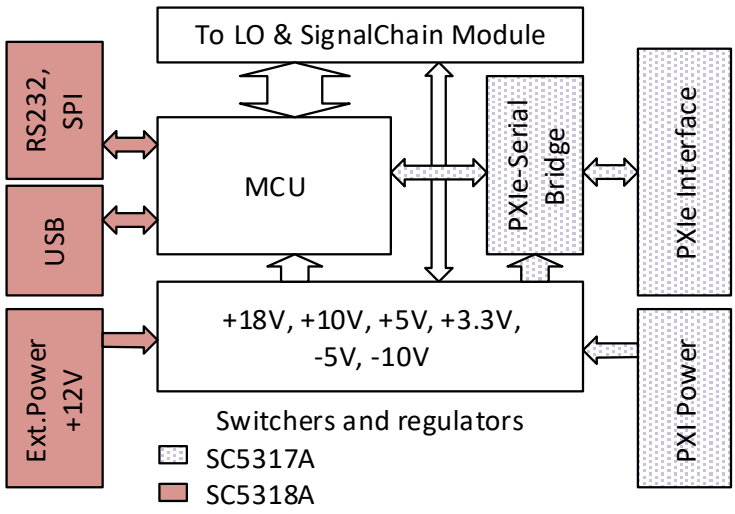


Figure 6. Block diagram of the power and interface module

4 Hardware Registers

The set of hardware registers of the downconverter may be divided into a configuration set and a query set; the configuration registers are write-only registers to set up the states of the device, while the query registers request the device to prepare and send back data associated with them. The registers are identical for all communication interfaces: PXIe, USB, RS232, and SPI. Data communication to all interfaces are sent in 1 byte wide, that is, data is transferred byte-by-byte. Register data lengths vary in size, and it is important that the correct number of bytes are sent for the corresponding register. Failure to do so may cause communication to lock-up and the device to become unresponsive.

4.1 Configuration Registers

These are write-only registers to configure the device. The registers vary in length to reduce redundant data and improve the communication speed, especially for SPI and RS232 interfaces. Furthermore, it is vitally important that the length of data written to a register is exact because failure to do so will cause the interfaces to misinterpret the incoming data, leaving the device in a stalled state. The total number of bytes is the sum of the register address (1 Byte) and its corresponding data bytes. For example, to set the RF frequency value, eight bytes must be written; the sum of the 1 register byte and 7 data bytes. See the RF_FREQUENCY register of Table 5. The table provides a summary of the configuration registers, and each register is explained in detail.

Table 4. Configuration Registers

Register Name	Reg Add	Serial Range	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INITIALIZE	0x01	[7:0]	Set to zeros [7:1]	Mode						
SYSTEM_ACTIVE	0x02	[7:0]	Set to zeros [7:1]	Enable “active” LED						
SYNTH_MODE	0x03	[7:0]	Set to zeros [7:2]			Fast-tune		Loop Gain		
RF_FREQUENCY	0x10	[7:0]	Frequency Word (Hz) [7:0]							
		[15:8]	Frequency Word (Hz) [15:8]							
		[23:16]	Frequency Word (Hz) [23:16]							
		[31:24]	Frequency Word (Hz) [31:24]							
		[39:32]	Frequency Word (Hz) [39:32]							
		[47:40]	Frequency Word (Hz) [47:40]							
		[55:48]	Frequency Word (Hz) [55:48]							
IF_FREQUENCY	0x11	[7:0]	Frequency Word (Hz) [7:0]							
		[15:8]	Frequency Word (Hz) [15:8]							
		[23:16]	Frequency Word (Hz) [23:16]							
		[31:24]	Frequency Word (Hz) [31:24]							
		[39:32]	Frequency Word (Hz) [39:32]							
		[47:40]	Frequency Word (Hz) [47:40]							
		[55:48]	Frequency Word (Hz) [55:48]							
LO_FREQUENCY	0x12	[7:0]	Frequency Word (Hz) [7:0]							
		[15:8]	Frequency Word (Hz) [15:8]							
		[23:16]	Frequency Word (Hz) [23:16]							
		[31:24]	Frequency Word (Hz) [31:24]							
		[39:32]	Frequency Word (Hz) [39:32]							
		[47:40]	Frequency Word (Hz) [47:40]							
		[55:48]	Frequency Word (Hz) [55:48]							

Register Name	Reg Add	Serial Range	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RF_AMP	0x14	[7:0]	Set to zeros [7:1]							Enable
ATTENUATOR	0x15	[7:0]	Attenuator Value							
		[15:8]	Set to zeros [15:11]	Attenuator Number						
		[23:16]	Set to zeros [23:16]							
SIGNAL_PATH	0x16	[7:0]	Set to zeros [7:4]				Invert Spectrm	IF Out Enable	Amp Enable	Bypass convert
STORE_DEFAULT_ STATE	0x18	[7:0]	Set all to zeros							
DEVICE_STANDY	0x19	[7:0]	Set to zeros [7:1]							Enable
REFERENCE_CLOCK	0x1A	[7:0]	[7:2] set to zeros					PXI 10 enable		Lock ext
REFERENCE_DAC	0x1B	[7:0]	DAC WORD [7:0]							
		[15:8]	zeros [15:14]	DAC WORD [13:8]						
		[23:16]	Set to zeros							
USER_EEPROM_ WRITE	0x1C	[7:0]	BYTE DATA [7:0]							
		[15:8]	ADDRESS [7:0]							
		[23:16]	ADDRESS [15:8]							
AUTO_CALC_GAIN	0x1D	[7:0]	[7:5] Set to 0			[4:2] Linearity Mode			Auto-amp enable	Auto-enable
		[15:8]	sign	RF Value (dB)						
		[23:16]	sign	Mixer Value (dB)						
		[31:24]	sign	IF Value (dB)						
		[47:32]	Set to zeros							
SYNTH_SELF_CAL	0x1F	[7:0]	Set all to zeros							

