ANALOG DEVICES

Dual, 16-Bit, 1000 MSPS, TxDAC+ Digital-to-Analog Converter
AD9125

FEATURES

Flexible CMOS interface allows dual-word, word, or byte load
Single-carrier W-CDMA ACLR = 80 dBc at 122.88 MHz IF
Analog output: adjustable 8.7 mA to 31.7 mA, R_L = 25 Ω to 50 Ω
Novel 2×/4×/8× interpolator/complex modulator allows carrier placement anywhere in the DAC bandwidth
Gain and phase adjustment for sideband suppression
Multichip synchronization interface
High performance, low noise PLL clock multiplier
Digital inverse sinc filter
Low power: 900 mW at 500 MSPS, full operating conditions
72-lead, exposed paddle LFCSP

APPLICATIONS

Wireless infrastructure
W-CDMA, CDMA2000, TD-SCDMA, WiMAX, GSM, LTE
Digital high or low IF synthesis
Transmit diversity
Wideband communications: LMDS/MMDS, point-to-point
Cable modem termination systems

GENERAL DESCRIPTION

The AD9125 is a dual, 16-bit, high dynamic range TxDAC+® digital-to-analog converter (DAC) that provides a sample rate of 1000 MSPS, permitting a multicarrier generation up to the Nyquist frequency. It includes features optimized for direct conversion transmit applications, including complex digital modulation, and gain and offset compensation. The DAC outputs are optimized to interface seamlessly with analog quadrature modulators, such as the ADL537x F-MOD series from Analog Devices, Inc. A 4-wire serial port interface allows programming/readback of many internal parameters. Full-scale output current can be programmed over a range of 8.7 mA to 31.7 mA. The AD9125 comes in a 72-lead LFCSP.

PRODUCT HIGHLIGHTS

1. Ultralow noise and intermodulation distortion (IMD) enable high quality synthesis of wideband signals from baseband to high intermediate frequencies.
2. A proprietary DAC output switching technique enhances dynamic performance.
3. The current outputs are easily configured for various single-ended or differential circuit topologies.
4. The flexible CMOS digital interface allows the standard 32-wire bus to be reduced to a 16-wire bus.
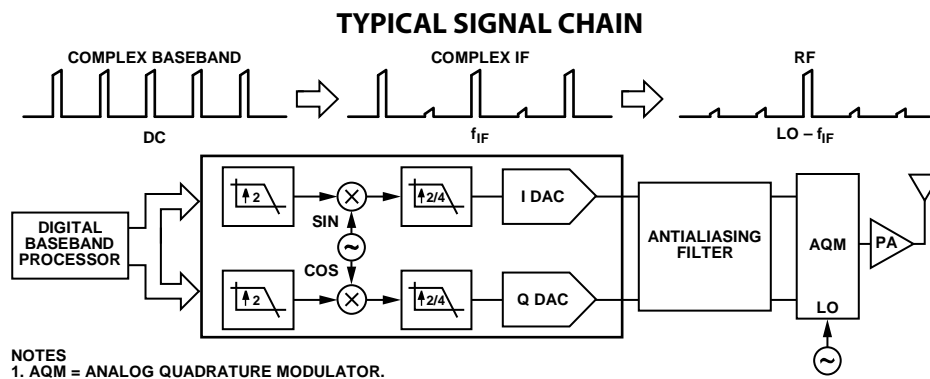
TYPICAL SIGNAL CHAIN

Figure 1.

AD9125

TABLE OF CONTENTS

REVISION HISTORY

6/10—Revision 0: Initial Version

FUNCTIONAL BLOCK DIAGRAM
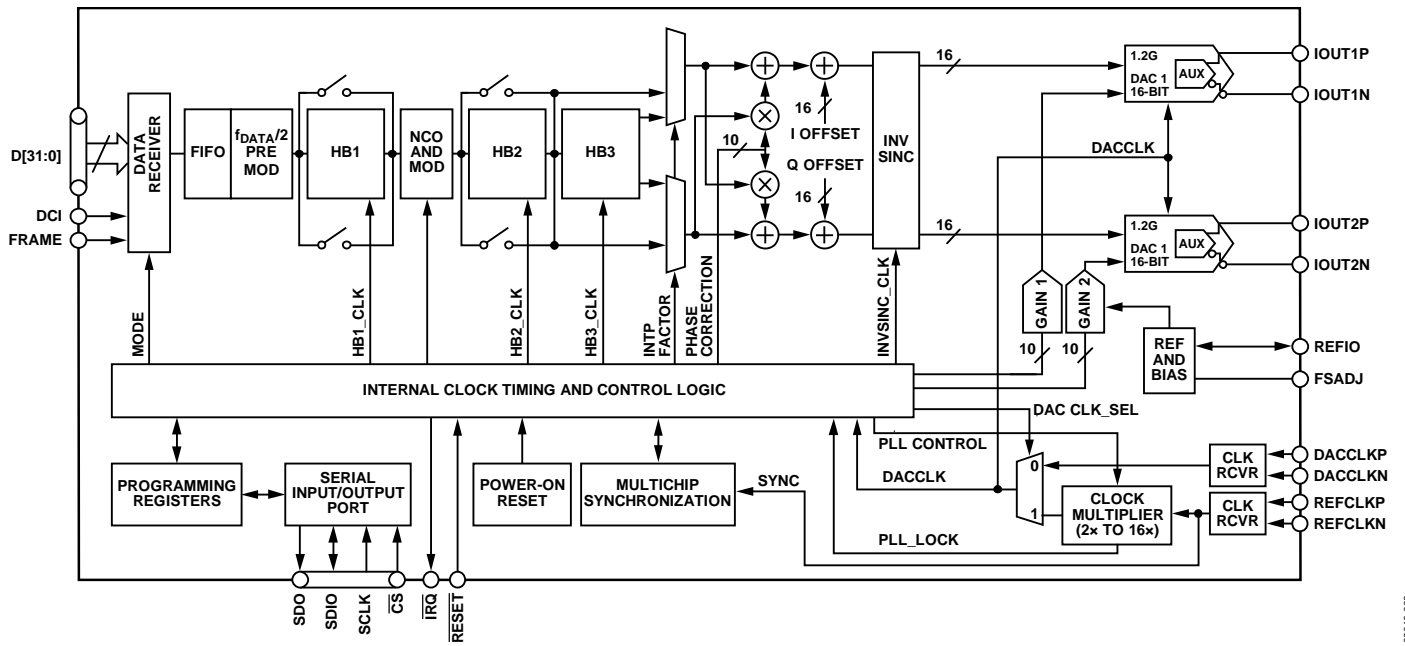
Figure 2. AD9125 Functional Block Diagram

AD9125

SPECIFICATIONS

DC SPECIFICATIONS

T$_{MIN}$ to T$_{MAX}$, AVDD33 = 3.3 V, DVDD18 = 1.8 V, CVDD18 = 1.8 V, I$_{OUTFS}$ = 20 mA, maximum sample rate, unless otherwise noted.

Table 1.

Parameter	Min	Typ	Max	Unit
RESOLUTION		16		Bits
ACCURACY				
Differential Nonlinearity (DNL)		±2.1		LSB
Integral Nonlinearity (INL)		±3.7		LSB
MAIN DAC OUTPUTS				
Offset Error	−0.001	0	+0.001	% FSR
Gain Error (with Internal Reference)	−3.6	±2	+3.6	% FSR
Full-Scale Output Current [1]	8.66	19.6	31.66	mA
Output Compliance Range	−1.0		+1.0	V
Output Resistance		10		MΩ
Gain DAC Monotonicity		Guaranteed		
Settling Time to Within ±0.5 LSB		20		ns
MAIN DAC TEMPERATURE DRIFT				
Offset		0.04		ppm/°C
Gain		100		ppm/°C
Reference Voltage		30		ppm/°C
REFERENCE				
Internal Reference Voltage		1.2		V
Output Resistance		5		kΩ
ANALOG SUPPLY VOLTAGES				
AVDD33	3.13	3.3	3.47	V
CVDD18	1.71	1.8	1.89	V
DIGITAL SUPPLY VOLTAGES				
DVDD18	1.71	1.8	1.89	V
IOVDD	1.71	1.8/3.3	3.47	V
POWER CONSUMPTION				
2× Mode, f$_{DAC}$ = 491.52 MSPS, IF = 10 MHz, PLL Off		834		mW
2× Mode, f$_{DAC}$ = 491.52 MSPS, IF = 10 MHz, PLL On		913		mW
8× Mode, f$_{DAC}$ = 800 MSPS, IF = 10 MHz, PLL Off		1114	1227	mW
AVDD33		55	58	mA
CVDD18		78	85	mA
DVDD18		440	490	mA
Power-Down Mode		1.5	2.7	mW
Power Supply Rejection Ratio, AVDD33	−0.3		+0.3	% FSR/V
OPERATING RANGE	−40	+25	+85	°C

[1] Based on a 10 kΩ external resistor.

DIGITAL SPECIFICATIONS

T_{MIN} to T_{MAX}, AVDD33 = 3.3 V, IOVDD = 3.3 V, DVDD18 = 1.8 V, CVDD18 = 1.8 V, I_{OUTFS} = 20 mA, maximum sample rate, unless otherwise noted.

Table 2.

Parameter	Conditions	Min	Typ	Max	Unit
CMOS DATA INPUTS					
Input V_{IN} Logic High		1.2			V
Input V_{IN} Logic Low				0.6	V
Maximum Bus Speed		250			MHz
SERIAL PORT OUTPUT LOGIC LEVELS					
Output V_{OUT} Logic High	IOVDD = 1.8 V	1.4			V
	IOVDD = 2.5 V	1.8			V
	IOVDD = 3.3 V	2.0			V
Output V_{OUT} Logic Low	IOVDD = 1.8 V			0.4	
	IOVDD = 2.5 V			0.4	V
	IOVDD = 3.3 V			0.4	V
SERIAL PORT INPUT LOGIC LEVELS					
Input V_{IN} Logic High	IOVDD = 1.8 V	1.2			V
	IOVDD = 2.5 V	1.6			V
	IOVDD = 3.3 V	2.4			V
Input V_{IN} Logic Low	IOVDD = 1.8 V			0.6	V
	IOVDD = 2.5 V			0.8	V
	IOVDD = 3.3V			0.8	V
DACCLK INPUT (DACCLKP, DACCLKN)					
Differential Peak-to-Peak Voltage		100	500	2000	mV
Common-Mode Voltage	Self biased input, ac couple		1.25		V
Maximum Clock Rate		1000			MHz
REFCLK INPUT (REFCLKP, REFCLKN)					
Differential Peak-to-Peak Voltage		100	500	2000	mV
Common-Mode Voltage			1.25		V
REFCLKx Frequency, PLL Mode	1 GHz ≤ f_{VCO} ≤ 2.1 GHz	15.625		600	MHz
REFCLKx Frequency, SYNC Mode	See the Multichip Synchronization section for conditions	0		600	MHz
SERIAL PERIPHERAL INTERFACE					
Maximum Clock Rate (SCLK)		40			MHz
Minimum Pulse Width High (t_{PWH})				12.5	ns
Minimum Pulse Width Low (t_{PWOL})				12.5	ns
Setup Time, SDI to SCLK (t_{DS})		1.9			ns
Hold Time, SDI to SCLK (t_{DH})		0.2			ns
Data Valid, SDO to SCLK (t_{DV})		2.3			ns
Setup Time, $\overline{CS}$ to SCLK ($t_{D\overline{CS}}$)			1.4		ns

LATENCY AND POWER-UP TIMING SPECIFICATIONS

Table 3.

Parameter	Min	Typ	Max	Unit
LATENCY (DACCLK Cycles)				
1× Interpolation (with or Without Modulation)		64		Cycles
2× Interpolation (with or Without Modulation)		135		Cycles
4× Interpolation (with or Without Modulation)		292		Cycles
8× Interpolation (with or Without Modulation)		608		Cycles
Inverse Sinc		20		Cycles
Fine Modulation		8		Cycles
Power-Up Time		260		ms

AD9125

AC SPECIFICATIONS

T_{MIN} to T_{MAX}, AVDD33 = 3.3 V, DVDD18 = 1.8 V, CVDD18 = 1.8 V, I_{OUTFS} = 20 mA, maximum sample rate, unless otherwise noted.

Table 4.

Parameter	Min	Typ	Max	Unit
SPURIOUS-FREE DYNAMIC RANGE (SFDR)				
f_{DAC} = 100 MSPS, f_{OUT} = 20 MHz		78		dBc
f_{DAC} = 200 MSPS, f_{OUT} = 50 MHz		80		dBc
f_{DAC} = 400 MSPS, f_{OUT} = 70 MHz		69		dBc
f_{DAC} = 800 MSPS, f_{OUT} = 70 MHz		72		dBc
TWO-TONE INTERMODULATION DISTORTION (IMD)				
f_{DAC} = 200 MSPS, f_{OUT} = 50 MHz		84		dBc
f_{DAC} = 400 MSPS, f_{OUT} = 60 MHz		86		dBc
f_{DAC} = 400 MSPS, f_{OUT} = 80 MHz		84		dBc
f_{DAC} = 800 MSPS, f_{OUT} = 100 MHz		81		dBc
NOISE SPECTRAL DENSITY (NSD) EIGHT-TONE, 500 kHz TONE SPACING				
f_{DAC} = 200 MSPS, f_{OUT} = 80 MHz		−162		dBm/Hz
f_{DAC} = 400 MSPS, f_{OUT} = 80 MHz		−163		dBm/Hz
f_{DAC} = 800 MSPS, f_{OUT} = 80 MHz		−164		dBm/Hz
W-CDMA ADJACENT CHANNEL LEAKAGE RATIO (ACLR), SINGLE CARRIER				
f_{DAC} = 491.52 MSPS, f_{OUT} = 10 MHz		82		dBc
f_{DAC} = 491.52 MSPS, f_{OUT} = 122.88 MHz		80		dBc
f_{DAC} = 983.04 MSPS, f_{OUT} = 122.88 MHz		81		dBc
W-CDMA SECOND ACLR, SINGLE CARRIER				
f_{DAC} = 491.52 MSPS, f_{OUT} = 10 MHz		88		dBc
f_{DAC} = 491.52 MSPS, f_{OUT} = 122.88 MHz		86		dBc
f_{DAC} = 983.04 MSPS, f_{OUT} = 122.88 MHz		88		dBc

Table 5. Interface Speeds

Mode	Interpolation	f_{BUS}	f_{DATA}	f_{DAC}
Byte Mode	1×	250	62.5	62.5
	2× (HB1)	250	62.5	125
	2× (HB2)	250	62.5	125
	4×	250	62.5	250
	8×	250	62.5	500
Word Mode	1×	250	125	125
	2× (HB1)	250	125	250
	2× (HB2)	250	125	250
	4×	250	125	500
	8×	250	125	1000
Dual-Word Mode	1×	250	250	250
	2× (HB1)	250	250	500
	2× (HB2)	250	250	500
	4×	250	250	1000
	8×	125	125	1000

ABSOLUTE MAXIMUM RATINGS

Table 6.

Parameter	With Respect To	Rating
AVDD33	AVSS, EPAD, CVSS, DVSS	−0.3 V to +3.6 V
IOVDD	AVSS, EPAD, CVSS, DVSS	−0.3 V to +3.6 V
DVDD18, CVDD18	AVSS, EPAD, CVSS, DVSS	−0.3 V to +2.1 V
AVSS	EPAD, CVSS, DVSS	−0.3 V to +0.3 V
EPAD	AVSS, CVSS, DVSS	−0.3 V to +0.3 V
CVSS	AVSS, EPAD, DVSS	−0.3 V to +0.3 V
DVSS	AVSS, EPAD, CVSS	−0.3 V to +0.3 V
FSADJ, REFIO, IOUT1P/IOUT1N, IOUT2P/IOUT2N	AVSS	−0.3 V to AVDD33 + 0.3 V
D[31:0], FRAME, DCI	EPAD, DVSS	−0.3 V to DVDD18 + 0.3 V
DACCLKP/DACCLKN, REFCLKP/REFCLKN	DVSS	−0.3 V to CVDD18 + 0.3 V
$\overline{RESET}$, $\overline{IRQ}$, $\overline{CS}$, SCLK, SDIO, SDO	EPAD, DVSS	−0.3 V to IOVDD + 0.3 V
Junction Temperature		125°C
Storage Temperature Range		−65°C to +150°C

Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

THERMAL RESISTANCE

The exposed paddle (EPAD) must be soldered to the ground plane for the 72-lead LFCSP. The EPAD performs as an electrical and thermal connection to the board.

Typical θ_{JA}, θ_{JB}, and θ_{JC} values are specified for a 4-layer board in still air. Airflow increases heat dissipation, effectively reducing θ_{JA} and θ_{JB}.

Table 7. Thermal Resistance

Package	θ_{JA}	θ_{JB}	θ_{JC}	Unit	Conditions
72-Lead LFCSP	20.7	10.9	1.1	°C/W	EPAD soldered

ESD CAUTION

ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

PIN CONFIGURATION AND FUNCTION DESCRIPTIONS

Figure 3. Pin Configuration
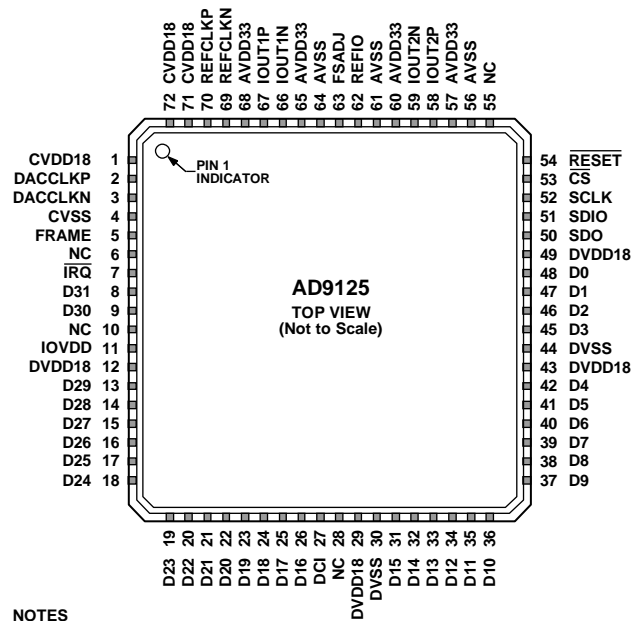
NOTES
1. NC = NO CONNECT.
2. EXPOSED PAD MUST BE CONNECTED TO AVSS.

Table 8. Pin Function Descriptions

Pin No.	Mnemonic	Description
1	CVDD18	1.8 V Clock Supply. Supplies clock receivers, clock distribution, and PLL circuitry.
2	DACCLKP	DAC Clock Input, Positive.
3	DACCLKN	DAC Clock Input, Negative.
4	CVSS	Clock Supply Common.
5	FRAME	Frame Input.
6	NC	No Connect
7	$\overline{IRQ}$ ($\overline{INT}$)	Interrupt Request. Open Drain, Active Low Output. Connect external pull-up to IOVDD.
8	D31	Data Bit 31.
9	D30	Data Bit 30.
10	NC	No Connect.
11	IOVDD	Supply for Serial Port Pin, $\overline{RESET}$ Pin, and $\overline{IRQ}$ Pin. 1.8 V to 3.3 V can be applied to this pin.
12	DVDD18	1.8 V Digital Supply. Supplies power to digital core and digital data ports.
13	D29	Data Bit 29.
14	D28	Data Bit 28.
15	D27	Data Bit 27.
16	D26	Data Bit 26.
17	D25	Data Bit 25.
18	D24	Data Bit 24.
19	D23	Data Bit 23.
20	D22	Data Bit 22.
21	D21	Data Bit 21.
22	D20	Data Bit 20.
23	D19	Data Bit 19.
24	D18	Data Bit 18.
25	D17	Data Bit 17.
26	D16	Data Bit 16.
27	DCI	Data Clock Input.

| Pin No.	Mnemonic	Description
28	NC	No Connect.
29	DVDD18	1.8 V Digital Supply.
30	DVSS	Digital Common.
31	D15	Data Bit 15.
32	D14	Data Bit 14.
33	D13	Data Bit 13.
34	D12	Data Bit 12.
35	D11	Data Bit 11.
36	D10	Data Bit 10.
37	D9	Data Bit 9.
38	D8	Data Bit 8.
39	D7	Data Bit 7.
40	D6	Data Bit 6.
41	D5	Data Bit 5.
42	D4	Data Bit 4.
43	DVDD18	1.8 V Digital Supply.
44	DVSS	Digital Supply Common.
45	D3	Data Bit 3.
46	D2	Data Bit 2.
47	D1	Data Bit 1.
48	D0	Data Bit 0.
49	DVDD18	1.8 V Digital Supply.
50	SDO	Serial Port Data Output (CMOS levels with respect to IOVDD).
51	SDIO	Serial Port Data Input/Output (CMOS levels with respect to IOVDD).
52	SCLK	Serial Port Clock Input (CMOS levels with respect to IOVDD).
53	$\overline{CS}$	Serial Port Chip Select. Active Low (CMOS levels with respect to IOVDD).
54	$\overline{RESET}$	Reset. Active Low (CMOS levels with respect to IOVDD).
55	NC	No Connect.
56	AVSS	Analog Supply Common.
57	AVDD33	3.3 V Analog Supply.
58	IOUT2P	Q DAC Positive Current Output.
59	IOUT2N	Q DAC Negative Current Output.
60	AVDD33	3.3 V Analog Supply.
61	AVSS	Analog Supply Common.
62	REFIO	Voltage Reference. Nominally 1.2 V output. Should be decoupled to analog common.
63	FSADJ	Full-Scale Current Output Adjust. Place a 10 kΩ resistor on the analog common.
64	AVSS	Analog Common.
65	AVDD33	3.3 V Analog Supply.
66	IOUT1N	I DAC Negative Current Output.
67	IOUT1P	I DAC Positive Current Output.
68	AVDD33	3.3 V Analog Supply.
69	REFCLKN	PLL Reference Clock Input, Negative. This pin has a secondary function as the SYNC input.
70	REFCLKP	PLL Reference Clock Input, Positive. This pin has a secondary function as the SYNC input.
71	CVDD18	1.8 V Clock Supply. Supplies clock receivers, clock distribution, and PLL circuitry.
72	CVDD18	1.8 V Clock Supply. Supplies clock receivers, clock distribution, and PLL circuitry.
	EPAD	Exposed pad must be connected to AVSS. This provides an electrical, thermal, and mechanical connection to the PCB.