4.1.1 Register 0x01 INITIALIZE

This register initializes the devices to the default state or reprograms all the device components with the current state parameters. Note, this register does not need to be called in order to program the device.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[0]	WO	Mode	1	0 = Resets the current state 1 = Resets the default to the default or startup state
[7:1]	WO	Unused	7	Set to zeros
[7:0]	RO	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.2 Register 0x02 SYSTEM_ACTIVE

This register turns on or off the active LED indicator on the front connector interface of the device. This register should be called when the device is opened or closed in software.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[0]	WO	Mode	1	0 = Turns off the active LED 1 = Turns on the active LED
[7:1]	WO	Unused	7	Set to zeros
[7:0]	RO	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.3 Register 0x03 SYNTH_MODE (2 Bytes)

This register configures the PLL loop gain of the local oscillator synthesizers. It also enables or disables faster tuning of the YIG based oscillator of LO1.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[1:0]	WO	Loop Gain	2	0 = Low loop gain, improves phase noise > 50 kHz 1 = Normal loop gain 2 = High loop gain, improves phase noise < 50 kHz
[2]	WO	Fast Tune	1	0 = Turns fast tune off, default 1 = Turns fast tune on, close in phase noise may degrade
[7:3]	WO	Unused	5	Set to zeros
[7:0]	RO	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.4 Register 0x10 RF_FREQUENCY

This register tunes the device to the input RF frequency.

Bytes written	8
Bytes read	1

Bits	Type	Name	Width	Description
[55:0]	W	Frequency word	56	Frequency word in milli-Hz (mHz), mHz used for future compatibility
[7:0]	RO	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.5 Register 0x11 IF_FREQUENCY

This register sets the final IF value.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[55:0]	W	Frequency word	56	Frequency word in milli-Hz (mHz), mHz used for future compatibility
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.6 Register 0x12 LO_FREQUENCY

This register sets the final IF value.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[55:0]	W	Frequency word	56	Frequency word in milli-Hz (mHz), mHz used for future compatibility
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.7 Register 0x14 RF_AMP

This register enables and disables the RF amplifier.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[0]	W	Frequency word	1	0 = Disables the RF amplifier

Bits	Type	Name	Width	Description
				1 = Enables the RF amplifier
[7:1]	W	Unused	7	Set to zeros
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.8 Register 0x15 ATTENUATOR

This register sets the value of the device attenuators.

Bytes written	4
Bytes read	1

Bits	Type	Name	Width	Description
[7:0]	W	Attenuator value	8	In 0.25 dB LSB for IF, 1 dB LSB for RF
[10:8]	W	The target attenuator	3	The attenuator number: 0 = RF Atten 1 = IF Atten
[23:11]	W	Unused	13	Set to zeros
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.9 Register 0x16 SIGNAL_PATH

This register configures how the signal chain is routed.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[0]	W	Bypass Converter	1	0 = Normal conversion path 1 = Bypasses the conversion, switches RF input directly to RF output
[1]	W	RF Amp Enable	1	0 = Turns off RF preamplifier 1 = Turns on RF preamplifier

Bits	Type	Name	Width	Description
[2]	W	IF Out Enable	1	0 = Disables the IF output port 1 = Enables the IF output port
[3]	W	Invert Spectrum	1	0 = Sets the LO to invert the IF spectrum 1 = Sets the LO to not invert the IF spectrum
[7:4]	W	Not used	14	Set to zeros
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.10 Register 0x18 STORE_DEFAULT_STATE

This register will store all current settings of the device into EEPROM and will become the default startup setting.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[7:0]	W	Unused	8	Set to zeros
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.11 Register 0x19 DEVICE_STANDBY

This register sets either the entire device or sections of the device into standby mode. Placing a section into standby involves powering down its circuitry. This conserves power and eliminates unwanted LO signals when they are not used.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[0]	W	State	1	0 = Powers down analog section 1 = Enables and powers on analog section
[7:1]	W	Unused	4	Set to zeros
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.12 Register 0x1A REFERENCE_CLOCK

This register configures the reference clock behavior.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[0]	W	LockEnable	1	0 = Uses internal 10 MHz TCXO
[1]	W	PXI10Enable	1	Only on SC5317A to enable export of the 10 MHz backplane clock
[7:2]	W	Unused	4	Set to zeros
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.13 Register 0x1B REFERENCE_DAC

This register makes adjustments to the 10 MHz TCXO accuracy via DAC to its tuning port.

Bytes written	4
Bytes read	1

Bits	Type	Name	Width	Description
[15:0]	W	Tuning DAC word	16	DAC WORD
[23:16]	W	Unused	8	Set to zeros
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.14 Register 0x1C USER_EEPROM_WRITE

This register writes a byte to the on board EEPROM.

Bytes written	4
Bytes read	1

Bits	Type	Name	Width	Description
[7:0]	W	Data	8	Byte to be written to the address

Bits	Type	Name	Width	Description
[23:8]	W	Address	16	EEPROM address
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.15 Register 0x1D AUTO_CALC_GAIN

This register stores the user configuration for auto-gain-conversion use.

Bytes written	6
Bytes read	1

Bits	Type	Name	Width	Description
[0]	W	Auto-gain-enable	1	1 sets the current state to enable auto gain setting
[1]	W	Auto-amp-set	1	1 calculations allows to automatically change the setting of the amplifier
[4:2]	W	Linearity Mode	3	0 – use mixer level 1 – balance of SNR to linearity 2 – better SNR 3 – best SNR 4 – better linearity 5 – best linearity
[15:8]	W	RF level	8	Bit 15 = sign bit, bits 14:8 Set the value in dB
[23:16]	W	Mixer level	8	Bit 15 = sign bit, bits 14:8 Set the value in dB
[31:24]	W	IF level	8	Bit 15 = sign bit, bits 14:8 Set the value in dB
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.1.16 Register 0x1F SYNTH_SELF_CAL

This register will start the YIG synthesizer calibration. Note that although the calibration procedure takes about 6-8 seconds to complete, the register returns a byte almost immediately.

Bytes written	2
Bytes read	1

Bits	Type	Name	Width	Description
[7:0]	W	Unused	8	Set to zeros
[7:0]	R	Read back byte	8	Read 1 byte back is required for PXIe and RS232

4.2 Query Registers

These are request for data registers, in that a request for certain data is made by writing to the specific register first, followed by reading back the requested data. Some registers may require instruction data to specify the type of data to return, while others do not need any. For example, the register GET_DEVICE_PARAM (0x30) returns the RF Frequency, IF1 Frequency, IF3 Frequency, etc.; the data returned depends on the parameter value of the instruction byte.

Returned data length is always 8 bytes (64 bits), with the first byte being the most significant (MSB). It is important that all 8 bytes are read to clear the interface buffers. We will examine how the different interface buses handle the return data in the *Device Information Parameters and Format* section.

Table 5. Query Registers

Register Name	Reg Add	Serial Range	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GET_DEVICE_PARAM	0x30	[7:0]					Parameter			
GET_TEMPERATURE	0x31	[7:0]	Zeros [7:0]							
GET_DEVICE_STATUS	0x32	[7:0]	Zeros [7:0]							
GET_DEVICE_INFO	0x33	[7:0]					Info			
CAL_EEPROM_READ	0x34	[7:0]	EEPROM ADDRESS [7:0]							
		[15:8]	EEPROM ADDRESS [15:8]							
USER_EEPROM_READ	0x35	[7:0]	EEPROM ADDRESS [7:0]							
		[15:8]	EEPROM ADDRESS [15:8]							
SERIAL_OUT_BUFFER	0x36	[7:0]	Zeros [7:0]							
		[15:8]	Zeros [15:8]							
		[23:16]	Zeros [23:16]							
		[31:24]	Zeros [31:24]							
		[39:32]	Zeros [39:32]							
		[47:40]	Zeros [47:40]							
		[55:48]	Zeros [55:48]							

4.2.1 Register 0x30 GET_DEVICE_PARAM

Write to this register the required device parameter to query from the device.

Bytes written	2
Bytes read	8

Bits	Type	Name	Width	Description
[3:0]	W	Parameter	1	0: Returns current RF frequency 1: Current IF frequency 2: Current LO frequency 3: Current Other parameters
[7:4]	W	Unused	4	Set to zeros
[63:0]	R	Read back bytes	64	Returned Data (See Device Parameter Data section for more info)

4.2.2 Register 0x31 GET_TEMPERATURE

Write to this register to query the device temperature.

Bytes written	2
Bytes read	8

Bits	Type	Name	Width	Description
[7:0]	W	Unused	7	Set to zeros
[31:0]	R	Temperature Data	32	These 32 bits of data need to be type casted back to float. i.e. var_float = *(float*)&var_u32 where var_u32 is some unsigned integer that holds the 32 bits of read data.
[63:32]	R	Invalid data	32	Ignore

4.2.3 Register 0x32 GET_DEVICE_STATUS

Write to this register to query the current operating conditions.

Bytes written	2
Bytes read	8

Bits	Type	Name	Width	Description
[7:0]	W	Unused	7	Set to zeros
[0]	R	Pll status: LO1 sum	1	The summing PLL of LO
[1]	R	Pll status: LO1 crs	1	The coarse tuning PLL of LO
[2]	R	Pll status: LO1 fine	1	The fine tuning PLL of LO
[3]	R	Pll status: VCXO	1	100 MHz VCXO
[4]	R	Pll status: TCXO	1	TCXO, only valid when lock to external reference is enabled
[6:5]	R	Loop Gain	2	Low, Normal, High (0,1,2)
[8]	R	Device Accessed	1	When the SET SYS ACTIVE register is accessed
[9]	R	ExtRefDetected	1	If an external reference is detected
[10]	R	LockExtRef	1	Enable device to lock to external reference
[11]	R	LoPwrEnable	1	LO section power status
[12]	R	ExtLoEnable	1	Select (0/1) internal/external LO
[13]	R	extLoSelect	1	Select (0/1) front/rear LO ports
[14]	R	loMode	1	Using RF&IF to calc LO value(0), use LO direct(1)
[15]	R	loDoublerEnable	1	The frequency doubler status
[16]	R	deviceStandby	1	Power down status
[17]	R	bypassConv	1	RF conversion bypassed
[18]	R	ifOutEnable	1	Enable IF output port
[19]	R	invertSpectrum	1	IF spectrum inverted
[20]	R	rfAmpEnable	1	RF pre-amp enabled

Bits	Type	Name	Width	Description
[21]	R	autoGainEnable	1	Only reflects the software setting for gain calculation, does not affect device.
[22]	R	autoAmpEnable	1	Only reflects the software setting for gain calculation, does not affect device.
[63:23]	R	Invalid data	29	Ignore

4.2.4 Register 0x33 GET_DEVICE_INFO

Write to this register to query the device information such as serial number.