AD9125

TYPICAL PERFORMANCE CHARACTERISTICS
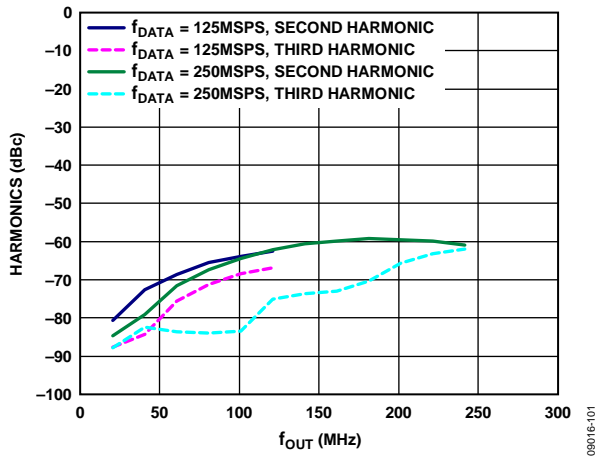
Figure 4. Harmonics vs. f_{OUT} over f_{DATA}, 2× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

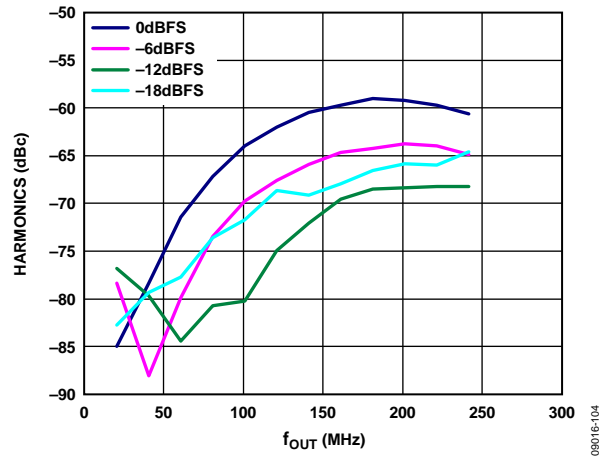Figure 5. Harmonics vs. f_{OUT} over f_{DATA}, 4× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

Figure 6. Harmonics vs. f_{OUT} over f_{DATA}, 8× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

Figure 7. Second Harmonic vs. f_{OUT} over Digital Scale, 2× Interpolation, f_{DATA} = 250 MSPS, f_{SC} = 20 mA

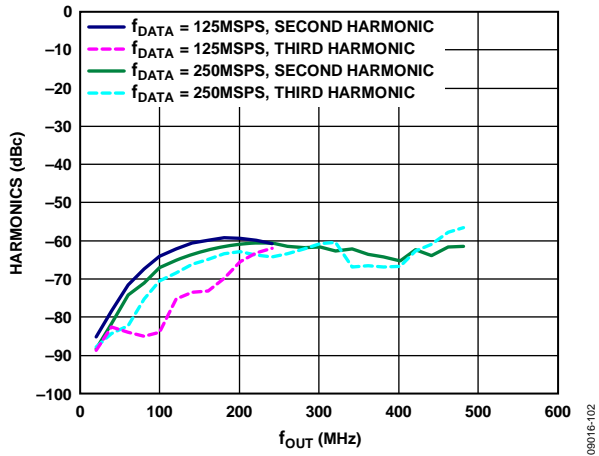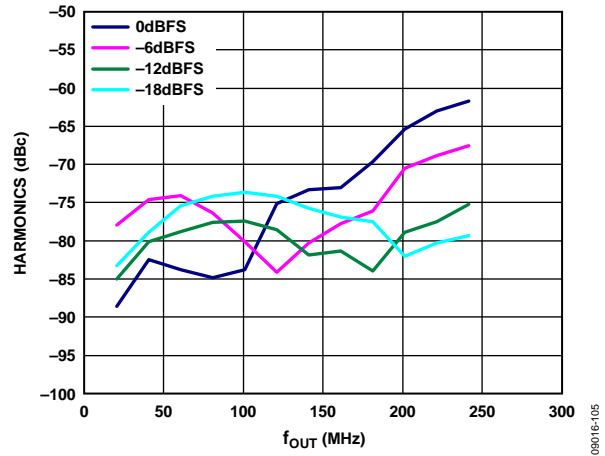Figure 8. Third Harmonic vs. f_{OUT} over Digital Scale, 2× Interpolation, f_{DATA} = 250 MSPS, f_{SC} = 20 mA

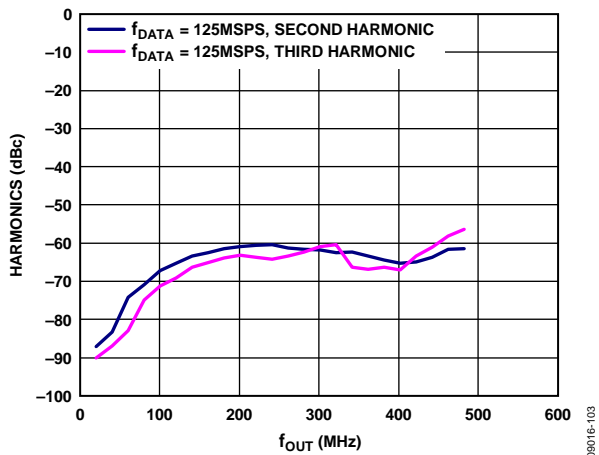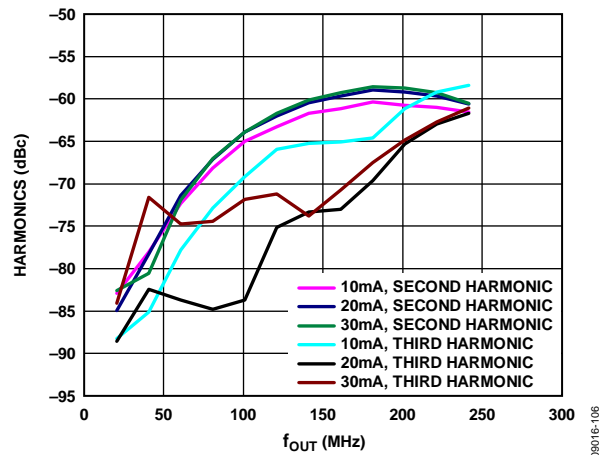Figure 9. Harmonics vs. f_{OUT} over f_{SC}, 2× Interpolation, f_{DATA} = 250 MSPS, Digital Scale = 0 dBFS

Figure 10. Highest Digital Spur vs. f_{OUT} over f_{DATA}, 2× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA
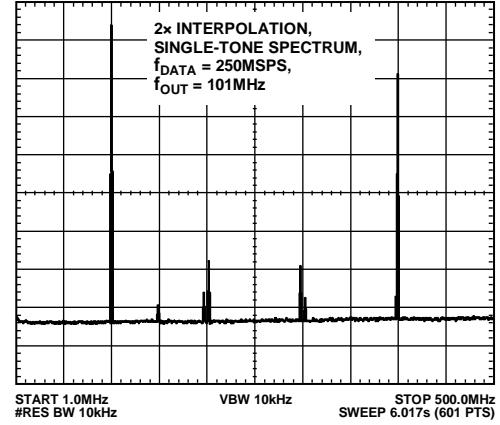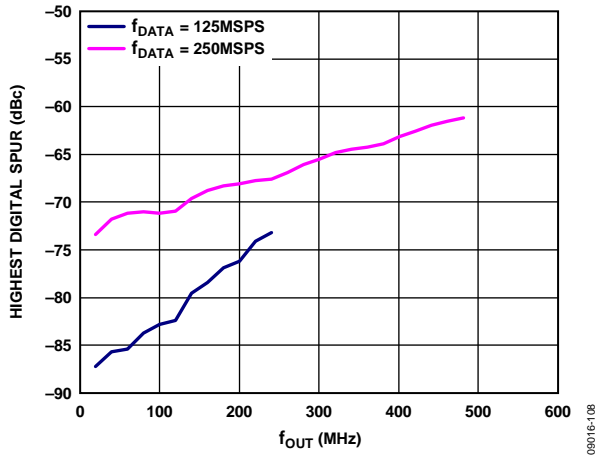
Figure 11. Highest Digital Spur vs. f_{OUT} over f_{DATA}, 4× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA
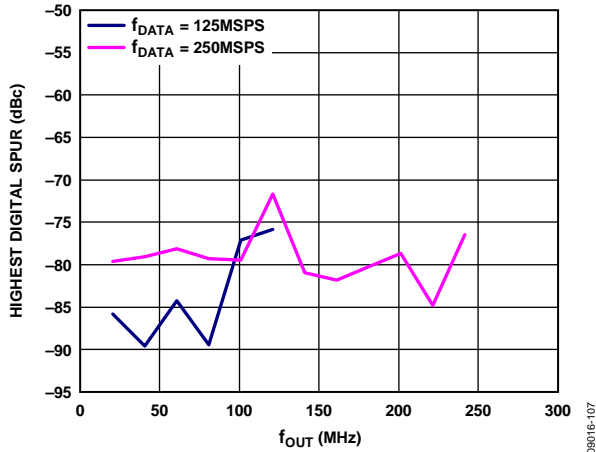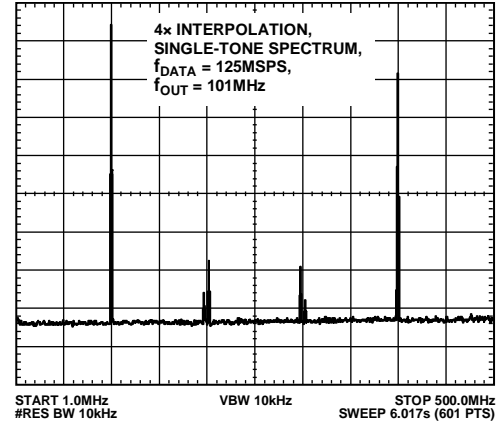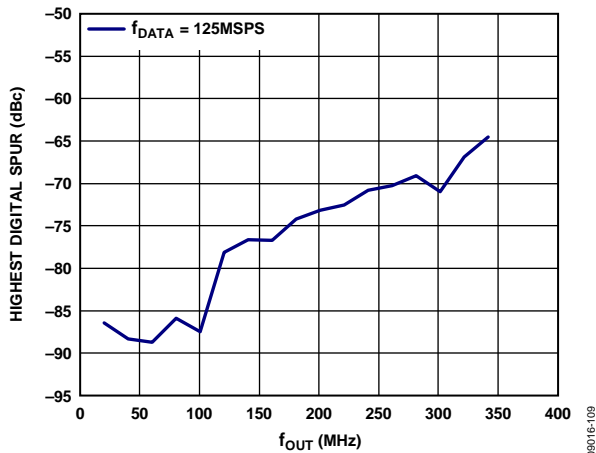

Figure 12. Highest Digital Spur vs. f_{OUT} over f_{DATA}, 8× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA

Figure 13. 2× Interpolation, Single-Tone Spectrum

Figure 14. 4× Interpolation, Single-Tone Spectrum

Figure 15. 8× Interpolation, Single-Tone Spectrum

Rev. 0 | Page 11 of 56
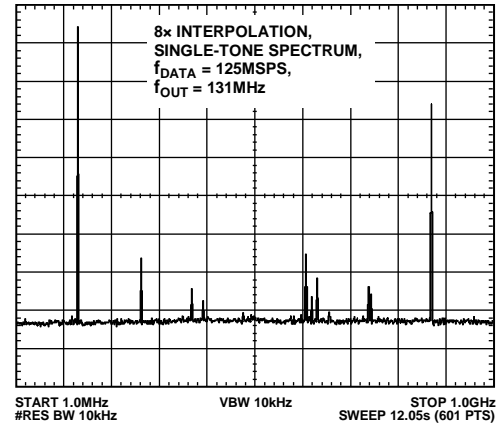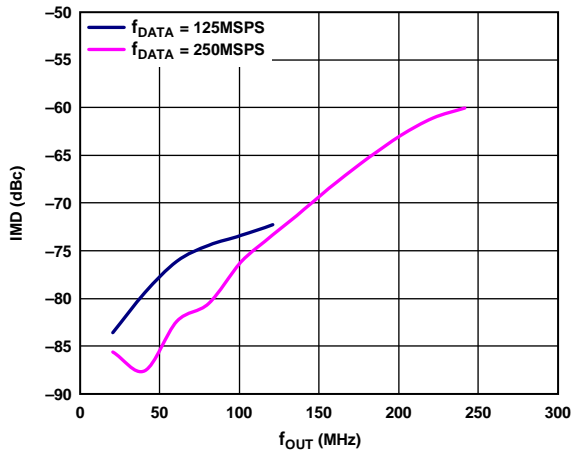
Figure 16. IMD vs. f_{OUT} over f_{DATA}, 2× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA
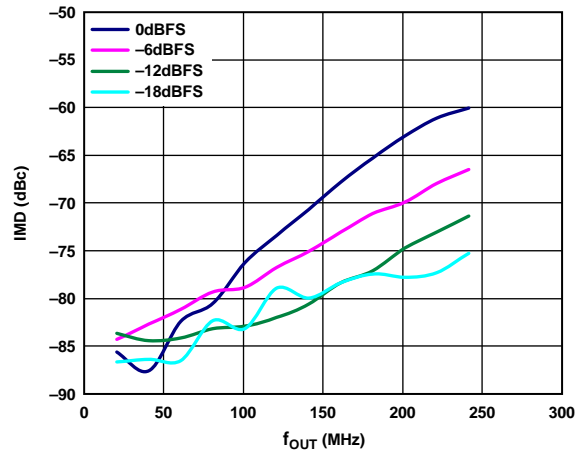
Figure 19. IMD vs. f_{OUT} over Digital Scale, 2× Interpolation, f_{DATA} = 250 MSPS, f_{SC} = 20 mA
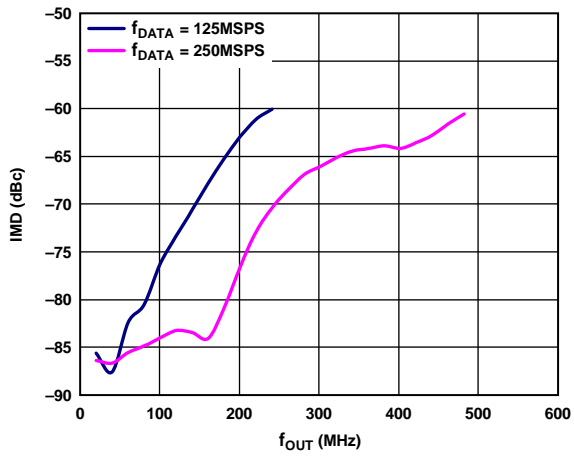
Figure 17. IMD vs. f_{OUT} over f_{DATA}, 4× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA
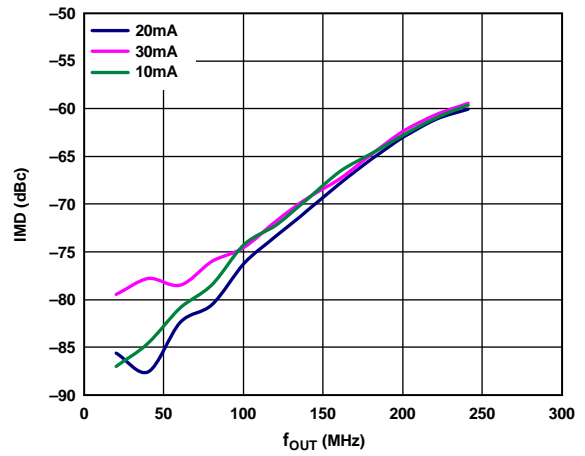
Figure 20. IMD vs. f_{OUT} over f_{SC}, 2× Interpolation, f_{DATA} = 250 MSPS, Digital Scale = 0 dBFS
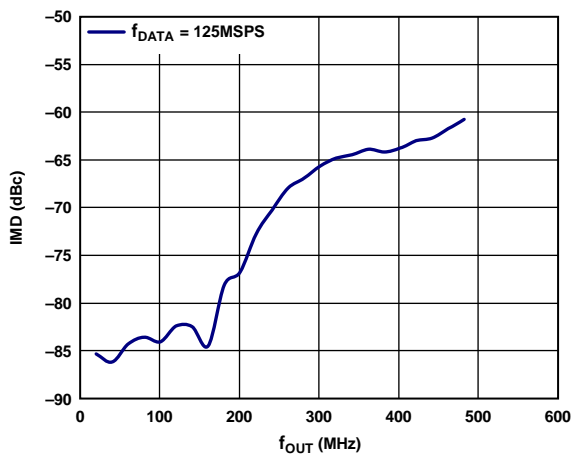
Figure 18. IMD vs. f_{OUT} over f_{DATA}, 8× Interpolation, Digital Scale = 0 dBFS, f_{SC} = 20 mA
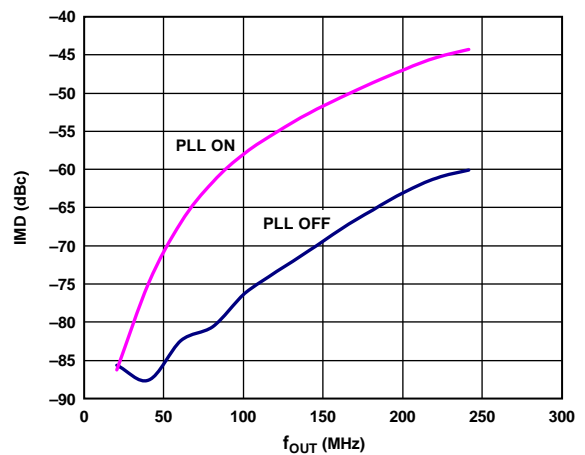
Figure 21. IMD vs. f_{OUT}, PLL On vs. PLL Off

Rev. 0 | Page 12 of 56
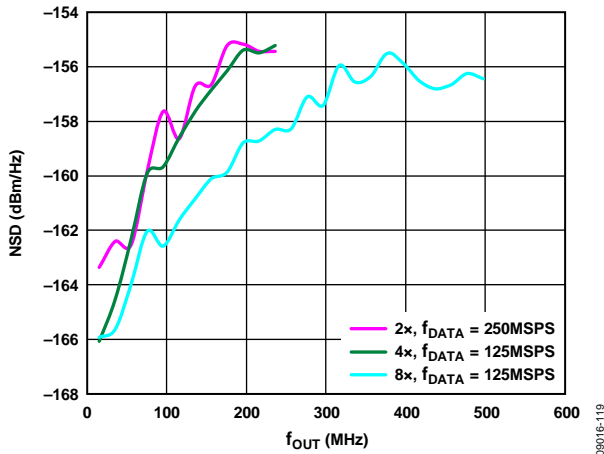
Figure 22. One-Tone NSD vs. f_{OUT} over Interpolation Rate and f_{DATA}, Digital Scale = 0 dBFS, f_{SC} = 20 mA, PLL Off
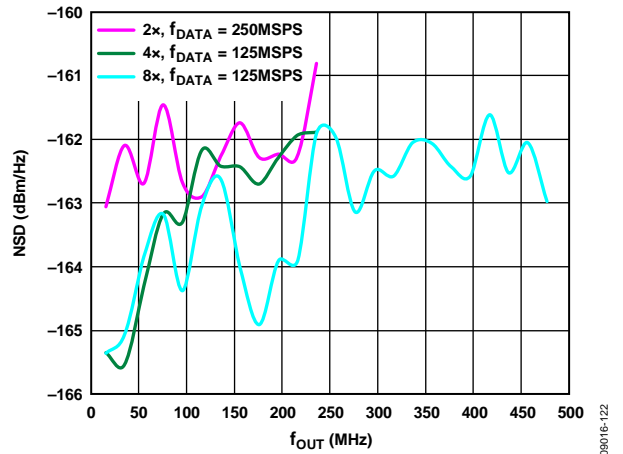
Figure 25. Eight-Tone NSD vs. f_{OUT} over Interpolation Rate and f_{DATA}, Digital Scale = 0 dBFS, f_{SC} = 20 mA, PLL Off
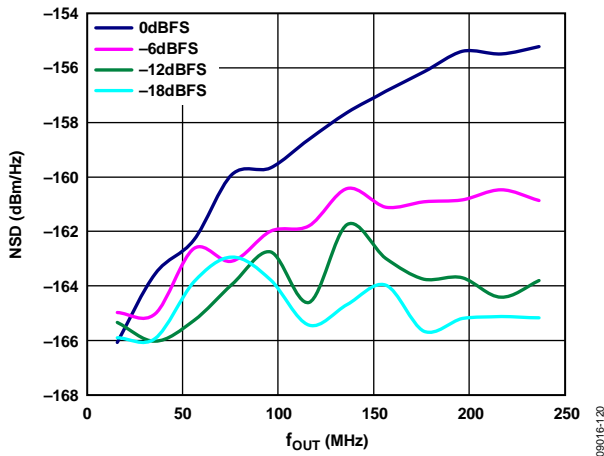
Figure 23. One-Tone NSD vs. f_{OUT} over Digital Scale, f_{DATA} = 200 MSPS, 4× Interpolation, f_{SC} = 20 mA, PLL Off
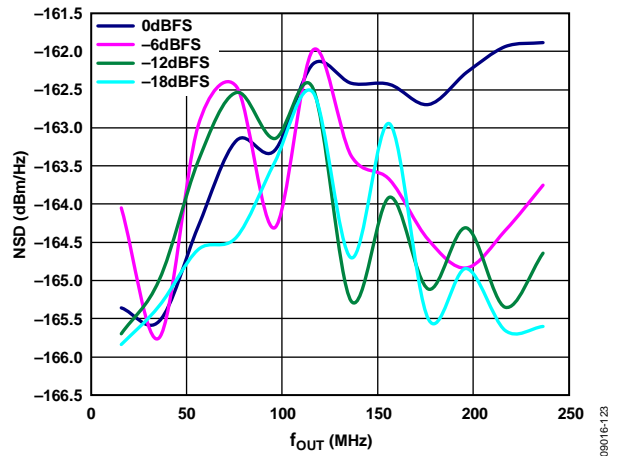
Figure 26. Eight-Tone NSD vs. f_{OUT} over Digital Scale, f_{DATA} = 200 MSPS, 4× Interpolation, f_{SC} = 20 mA, PLL Off
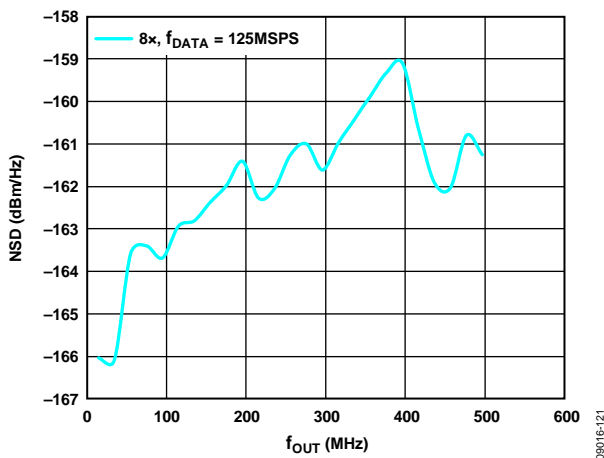
Figure 24. One-Tone NSD vs. f_{OUT} over Interpolation Rate and f_{DATA}, Digital Scale = 0 dBFS, f_{SC} = 20 mA, PLL On
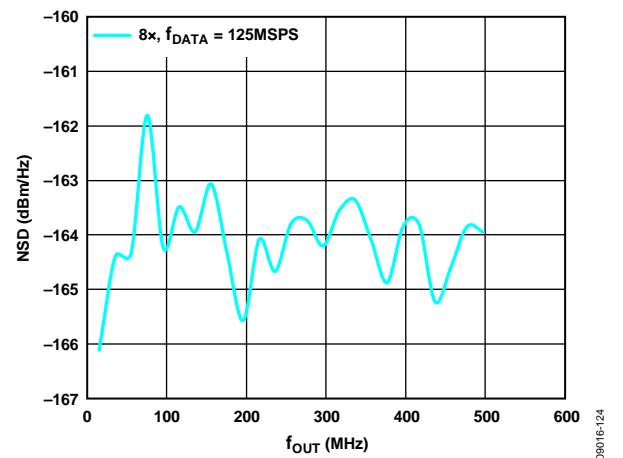
Figure 27. Eight-Tone NSD vs. f_{OUT} over Interpolation Rate and f_{DATA}, Digital Scale = 0 dBFS, f_{SC} = 20 mA, PLL On
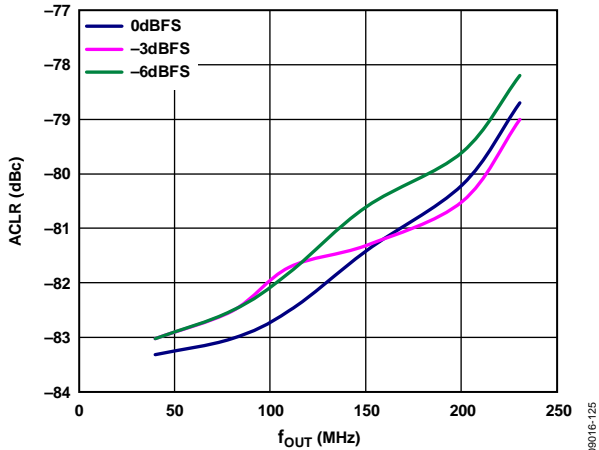
Figure 28. One-Carrier W-CDMA ACLR vs. f_{OUT} over Digital Cutback, Adjacent Channel, PLL Off