Bytes written	2
Bytes read	8

Bits	Type	Name	Width	Description
[3:0]	W	DeviceInfo	4	0: Product serial number and Interface type 1: Firmware and hardware revision 2: Calibration and manufacture date
[7:4]	W	Unused	4	Ignore
[63:0]	R	Data	64	Information data (see Device Info data)

4.2.5 Register 0x34 CAL_EEPROM_READ

Write to this register to query 8 bytes of data from the calibration EEPROM at the starting address.

Bytes written	4
Bytes read	8

Bits	Type	Name	Width	Description
[15:0]	W	Address	16	Starting EEPROM address
[23:16]	W	Unused	8	Zeros
[63:0]	R	Data	64	8 bytes of data, LSB is the byte at the start address

4.2.6 Register 0x35 USER_EEPROM_READ

Write to this register to query 8 bytes of data from the user EEPROM at the starting address.

Bytes written	4
Bytes read	8

Bits	Type	Name	Width	Description
[15:0]	W	Address	16	Starting EEPROM address
[23:16]	W	Unused	8	Zeros
[63:0]	R	Data	64	8 bytes of data, LSB is the byte at the start address

4.2.7 Register 0x36 SERIAL_OUT_BUFFER

Writing to this register only provides the 64 clock edges (Reg + 7 data bytes) to transfer serial data from the device through SPI. Other interfaces do not use this register.

Bytes read	8

Bits	Type	Name	Width	Description
[55:0]	W	Unused	56	Zeros, just to provide clocking for SPI data from device
[63:0]	R	Data	64	8 bytes of data

4.2.8 Device Parameters Data and Format

The data read back from the GET_DEVICE_PARAM register has a total of 8 bytes, however not all bytes contain valid data. The table below shows the valid data for each of the parameters.

Table 6. Device parameter data

Device Parameter Name	Param num	Data type	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0
RF FREQ	0	U64	zeros	Data						
IF1 FREQ	1	U64	zeros	Data						
LO FREQ	2	U64	zeros	Data						
OTHER	3	Mixed	zeros	zeros	zeros	zeros	zeros	Sig Path	RF Atten	IF Atten

4.2.8.1 *Frequency parameter values*

Frequency parameter values are returned in the first 7 bytes with the last byte padded with zeros. The least significant bit (LSB) is 1/1000 Hertz (the data is returned in 1 one thousandth of a Hertz). Divide the data by 1000 to obtain the result in Hertz.

4.2.8.2 *Attenuator values*

Each attenuator value is returned as one byte, and the LSB is in 0.25 dB. Divide each byte by 4 to obtain the result in dB.

4.2.8.3 *Signal Chain Configuration*

The signal chain configuration data is in the first 2 bytes, and their data is represented in the following table.

Table 7. Signal chain configuration data representation

Bit	Description
[0]	Bypass Conversion
[1]	RF Amp Enable
[2]	IF Out Enable
[4:3]	Inverted Spectrum

4.2.9 Device Information Parameters and Format

Not all the 8 bytes read back contain valid data. The following table shows the valid bytes of data for each of the parameters.

Device Parameter Name	Param num	Data type	Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0
INTERFACE/SN	0	U32				Intrfce	Serial Number			
REVISIONS	1	F32	hardware revision				Firmware revision			
DATES	2	U32	Mfg year		Month	Day	Cal year		Month	Day

4.2.9.1 *Interface Information*

The first 4 bytes contain the device serial number as an unsigned 32-bit integer. Byte [4] contains the interface as represented in the following table.

Table 8. Interface ID

Bit	Description
[0]	PXI/PXIe (1 if available)
[1]	USB
[2]	SPI
[3]	RS232
[7:4]	Undefined

4.2.9.2 *Revision Information*

The first 4 bytes represent the hardware revision and last 4 bytes represent the firmware revision of the device. These 4 bytes encompass a 32-bit floating point number so the data needs to be type casted from an unsigned 32-bit value to float value.

4.2.9.3 *Date Information*

The first 4 bytes represent the manufactured date and the last 4 bytes represent the last calibration date. The date format is outlined in the following table.

Bit	Type	Description
[0]	U8	Day
[1]	U8	Month
[4:3]	U16	Year (i.e. 2016)

Section 2

Communication Interfaces and Calibration

5 Communication Interfaces

The SC5317A has a PXI express interface, while the SC5318A has 2 communication interfaces:

1. USB and SPI
2. USB and RS232

This section will examine the communication aspects of the product, focusing on data transfer to and from the device on each interface. Although the registers are identical for all interfaces, there are subtle differences in the implementation of the interfaces to transfer the data.

5.1 Communication Data Format

All data sent and received by all interfaces is sent as buffers of unsigned bytes. For example, to change RF frequency of the device to 12 GHz we perform the following:

1. Frequency is sent in 1000^{th} of Hertz, so the data that represents the frequency is 12,000,000,000,000 milli-Hertz.
2. This number can be represented by a 64-bit unsigned long, and in Hexadecimal is 0x 0000 0574 FBDE 6000. Only the least 6 bytes are needed to represent all frequencies allowable for this device.
3. A byte data buffer needs to be 8 bytes for register RF_FREQUENCY (address 0x10), so the byte array buffer to be sent would be:

   [0x10][0x00][0x05][0x74][0xFB][0xDE][0x60][0x00]

   The register address byte [0x10] is the first member of the buffer to be sent.

5.2 USB Interface

There are 2 transfer types for the USB interface.

- Control transfer
- Bulk transfer

5.2.1 Control Transfer

The USB control transfer parameters are:

ENDPOINT_IN	0x80
ENDPOINT_OUT	0x00
TYPE_VENDOR	0x40
RECIP_INTERFACE	0x01

5.2.2 Bulk Transfer

The USB bulk transfer parameters are:

ENDPOINT_IN	0x83
ENDPOINT_OUT	0x04

The bulk transfer from the host to the device operates on a loopback with a data buffer of 8 bytes. When a device register is addressed, and upon completion of the register task, such as changing frequency, it will send back 8 bytes, which the host must read to clear the transfer buffers. Unlike the other interface methods, where only the required number of bytes needs to be sent for a given register, 8 bytes are needed for every USB bulk transfer. For example, if a configuration register requires only 4 bytes to be sent, these bytes will be the first of the 8 bytes and the last 4 bytes are zeros. The returned 8 bytes do not carry valid data for a configuration register. However, they do carry valid data for query registers.

5.3 SPI Interface

The SPI interface on the device is implemented using an 8-bit (single Byte) buffer for both the input and output, hence, it needs to be read and cleared by the device before consecutive bytes can be transferred to and from it. The process of clearing the SPI buffer and decisively moving it into the appropriate register takes CPU time, so a time delay is required between consecutive bytes written to or read from the device by the host. The chip-select pin ($\overline{CS}$) must be asserted low before data is clocked in or out of the product. Furthermore, pin $\overline{CS}$ must be asserted low for the entire duration of a register transfer.

Once a full transfer has been received, the device will proceed to process the command and de-assert low the SRDY pin. The status of this pin may be monitored by the host because when it is de-asserted low, the device will ignore any incoming data. The device SPI is ready when the previous command is fully processed and the SRDY pin is re-asserted high. It is important that the host either monitors the SRDY pin or waits for 500 μs between register writes.

There are 2 SPI modes: 0 and 1. The default mode is 1 (C_polarity = 0, C_phase = 1), where data is clocked in and out of the device on the falling edge of the clock signal. In mode 0 (C_polarity = 0, C_phase = 0), data is clocked in and out on the rising edge. To select mode 0, pin 8 of the interface connector must be pulled low to ground as the device is powered on or as the reset line (pin 12) is toggled low-high. If pin 8 is pulled high or left unconnected, mode 1 is selected.

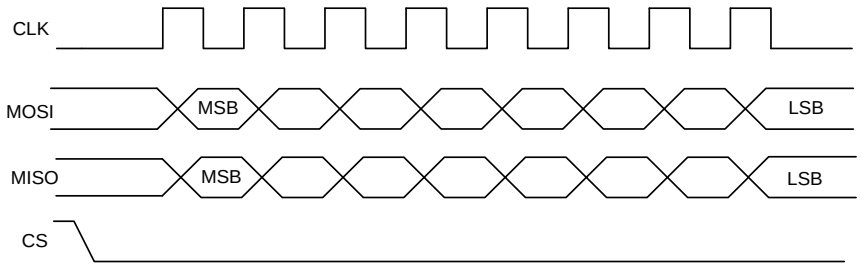


Figure 7. SPI Mode 1 shown.

Register writes are accomplished in a single write operation. Register lengths vary depending on the register. They vary in lengths of 2 to 8 bytes, with the first byte sent being the register address followed by the data associated with that register. The ($\overline{CS}$) pin must be asserted low for a minimum period of 1 μs (T_S, see *Figure 8*) before data is clocked in, and must remain low for the entire register write. The clock rate may be as high as 2.0 MHz (T_C = 0.5 μs), however, if the external SPI signals do not have sufficient integrity due to trace issues, the rate should be lowered.

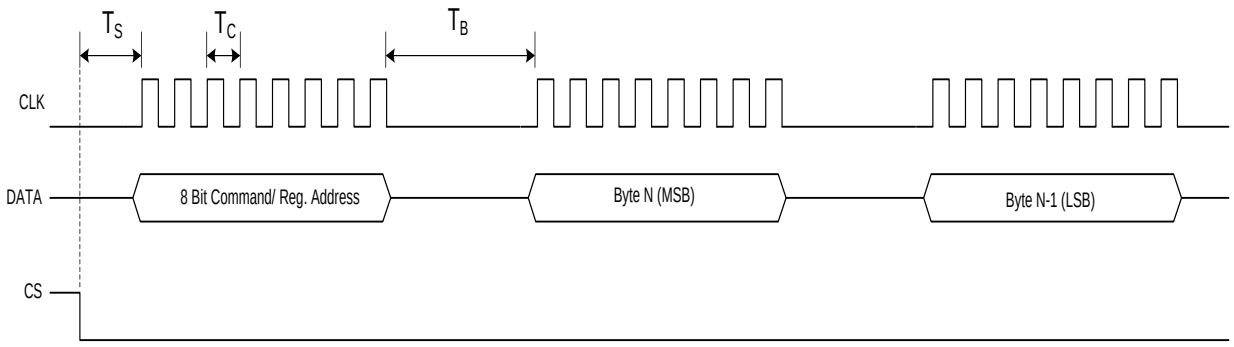


Figure 8. SPI timing.