Figure 29. One-Carrier W-CDMA ACLR vs. f_{OUT} over f_{DAC}, Alternate Channel, PLL Off

Figure 30. One-Carrier W-CDMA ACLR vs. f_{OUT} over f_{DAC}, Second Alternate Channel, PLL Off

Figure 31. One-Carrier W-CDMA ACLR vs. f_{OUT} over Interpolation Rate, Adjacent Channel, PLL On vs. PLL Off

Figure 32. One-Carrier W-CDMA ACLR vs. f_{OUT} over Interpolation Rate, Alternate Channel, PLL On vs. PLL Off
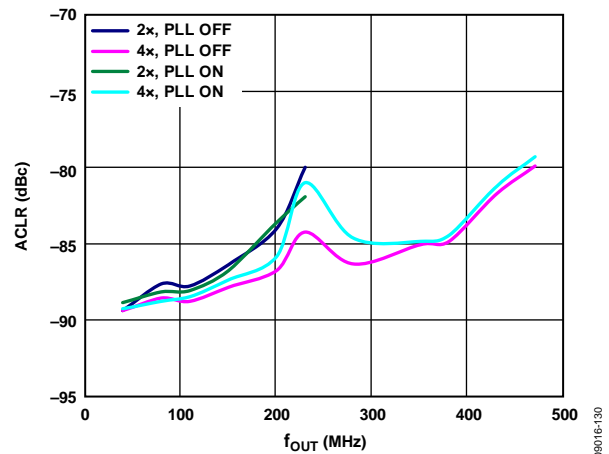
Figure 33. One-Carrier W-CDMA ACLR vs. f_{OUT} over Interpolation Rate, Second Alternate Channel, PLL On vs. PLL Off
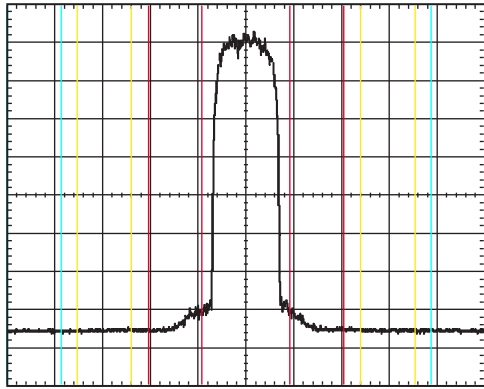
START 133.06MHz   **VBW 30kHz**   **STOP 166.94MHz**
#RES BW 30kHz   **SWEEP 143.6ms (601 PTS)**

RMS RESULTS	FREQ OFFSET	REF BW	LOWER dBc	LOWER dBm	UPPER dBc	UPPER dBm
CARRIER POWER	5.00MHz	3.840MHz	−75.96	−85.96	−77.13	−87.13
−10.00dBm/	10.00MHz	3.840MHz	−85.33	−95.33	−85.24	−95.25
3.840MHz	15.00MHz	2.888MHz	−95.81	−95.81	−85.43	−95.43

Figure 34. Four-Carrier W-CDMA ACLR Performance, IF ≈150 MHz
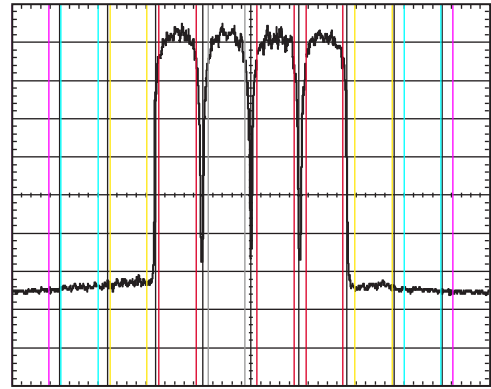
START 125.88MHz   **VBW 30kHz**   **STOP 174.42MHz**
#RES BW 30kHz   **SWEEP 206.9ms (601 PTS)**

TOTAL CARRIER POWER: −11.19dBm/15.3600MHz
RRC FILTER: OFF FILTER ALPHA 0.22
REF CARRIER POWER: −16.89dBm/3.84000MHz

			LOWER		UPPER	
		OFFSET FREQ	INTEG BW	dBc	dBm	dBc	dBm
1	−16.92dBm	5.000MHz	3.840MHz	−65.88	−82.76	−67.52	−84.40
2	−16.89dBm	10.00MHz	3.840MHz	−68.17	−85.05	−69.91	−86.79
3	−17.43dBm	15.00MHz	3.840MHz	−70.42	−87.31	−71.40	−88.28
4	−17.64dBm						

Figure 35. One-Carrier W-CDMA ACLR Performance, IF ≈150 MHz

TERMINOLOGY

Integral Nonlinearity (INL)
INL is defined as the maximum deviation of the actual analog output from the ideal output, determined by a straight line drawn from zero scale to full scale.

Differential Nonlinearity (DNL)
DNL is the measure of the variation in analog value, normalized to full scale, associated with a 1 LSB change in digital input code.

Offset Error
The deviation of the output current from the ideal of zero is called offset error. For IOUT1P, 0 mA output is expected when the inputs are all 0s. For IOUT1N, 0 mA output is expected when all inputs are set to 1.

Gain Error
The difference between the actual and ideal output span. The actual span is determined by the difference between the outputs when all inputs are set to 1 vs. when all inputs are set to 0.

Output Compliance Range
The range of allowable voltage at the output of a current output DAC. Operation beyond the maximum compliance limits can cause either output stage saturation or breakdown, resulting in nonlinear performance.

Temperature Drift
Temperature drift is specified as the maximum change from the ambient (25°C) value to the value at either T_{MIN} or T_{MAX}. For offset and gain drift, the drift is reported in ppm of full-scale range (FSR) per degree Celsius. For reference drift, the drift is reported in ppm per degree Celsius.

Power Supply Rejection (PSR)
The maximum change in the full-scale output as the supplies are varied from minimum to maximum specified voltages.

Settling Time
The time required for the output to reach and remain within a specified error band around its final value, measured from the start of the output transition.

Spurious-Free Dynamic Range (SFDR)
The difference in decibels between the peak amplitude of the output signal and the peak spurious signal within the dc to the Nyquist frequency of the DAC. Typically, energy in this band is rejected by the interpolation filters. This specification, therefore, defines how well the interpolation filters work and the effect of other parasitic coupling paths to the DAC output.

Signal-to-Noise Ratio (SNR)
SNR is the ratio of the rms value of the measured output signal to the rms sum of all other spectral components below the Nyquist frequency, excluding the first six harmonics and dc. The value for SNR is expressed in decibels.

Interpolation Filter
If the digital inputs to the DAC are sampled at a multiple rate of f_{DATA} (interpolation rate), a digital filter can be constructed that has a sharp transition band near $f_{DATA}/2$. Images that typically appear around f_{DAC} (output data rate) can be greatly suppressed.

Adjacent Channel Leakage Ratio (ACLR)
The ratio in decibels relative to the carrier (dBc) between the measured power within a channel and that of its adjacent channel.

Complex Image Rejection
In a traditional two-part upconversion, two images are created around the second IF frequency. These images have the effect of wasting transmitter power and system bandwidth. By placing the real part of a second complex modulator in series with the first complex modulator, either the upper or lower frequency image near the second IF can be rejected.

Figure 36. Defining Data Rates

THEORY OF OPERATION

The AD9125 combines many features that make it a very attractive DAC for wired and wireless communications systems. The dual digital signal path and dual DAC structure allow an easy interface to common quadrature modulators when designi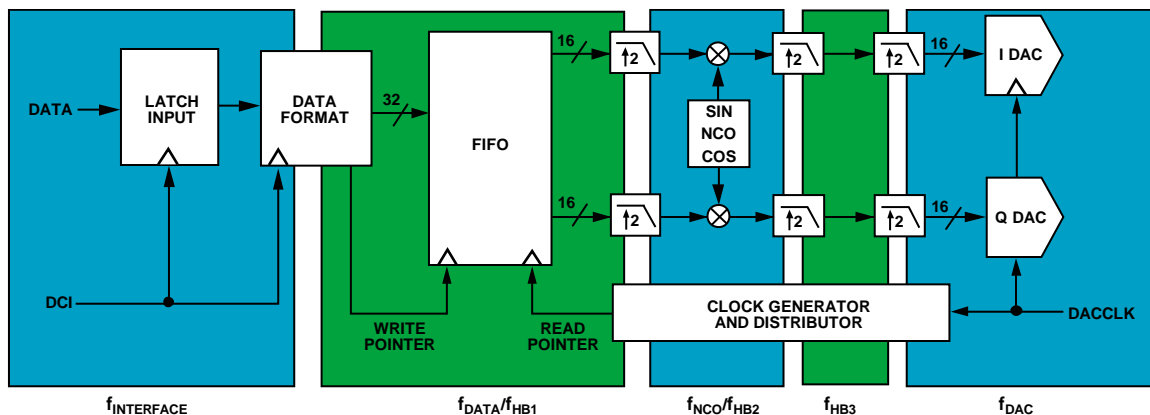ng single sideband transmitters. The speed and performance of the AD9125 allows wider bandwidths and more carriers to be synthesized than in previously available DACs. In addition, these devices include an innovative low power, 32-bit, complex NCO that greatly increases the ease of frequency placement.

The AD9125 offers features that allow simplified synchronization with incoming data and between multiple devices. Auxiliary DACs are also provided on chip for output dc offset compensation (for local oscillator [LO] compensation in single sideband [SSB] transmitters) and for gain matching (for image rejection optimization in SSB transmitters).

SERIAL PORT OPERATION

The serial port is a flexible, synchronous serial communication port, allowing easy interface to many industry-standard micro-controllers and microprocessors. The serial I/O is compatible with most synchronous transfer formats, including both the Motorola SPI® and Intel® SSR protocols. The interface allows read/write access to all registers that configure the AD9125. Single- or multiple-byte transfers are supported, as well as MSB-first or LSB-first transfer formats. The serial interface ports can be configured as a single-pin I/O (SDIO) or two unidirectional pins for input/output (SDIO/SDO).
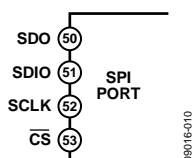
Figure 37. Serial Port Interface Pins

There are two phases of a communication cycle with the AD9125. Phase 1 is the instruction cycle (the writing of an instruction byte into the device), which is coincident with the first eight SCLK rising edges. The instruction byte provides the serial port controller with information regarding the data transfer cycle, which is Phase 2 of the communication cycle. The Phase 1 instruction byte defines whether the upcoming data transfer is a read or write and the starting register address for the first byte of the data transfer. The first eight SCLK rising edges of each communication cycle are used to write the instruction byte into the device.

A logic high on the $\overline{CS}$ pin followed by a logic low resets the serial port timing to the initial state of the instruction cycle. From this state, the next eight rising SCLK edges represent the instruction bits of the current I/O operation.

The remaining SCLK edges are for Phase 2 of the communication cycle. Phase 2 is the actual data transfer between the device and the system controller. Phase 2 of the communication cycle is a transfer of one or more data bytes. Registers change immediately upon writing to the last bit of each transfer byte, except for the frequency tuning word and NCO phase offsets, which only change when the frequency update bit (Register 0x36, Bit 0) is set.

DATA FORMAT

The instruction byte contains the information shown in Table 9.

Table 9. Serial Port Instruction Byte

I7 (MSB)	I6	I5	I4	I3	I2	I1	I0 (LSB)
R/$\overline{W}$	A6	A5	A4	A3	A2	A1	A0

R/$\overline{W}$, Bit 7 of the instruction byte, determines whether a read or write data transfer occurs after the instruction byte write. Logic 1 indicates a read operation, and Logic 0 indicates a write operation.

A6 to A0, Bit 6 to Bit 0 of the instruction byte, determine the register that is accessed during the data transfer portion of the communication cycle. For multibyte transfers, A6 is the starting byte address. The remaining register addresses are generated by the device based on the LSB_FIRST bit (Register 0x00, Bit 6).

SERIAL PORT PIN DESCRIPTIONS

Serial Clock (SCLK)

The serial clock pin synchronizes data to and from the device and runs the internal state machines. The maximum frequency of SCLK is 40 MHz. All data input is registered on the rising edge of SCLK. All data is driven out on the falling edge of SCLK.

Chip Select ($\overline{CS}$)

An active low input starts and gates a communication cycle. It allows more than one device to be used on the same serial communication lines. The SDO and SDIO pins go to a high impedance state when this input is high. During the communication cycle, the $\overline{CS}$ pin should stay low.

Serial Data I/O (SDIO)

Data is always written into the device on this pin. However, this pin can be used as a bidirectional data line. The configuration of this pin is controlled by Register 0x00, Bit 7. The default is Logic 0, configuring the SDIO pin as unidirectional.

Serial Data Out (SDO)

Data is read from this pin for protocols that use separate lines for transmitting and receiving data. In the case where the device operates in a single bidirectional I/O mode, this pin does not output data and is set to a high impedance state.

SERIAL PORT OPTIONS

The serial port can support both MSB-first and LSB-first data formats. This functionality is controlled by the LSB_FIRST bit (Register 0x00, Bit 6). The default is MSB-first (LSB_FIRST = 0).

When LSB_FIRST = 0 (MSB-first), the instruction and data bit must be written from MSB to LSB. Multibyte data transfers in MSB-first format start with an instruction byte that includes the register address of the most significant data byte. Subsequent data bytes should follow from the high address to the low address. In MSB-first mode, the serial port internal byte address generator decrements for each data byte of the multibyte communication cycle.

When LSB_FIRST = 1 (LSB-first), the instruction and data bit must be written from LSB to MSB. Multibyte data transfers in LSB-first format start with an instruction byte that includes the register address of the least significant data byte followed by multiple data bytes. The serial port internal byte address generator increments for each byte of the multibyte communication cycle.

The serial port controller data address decrements from the data address written toward 0x00 for multibyte I/O operations if the MSB-first mode is active. The serial port controller address increments from the data address written toward 0x7F for multibyte I/O operations if the LSB-first mode is active.
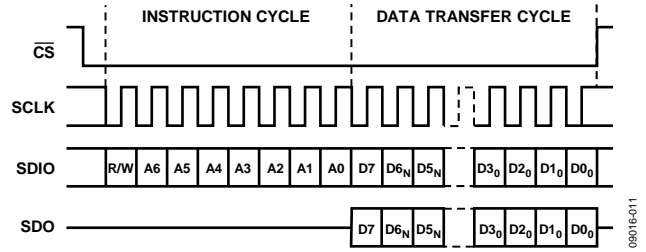
Figure 38. Serial Register Interface Timing, MSB First
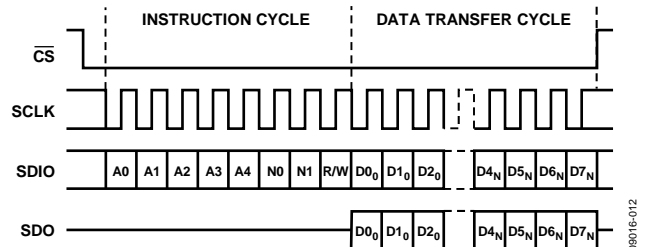
Figure 39. Serial Register Interface Timing, LSB First
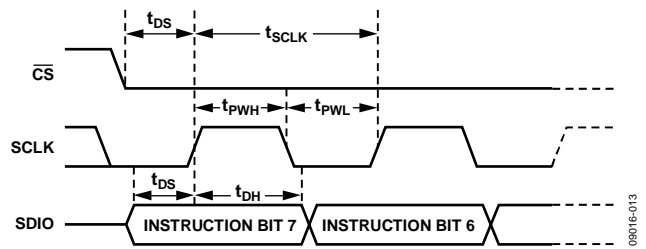
Figure 40. Timing Diagram for Serial Port Register Write (t_{DS} to $t_{D\overline{CS}}$)
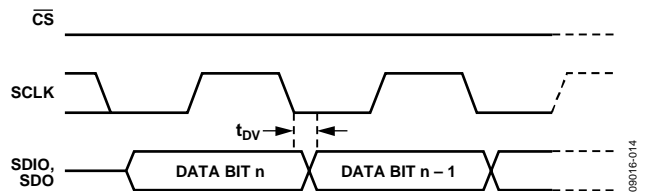
Figure 41. Timing Diagram for Serial Port Register Read

DEVICE CONFIGURATION REGISTER MAP

Table 10. Device Configuration Register Map

Register Name	Addr (Hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default
Comm	0x00	SDIO	LSB_FIRST	Reset						0x00
Power Control	0x01	Power-down DAC I	Power-down DAC Q	Power-down data receiver	Power-down aux ADC				PLL lock status	0x10
Data Format	0x03	Binary data format	Q data first	MSB swap				Data bus width[1:0]		0x00
Interrupt Enable 1	0x04	Enable PLL lock lost	Enable PLL lock	Enable sync signal lost	Enable sync signal lock	Enable sync phase lock	Enable soft FIFO sync	Enable FIFO Warning 1	Enable FIFO Warning 2	0x00
Interrupt Enable 2	0x05	0	0	0	Enable AED compare pass	Enable AED compare fail	Enable SED compare fail	0	0	0x00
Event Flag 1	0x06	PLL lock lost	PLL locked	Sync signal lost	Sync signal locked	Sync phase locked	Soft FIFO sync	FIFO Warning 1	FIFO Warning 2	N/A
Event Flag 2	0x07				AED compare pass	AED compare fail	SED compare fail			N/A
Clock Receiver Control	0x08	DACCLK duty correction	REFCLK duty correction	DACCLK cross-correction	REFCLK cross-correction	1	1	1	1	0x3F
PLL Control 1	0x0A	PLL enable	PLL manual enable	Manual VCO band[5:0]						0x40
PLL Control 2	0x0C	PLL loop bandwidth[2:0]			PLL charge pump current[4:0]					0xD1
PLL Control 3	0x0D	N2[1:0]			PLL cross control enable	N0[1:0]		N1[1:0]		0xD9
PLL Status 1	0x0E	PLL lock				VCO control voltage[3:0]				0x00
PLL Status 2	0x0F			VCO band readback[5:0]						0x00
Sync Control 1	0x10	Sync enable	Data/FIFO rate toggle			Rising edge sync	Sync Averaging[2:0]			0x48
Sync Control 2	0x11			Sync phase request[5:0]						0x00
Sync Status 1	0x12	Sync lost	Sync locked							N/A
Sync Status 2	0x13	Sync phase readback[7:0] (6.2 format)								N/A
FIFO Control	0x17						FIFO phase offset[2:0]			0x04
FIFO Status 1	0x18	FIFO Warning 1	FIFO Warning 2				FIFO soft align ack	FIFO soft align request	FIFO reset aligned	N/A
FIFO Status 2	0x19	FIFO level[7:0]								N/A
Datapath Control	0x1B	Bypass premod	Bypass sinc^{-1}	Bypass NCO		NCO gain	Bypass phase compensation and dc offset	Select sideband	Send I data to Q data	0xE4
HB1 Control	0x1C						HB1[1:0]		Bypass HB1	0x00
HB2 Control	0x1D		HB2[5:0]						Bypass HB2	0x00
HB3 Control	0x1E		HB3[5:0]						Bypass HB3	0x00
Chip ID	0x1F	Chip ID[7:0]								0x08

Register Name	Addr (Hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default
FTW 1 (LSB)	0x30	FTW[7:0]								0x00
FTW 2	0x31	FTW[15:8]								0x00
FTW 3	0x32	FTW[23:16]								0x00
FTW 4 (MSB)	0x33	FTW[31:24]								0x00
NCO Phase Offset LSB	0x34	NCO phase offset[7:0]								0x00
NCO Phase Offset MSB	0x35	NCO phase offset[15:8]								0x00
NCO FTW Update	0x36			Frame FTW ack	Frame FTW request			Update FTW ack	Update FTW request	0x00
I Phase Adj LSB	0x38	I phase adjust[7:0]								0x00
I Phase Adj MSB	0x39							I phase adjust[9:8]		0x00
Q Phase Adj LSB	0x3A	Q phase adjust[7:0]								0x00
Q Phase Adj MSB	0x3B							Q phase adjust[9:8]		0x00
I DAC Offset LSB	0x3C	I DAC offset[7:0]								0x00
I DAC Offset MSB	0x3D	I DAC offset[15:8]								0x00
Q DAC Offset LSB	0x3E	Q DAC offset[7:0]								0x00
Q DAC Offset MSB	0x3F	Q DAC offset[15:8]								0x00
I DAC FS Adjust	0x40	I DAC FS adjust[7:0]								0xF9
I DAC Control	0x41	I DAC sleep						I DAC FS adjust[9:8]		0x01
Aux DAC I Data	0x42	I aux DAC[7:0]								0x00
I Aux DAC Control	0x43	I Aux DAC sign	I Aux DAC current direction	I aux DAC sleep				I aux DAC[9:8]		0x00
Q DAC FS Adjust	0x44	Q DAC FS adjust[7:0]								0xF9
Q DAC Control	0x45	Q DAC sleep						Q DAC FS adjust[9:8]		0x01
Aux DAC Q Data	0x46	Q aux DAC[7:0]								0x00
Q Aux DAC Control	0x47	Q Aux DAC sign	Q Aux DAC current direction	Q aux DAC sleep				Q aux DAC[9:8]		0x00
Die Temperature Range Control	0x48		FS current[2:0]			Reference current[2:0]			Capacitor value	0x02
Die Temperature LSB	0x49	Die temperature[7:0]								N/A
Die Temperature MSB	0x4A	Die temperature[15:8]								N/A
SED Control	0x67	SED compare enable		Sample error detected		Auto-clear enable		Compare fail	Compare pass	0x00
Compare I0 LSBs	0x68	Compare Value I0[7:0]								0xB6
Compare I0 MSBs	0x69	Compare Value I0[15:8]								0x7A
Compare Q0 LSBs	0x6A	Compare Value Q0[7:0]								0x45
Compare Q0 MSBs	0x6B	Compare Value Q0[15:8]								0xEA
Compare I1 LSBs	0x6C	Compare Value I1[7:0]								0x16
Compare I1 MSBs	0x6D	Compare Value I1[15:8]								0x1A
Compare Q1 LSBs	0x6E	Compare Value Q1[7:0]								0xC6

Register Name	Addr (Hex)	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default
Compare Q1 MSBs	0x6F	Compare Value Q1[15:8]								0xAA
SED I LSBs	0x70	Errors detected I_BITS[7:0]								0x00
SED I MSBs	0x71	Errors detected I_BITS[15:8]								0x00
SED Q LSBs	0x72	Errors detected Q_BITS[7:0]								0x00
SED Q MSBs	0x73	Errors detected Q_BITS[15:8]								0x00
Die Revsion	0x7F			Revision[3:0]						0x0C