As mentioned above, the SPI architecture limits the byte rate since after every byte transfer the input and output SPI buffers need to be cleared and loaded respectively by the device SPI engine. Data is transferred between the input buffer and internal register buffers. The time required to perform this task is indicated by T_B, which is the time interval between the end of one byte transfer and the beginning of another. The recommended minimum time delay for T_B is 5 μs. The number of bytes transferred depends on the register. It is important that the correct number of bytes is transferred for the associated register, because once the first byte (MSB) containing the device register address is received, the device will wait for the desired number of associated data bytes. The device will hang if an insufficient number of bytes are written to the register. To clear a hung condition, the device will need an external hard reset. The time required to process a command is also dependent on the command itself. Measured times for command completions are typically between 50 μs to 300 μs after reception.

5.3.1 Writing the SPI Bus

The SPI transfer size (in bytes) depends on the register being targeted. The first byte sent is the register address and subsequent bytes contain the data associated with the register. As data from the host is being transferred to the device via the MOSI line, data present on its SPI output buffer is simultaneously transferred back, MSB first, via the MISO line. The data returned is invalid for configuration registers. The following figure shows the contents of a single 3-byte SPI command written to the device. The Hardware Registers section provides information on the number of data bytes and their contents for an associated register. There is a minimum of 1 data byte for each register even if the data contents are “zeros”.

23								15								7							0
	Register Address								Byte 1								Byte 0						

Figure 9. Example of a 3-byte write

5.3.2 Reading the SPI Bus

Data is simultaneously read back during an SPI transfer cycle. Requested data from a prior command is available on the device SPI output buffers, and these are transferred back to the user host via the MISO pin. To obtain valid requested data would require querying the SERIAL_OUT_BUFFER, which requires 8 bytes or 64 clock cycles; 1 byte for the device register

(0x36) and 7 empty bytes (MOSI) to clock out the returned data (MISO). An example of reading the device RF parameters (IF3 frequency) from the device is shown in the following figure.

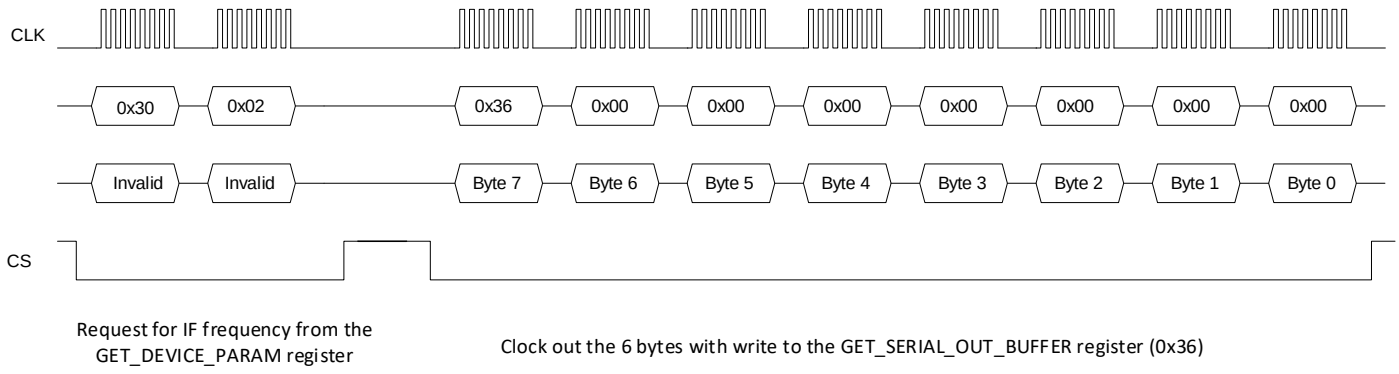


Figure 10. Query example: Write followed by Read to the GET_SERIAL_BUFFER

In the figure above, the first transfer cycle is to make the request for IF3 frequency data through the GET_DEVICE_PARAM register. The subsequent cycle is to clock the data that was requested by sending 64 clocks into the GET_SERIAL_BUFFER register.

5.4 RS232 Interface

The RS232 version of the SC5318A has a standard interface buffered by an RS232 transceiver so that it may interface directly with many host devices, such as a desktop computer. The interface connector for RS232 communication is labeled "Digital I/O" on the front of the panel. Refer to *Table 3. Interface connector pin out description* for position and pin-out information. The RS232 device communication control set is provided in the following table.

Table 9. RS232 Control Setting

Baud rate	Rate of transmission *Pin 8 of the Digital IO connector selects the rate. By default, if the pin is pulled high or open, the rate is set to 56700 at power up or upon HW reset. When the pin is pulled low or grounded, the rate is set to 115200 upon reset or power up.
Data bits	The number of bits in the data is fixed at 8.
Parity	None
Stop bits	1 stop bit
Flow control	none

Only 3-wire RS232 is required since hardware flow control is not used. These connections are the Tx, Rx, and Gnd. This interface is common on most host computers and microcontrollers, so user access to host ports is readily provided by the computer OS or microcontroller hardware registers.

5.4.1 Writing to the Device Via RS232

It is important that all necessary bytes associated with any one register are fully sent. In other words, if a register requires a total of 6 bytes (address plus data), then all 6 bytes must be sent even though the last byte may be null. The device, upon receiving the first register addressing byte, will wait for all the associated data bytes before acting on the register instruction. Failure to complete the register transmission will cause the device to behave erratically or hang. Information for writing to the configuration registers is provided in *Table 4. Configuration Registers*. Upon the execution of the register that was sent, the device will return one (1) byte of data with bit 1 high to indicate success. This byte must be read by the host to clear its receive buffer so that reading subsequent registers will not contain corrupted data. Furthermore, reading back this byte will ensure that the device is ready for the next register command.

5.4.2 Reading from the Device Via RS232

To query information from the device, the query registers are addressed, and data is returned. *Table 5. Query Registers* contains the query register information. As with the configuration registers, it is important that all data byte(s) (write) associated with the query registers are sent even if they are null. All queries will return 8 bytes of data (read) with the first received byte being the most significant (MSB). Section *4.2 Query Registers* provides the format details of the received data.

5.5 PXI Express

The PXIe interface contains a high-speed PCIe-to-Serial bridge chip. This bridge chip communicates with the onboard microcontroller serially. The interface on the bridge chip resides at offset addresses between 0x00 and 0xFF from BAR0; which is memory mapped. A kernel level driver for the operating system is needed to access this memory address. A simple driver using IO controls should be sufficient to read and write byte data to this block of addresses. Although SignalCore provides the driver and API for these products, information is provided here for users who may need to write drivers for a different operating system or a different driver. An example would be writing the API for the Linux operating system.

5.5.1 Setting Up the PCI to Serial Bridge

The serial function of the bridge chip must first be initialized before it can communicate with the onboard microcontroller, and hence communication between the microcontroller and the PXIe bus. The initialization can be done at the kernel level mode or at the user level mode, the decision is left to the user. The following table lists the programing order of the bridge register addresses to initialize and setup the serial port function.

Step	BAR0 Register Address	Data (Byte)
1	0x88	0x01
2	0x04	0x00
3	0x03	0x80
4	0x00	0x07
5	0x01	0x00

Step	BAR0 Register Address	Data (Byte)
6	0x03	0x0D
7	0x02	0x07

5.5.2 Writing to the Device

Bytes that are written to the device must go through the bridge chip. In this section, we will first look at the write cycle of each byte, and then the write cycle of each device register. Do note the difference between the bridge register addresses and the device register addresses.

5.5.2.1 *Single Byte Write*

The serial transfer buffer register address is located at 0x00 offset from BAR0 of the bridge chip, however, before writing byte data to this register, its status needs to be checked to confirm that it is ready to accept a new buffer of bytes. The status register is located at 0x05; it must be read and bit 7 must be **high** to indicate that the transfer register is ready to receive the next byte buffer. Checking the status register of the serial bridge chip is required before every new command write.

5.5.2.2 *Device Register Write*

The process of writing the device registers is the same as writing an RS232 port, so the description of Section *5.4.1 Writing to the Device Via RS232* is applicable. Writing the device registers involves sending byte-by-byte data as described previously. Section *4.1 Configuration Registers* provides information on the number of configuration write bytes needed for each device register. The first byte sent is the device register address, followed by the most significant byte of the register's associated data. When a device register is fully written, that is, all its data has been sent to the device, it will return 1 byte. This returned byte must be read (by the host) to clear the transfer buffer so that later received data are not corrupted. Section *5.5.3.1 Single Byte Read* describes how a byte read cycle is performed.

5.5.3 Reading from the Device

Device data is passed back to the host via the bridge chip byte-by-byte, so we will discuss a single byte read process and an entire register read process.

5.5.3.1 *Single Byte Read*

The serial transfer buffer register address is located at 0x00 offset from BAR0 of the bridge chip. Before valid data can be read from the transfer register, its ready status must first be confirmed. The status register is located at 0x05; it must be read and bit 0 must be **high** to indicate that valid data is available. Checking the status register for available data is required before **every byte** read.

5.5.3.2 *Device Register Read*

After a write request to the device is made, 8 bytes of data is available to be read back. Use the single byte read process, as mentioned previously, to read all the bytes. See Section 4.2 for information of the exact number of request write bytes and the number of request read bytes, which is 8. All 8 bytes must be read to fully clear the transfer buffer; the first byte read is the most significant byte.

6 Calibration

6.1 Calibration EEPROM Map

Table 10 represents the EEPROM map of the device calibration values. All values are stored as little-endian 4-byte floating point numbers. Every point is 4-bytes long. Access to the data is possible through the CAL_EEPROM_READ register, which reads 8 bytes starting at the address pointed to by the register input.