DEVICE CONFIGURATION REGISTER DESCRIPTIONS

Table 11. Device Configuration Register Descriptions

Register Name	Address (Hex)	Bits	Name	Description	Default
Comm	0x00	7	SDIO	SDIO operation. 0 = SDIO operates as an input only. 1 = SDIO operates as a bidirectional input/output.	0
		6	LSB_FIRST	Serial port communication LSB or MSB first. 0 = MSB first. 1 = LSB first.	0
		5	Reset	1 = device is held in reset when this bit is written high and is held there until the bit is written low.	0
Power Control	0x01	7	Power-down DAC I	1 = powers down DAC I.	0
		6	Power-down DAC Q	1 = powers down DAC Q.	0
		5	Power-down data receiver	1 = powers down the input data receiver.	0
		4	Power-down auxiliary ADC	1 = powers down the auxiliary ADC for temperature sensor.	0
		0	PLL lock status	1 = PLL is locked.	0
Data Format	0x03	7	Binary data format	0 = input data is in twos complement format. 1 = input data is in binary format.	0
		6	Q data first	Indicates I/Q data pairing on data input. 0 = I data sent to data receiver first. 1 = Q data sent to data receiver first.	0
		5	MSB swap	Swaps the bit order of the data input port. 0 = order of the data bits corresponds to the pin descriptions. 1 = bit designations are swapped; most significant bits become the least significant bits.	0
		[1:0]	Data bus width	Data receiver interface mode. 00 = dual-word mode; 32-bit interface bus width. 01 = word mode; 16-bit interleaved interface bus width. 10 = byte mode; 8-bit interleaved interface bus width. 11 = invalid. See the CMOS Input Data Ports section for details on the operation of the different interface modes.	0
Interrupt Enable 1	0x04	7	Enable PLL lock lost	1 = enables interrupt for PLL lock lost.	0
		6	Enable PLL lock	1 = enables interrupt for PLL lock.	0
		5	Enable sync signal lost	1 = enables interrupt for sync signal lock lost.	0
		4	Enable sync signal lock	1 = enables interrupt for sync signal lock.	0
		3	Enable sync phase locked	1 = enables interrupt for clock generation ready.	0
		2	Enable soft FIFO sync	1 = enables interrupt for soft FIFO reset.	0

Register Name	Address (Hex)	Bits	Name	Description	Default
		1	Enable FIFO Warning 1	1 = enables interrupt for FIFO Warning 1.	0
		0	Enable FIFO Warning 2	1 = enables interrupt for FIFO Warning 2.	0
Interrupt Enable 2	0x05	7	Set to 0	Set this bit to 0.	0
		6	Set to 0	Set this bit to 0.	0
		5	Set to 0	Set this bit to 0.	0
		4	Enable AED comparison pass	1 = enables interrupt for AED comparison pass.	0
		3	Enable AED comparison fail	1 = enables interrupt for AED comparison fail.	0
		2	Enable SED comparison fail	1 = enables interrupt for SED comparison fail.	0
		1	Set to 0	Set this bit to 0.	0
		0	Set to 0	Set this bit to 0.	0
Event Flag 1[1]	0x06	7	PLL lock lost	1 = indicates that the PLL, which had been previously locked, has unlocked from the reference signal. This is a latched signal.	0
		6	PLL locked	1 = indicates that the PLL has locked to the reference clock input.	0
		5	Sync signal lost	1 = indicates that the sync logic, which had been previously locked, has lost alignment. This is a latched signal.	0
		4	Sync signal locked	1 = indicates that the sync logic did achieve sync alignment. This is indicated when no phase changes were requested for at least a few full averaging cycles.	0
		3	Sync phase locked	1 = indicates that the internal digital clock generation logic is ready. This occurs when internal clocks are present and stable.	0
		2	Soft FIFO sync	1 = indicates that a FIFO reset originating from a serial port-based request has successfully completed. This is a latched signal.	0
		1	FIFO Warning 1	1 = indicates that the difference between the FIFO read and write pointers is 1.	0
		0	FIFO Warning 2	1 = indicates that the difference between the FIFO read and write pointers is 2.	0
Event Flag 2[1]	0x07	4	AED comparison pass	1 = indicates that the SED logic detected a valid input data pattern compared with the preprogrammed expected values. This is a latched signal.	0
		3	AED comparison fail	1 = indicates that the SED logic detected an invalid input data pattern compared with the preprogrammed expected values. This is a latched signal that automatically clears when eight valid I/Q data pairs are received.	0
		2	SED comparison fail	1 = indicates that the SED logic detected an invalid input data pattern compared with the preprogrammed expected values. This is a latched signal.	
Clock Receiver Control	0x08	7	DACCLK duty correction	1 = enables duty-cycle correction on the DACCLK input.	0
		6	REFCLK duty correction	1 = enables duty-cycle correction on the REFCLK input.	0
		5	DACCLK cross-correction	1 = enables differential crossing correction on the DACCLK input.	1
		4	REFCLK cross-correction	1 = enables differential crossing correction on the REFCLK input.	1

Register Name	Address (Hex)	Bits	Name	Description	Default
PLL Control 1	0x0A	7	PLL enable	1 = enables the PLL clock multiplier. REFCLK input is used as the PLL reference clock signal.	0
		6	PLL manual enable	Enables the manual selection of the VCO band. 1 = manual mode; the correct VCO band must be determined by the user.	1
		[5:0]	Manual VCO band	Selects the VCO band to be used.	0
PLL Control 2	0x0C	[7:5]	PLL loop bandwidth[2:0]	Selects the PLL loop filter bandwidth. 000 = loop bandwidth is nominally 200 kHz 010 = loop bandwidth is nominally 450 kHz 100 = loop bandwidth is nominally 950 kHz 110 = loop bandwidth is nominally 2 MHz	110
		[4:0]	PLL charge pump current[4:0]	Sets the nominal PLL charge-pump current. 00000 = lowest current setting. 11111 = highest current setting.	10001
PLL Control 3	0x0D	[7:6]	N2[1:0]	PLL control clock divider. These bits determine the ratio of the DACCLK rate to the PLL controller clock rate. f_{PC_CLK} must always be less than 80 MHz. 00 = f_{DACCLK}/f_{PC_CLK} = 2. 01 = f_{DACCLK}/f_{PC_CLK} = 4. 10 = f_{DACCLK}/f_{PC_CLK} = 8. 11 = f_{DACCLK}/f_{PC_CLK} = 16.	3
		4	PLL cross control enable	Enables PLL cross-point controller.	1
		[3:2]	N0[1:0]	PLL VCO divider. These bits determine the ratio of the VCO output to the DACCLK frequencies. 00 = f_{VCO}/f_{DACCLK} = 1. 01 = f_{VCO}/f_{DACCLK} = 2. 10 = f_{VCO}/f_{DACCLK} = 4. 11 = f_{VCO}/f_{DACCLK} = 4.	10
		[1:0]	N1[1:0]	PLL loop divider. These bits determine the ratio of the DACCLK to the REFCLK frequencies. 00 = f_{DACCLK}/f_{REFCLK} = 2. 01 = f_{DACCLK}/f_{REFCLK} = 4. 10 = f_{DACCLK}/f_{REFCLK} = 8. 11 = f_{DACCLK}/f_{REFCLK} = 16.	01
PLL Status 1	0x0E	7	PLL lock	The PLL generated clock is tracking the REFCLK input signal.	R
		[3:0]	VCO control voltage[3:0]	VCO control voltage readback (see Table 25).	R
PLL Status 2	0x0F	[5:0]	VCO band readback[5:0]	Indicates the VCO band currently selected.	R
Sync Control 1	0x10	7	Sync enable	1 = enables the synchronization logic.	0
		6	Data/FIFO rate toggle	0 = operates the synchronization at the FIFO reset rate. 1 = operates the synchronization at the data rate.	1
		3	Rising edge sync	0 = sync is initiated on the falling edge of the sync input. 1 = sync is initiated on the rising edge of the sync input.	1
		[2:0]	Sync averaging[2:0]	Sets the number of input samples that are averaged for determining the sync phase. 000 = 1. 001 = 2. 010 = 4. 011 = 8. 100 = 16. 101 = 32.	0

AD9125

Register Name	Address (Hex)	Bits	Name	Description	Default
				110 = 64.	
				111 = 128.	
Sync Control 2	0x11	5:0	Sync phase request[5:0]	This sets the requested clock phase offset after sync. The offset unit is in DACCLK cycles. This enables repositioning of the DAC output with respect to the sync input. The offset can also be used to skew the DAC outputs between the synchronized DACs. 000000 = 0 DACCLK cycles. 000001 = 1 DACCLK cycle. … 111111 = 63 DACCLK cycles.	0
Sync Status 1	0x12	7	Sync lost	1 = indicates that synchronization had been attained but was subsequently lost.	R
		6	Sync locked	1 = indicates that synchronization has been attained.	R
Sync Status 2	0x13	[7:0]	Sync phase readback[7:0]	Indicates the averaged sync phase offset (6.2 format). If the value differs from the requested sync phase value, this indicates sync timing errors. 00000000 = 0.0. 00000001 = 0.25. … 11111110 = 63.50. 11111111 = 63.75.	R
FIFO Control	0x17	[2:0]	FIFO phase offset[2:0]	FIFO write pointer phase offset following FIFO reset. This is the difference between the read pointer and the write pointer values upon FIFO reset. The optimal value is nominally 4. 000 = 0. 001 = 1. … 111 = 7.	4
FIFO Status 1	0x18	7	FIFO Warning 1	FIFO read and write pointers within ±1.	0
		6	FIFO Warning 2	FIFO read and write pointers within ±2.	0
		2	FIFO soft align acknowledge	FIFO read and write pointers are aligned after a serial port initiated FIFO reset.	
		1	FIFO soft align request	Request FIFO read and write pointers alignment via the serial port.	0
		0	FIFO reset aligned	FIFO read and write pointers aligned after a hardware reset.	0
FIFO Status 2	0x19	[7:0]	FIFO level[7:0]	Thermometer encoded measure of the FIFO level.	0
Datapath Control	0x1B	7	Bypass premod	1 = bypasses $f_S/2$ premodulator.	1
		6	Bypass sinc^{-1}	1 = bypasses inverse sinc filter.	1
		5	Bypass NCO	1 = bypasses NCO.	1
		3	NCO gain	0 = default. No gain scaling is applied to the NCO input to the internal digital modulator. 1 = gain scaling of 0.5 is applied to the NCO input to the internal digital modulator. This can eliminate saturation of the modulator output for some combinations of data inputs and NCO signals.	0
		2	Bypass phase compensation and dc offset	1 = bypasses phase compensation and dc offset.	1

Register Name	Address (Hex)	Bits	Name	Description	Default
		1	Select sideband	0 = the modulator outputs high-side image.	0
				1 = the modulator outputs low-side image. The image is spectrally inverted compared with the input data.	
		0	Send I data to Q data	1 = ignores Q data from the interface and disables the clocks to the Q datapath. Sends I data to both the I and Q DACs.	0
HB1 Control	0x1C	[2:1]	HB1[1:0]	00 = input signal is not modulated; filter pass band is from −0.4 to +0.4 of f_{IN1}.	0
				01 = input signal is not modulated; filter pass band is from 0.1 to 0.9 of f_{IN1}.	
				10 = input signal is modulated by f_{IN1}; filter pass band is from 0.6 to 1.4 of f_{IN1}.	
				11 = input signal is modulated by f_{IN1}; filter pass band is from 1.1 to 1.9 of f_{IN1}.	
		0	Bypass HB1	1 = bypasses first-stage interpolation filter.	0
HB2 Control	0x1D	[6:1]	HB2[5:0]	Modulation mode for I Side Half-Band Filter 2.	0
				000000 = input signal is not modulated; filter pass band is from −0.25 to +0.25 of f_{IN2}.	
				001001 = input signal is not modulated; filter pass band is from 0.0 to 0.5 of f_{IN2}.	
				010010 = input signal is not modulated; filter pass band is from 0.25 to 0.75 of f_{IN2}.	
				011011 = input signal is not modulated; filter pass band is from 0.5 to 1.0 of f_{IN2}.	
				100100 = input signal is modulated by f_{IN2}; filter pass band is from 0.75 to 1.25 of f_{IN2}.	
				101101 = input signal is modulated by f_{IN2}; filter pass band is from 1.0 to 1.5 of f_{IN2}.	
				110110 = input signal is modulated by f_{IN2}; filter pass band is from 1.25 to 1.75 of f_{IN2}.	
				111111 = input signal is modulated by f_{IN2}; filter pass band is from 1.5 to 2.0 of f_{IN2}.	
		0	Bypass HB2	1 = bypasses second stage interpolation filter.	0
HB3 Control	0x1E	[6:1]	HB3[5:0]	Modulation mode for I Side Half-Band Filter 3.	0
				000000 = input signal is not modulated; filter pass band is from −0.2 to +0.2 of f_{IN3}.	
				001001 = input signal is not modulated; filter pass band is from 0.05 to 0.45 of f_{IN3}.	
				010010 = input signal is not modulated; filter pass band is from 0.3 to 0.7 of f_{IN3}.	
				011011 = input signal is not modulated; filter pass band is from 0.55 to 0.95 of f_{IN3}.	
				100100 = input signal is modulated by f_{IN3}; filter pass band is from 0.8 to 1.2 of f_{IN3}.	
				101101 = input signal is modulated by f_{IN3}; filter pass band is from 1.05 to 1.45 of f_{IN3}.	
				110110 = input signal is modulated by f_{IN3}; filter pass band is from 1.3 to 1.7 of f_{IN3}.	
				111111 = input signal is modulated by f_{IN3}; filter pass band is from 1.55 to 1.95 of f_{IN3}.	
		0	Bypass HB3	1 = bypasses third-stage interpolation filter.	0
Chip ID	0x1F	[7:0]	Chip ID[7:0]	This register identifies the device as an AD9125.	8

AD9125

Register Name	Address (Hex)	Bits	Name	Description	Default
FTW 1 (LSB)	0x30	[7:0]	FTW[7:0]	FTW[31:0] is the 32-bit frequency tuning word that determines the frequency of the complex carrier generated by the on-chip NCO. The frequency is not updated when the FTW registers are written. The values are only updated when Bit 0 of Register 0x36 transitions from 0 to 1.	0
FTW 2	0x31	[7:0]	FTW[15:8]	See Register 0x30.	0
FTW 3	0x32	[7:0]	FTW[23:16]	See Register 0x30.	0
FTW 4 (MSB)	0x33	[7:0]	FTW[31:24]	See Register 0x30.	0
NCO Phase Offset LSB	0x34	[7:0]	NCO phase offset[7:0]	NCO phase offset[15:0] sets the phase of the complex carrier signal when the NCO is reset. The phase offset spans between 0° and 360°. Each bit represents an offset of 0.0055°. The value is in twos complement format.	0
NCO Phase Offset MSB	0x35	[7:0]	NCO phase offset[15:8]	See Register 0x34.	0
NCO FTW Update	0x36	5	FRAME FTW acknowledge	1 = indicates that the NCO has been reset due to an extended FRAME pulse signal.	0
		4	FRAME FTW request	0 → 1 = the NCO is reset on the first extended FRAME pulse after this bit transitions from 0 to 1.	0
		1	Update FTW acknowledge	1 = indicates that the FTW has been updated.	0
		0	Update FTW request	0 → 1 = the FTW is updated on 0-to-1 transition of this bit.	0
I Phase Adj LSB	0x38	[7:0]	I phase adjust[7:0]	I phase adjust[9:0] is used to insert a phase offset between the I and Q datapaths. This can be used to correct for phase imbalance in a quadrature modulator. See the Quadrature Phase Correction section for details.	0
I Phase Adj MSB	0x39	[1:0]	I phase adjust[9:8]	Register 0x38.	0
Q Phase Adj LSB	0x3A	[7:0]	Q phase adjust[7:0]	Q phase adjust[9:0] is used to insert a phase offset between the I and Q datapaths. This can be used to correct for phase imbalance in a quadrature modulator. See the Quadrature Phase Correction section for details.	0
Q Phase Adj MSB	0x3B	[1:0]	Q phase adjust[9:8]	See Register 0x3A.	0
I DAC Offset LSB	0x3C	[7:0]	I DAC offset[7:0]	I DAC offset[15:0] is a value added directly to the samples written to the I DAC.	0
I DAC Offset MSB	0x3D	[7:0]	I DAC offset[15:8]	See Register 0x3C.	0
Q DAC Offset LSB	0x3E	[7:0]	Q DAC offset[7:0]	Q DAC offset[15:0] is a value added directly to the samples written to the Q DAC.	0
Q DAC Offset MSB	0x3F	[7:0]	Q DAC offset[15:8]	See Register 0x3E.	0
I DAC FS Adjust	0x40	[7:0]	I DAC FS adjust[7:0]	I DAC FS adjust[9:0] sets the full-scale current of the I DAC. The full-scale current can be adjusted from 8.64 mA to 31.6 mA in step sizes of approximately 22.5 μA. 0x000 = 8.64 mA. … 0x200 = 20.14 mA. … 0x3FF = 31.66 mA.	F9
I DAC Control	0x41	7	I DAC sleep	1 = puts the I-channel DAC into sleep mode (fast wake-up mode).	0
		[1:0]	I DAC FS adjust[9:8]	See Register 0x40.	1
Aux DAC I Data	0x42	[7:0]	I aux DAC[7:0]	I aux DAC[9:0] sets the magnitude of the auxiliary DAC current. The range is 0 mA to 2 mA, and the step size is 2 μA. 0x000 = 0.000 mA. 0x001 = 0.002 mA. … 0x3FF = 2.046 mA.	0

Register Name	Address (Hex)	Bits	Name	Description	Default
I Aux DAC Control	0x43	7	I aux DAC sign	0 = the auxiliary DAC I sign is positive, and the current is directed to the IOUT1P pin (Pin 67).	0
				1 = the auxiliary DAC I sign is negative, and the current is directed to the IOUT1N pin (Pin 66).	
		6	I aux DAC current direction	0 = the auxiliary DAC I sources current.	0
				1 = the auxiliary DAC I sinks current.	
		5	I aux DAC sleep	I channel auxiliary DAC sleep.	0
		[1:0]	I Aux DAC[9:8]	See Register 0x42.	0
Q DAC FS Adjust	0x44	[7:0]	Q DAC FS adjust[7:0]	Q DAC FS adjust[9:0] sets the full-scale current of the I DAC. The full-scale current can be adjusted from 8.64 mA to 31.6 mA in step sizes of approximately 22.5 µA.	F9
				0x000 = 8.64 mA.	
				…	
				0x200 = 20.14 mA.	
				…	
				0x3FF = 31.66 mA.	
Q DAC Control	0x45	7	Q DAC sleep	1 = puts the Q-channel DAC into sleep mode (fast wake-up mode).	0
		[1:0]	Q DAC FS adjust[9:8]	See Register 0x44.	1
Aux DAC Q Data	0x46	[7:0]	Q aux DAC[7:0]	Q aux DAC[9:0] sets the magnitude of the aux DAC current. The range is 0 mA to 2 mA, and the step size is 2 µA.	0
				0x000 = 0.000 mA.	
				0x001 = 0.002 mA.	
				…	
				0x3FF = 2.046 mA.	
Q Aux DAC Control	0x47	7	Q aux DAC sign	0 = the auxiliary DAC Q sign is positive, and the current is directed to the IOUT2P pin (Pin 58).	0
				1 = the auxiliary DAC Q sign is negative, and the current is directed to the IOUT2N pin (Pin 59).	
		6	Q aux DAC current direction	0 = the auxiliary DAC Q sources current.	0
				1 = the auxiliary DAC Q sinks current.	
		5	Q aux DAC sleep	Q-channel auxiliary DAC sleep	0
		[1:0]	Q aux DAC[9:8]	See Register 0x46.	0
Die Temp Range Control	0x48	[6:4]	FS current[2:0]	Auxiliary ADC full-scale current.	0
				000 = lowest current.	
				…	
				111 = highest current.	
		[3:1]	Reference current[2:0]	Auxiliary ADC reference current.	1
				000 = lowest current.	
				111 = highest current.	
		0	Capacitor value	Auxiliary ADC internal capacitor value.	0
				0 = 5 pF.	
				1 = 10 pF.	
Die Temp LSB	0x49	[7:0]	Die temp[7:0]	Die Temp[15:0] indicates the approximate die temperature.	R
				0xADCC = −39.9°C.	
				0xC422 = 25.1°C.	
				…	
				0xD8A8 = 84.8°C (see the Temperature Sensor section for details).	
Die Temp MSB	0x4A	[7:0]	Die temp[15:8]	See Register 0x49.	R

Register Name	Address (Hex)	Bits	Name	Description	Default
SED Control	0x67	7	SED compare enable	1 = enables the SED circuitry. None of the flags in this register or the values in Register 0x70 through Register 0x73 are significant if the SED is not enabled.	0
		5	Sample error detected	1 = indicates an error is detected. The bit remains set until cleared. Any write to this register clears this bit to 0.	0
		3	Autoclear enable	1 = enables autoclear mode. This activates Bit 1 and Bit 0 of this register and causes Register 0x70 through Register 0x73 to be autocleared whenever eight consecutive error-free sample data sets are received.	0
		1	Compare fail	1 = indicates an error has been detected. This bit remains high until it is autocleared by the reception of eight consecutive error-free comparisons or until it is cleared by writing to this register.	0
		0	Compare pass	1 = indicates that the last sample comparison was error free.	0
Compare I0 LSBs	0x68	[7:0]	Compare Value I0[7:0]	Compare Value I0[15:0] is the word that is compared with the I0 input sample captured at the input interface.	B6
Compare I0 MSBs	0x69	[7:0]	Compare Value I0[15:8]	See Register 0x68.	7A
Compare Q0 LSBs	0x6A	[7:0]	Compare Value Q0[7:0]	Compare Value Q0[15:0] is the word that is compared with the Q0 input sample captured at the input interface.	45
Compare Q0 MSBs	0x6B	[7:0]	Compare Value Q0[15:8]	See Register 0x6A	EA
Compare I1 LSBs	0x6C	[7:0]	Compare Value I1[7:0]	Compare Value I1[15:0] is the word that is compared with the I1 input sample captured at the input interface.	16
Compare I1 MSBs	0x6D	[7:0]	Compare Value I1[15:8]	See Register 0x6C.	1A
Compare Q1 LSBs	0x6E	[7:0]	Compare Value Q1[7:0]	Compare Value Q1[15:0] is the word that is compared with the Q1 input sample captured at the input interface.	C6
Compare Q1 MSBs	0x6F	[7:0]	Compare Value Q1[15:8]	See Register 0x6E.	AA
SED I LSBs	0x70	[7:0]	Errors Detected I_BITS[7:0]	Errors detected I_BITS[15:0] indicates which bits were received in error.	0
SED I MSBs	0x71	[7:0]	Errors detected I_BITS[15:8]	See Register 0x70.	0
SED Q LSBs	0x72	[7:0]	Errors detected Q_BITS[7:0]	Errors detected Q_BITS[15:0] indicates which bits were received in error.	0
SED Q MSBs	0x73	[7:0]	Errors detected Q_BITS[15:8]	See Register 0x72.	0
Die Revision	0x7F	[5:2]	Revision[3:0]	Corresponds to device die revision.	3

[1] All bit event flags are cleared by writing the respective bit high.

CMOS INPUT DATA PORTS

The AD9125 input data port consists of a data clock (DCI), data bus, and FRAME signal. The data port can be configured to operate in three modes: dual-word mode, word mode, and byte mode.

In dual-word mode, I and Q data is received simultaneously on two 16-pin buses. One bus receives I datapath input words, and the other bus receives Q datapath input words. In word mode, one 16-pin bus is used to receive interleaved I and Q input words. In byte mode, an 8-pin bus is used to receive interleaved I and Q input bytes. The pin assignments of the bus in each mode is described in Table 12.

Table 12. Data Bit Pin Assignments for Data Input Modes

Mode	Data Bus Pin Assignments
Dual Word	I data: D[31:16]
	Q data: D[15:0]
Word	I and Q data: D[29:28], D[25:24], D[21:20], D[17:16], D[15:14], D[11:10], D[7:6], D[3:2]
Byte	I and Q data: D[21:20], D[17:16], D[15:14], D[11:10]

In byte and word modes, a FRAME signal is required for controlling which DAC receives the data. In dual-word mode, the FRAME signal is not required because each DAC has a dedicated bus.

DUAL-WORD MODE

In dual-word mode, the DCI signal is supplied as a qualifying clock that is time aligned with the input data. The rising edge of the DCI signal should be aligned with the changing data of each of the I and Q input data streams.
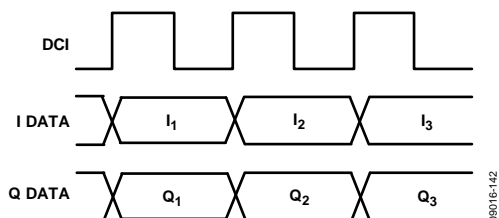
Figure 42. Timing Diagram for Dual-Word Mode

WORD MODE

In word mode, the DCI signal is supplied as a qualifying clock that is time aligned with the input data. The rising edge of the DCI signal should be aligned with the changing data of the interleaved I and Q input data stream. The FRAME signal indicates to which DAC the data is sent. When FRAME is high, data is sent to the I DAC. When FRAME is low, data is sent to the Q DAC. For 14- and 12-bit resolution devices, the two and four LSBs are not significant, respectively. The complete timing diagram is shown in Figure 43.
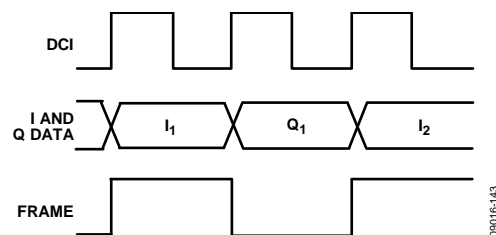

Figure 43. Timing Diagram for Word Mode

BYTE MODE

In byte mode, the DCI signal is supplied as a qualifying clock that is time aligned with the input data. The rising edge of the DCI signal should be aligned with the changing data of the interleaved I and Q input data stream. The FRAME signal indicates to which DAC the data is sent. When FRAME is high, data is sent to the I DAC. When FRAME is low, data is sent to the Q DAC. Both bytes must be written to each datapath for proper operation. For 14- and 12-bit resolution devices, the LSBs in the second byte are not significant. The complete timing diagram is shown in Figure 44.
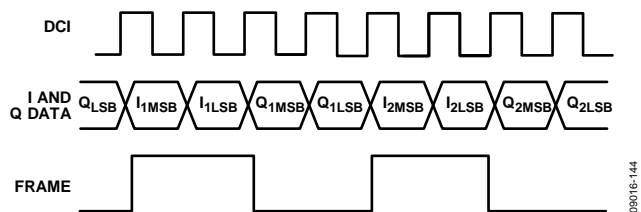

Figure 44. Timing Diagram for Byte Mode

AD9125

INTERFACE TIMING

The timing diagram for the digital interface port is shown in Figure 45. The sampling point of the data bus occurs on the falling edge of the DCI signal and has an uncertainty of 2.1 ns, as illustrated by the sampling interval shown in Figure 45. The D[31:0] and FRAME signals must be valid throughout this sampling interval.

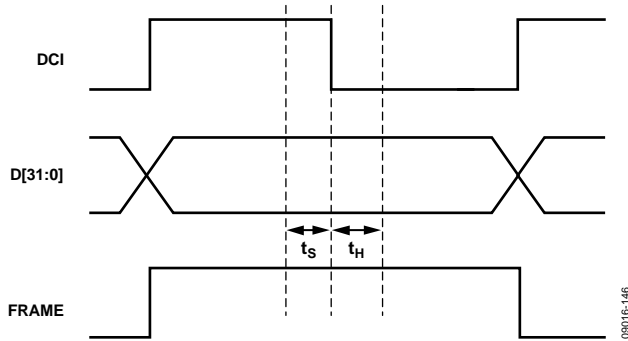The setup (t_S) and hold (t_H) times with respect to the edges are shown in Figure 45. The minimum setup and hold times are shown in Table 13.

Figure 45. Timing Diagram for Input Data Ports

Table 13. Data Port Setup and Hold Times

Minimum Setup Time, t_S (ns)	Minimum Hold Time, t_H (ns)
0.86	1.24
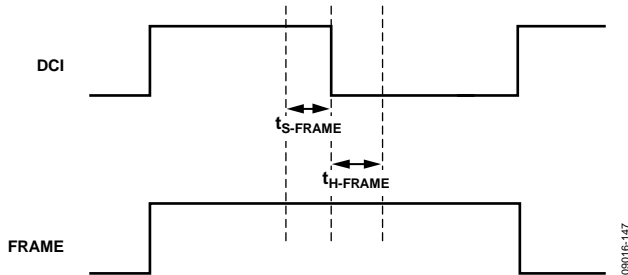
Figure 46. Timing Diagram for Frame input

Table 14. FRAME Setup and Hold Times

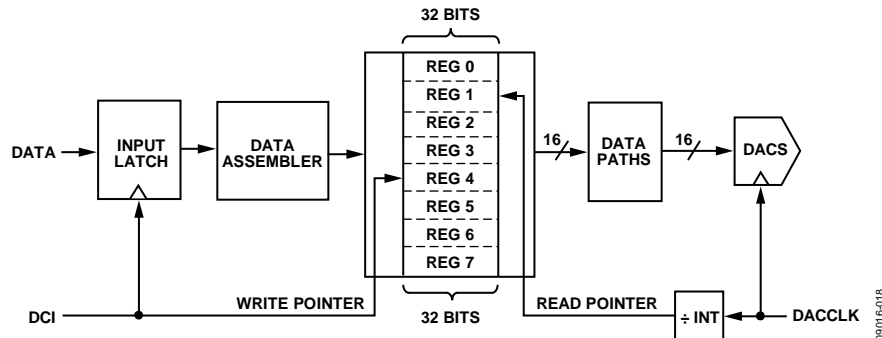| Minimum Setup Time, $t_{S-FRAME}$ (ns)	Minimum Hold Time, t_{H_FRAME} (ns)
−0.04	+1.05

The data interface timing can be verified by using the sample error detection (SED) circuitry. See the Interface Timing Validation section for details.

FIFO OPERATION

The AD9125 contains a 2-channel, 16-bit wide, eight-word-deep FIFO designed to relax the timing relationship between the data arriving at the DAC input ports and the internal DAC data rate clock. The FIFO acts as a buffer that absorbs timing variations between the data source and DAC, such as the clock-to-data variation of an FPGA or ASIC, which significantly increases the timing budget of the interface.

Figure 47 shows the block diagram of the datapath through the FIFO. The data is latched into the device and is formatted, and then it is written into the FIFO register determined by the FIFO write pointer. The value of the write pointer is incremented every time a new word is loaded into the FIFO. Meanwhile, data is read from the FIFO register determined by the read pointer and fed into the digital datapath. The value of the read pointer is updated every time data is read into the datapath from the FIFO. This happens at the data rate, that is, the DACCLK rate divided by the interpolation ratio.

Valid data is transmitted through the FIFO as long as the FIFO does not overflow or become empty. Note that an overflow or empty condition of the FIFO is the same as the write pointer and read pointer being equal. When both pointers are equal, an attempt is made to read and write a single FIFO register simultaneously. This simultaneous register access leads to unreliable data transfer through the FIFO and must be avoided.

Nominally, data is written to the FIFO at the same rate that data is read from the FIFO, which keeps the data level in the FIFO constant. If data is written to the FIFO faster than data is read, the data level in the FIFO increases. If data is written to the device slower than data is read, the data level in the FIFO decreases. For an optimum timing margin, the FIFO level should be maintained near half full, which is the same as maintaining a difference of four between the write pointer and read pointer values.

Figure 47. Block Diagram of Datapath Through FIFO

Initializing the FIFO Data Level

To avoid a concurrent read and write to the same FIFO address and to ensure a fixed pipeline delay, it is important to initialize the FIFO pointers to known states. The FIFO pointers can be initialized in two ways: via a write sequence to the serial port or by strobing the FRAME input. There are two types of FIFO pointer resets: a relative reset and an absolute reset. A relative reset enforces a defined FIFO depth. An absolute reset enforces a particular write pointer value when the reset is initiated. A serial port initiated FIFO reset is always a relative reset. A FRAME strobe initiated reset can be either a relative or an absolute reset.

The operation of the FRAME initiated FIFO reset depends on the synchronization mode chosen. When synchronization is disabled or when the device is configured for data rate mode synchronization, the FRAME strobe initiates a relative FIFO reset. When FIFO mode synchronization is chosen, the FRAME strobe initiates an absolute FIFO reset. More details on the synchronization function can be found in the Multichip Synchronization section.

A summary of the synchronization modes and the type of FIFO reset employed is listed in Table 15.

Table 15. Summary of FIFO Resets

FIFO Reset Signal	Synchronization Mode		
	Disabled	**Data Rate**	**FIFO Rate**
Serial Port	Relative reset	Relative reset	Relative reset
FRAME	Relative reset	Relative reset	Absolute reset

FIFO Level Initialization via Serial Port

A serial port initiated FIFO reset can be issued in any mode and always results in a relative FIFO reset. To initialize the FIFO data level through the serial port, Bit 1 of Register 0x18 should be toggled from 0 to 1 and then back to 0. When the write to the register is complete, the FIFO data level is initialized. When the initialization is triggered, the next time the read pointer becomes 0, the write pointer is set to the value of the FIFO phase offset level (Register 0x17, Bits[2:0]) variable upon initialization. By default, this is 4, but it can be programmed to a value between 0 and 7.

The recommended procedure for a serial port FIFO data level initialization is as follows:

1. Request FIFO level reset by setting Register 0x18, Bit 1, to 1.
2. Verify that the part acknowledges the request by ensuring that Register 0x18, Bit 2, is 1.
3. Remove the request by setting Register 0x18, Bit 1, to 0.
4. Verify that the part drops the acknowledge signal by ensuring that Register 0x18, Bit 2, is 0.

FIFO Level Initialization via FRAME Signal

The primary function of the FRAME input is indicating to which DAC the input data is written. Another function of the FRAME input is initializing the FIFO data level value. This is done by asserting the FRAME signal high for at least the time interval needed to load complete data to the I and Q DACs. This

corresponds to one DCI period in dual-word mode, two DCI periods in word mode, and four DCI periods in byte mode.

To initiate a relative FIFO reset with the FRAME signal, the device must be configured in data rate mode (Register 0x10, Bit 6). When FRAME is asserted in data rate mode, the write pointer is set to 4 (by default or to the FIFO start level) the next time the read pointer becomes 0 (see Figure 48).
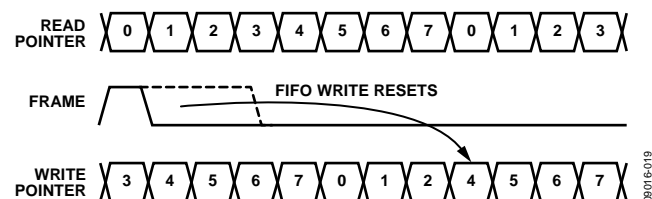

Figure 48. FRAME Input vs. Write Pointer Value, Data Rate Mode

Write Pointer Initialization via FRAME Signal

In FIFO rate synchronization mode, the REFCLK/SYNC signal is used to reset the FIFO read pointer to 0. The edge of the DAC clock used to sample the SYNC signal is selected by Bit 3 of Register 0x10. The FRAME signal is used to reset the FIFO write pointer. In the FIFO rate synchronization mode, the FIFO write pointer is reset immediately after the FRAME signal is asserted high for at least the time interval needed to load complete data to the I and Q DACs. The FIFO write pointer is initialized to the value of the FIFO phase offset[2:0] (Register 0x17). FIFO rate synchronization is selected by setting Bit 6 of Register 0x10 to 0.
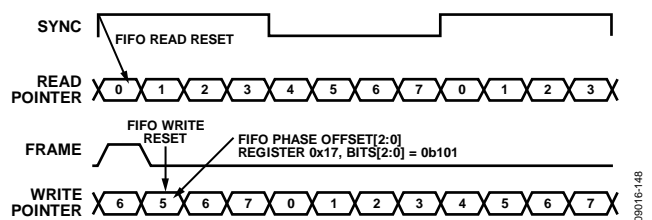

Figure 49. FRAME Input vs. Write Pointer Value, FIFO Rate Mode

Monitoring the FIFO Status

The FIFO initialization and status can be read from Register 0x18. This register provides information on the FIFO initialization method and whether the initialization was successful. The MSB of Register 0x18 is a FIFO warning flag that can optionally trigger a device interrupt ($\overline{IRQ}$). This flag is an indication that the FIFO is close to emptying (FIFO level is 1) or overflowing (FIFO level is 7). This is an indication that data may soon be corrupted and action should be taken.

The FIFO data level can be read from Register 0x19 at any time. The FIFO data level reported by the serial port is denoted as a 7-bit thermometer code of the write counter state relative to the absolute read counter being at 0. The optimum FIFO data level of 4 is, therefore, reported as a value of 00001111 in the status register. It should be noted that, depending on the timing relationship between DCI and the main DACCLK, the FIFO level value can be off by ±1 count. Therefore, it is important to keep the difference between the read and write pointers to at least 2.

DIGITAL DATAPATH

The block diagram in Figure 50 shows the functionality of the digital datapath. The digital processing includes a premodulation block, three half-band interpolation filters, a quadrature modulator with a fine resolution NCO, a phase and offset adjustment block, and an inverse sinc filter.
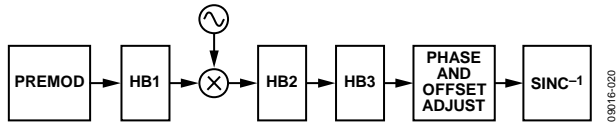

Figure 50. Block Diagram of Digital Datapath

The digital datapath accepts I and Q data streams and processes them as a quadrature data stream. The signal processing blocks can be used when the input data stream is represented as complex data.

The datapath can be used to process an input data stream representing two independent real data streams as well, but the functionality is somewhat restricted. The premodulation block can be used, as well as any of the nonshifted interpolation filter modes (see the Premodulation section for more details).

PREMODULATION

The half-band interpolation filters have selectable pass bands that allow the center frequencies to be moved in increments of ½ of their input data rate. The premodulation block provides a digital upconversion of the incoming waveform by ½ of the incoming data rate, f_{DATA}. This can be used to frequency-shift baseband input data to the center of the interpolation filters' pass band.

INTERPOLATION FILTERS

The transmit path contains three interpolation filters. Each of the three interpolation filters provides a 2× increase in output data rate. The half-band (HB) filters can be individually bypassed or cascaded to provide 1×, 2×, 4×, or 8× interpolation ratios. Each of the half-band filter stages offers a different combination of bandwidths and operating modes.

The bandwidth of the three half-band filters with respect to the data rate at the filter input is as follows:

- Bandwidth of HB1 = $0.8 \times f_{IN1}$
- Bandwidth of HB2 = $0.5 \times f_{IN2}$
- Bandwidth of HB3 = $0.4 \times f_{IN3}$

The usable bandwidth is defined as the frequency over which the filters have a pass-band ripple of less than ±0.001 dB and an image rejection of greater than +85 dB. As is discussed in the Half-Band Filter 1 (HB1) section, the image rejection usually sets the usable bandwidth of the filter, not the pass-band flatness.

The half-band filters operate in several modes, providing programmable pass-band center frequencies as well as signal modulation. The HB1 filter has four modes of operation, and the HB2 and HB3 filters each have eight modes of operation.

Half-Band Filter 1 (HB1)

HB1 has four modes of operation, as shown in Figure 51. The shape of the filter response is identical in each of the four modes. The four modes are distinguished by two factors: the filter center frequency and whether the input signal is modulated by the filter.
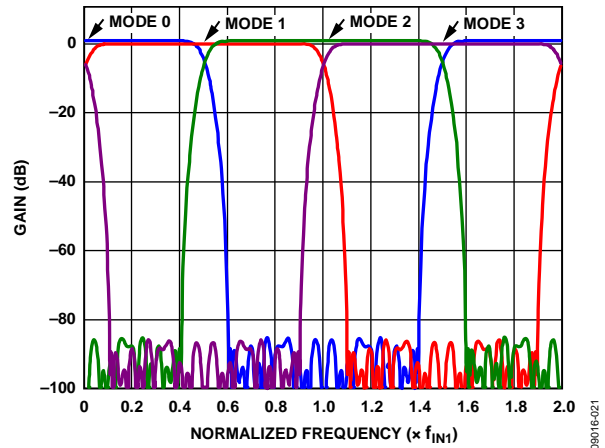

Figure 51. HB1 Filter Modes

As shown in Figure 51, the center frequency in each mode is offset by ½ the input data rate (f_{IN1}) of the filter. Mode 0 and Mode 1 do not modulate the input signal. Mode 2 and Mode 3 modulate the input signal by f_{IN1}. When HB1 operates in Mode 0 and Mode 2, the I and Q paths operate independently and no mixing of the data between channels occurs. When HB1 operates in Mode 1 and Mode 3, mixing of the data between the I and Q paths occurs; therefore, the data input into the filter is assumed complex. Table 16 summarizes the HB1 modes.

Table 16. HB1 Filter Mode Summary

Mode	f_{CENTER}	f_{MOD}	Input Data
0	DC	None	Real or complex
1	$f_{IN}/2$	None	Complex
2	f_{IN}	f_{IN}	Real or complex
3	$3f_{IN}/2$	f_{IN}	Complex

Figure 52 shows the pass-band filter response for HB1. In most applications, the usable bandwidth of the filter is limited by the image suppression provided by the stop-band rejection, not by the pass-band flatness. Table 17 shows the pass-band flatness and stop-band rejection that the HB1 filter supports at different bandwidths.
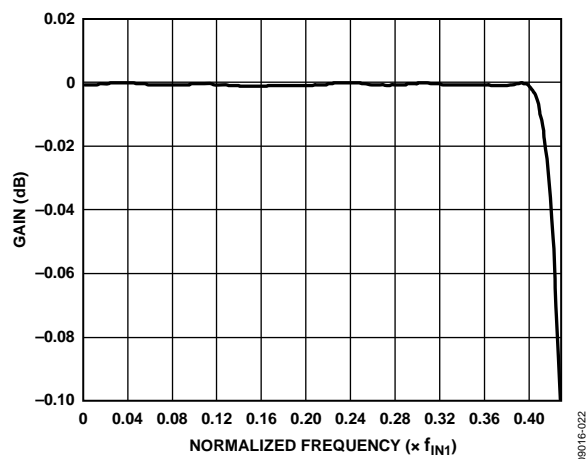

Figure 52. Pass-Band Detail of HB1

Table 17. HB1 Pass-Band Flatness and Stop-Band Rejection

Bandwidth (% of f_{IN1})	Pass-Band Flatness (dB)	Stop-Band Rejection (dB)
80	0.001	85
80.4	0.0012	80
81.2	0.0033	70
82.0	0.0076	60
83.6	0.0271	50
85.6	0.1096	40

Half-Band Filter 2 (HB2)

HB2 has eight modes of operation, as shown in Figure 53 and Figure 54. The shape of the filter response is identical in each of the eight modes. The eight modes are distinguished by two factors: the filter center frequency and whether the input signal is modulated by the filter.
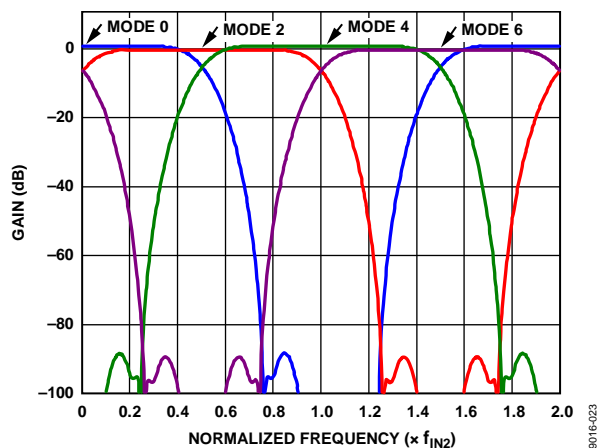

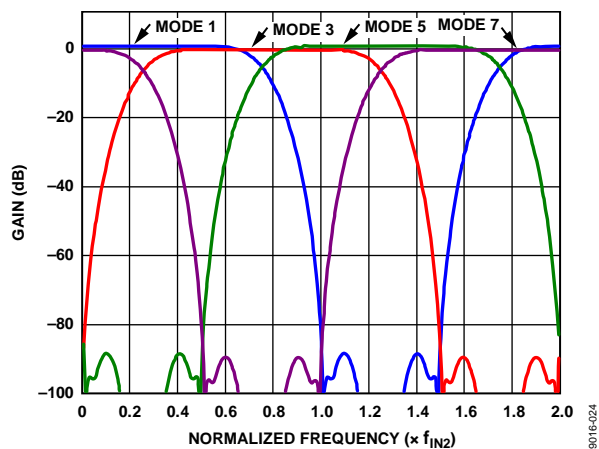*Figure 53. HB2, Even Filter Modes*

Figure 54. HB2, Odd Filter Modes

As shown in Figure 53 and Figure 54, the center frequency in each mode is offset by ¼ of the input data rate (f_{IN2}) of the filter. Mode 0 through Mode 3 do not modulate the input signal. Mode 4 through Mode 7 modulate the input signal by f_{IN2}. When HB2 operates in Mode 0 and Mode 4, the I and Q paths operate independently and no mixing of the data between channels occurs. When HB2 operates in the other six modes, mixing of the data between the I and Q paths occurs; therefore, the data input to the filter is assumed complex.

Table 18 summarizes the HB2 and HB3 modes.