Table 10. Calibration EEPROM Map

Offset address	Points	Length (Bytes)	Data Type	Description
0x00	664	664	U_8	Factory Reserved
0x298	1	4	Float_32	Calibration Temperature
0x29C	6	12	Float_32	Temp coeffs, band1 = <13 GHz, <13Ghz, band2 < 20GHz, and band3 > 20 GHz. (c1, c2, c1, c2, c1, c2)
0x338	35	140	Float_32	IF response cal frequencies 100, 200, 300, …, 3500 in MHz
0x4B0	35	140	Float_32	IF3_response, relative gain to 1 GHz at each IF cal frequency
0x5C8	30	120	Float_32	Relative IF atten 1 dB…..30 dB, 1 dB step
0x6B8	60	240	Float_32	Bypass response cal frequencies 100, 200, 300, …, 6000 in MHz
0x7A8	60	240	Float_32	Bypass absolute gain at each bypass cal freq
0x898	83	332	Float_32	RF cal frequencies, 6 GHz to 26.5 GHz, 250 MHz step
0xBD0	83	332	Float_32	RF non-invert abs gain @ each rf cal freq
0xF08	83	332	Float_32	RF invert abs gain @ each rf cal freq
0x1240	83	332	Float_32	Rf rel amplifier gain @ each rf cal freq
0x1578	2490	9960	Float_32	RF atten, 1-30 dB, 1 dB step for each freq a1_f0, a1_f1,…,a1_fN a2_f0, a2_f1,…,a2_fN a30_f0, a30_f1,…,a30_fN

6.2 Absolute Conversion Gain

It is difficult to make gain measurements of every configuration setting of the downconverter as there are many combinations of attenuator and frequency settings, not to mention the various operating temperatures. However, the gain of the device can be computed to a reasonable level of accuracy with less data if the assumption is that every component setting is independent from one another. For example, assume the frequency response of the IF_ATTEN attenuator is the same for all attenuation settings. Another assumption we made, which may introduce the most error is the IF response is the same for all LO and RF values. As an example, the IF value at 2 GHz is the same for the sets (LO = 12 GHz, RF = 10 GHz) and (LO = 16 GHz, RF = 14 GHz).

6.3 Absolute Gain of the RF Conversion Path

Assuming independency, and to be able to compute the gain of the downconverter at any setting, an absolute reference state of the device must be determined first. Once this reference state is established, compensation can be applied to other relative configured states of the device. The established reference state of the device is as follows:

- All attenuators are set to 0 dB
- IF value is set at 1.25 GHz
- RF preamplifier is disabled.
- RF is tuned from 6000 MHz to 26500 MHz in 250 MHz steps

The gain of the device in this state is measured at different frequencies in the range of 6000 MHz to 26500 MHz. This set of measurements is the RF absolute gain stored at starting address 0xBD0 of the calibration EEPROM. All other measurements taken as deviations from this reference setting are relative.

6.4 Gain Through the Bypass RF Conversion Path

The absolute gain through the bypass path is stored beginning at address 0x7A8. The absolute gain measurement is made every 250 MHz, between 0.1 MHz and 8000 MHz.

6.4.1 Applying Calibration

The gain and attenuator values are relative measurements from the absolute reference values made over various RF frequencies. These relative values are either subtracted (attenuation) or added (gain) to the absolute gain value to determine the gain of the relative configuration. For example, assume the RF_ATTEN is at 5 dB, IF3_ATTEN is at 2 dB, IF frequency set to 1950 MHz, and the RF frequency is set to 13.1 GHz. A possible systematic approach would be:

1. Compute the absolute gain value by interpolation because there is no measurement value at 13.1 GHz. A simple linear interpolation between the measured 13.0 GHz and 13.25 GHz values would provide a good estimate. A local spline interpolation over 5-6 surrounding points would provide better accuracy. Let us call this value $G(rf)$.

2. Compute the relative RF attenuator value by interpolation because there is no measurement value at 13.1 GHz. A simple linear interpolation between the measured 13.0 GHz and 13.25 GHz values would provide a good estimate. A local spline interpolation over 5-6 surrounding points would provide better accuracy. Let us call this value $A(5dB, rf)_{rf\_atten}$.

3. The IF Attenuator value is read directly from memory as $A(2dB)_{if\_atten}$, however it must be corrected for frequency offset from 1250 MHz. There are offset gain response values measured at 1900 MHz and 2000 MHz, so a simple linear interpolation between 2 points should be sufficient. Call this correct gain value $G(if)$.

4. The gain for this configuration is calculated using:

   $$G = G(rf)_{abs} + G(if) - A(5dB, rf)_{rf\_atten} - A(2dB)_{if\_atten}$$

   If the current device temperature is different from the calibration temperature, the gain correction due to temperature difference is computed using

   $$\Delta G_{temp} = C(b1)_1(T - T_0) + C(b1)_2(T - T_0)^2 .$$

   ΔG_{temp} is the gain correction, $C(b1)_1$ and $C(b1)_2$ are the first and second order temperature gain coefficients respectively for RF band 1, and T and T_0 are the current temperature and calibration temperature respectively. Adding this correction to the previously calculated gain will compensate the value for temperature deviation.

   $$G = G(rf)_{abs} + G(if) - A(5dB, rf)_{rf_{atten}} - A(2dB)_{if_{atten}} + \Delta G_{temp}$$

Revision Table

Revision	Revision Date	Description
0.1	6/19/18	Document Created
1.0	9/12/18	Initial Release
1.1	9/24/18	Edited for clarity
1.2	11/22/19	Update to table 3, row 3. Changed 19 to 18.
1.3	3/2/20	Updated 6.4 – Gain measurement
1.4	11/24/20	1. Updated table 3 2. Corrected pin references for SPI in Communications section
1.5	6/4/21	Updated SPI settings
1.6	12/7/22	Corrected torque range
1.7	6/4/24	Corrected register 0x15
1.8	9/9/24	Correct baud rate pin number on table 9
1.9	2/3/24	Updated with register AUTO_CALC_GAIN, 0x1D

SignalCore™
PRESERVING SIGNAL INTEGRITY