Table 18. HB2 and HB3 Filter Mode Summary

Mode	f_{CENTER}	f_{MOD}	Input Data
0	DC	None	Real or complex
1	$f_{IN}/4$	None	Complex
2	$f_{IN}/2$	None	Complex
3	$3f_{IN}/4$	None	Complex
4	f_{IN}	f_{IN}	Real or complex
5	$5f_{IN}/4$	f_{IN}	Complex
6	$6f_{IN}/4$	f_{IN}	Complex
7	$7f_{IN}/4$	f_{IN}	Complex

Figure 55 shows the pass-band filter response for HB2. In most applications, the usable bandwidth of the filter is limited by the image suppression provided by the stop-band rejection, not by the pass-band flatness. Table 19 shows the pass-band flatness and stop-band rejection that the HB2 filter supports at different bandwidths.
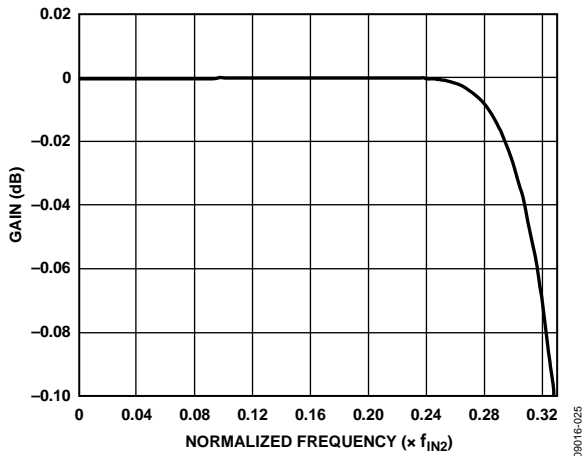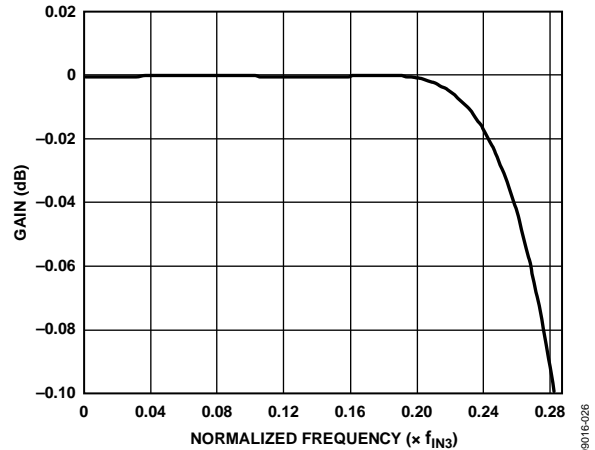

Figure 55. Pass-Band Detail of HB2

Table 19. HB2 Pass-Band Flatness and Stop-Band Rejection

Complex Bandwidth (% of f_{IN2})	Pass-Band Flatness (dB)	Stop-Band Rejection (dB)
50	0.001	85
50.8	0.0012	80
52.8	0.0028	70
56.0	0.0089	60
60	0.0287	50
64.8	0.1877	40

Half-Band Filter 3 (HB3)

HB3 has eight modes of operation that function the same as HB2. The primary difference between HB2 and HB3 is the filter bandwidths.

Figure 56 shows the pass-band filter response for HB3. In most applications, the usable bandwidth of the filter is limited by the image suppression provided by the stop-band rejection, not by the pass-band flatness. Table 20 shows the pass-band flatness and stop-band rejection that the HB3 filter supports at different bandwidths.

Figure 56. Pass-Band Detail of HB3

Table 20. HB3 Pass-Band Flatness and Stop-Band Rejection

Complex Bandwidth (% of f_{IN3})	Pass-Band Flatness (dB)	Stop-Band Rejection (dB)
40	0.001	85
40.8	0.0014	80
42.4	0.002	70
45.6	0.0093	60
49.8	0.03	50
55.6	0.1	40

NCO MODULATION

The digital quadrature modulator makes use of a numerically controlled oscillator, a phase shifter, and a complex modulator to provide a means for modulating the signal by a programmable carrier signal. A block diagram of the digital modulator is shown in Figure 57. The fine modulation provided by the digital modulator, in conjunction with the coarse modulation of the interpolation filters and premodulation block, allows the signal to be placed anywhere in the output spectrum with very fine frequency resolution.
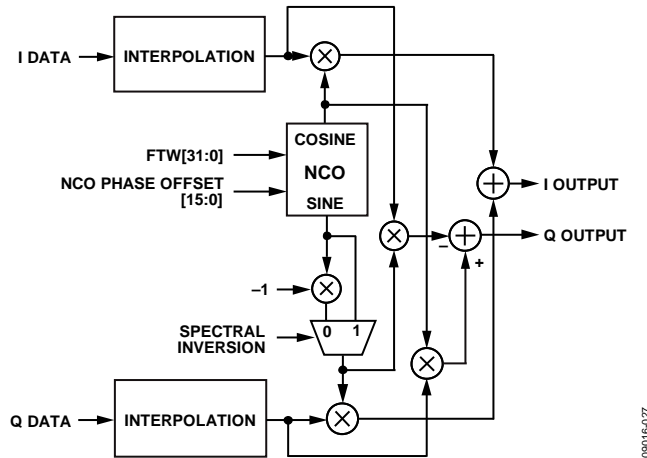
Figure 57. Digital Quadrature Modulator Block Diagram

The quadrature modulator is used to mix the carrier signal generated by the NCO with the I and Q signal. The NCO produces a quadrature carrier signal to translate the input signal to a new center frequency. A complex carrier signal is a pair of sinusoidal waveforms of the same frequency, offset 90° from each other. The frequency of the complex carrier signal is set via FTW[31:0] in Register 0x30 through Register 0x33.

The NCO operating frequency, f_{NCO}, is at either f_{DATA} (HB1 bypassed) or twice f_{DATA} (HB1 enabled). The frequency of the complex carrier signal can be set from dc up to f_{NCO}. The frequency tuning word (FTW) is calculated as

$$FTW = \frac{f_{CARRIER}}{f_{NCO}} \times 2^{32}$$

The generated quadrature carrier signal is mixed with the I and Q data. The quadrature products are then summed into the I and Q datapaths, as shown in Figure 57.

Updating the Frequency Tuning Word

The frequency tuning word registers do not update immediately upon writing as other configuration registers. After loading the FTW registers with the desired values, Bit 0 of Register 0x36 must transition from 0 to 1 for the new FTW to take effect.

DATAPATH CONFIGURATION

Configuring the AD9125 datapath starts with the application requirements of the input data rate, the interpolation ratio, the output signal bandwidth, and the output signal center frequency.

Given these four parameters, the first step in configuring the datapath is to verify that the device supports the bandwidth requirements. The modes of the interpolation filters are then chosen. Finally, any additional frequency offset requirements are determined and applied with premodulation and NCO modulation.

Determining Datapath Signal Bandwidth

The available signal bandwidth of the datapath is dependent on the center frequency of the output signal in relation to the center frequency of the interpolation filters used. Signal center frequencies that are offset from the center frequencies of the half-band filters lower the available signal bandwidth.

When correctly configured, the available complex signal bandwidth for 2× interpolation is always 80% of the input data rate. The available signal bandwidth for 4× interpolation vs. output frequency varies between 50% and 80% of the input data rate, as shown in Figure 58. Note that in 4× interpolation mode, $f_{DAC} = 4 \times f_{DATA}$; therefore, the data shown in Figure 58 repeats four times from dc to f_{DAC}.
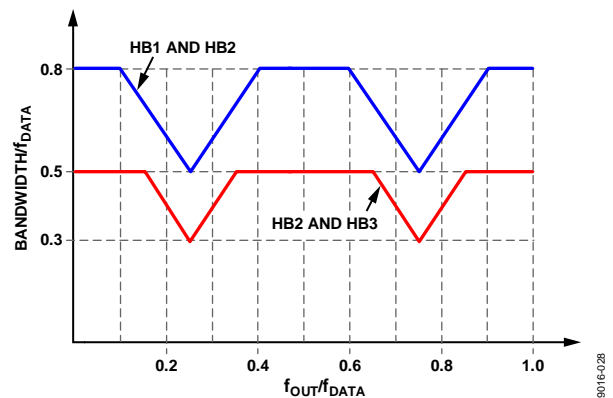

Figure 58. Signal Bandwidth vs. Center Frequency of the Output Signal, 4× Interpolation

Configuring 4× interpolation using the HB2 and HB3 filters can lower the power consumption of the device at the expense of reduced bandwidth. The lower curve in Figure 58 shows that the supported bandwidth in this mode varies from 30% to 50% of f_{DATA}.

The available signal bandwidth for 8× interpolation vs. output frequency varies between 50% and 80% of the input data rate, as shown in Figure 59. Note that in 8× interpolation mode, $f_{DAC} = 8 \times f_{DATA}$; therefore, the data shown in Figure 59 repeats eight times from dc to f_{DAC}.
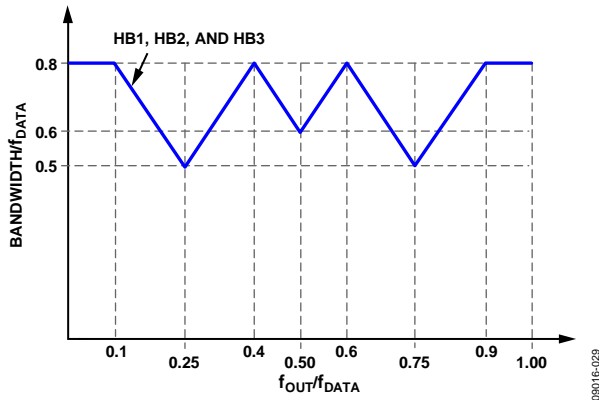
AD9125

Figure 59. Signal Bandwidth vs. Center Frequency of the Output Signal,
8× Interpolation

DETERMINING INTERPOLATION FILTER MODES

Table 21 shows the recommended interpolation filter settings for a variety of filter interpolation factors, filter center frequencies, and signal modulation. The interpolation modes were chosen based on the final center frequency of the signal and by determining the frequency shift of the signal required. When these are known and put in terms of the input data rate (f_{DATA}), the filter configuration that comes closest to matching should be chosen from Table 21.

Table 21. Recommended Interpolation Filter Modes (Register 0x1C through Register 0x1E)

| Interpolation Factor | Filter Modes | | | f_{SIGNAL} Modulation | f_{CENTER} Shift |
	HB1[1:0]	HB2[5:0]	HB3[5:0]		
8	00 (0)	000000	000000	DC	0
8	01 (1)	001001	000000	DC[1]	$f_{DATA}/2$
8[2]	10 (2)	010010	001001	f_{DATA}	f_{DATA}
8	11 (3)	011011	001001	f_{DATA}[1]	$3 \times f_{DATA}/2$
8	00 (0)	100100	010010	$2 \times f_{DATA}$	$2 \times f_{DATA}$
8	01 (1)	101101	010010	$2 \times f_{DATA}$[1]	$5 \times f_{DATA}/2$
8	10 (2)	110110	011011	$3 \times f_{DATA}$	$3 \times f_{DATA}$
8	11 (3)	111111	011011	$3 \times f_{DATA}$[1]	$7 \times f_{DATA}/2$
8	00 (0)	000000	100100	$4 \times f_{DATA}$	$4 \times f_{DATA}$
8	01 (1)	001001	100100	$4 \times f_{DATA}$[1]	$9 \times f_{DATA}/2$
8	10 (2)	010010	101101	$5 \times f_{DATA}$	$5 \times f_{DATA}$
8	11 (3)	011011	101101	$5 \times f_{DATA}$[1]	$11 \times f_{DATA}/2$
8	00 (0)	100100	110110	$6 \times f_{DATA}$	$6 \times f_{DATA}$
8	01 (1)	101101	110110	$6 \times f_{DATA}$[1]	$13 \times f_{DATA}/2$
8	10 (2)	110110	111111	$7 \times f_{DATA}$	$7 \times f_{DATA}$
8	11 (3)	111111	111111	$7 \times f_{DATA}$[1]	$15 \times f_{DATA}/2$
4	00 (0)	000000	Bypass	DC	0
4[3]	01 (1)	001001	Bypass	DC[1]	$f_{DATA}/2$
4	10 (2)	010010	Bypass	f_{DATA}	f_{DATA}
4	11 (3)	011011	Bypass	f_{DATA}[1]	$3 \times f_{DATA}/2$
4	00 (0)	100100	Bypass	$2 \times f_{DATA}$	$2 \times f_{DATA}$
4	01 (1)	101101	Bypass	$2 \times f_{DATA}$[1]	$5 \times f_{DATA}/2$
4	10 (2)	110110	Bypass	$3 \times f_{DATA}$	$3 \times f_{DATA}$
4	11 (3)	111111	Bypass	$3 \times f_{DATA}$[1]	$7 \times f_{DATA}/2$
2	00 (0)	Bypass	Bypass	DC	0
2	01 (1)	Bypass	Bypass	DC[1]	$f_{DATA}/2$
2	10 (2)	Bypass	Bypass	f_{DATA}	f_{DATA}
2	11 (3)	Bypass	Bypass	f_{DATA}[1]	$3 \times f_{DATA}/2$

[1] When HB1 Mode 1 or Mode 3 is used, enabling premodulation provides an addition frequency translation of the input signal by $f_{DATA}/2$, which centers a baseband input signal in the filter pass band.
[2] This configuration was used in the 8× interpolation without NCO example. In addition, see the 8× Interpolation Without NCO section.
[3] This configuration was used in the 4× interpolation with NCO example. In addition, see the 4× Interpolation with NCO section

DATAPATH CONFIGURATION EXAMPLE

8× Interpolation Without NCO

Given the following conditions, the desired 75 MHz of bandwidth is 75% of f_{DATA}:

- f_{DATA} = 100 MSPS
- 8× interpolation
- f_{BW} = 75 MHz
- f_{CENTER} = 100 MHz

In this case, the ratio of f_{OUT}/f_{DATA} = 100/100 = 1.0. From Figure 59, the bandwidth supported at f_{DATA} is 0.8, which verifies that the AD9125 supports the bandwidth required in this configuration.

The signal center frequency is f_{DATA}, and assuming the input signal is at baseband, the frequency shift required is also f_{DATA}. Using the settings detailed in the third row of the IF column from Table 21 (these settings use the configuration in the 8× interpolation without NCO example) selects filter modes that result in a center frequency of f_{DATA} and a frequency translation of f_{DATA}. The selected modes for the three half-band filters are HB1, Mode 2; HB2, Mode 2; and HB3, Mode 1. Figure 60 shows how the signal propagates through the interpolation filters.

Because $2 \times f_{IN1} = f_{IN2}$ and $2 \times f_{IN2} = f_{IN3}$, the signal appears frequency scaled by ½ into each consecutive stage. The output signal band spans 0.15 to 0.35 of f_{IN3} (400 MHz). Therefore, the output frequency supported is 60 MHz to 140 MHz, which covers the 75 MHz bandwidth centered at 100 MHz, as desired.

4× Interpolation with NCO

Given the following conditions, the desired 140 MHz of bandwidth is 56% of f_{DATA}:

- f_{DATA} = 250 MSPS
- 4× interpolation
- f_{BW} = 140 MHz
- f_{CENTER} = 175 MHz

As shown in Figure 58, the value at $0.7 \times f_{DATA}$ is 0.6. This is calculated as $0.8 - 2(0.7 - 0.6) = 0.6$. Therefore, the AD9125 supports a bandwidth of 60% of f_{DATA}, which exceeds the required 56%.

The signal center frequency is $0.7 \times f_{DATA}$, and assuming the input signal is at baseband, the frequency shift required is also $0.7 \times f_{DATA}$. Using the settings detailed in the second row in the IF column in the 4× interpolation section in Table 21 selects the filter modes that give a center frequency of $f_{DATA}/2$ and no frequency translation. The selected modes for the three half-band filters are HB1, Mode 1; HB2, Mode 1; and HB3, bypassed.

Because Mode 1 of HB1 was selected, the premodulation block should be enabled. This provides $f_{DATA}/2$ modulation, which centers the baseband input data at the center frequency of HB1. The digital modulator can be used to provide the final frequency translation of $0.2 \times f_{DATA}$ to place the output signal at $0.7 \times f_{DATA}$, as desired.

The formula for calculating the FTW of the NCO is

$$FTW = \frac{f_{CARRIER}}{f_{NCO}} \times 2^{32}$$

where:
$f_{CARRIER} = 0.2 \times f_{DATA}$.
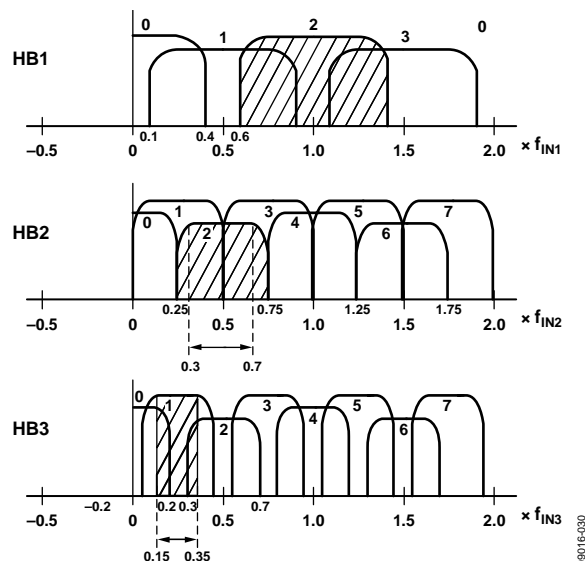$f_{NCO} = 2 \times f_{DATA}$. Therefore, $FTW = 2^{32}/10$.

Figure 60. Signal Propagation for 8× Interpolation (f_{DATA} Modulation)

Rev. 0 | Page 37 of 56

DATA RATES VS. INTERPOLATION MODES

Table 23 summarizes the maximum bus speed (f_BUS), the supported input data rates, and the signal bandwidths for various combinations of bus width modes and interpolation rates. The maximum bus speed in any mode is 250 MHz. The maximum DAC update rate (f_DAC) in any mode is 1000 MHz. The real signal bandwidth supported is a fraction of the input data rate, which depends on the interpolation filter (HB1, HB2, or HB3) selected. The complex signal bandwidth supported is twice the real signal bandwidth.

In general, 2× interpolation is best supported by enabling HB1, and 4× interpolation is best supported enabling HB1 and HB2. In some cases, power dissipation can be lowered by avoiding HB1. If the bandwidth required is low enough, 2× interpolation can be supported by using HB2, and 4× interpolation can be supported by using HB2 and HB3.

COARSE MODULATION MIXING SEQUENCES

The coarse digital quadrature modulation occurs within the interpolation filters. The modulation shifts the frequency spectrum of the incoming data by the frequency offset selected. The frequency offsets available are multiples of the input data rate. The modulation is equivalent to multiplying the quadrature input signal by a complex carrier signal, C(t), of the form

$$C(t) = \cos(\omega_c t) + j \sin(\omega_c t)$$

In practice, this modulation results in mixing functions as shown in Table 22.

Table 22. Modulation Mixing Sequences

Modulation	Mixing Sequence
$f_S/2$	I = I, −I, I, −I, … Q = Q, −Q, Q, −Q, …
$f_S/4$	I = I, Q, −I, −Q, … Q = Q, −I, −Q, I, …
$3 \times f_S/4$	I = I, −Q, −I, Q, … Q = Q, I, −Q, −I, …
$f_S/8$	I = I, r(I + Q), Q, r(−I + Q), −I, −r(I + Q), −Q, r(I − Q), … Q = Q, r(Q − I), −I, −r(Q + I), −Q, r(−Q + I),I, r(Q + I), …

Note that $r = \dfrac{\sqrt{2}}{2}$

As shown in Table 22, the mixing functions of most of the modes result in cross coupling samples between the I and Q channels. The I and Q channels only operate independently in $f_S/2$ mode. This means that real modulation using both the I and Q DAC outputs can only be done in $f_S/2$ mode. All other modulation modes require complex input data and produce complex output signals.

Table 23. Summary of Data Rates and Bandwidths vs. Interpolation Modes

Bus Width	Filter Modes			f_BUS (Mbps)	f_DATA (Mbps)	Real Signal Bandwidth (MHz)	f_DAC (MHz)
	HB3	HB2	HB1				
Byte Mode (8 Bits)	0	0	0	250	62.5	31.25	62.5
	0	0	1	250	62.5	25	125
	0	1	0	250	62.5	15.625	125
	0	1	1	250	62.5	25	250
	1	1	0	250	62.5	15.625	250
	1	1	1	250	62.5	25	500
Word Mode (16 Bits)	0	0	0	250	125	62.5	125
	0	0	1	250	125	50	250
	0	1	0	250	125	31.25	250
	0	1	1	250	125	50	500
	1	1	0	250	125	31.25	500
	1	1	1	250	125	50	1000
Dual-Word Mode (32 Bits)	0	0	0	250	250	125	250
	0	0	1	250	250	100	500
	0	1	0	250	250	62.5	500
	0	1	1	250	250	100	1000
	1	1	0	250	250	62.5	1000
	1	1	1	125	125	50	1000

QUADRATURE PHASE CORRECTION

The purpose of the quadrature phase correction block is to enable compensation of the phase imbalance of the analog quadrature modulator following the DAC. If the quadrature modulator has a phase imbalance, the unwanted sideband appears with significant energy. Tuning the quadrature phase adjust value can optimize image rejection in single sideband radios.

Ordinarily, the I and Q channels have an angle of precisely 90° between them. The quadrature phase adjustment is used to change the angle between the I and Q channels. When the I phase adjust[9:0] is set to 1000000000, the I DAC output moves approximately 1.75° away from the Q DAC output, creating an angle of 91.75° between the channels. When the I phase adjust[9:0] is set to 0111111111, the I DAC output moves approximately 1.75° toward the Q DAC output, creating an angle of 88.25° between the channels.

The Q phase adjust bits (Bits[9:0]) work in a similar fashion. When the Q phase adjust[9:0]) is set to 1000000000, the Q DAC output moves approximately 1.75° away from the I DAC output, creating an angle of 91.75° between the channels. When the Q phase adjust[9:0] is set to 0111111111, the Q DAC output moves approximately 1.75° toward the I DAC output, creating an angle of 88.25° between the channels.

Based on these two endpoints, the combined resolution of the phase compensation register is approximately 3.5°/1024, or 0.00342°, per code.

DC OFFSET CORRECTION

The dc value of the I datapath and the Q datapath can be independently controlled by adjusting the I DAC offset[15:0] and Q DAC offset[15:0] values in Register 0x3C through Register 0x3F. These values are added directly to the datapath values. Care should be taken not to overrange the transmitted values.

Figure 61 shows how the DAC offset current varies as a function of the I DAC offset[15:0] and Q DAC offset[15:0] values. With the digital inputs fixed at midscale (0x0000, twos complement data format), Figure 61 shows the nominal I_{OUTxP} and I_{OUTxN} currents as the DAC offset value is swept from 0 to 65,535. Because I_{OUTxP} and I_{OUTxN} are complementary current outputs, the sum of I_{OUTxP} and I_{OUTxN} is always 20 mA.
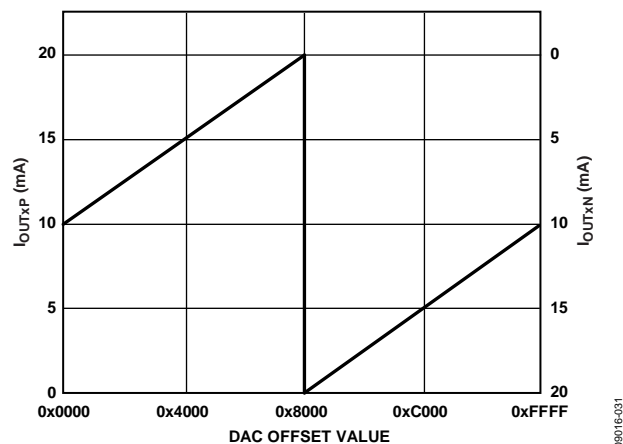

Figure 61. DAC Output Currents vs. DAC Offset Value

INVERSE SINC FILTER

The inverse sinc (sinc^{-1}) filter is a nine-tap FIR filter. The composite response of the sinc^{-1} and the sin(x)/x response of the DAC is shown in Figure 62. The composite response has less than ±0.05 dB pass-band ripple up to a frequency of $0.4 \times f_{DACCLK}$. To provide the necessary peaking at the upper end of the pass band, the inverse sinc filters have an intrinsic insertion loss of about 3.2 dB. Figure 62 shows the composite frequency response.
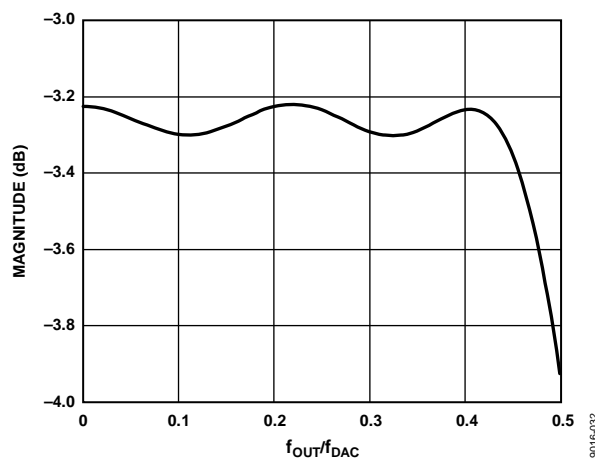

Figure 62. Sample Composite Responses of the Sinc^{-1} Filter with Sin(x)/x Roll-Off

The sinc^{-1} filter is enabled by default. It can be bypassed by setting the bypass sinc^{-1} bit (Register 0x1B, Bit 6).

DAC INPUT CLOCK CONFIGURATIONS

DAC INPUT CLOCK CONFIGURATIONS

The AD9125 DAC sample clock (DACCLK) can be sourced directly or by clock multiplying. Clock multiplying employs the on-chip phased-locked loop (PLL) that accepts a reference clock operating at a submultiple of the desired DACCLK rate, most commonly the data input frequency. The PLL then multiplies the reference clock up to the desired DACCLK frequency, which can be used to generate all the internal clocks required by the DAC. The clock multiplier provides a high quality clock that meets the performance requirements of most applications. Using the on-chip clock multiplier removes the burden of generating and distributing the high speed DACCLK.

The second mode bypasses the clock multiplier circuitry and allows DACCLK to be sourced directly to the DAC core. This mode enables the user to source a very high quality clock directly to the DAC core. Sourcing the DACCLK directly through the REFCLKP, REFCLKN, DACCLKP, and DACCLKN pins may be necessary in demanding applications that require the lowest possible DAC output noise, particularly when directly synthesizing signals above 150 MHz.

Driving the DACCLK and REFCLK Inputs

The REFCLK and DACCLK differential inputs share similar clock receiver input circuitry. Figure 63 shows a simplified circuit diagram of the input. The on-chip clock receiver has a differential input impedance of about 10 kΩ. It is self-biased to a common-mode voltage of about 1.25 V. The inputs can be driven by directly coupling differential PECL or LVDS drivers. The inputs can also be ac-coupled if the driving source cannot meet the input compliance voltage of the receiver.
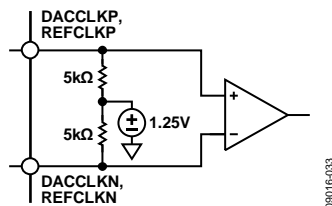

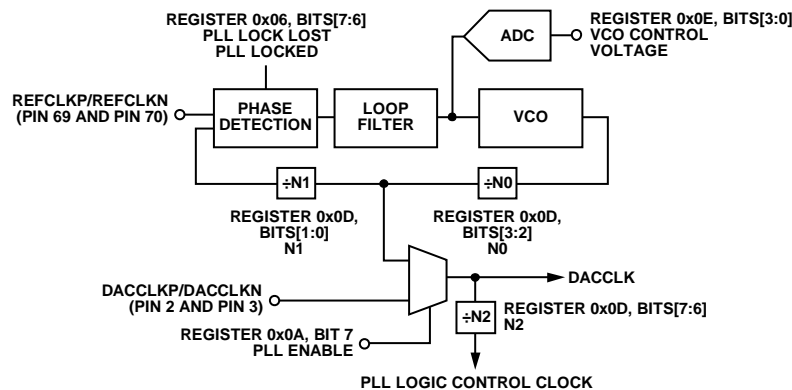Figure 63. Clock Receiver Input Equivalent Circuit

The minimum input drive level to either clock input is 200 mV p-p differential. The optimal performance is achieved when the clock input signal is between 800 mV p-p differential and 1.6 V p-p differential. Whether using the on-chip clock multiplier or sourcing the DACCLK directly, it is necessary that the input clock signal to the device has low jitter and fast edge rates to optimize the DAC noise performance.

Direct Clocking

Direct clocking with a low noise clock produces the lowest noise spectral density at the DAC outputs. To select the differential clock inputs as the source for the DAC sampling clock, set the PLL enable bit (Register 0x0A, Bit 7) to 0. This powers down the internal PLL clock multiplier and selects the input from the DACCLKP and DACCLKN pins as the source for the internal DAC sample clock.

The device also has duty-cycle correction circuitry and differential input-level correction circuitry. Enabling these circuits can provide improved performance in some cases. The control bits for these functions can be found in Register 0x08 (see Table 11).

Clock Multiplication

The on-chip PLL clock multiplier circuit can be used to generate the DAC sample rate clock from a lower frequency reference clock. When the PLL enable bit (Register 0x0A, Bit 7) is set to 1, the clock multiplication circuit generates the DAC sample clock from the lower rate REFCLK input. The functional diagram of the clock multiplier is shown in Figure 64.

The clock multiplication circuit operates such that the VCO outputs a frequency, f_{VCO}, equal to the REFCLK input signal frequency multiplied by $N1 \times N0$.

$$f_{VCO} = f_{REFCLK} \times (N1 \times N0)$$

The DAC sample clock frequency, f_{DACCLK}, is equal to

$$f_{DACCLK} = f_{REFCLK} \times N1$$

The output frequency of the VCO must be chosen to keep f_{VCO} in the optimal operating range of 1.0 GHz to 2.1 GHz. The frequency of the reference clock and the values of N1 and N0 must be chosen so that the desired DACCLK frequency can be synthesized and the VCO output frequency is in the correct range.

Figure 64. PLL Clock Multiplication Circuit

Rev. 0 | Page 40 of 56

PLL Settings

There are three settings for the PLL circuitry that should be programmed to their nominal values. Table 24 lists the recommended PLL settings for these parameters.

Table 24. PLL Settings

PLL SPI Control	Address Register	Bits	Optimal Setting
PLL Loop Bandwidth[2:0]	0x0C	[7:5]	110
PLL Charge Pump Current[4:0]	0x0C	[4:0]	10001
PLL Cross Control Enable	0x0D	4	1

Configuring the VCO Tuning Band

The PLL VCO has a valid operating range from approximately 1.0 GHz to 2.1 GHz, covered in 63 overlapping frequency bands. For any desired VCO output frequency, there may be several valid PLL band select values. The frequency bands of a typical device are shown in Figure 65. Device-to-device variations and operating temperature affect the actual band frequency range. Therefore, it is required that the optimal PLL band select value be determined for each individual device.

Automatic VCO Band Select

The device has an automatic VCO band select feature on chip. Using the automatic VCO band select feature is a simple and reliable method of configuring the VCO frequency band. This feature is enabled by writing 0x80 to Register 0x0A. When this value is written, the device executes an automated routine that determines the optimal VCO band setting for the device. The setting selected by the device ensures that the PLL remains locked over the full −40°C to +85°C operating temperature range of the device without further adjustment. (The PLL remains locked over the full temperature range even if the temperature during initialization is at one of the temperature extremes.)
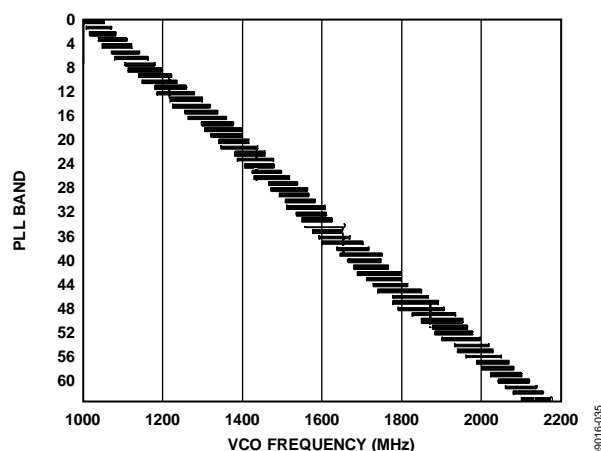

Figure 65. PLL Lock Range over Temperature for a Typical Device

Manual VCO Band Select

The device also has a manual band select mode (PLL manual enable, Register 0x0A, Bit 6 = 1) that allows the user to select the VCO tuning band. When in manual mode, the VCO band is set directly with the value written to the manual VCO band, (Register 0x0A, Bits[5:0]). To properly select the VCO band, follow these steps:

1. Put the device in manual band select mode.
2. Sweep the VCO band over a range of bands that result in the PLL being locked.
3. For each band, verify that the PLL is locked and read the PLL using the VCO control voltage (Register 0x0E, Bits[3:0]).
4. Select the band that results in the control voltage being closest to the center of the range, that is, 0000 or 1000 (see Table 25 for more details). The resulting VCO band should be the optimal setting for the device. Write this band to the manual VCO band (Register 0x0A, Bits[5:0]) value.
5. If desired, an indication of where the VCO is within the operating frequency band can be determined by querying the VCO control voltage. Table 25 shows how to interpret the PLL VCO control voltage (Register 0x0E, Bits[3:0]) value.

Table 25. VCO Control Voltage Range Indications

VCO Control Voltage	Indication
1111 1110	Move to a higher VCO band
1101 1100 1011 1010	VCO is operating in the higher end of the frequency band
1001 1000 0111 0110	VCO is operating within an optimal region of the frequency band
0101 0100 0011 0010	VCO is operating in the lower end of the frequency band
0001 0000	Move to a lower VCO band

ANALOG OUTPUTS

TRANSMIT DAC OPERATION

Figure 66 shows a simplified block diagram of the transmit path DACs. The DAC core consists of a current source array, a switch core, a digital control logic, and a full-scale output current control. The DAC full-scale output current (I_{OUTFS}) is nominally 20 mA. The output currents from the IOUT1P/IOUT2P and IOUT1N/ IOUT2N pins are complementary, meaning that the sum of the two currents always equals the full-scale current of the DAC. The digital input code to the DAC determines the effective differential current delivered to the load.
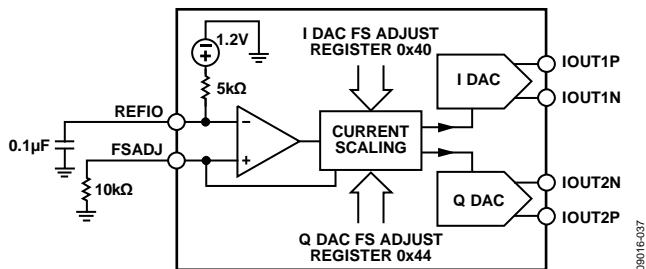
Figure 66. Simplified Block Diagram of DAC Core

The DAC has a 1.2 V band gap reference with an output impedance of 5 kΩ. The reference output voltage appears on the REFIO pin. When using the internal reference, the REFIO pin should be decoupled to AVSS with a 0.1 μF capacitor. Only use the internal reference for external circuits that draw dc currents of 2 μA or less. For dynamic loads or static loads greater than 2 μA, buffer the REFIO pin. If desired, an external reference (between 1.10 V and 1.30 V) can be applied to the REFIO pin. The internal reference can either be overdriven or powered down by setting Register 0x43, Bit 5.

A 10 kΩ external resistor, R_{SET}, must be connected from the FSADJ pin to AVSS. This resistor, along with the reference control amplifier, sets up the correct internal bias currents for the DAC. Because the full-scale current is inversely proportional to this resistor, the tolerance of R_{SET} is reflected in the full-scale output amplitude.

The full-scale current equation, where the DAC gain is set individually for the I and Q DACs in Register 0x40 and Register 0x44, respectively, follows:

$$I_{FS} = \frac{V_{REF}}{R_{SET}} \times \left(72 + \left(\frac{3}{16} \times DAC\ gain \right) \right)$$

For nominal values of V_{REF} (1.2 V), R_{SET} (10 kΩ), and DAC gain (512), the full-scale current of the DAC is typically 20.16 mA. The DAC full-scale current can be adjusted from 8.66 mA to 31.66 mA by setting the DAC gain code, as shown in Figure 67.
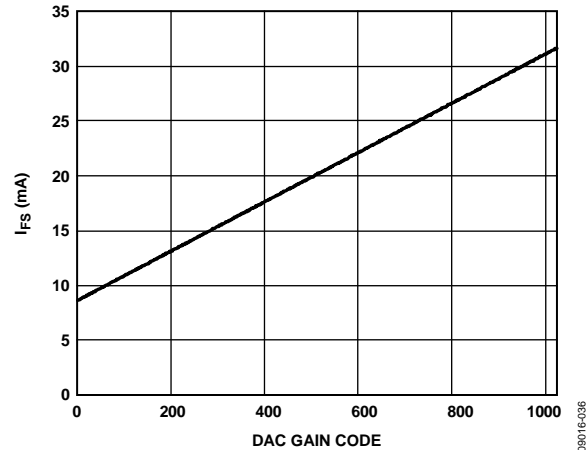

Figure 67. DAC Full-Scale Current vs. DAC Gain Code

Transmit DAC Transfer Function

The output currents from the IOUT1P/IOUT2P and IOUT1N/ IOUT2N pins are complementary, meaning that the sum of the two currents always equals the full-scale current of the DAC. The digital input code to the DAC determines the effective differential current delivered to the load. IOUT1P/IOUT2P provide maximum output current when all bits are high. The output currents vs. DACCODE for the DAC outputs are expressed as

$$I_{OUTP} = \left[\frac{DACCODE}{2^N} \right] \times I_{OUTFS} \tag{1}$$

$$I_{OUTN} = I_{OUTFS} - I_{OUTP} \tag{2}$$

where $DACCODE = 0$ to $2^N - 1$.

Transmit DAC Output Configurations

The optimum noise and distortion performance of the AD9125 is realized when it is configured for differential operation. The common-mode error sources of the DAC outputs are significantly reduced by the common-mode rejection of a transformer or differential amplifier. These common-mode error sources include even-order distortion products and noise. The enhancement in distortion performance becomes more significant as the frequency content of the reconstructed waveform increases and/or its amplitude increases. This is due to the first-order cancellation of various dynamic common-mode distortion mechanisms, digital feedthrough, and noise.

Figure 68 shows the most basic DAC output circuitry. A pair of resistors, R_O, is used to convert each of the complementary output currents to a differential voltage output, V_{OUT}. Because the current outputs of the DAC are high impedance, the differential driving point impedance of the DAC outputs, R_{OUT}, is equal to $2 \times R_O$. Figure 69 illustrates the output voltage waveforms.
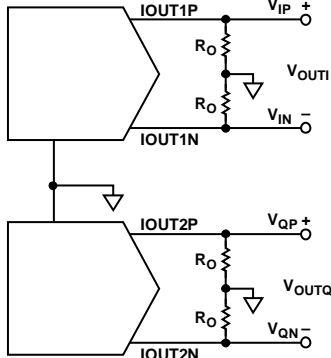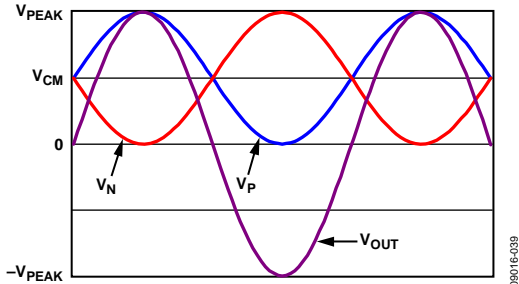

Figure 68. Basic Transmit DAC Output Circuit

Figure 69. Voltage Output Waveforms

The common-mode signal voltage, V_{CM}, is calculated as

$$V_{CM} = \frac{I_{FS}}{2} \times R_O$$

The peak output voltage, V_{PEAK}, is calculated as

$$V_{PEAK} = I_{FS} \times R_O$$

With this circuit configuration, the single-ended peak voltage is the same as the peak differential output voltage.

Transmit DAC Linear Output Signal Swing

To achieve optimum performance, the DAC outputs have a linear output compliance voltage range that must be adhered to. The linear output signal swing is dependent on the full-scale output current, I_{OUTFS}, and the common-mode level of the output. Figure 70 and Figure 71 show the IMD performance vs. the common-mode output voltage at various full-scale currents and output frequencies.
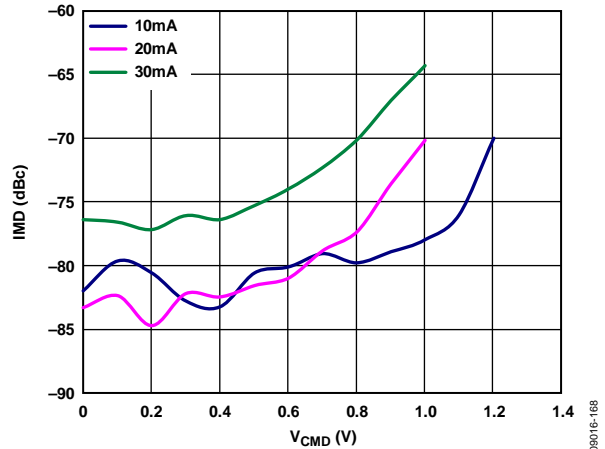

Figure 70. IMD vs. Common-Mode Output Voltage (f_{OUT} = 61 MHz, R_{LOAD} = 50 Ω differential, I_{FS} = 10 mA, 20 mA, and 30 mA)
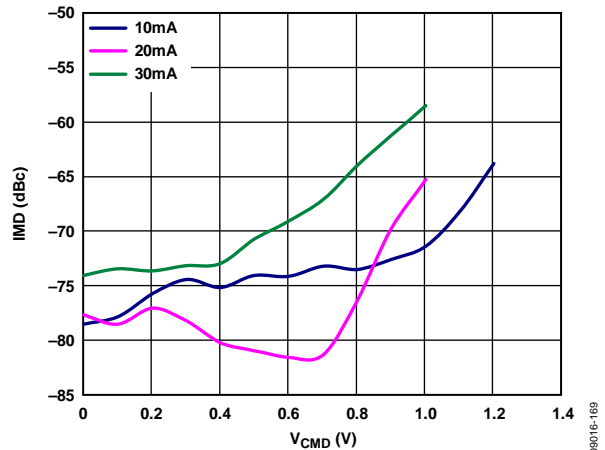

Figure 71. IMD vs. Common-Mode Output Voltage (f_{OUT} = 161 MHz, R_{LOAD} = 50 Ω differential, I_{FS} = 10 mA, 20 mA, and 30 mA)

AUXILIARY DAC OPERATION

The AD9125 has two auxiliary DACs; one is associated with the I path, and the other is associated with the Q path. These auxiliary DACs can be used to compensate for dc offsets in the transmitted signal. Each auxiliary DAC has a single-ended current that can sink or source current into either the P or N output of the associated transmit DAC. The auxiliary DAC structure is shown in Figure 72.
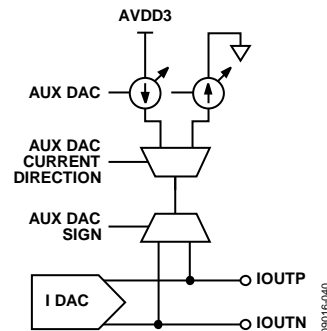

Figure 72. Auxiliary DAC Structure

The control registers for controlling the I and Q auxiliary DACs are in Register 0x42, Register 0x43, Register 0x46, and Register 0x47.

Interfacing to Modulators

The AD9125 interfaces to the ADL537x family of modulators with a minimal number of components. An example of the recommended interface circuitry is shown in Figure 73.

The baseband inputs of the ADL537x family require a dc bias of 500 mV. The nominal midscale output current on each output of the DAC is 10 mA (½ the full-scale current). Therefore, a single 50 Ω resistor to ground from each DAC output results in the desired 500 mV dc common-mode bias for the inputs to the ADL537x. The signal level can be reduced through the addition of the load resistor in parallel with the modulator inputs. The peak-to-peak voltage swing of the transmitted signal is

$$V_{SIGNAL} = I_{FS} \times \frac{[2 \times R_B \times R_L]}{[2 \times R_B + R_L]}$$
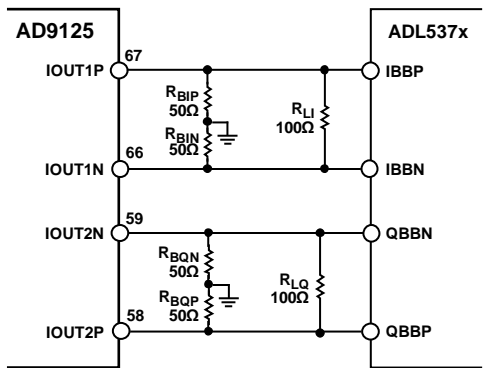

Figure 73. Typical Interface Circuitry Between the AD9125 and the ADL537x Family of Modulators

BASEBAND FILTER IMPLEMENTATION

Most applications require a baseband anti-imaging filter between the DAC and the modulator to filter out Nyquist images and broadband DAC noise. The filter can be inserted between the I-V resistors at the DAC output and the signal-level setting resistor across the modulator input. This configuration establishes the input and output impedances for the filter.

Figure 75 shows a fifth-order low-pass filter. A common-mode choke is used between the I-V resistors and the remainder of the filter. This removes the common-mode signal produced by the DAC and prevents the common-mode signal from being converted to a differential signal, which can appear as unwanted spurious signals in the output spectrum. Splitting the first filter capacitor into two and grounding the center point creates a common-mode low-pass filter, providing additional common-mode rejection of high frequency signals. A purely differential filter can pass common-mode signals.
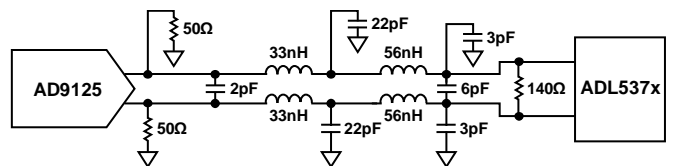
DRIVING THE ADL5375-15

The ADL5375-15 is the version of the ADL5375 that offers an input baseband bias levels of 1500 mV. Because the ADL5375-15 requires a 1500 mV dc bias, it requires a slightly more complex interface than most other Analog Devices, Inc., modulators. The DAC output must be level-shifted from a 500 mV dc bias to the 1500 mV dc bias. Level-shifting can be achieved with a purely passive network, as shown in Figure 74. In this network, the dc bias of the DAC remains at 500 mV while the input to the ADL5375-15 is 1500 mV. This passive level-shifting network introduces approximately 2 dB of loss in the ac signal.
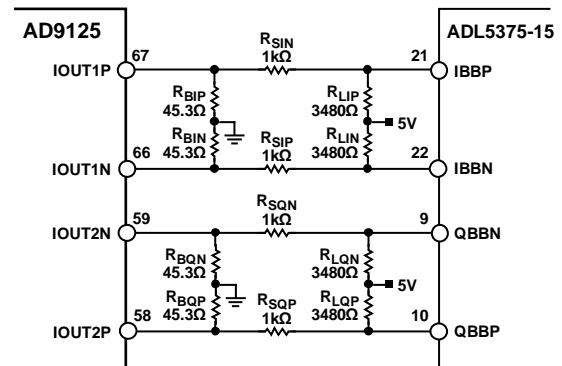

Figure 74. Passive Level-Shifting Network for Biasing ADL5375-15

REDUCING LO LEAKAGE AND UNWANTED SIDEBANDS

Analog quadrature modulators can introduce unwanted signals at the LO frequency due to dc offset voltages in the I and Q baseband inputs, as well as feedthrough paths from the LO input to the output. The LO feedthrough can be nulled by applying the correct dc offset voltages at the DAC output. This can be done using the auxiliary DACs (Register 0x42, Register 0x43, Register 0x46, and Register 0x47) or by using the digital dc offset adjustments (Register 0x3C through Register 0x3F).

The advantage of using the auxiliary DACs is that none of the main DAC dynamic range is used to perform the dc offset adjustment. However, the disadvantage is that the common-mode level of the output signal changes as a function of the auxiliary DAC current. The opposite is true when the digital offset adjustment is used.

Good sideband suppression requires both gain and phase matching of the I and Q signals. The I phase adjust (Register 0x38 and Register 0x39), Q phase adjust (Register 0x3A and Register 0x3B), I DAC FS adjust (Register 0x40 and Register 0x41), and Q DAC FS adjust (Register 0x44 and Register 0x45) registers can be used to calibrate I and Q transmit paths to optimize the sideband suppression.

Figure 75. DAC Modulator Interface with Fifth-Order, Low Pass Filter

DEVICE POWER DISSIPATION

The AD9125 has four supply rails: AVDD33, IOVDD, DVDD18, and CVDD18.

The AVDD33 supply powers the DAC core circuitry. The power dissipation of the AVDD33 supply rail is independent of the digital operating mode and sample rate. The current drawn from the AVDD33 supply rail is typically 57 mA (188 mW) when the full-scale current of the I and Q DACs is set to the nominal value of 20 mA. Changing the full-scale current directly impacts the supply current drawn from the AVDD33 rail. For example, if the full-scale current of the I DAC and the Q DAC is changed to 10 mA, the AVDD33 supply current drops by 20 mA to 37 mA.

The IOVDD voltage supplies the serial port I/O pins, the $\overline{RESET}$ pin, and the $\overline{IRQ}$ pin. The voltage applied to the IOVDD pin can range from 1.8 V to 3.3 V. The current drawn by the IOVDD supply pin is typically 3 mA.

The DVDD18 supply powers all of the digital signal processing blocks of the device. The power consumption from this supply is a function of which digital blocks are enabled and the frequency at which the device is operating.

The CVDD18 supply powers the clock receiver and clock distribution circuitry. The power consumption from this supply varies directly with the operating frequency of the device. CVDD18 also powers the PLL. The power dissipation of the PLL is typically 80 mA when enabled.

Figure 76 through Figure 80 detail the power dissipation of the AD9125 under a variety of operating conditions. All of the graphs are taken with data being supplied to both the I and Q channels. The power consumption of the device does not vary significantly with changes in the coarse modulation mode selected or analog output frequency. Graphs of the total power dissipation are shown along with the power dissipation of the DVDD18 and CVDD18 supplies.

Maximum power dissipation can be estimated to be 20% higher than the typical power dissipation.
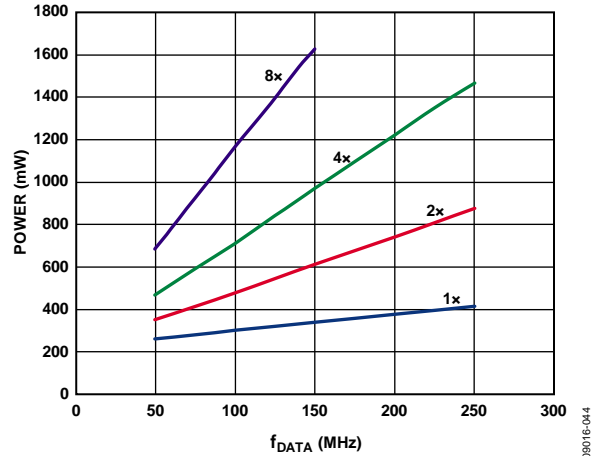

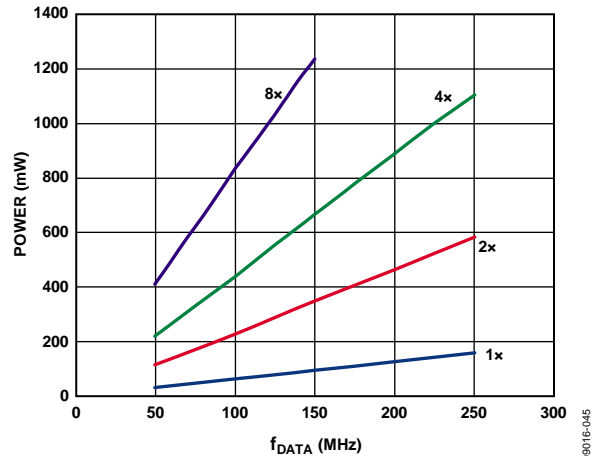Figure 76. Total Power Dissipation vs. f_{DATA} Without PLL, Fine NCO, and Inverse Sinc

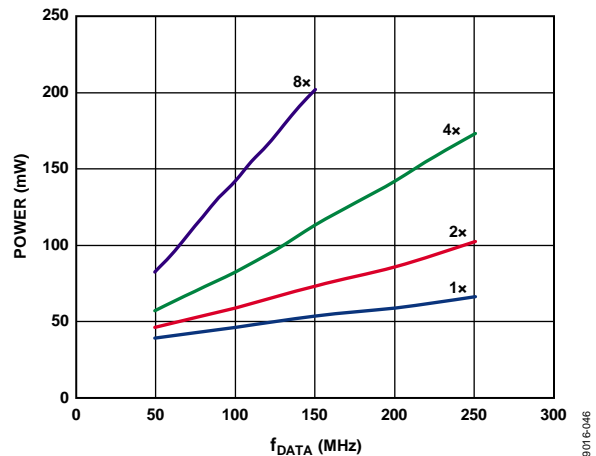Figure 77. DVDD18 Power Dissipation vs. f_{DATA} Without Fine NCO and Inverse Sinc

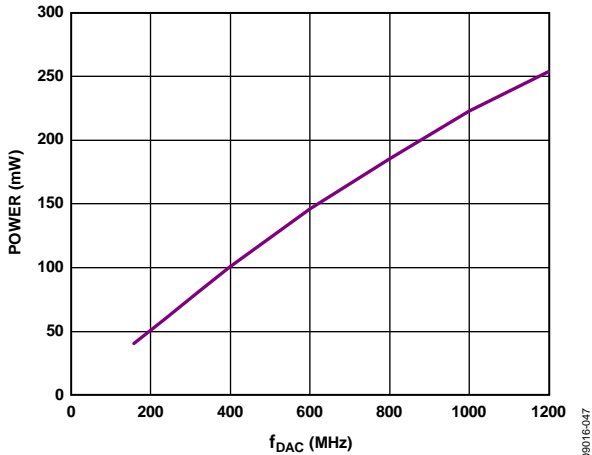Figure 78. CVDD18 Power Dissipation vs. f_{DATA} with PLL Disabled

Figure 79. DVDD18 Power Dissipation vs. f_{DAC} Due to Inverse Sinc Filter
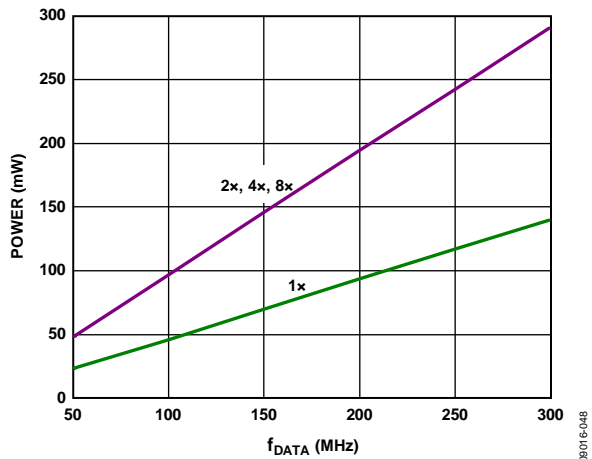

Figure 80. DVDD18 Power Dissipation vs. f_{DATA} Due to Fine NCO

TEMPERATURE SENSOR

The AD9125 has a diode-based temperature sensor for measuring the temperature of the die. The temperature reading is accessed through Register 0x49 and Register 0x4A. The temperature of the die can be calculated by

$$T_{DIE} = \frac{(Die\ Temp[15:0] - 47,925)}{88}$$

where T_{DIE} is the die temperature in °C. The temperature accuracy is ±5°C typical.

Estimates of the ambient temperature can be made if the power dissipation of the device is known. For example, if the device power dissipation is 800 mW and the measured die temperature is 50°C, then the ambient temperature can be calculated as

$$T_A = T_{DIE} - P_D \times T_{JA} = 50 - 0.8 \times 20.7 = 33.4°C$$

where:
T_A is the ambient temperature in °C.
T_{DIE} is the die temperature in °C.
P_D is the power dissipation.
T_{JA} is the thermal resistance from junction to ambient of the AD9125, as shown in Table 7.

To use the temperature sensor, it must be enabled by setting Register 0x01, Bit 4, to 0. In addition, to obtain accurate readings, the range control register (Register 0x48) should be set to 0x02.

MULTICHIP SYNCHRONIZATION

System demands may require that the outputs of multiple DACs be synchronized with each other or with a system clock. Systems that support transmit diversity or beam forming, where multiple antennas are used to transmit a correlated signal, require multiple DAC outputs to be phase aligned with each other. Systems with a time division multiplexing transmit chain may require one or more DACs to be synchronized with a system-level reference clock.

Multiple devices are considered synchronized to each other when the state of the clock generation state machine is identical for all parts and when time aligned data is being read from the FIFOs of all parts simultaneously. Devices are considered synchronized to a system clock when there is a constant, known relationship among the clock generation state machine, the data being read from the FIFO, and a particular clock edge of the system clock. The AD9125 has provisions for enabling multiple devices to be synchronized to each other or to a system clock.

The AD9125 supports synchronization in two modes: data rate mode and FIFO rate mode. Each of these modes has a different lowest rate clock that the synchronization logic attempts to synchronize to. In data rate mode, the input data rate represents the lowest synchronized clock. In FIFO rate mode, the FIFO rate, which is the data rate divided by the FIFO depth of 8, represents the lowest rate clock. The advantage of FIFO rate synchronization is increased time between keep-out windows for DCI changes relative to the DACCLK or REFCLK input.

When in data rate mode, the elasticity of the FIFO is not used to absorb timing variations between the data source and the DAC, resulting in keep-out widows repeating at the input data rate.

The method chosen for providing the DAC sampling clock directly impacts the synchronization methods available. When the device clock multiplier is used, only data rate mode is available. When the DAC sampling clock is sourced directly, both data rate mode and FIFO rate mode synchronization are available. This section details the synchronization methods for enabling both clocking modes and for querying the status of the synchronization logic.

SYNCHRONIZATION WITH CLOCK MULTIPLICATION

When using the clock multiplier to generate the DAC sample rate clock, the REFCLK input signal acts both as the reference clock for the PLL-based clock multiplier and as the synchronization signal. To synchronize devices, distribute the REFCLK signal with low skew to all of the devices that need to be synchronized. Skew between the REFCLK signals of the devices shows up directly as a timing mismatch at the DAC outputs.

The frequency of the REFCLK signal is typically equal to the input data rate. The FRAME and DCI signals, along with the data, can be created in the FPGA. A circuit diagram of a typical configuration is shown in Figure 81.
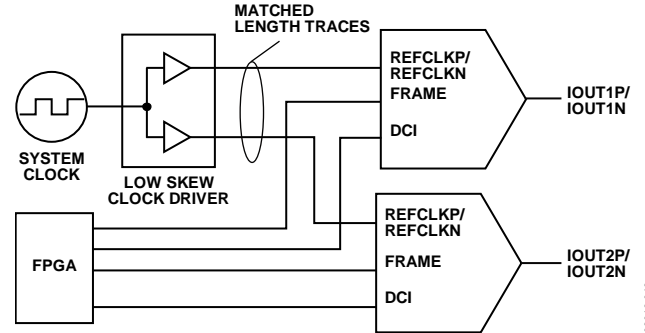

Figure 81. Typical Circuit Diagram for Synchronizing Devices

The Procedure for Synchronization when Using the PLL section outlines the steps required to synchronize multiple devices. The procedure assumes that the REFCLK signal is applied to all devices and that the PLL of each device is phase locked to this signal. This procedure must be carried out on each individual device.

Procedure for Synchronization when Using the PLL

To synchronize all devices,

1. Configure the device for data rate mode and periodic synchronization by writing 0xC0 to the Sync Control 1 register (Register 0x10). Additional synchronization options are available.
2. Read the Sync Status 1 register (Register 0x12) and verify that the sync locked bit (Bit 6) is set high, indicating that the device achieved back-end synchronization and that the sync lost bit (Bit 7) is low. These levels indicate that the clocks are running with a constant, known phase relative to the sync signal.
3. Reset the FIFO by strobing the FRAME signal high for the time required to write two complete input data-words. Resetting the FIFO ensures that the correct data is being read from the FIFO.

To maintain synchronization, the skew between the REFCLK signals of the devices must be less than t_{SKEW}. There are also setup and hold times to be observed among the DCI, the data of each device, and the REFCLK signal. When resetting the FIFO, the FRAME signal must be held high for the time interval required to write two complete input data-words. A timing diagram of the input signals is shown in Figure 82.

This example shows a REFCLK frequency equal to the data rate. Although this is the most common situation, it is not strictly required for proper synchronization. Any REFCLK frequency that satisfies the following equation is acceptable:

$$f_{SYNC\_I} = f_{DACCLK}/2^N \text{ and } f_{SYNC\_I} \leq f_{DATA}$$

where $N = 0, 1, 2,$ or 3.

As an example, a configuration with 4× interpolation and clock frequencies of $f_{VCO} = 1600$ MHz, $f_{DACCLK} = 800$ MHz, $f_{DATA} = 200$ MHz, and $f_{SYNC\_I} = 100$ MHz is a viable solution.
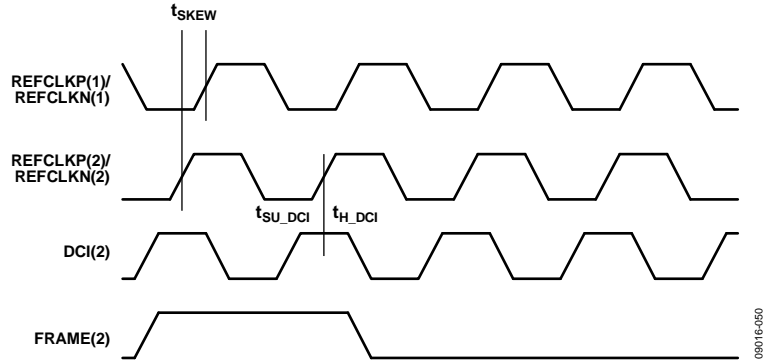
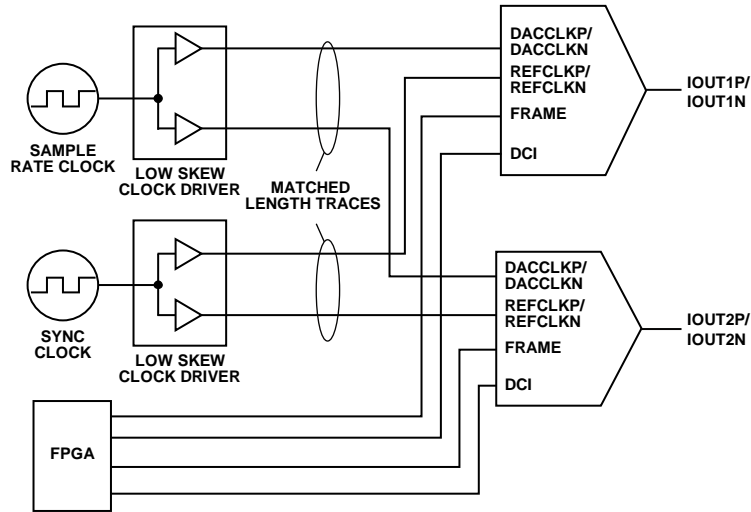Figure 82. Timing Diagram Required for Synchronizing Devices

Figure 83. Typical Circuit Diagram for Synchronizing Devices to a System Clock

SYNCHRONIZATION WITH DIRECT CLOCKING

When directly sourcing the DAC sample rate clock, a separate REFCLK input signal is required for synchronization. To synchronize devices, the DACCLK signal and the REFCLK signal must be distributed with low skew to all of the devices being synchronized. If the devices need to be synchronized to a master clock, then use the master clock directly for generating the REFCLK input (see Figure 83).

DATA RATE MODE SYNCHRONIZATION

The Procedure for Data Rate Synchronization when Directly Sourcing the DAC Sampling Clock section outlines the steps required to synchronize multiple devices in data rate mode. The procedure assumes that the DACCLK and REFCLK signals are applied to all of the devices. The procedure must be carried out on each individual device.

Procedure for Data Rate Synchronization When Directly Sourcing the DAC Sampling Clock

To synchronize all devices,

1. Configure the device for data rate mode and periodic synchronization by writing 0xC0 to the Sync Control 1 register (Register 0x10). Additional synchronization options are available and are described in the Additional Synchronization Features section.
2. Poll the sync locked bit (Register 0x12, Bit 6) to verify that the device is back-end synchronized. A high level on this bit indicates that the clocks are running with a constant, known phase relative to the sync signal.
3. Reset the FIFO by strobing the FRAME signal high for the time required to write two complete input data-words. Resetting the FIFO ensures that the correct data is being read from the FIFO of each of the devices simultaneously.

To ensure that each DAC is updated with the correct data on the same CLK edge, two timing relationships must be met on each DAC. DCI and D[31:0] must meet the setup and hold times with respect to the rising edge of DACCLK, and REFCLK must meet the setup and hold times with respect to the rising edge of DACCLK. When resetting the FIFO, the FRAME signal must be held high for the time required to input two complete input data-words. When these conditions are met, the outputs of the DACs are updated within $t_{SKEW} + t_{OUTDLY}$ of each other. A timing diagram that illustrates the timing requirements of the input signals is shown in Figure 84.

Figure 84 shows the synchronization signal timing with 2× interpolation; therefore, $f_{DCI} = \frac{1}{2} \times f_{CLK}$. The REFCLK input is shown to be equal to the data rate. The maximum frequency at which the device can be resynchronized in data rate mode can be expressed as

$$f_{SYNC\_I} = f_{DATA}/2^N$$

where N is any nonnegative integer.

Generally, for values of N equal to or greater than 3, select the FIFO rate synchronization mode.

Table 26. DCI-DAC Setup and Hold Times

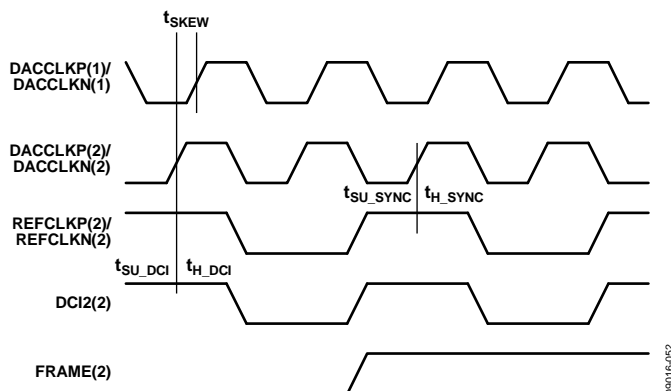| Minimum Setup Time, t_{SU_DCI} (ns)	Minimum Hold Time, t_{H_DCI} (ns)
0.16	0.59

Figure 84. Data Rate Synchronization Signal Timing Requirements, 2× Interpolation

FIFO RATE MODE SYNCHRONIZATION

The Procedure for FIFO Rate Synchronization when Directly Sourcing the DAC Sampling Clock section outlines the steps required to synchronize multiple devices in FIFO rate mode. The procedure assumes that the REFCLK and DACCLK signals are applied to all of the devices. The procedure must be carried out on each individual device.

Procedure for FIFO Rate Synchronization When Directly Sourcing the DAC Sampling Clock

To synchronize all devices,

1. Configure the device for FIFO rate mode and periodic synchronization by writing 0x80 to the Sync Control 1 register (Register 0x10). Additional synchronization options are available and are described in the Additional Synchronization Features section.
2. Poll the sync locked bit (Register 0x12, Bit 6) to verify that the device is back-end synchronized. A high level on this bit indicates that the clocks are running with a constant and known phase relative to the sync signal.
3. Reset the FIFO by strobing the FRAME signal high for the time required to write two complete input data-words. Resetting the FIFO ensures that the correct data is being read from the FIFO of each of the devices simultaneously.

To ensure that each DAC is updated with the correct data on the same CLK edge, two timing relationships must be met on each DAC. DCI and D[31:0] must meet the setup and hold times with respect to the rising edge of DACCLK, and REFCLK must meet the setup and hold times with respect to the rising edge of DACCLK. When resetting the FIFO, the FRAME signal must be held high for at least three data periods (that is, 1.5 cycles of DCI). When these conditions are met, the outputs of the DACs are updated within $t_{SKEW} + t_{OUTDLY}$ of each other. A timing diagram that illustrates the timing requirements of the input signals is shown in Figure 85.

Figure 85 shows the synchronization signal timing with 2× interpolation; therefore, $f_{DCI} = ½ × f_{CLK}$. The REFCLK input is shown to be equal to the FIFO rate. More generally, the maximum frequency at which the device can be resynchronized in FIFO rate mode can be expressed as

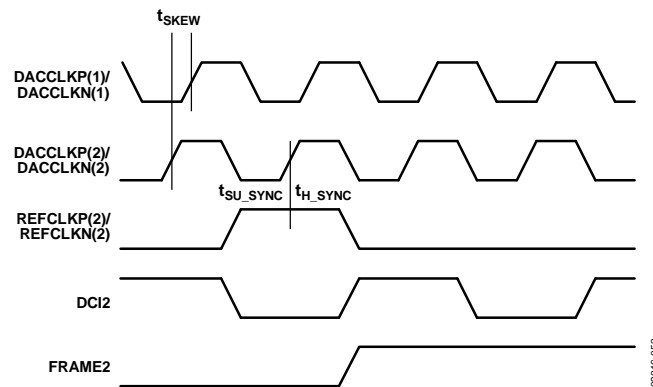$$f_{SYNC\_1} = f_{DATA}/(8 × 2^N)$$

where N is any nonnegative integer.

Figure 85. FIFO Rate Synchronization Signal Timing Requirements, 2× Interpolation

ADDITIONAL SYNCHRONIZATION FEATURES

The synchronization logic incorporates additional features that provide means for querying the status of the synchronization, improving the robustness of the synchronization, and enabling a one-shot synchronization mode. These features are detailed in the Sync Status Bits and Timing Optimization sections.

Sync Status Bits

When the sync locked bit (Register 0x12, Bit 6) is set, it indicates that the synchronization logic has reached alignment. This alignment is determined when the clock generation state machine phase is constant. It takes between $(11 + \text{averaging}) \times 64$ and $(11 + \text{averaging}) \times 128$ DACCLK cycles. This bit can optionally trigger an $\overline{\text{IRQ}}$, as described in the Interrupt Request Operation section.

When the sync lost bit (Register 0x12, Bit 7) is set, it indicates a previously synchronized device has lost alignment. This bit is latched and remains set until cleared by overwriting the register. This bit can optionally trigger an $\overline{\text{IRQ}}$, as described in the Interrupt Request Operation section.

The sync phase readback bits (Register 0x13, Bits[7:0]) report the current clock phase in a 6.2 format. Bits[7:2] report which of the 64 states (0 to 63) the clock is currently in. When averaging is enabled, Bits[1:0] provide ¼ state accuracy (for 0, ¼, ½, ¾). The lower two bits give an indication of the timing margin issues that may exist. If the sync sampling is error free, the fractional clock state should be 00.

Timing Optimization

The REFCLK signal is sampled by a version of the DACCLK. If sampling errors are being detected, the opposite sampling edge can be selected to improve the sampling point. The sampling edge can be selected by setting Register 0x10, Bit 3 (1 = rising and 0 = falling).

The synchronization logic resynchronizes when a phase change between the REFCLK signal and the state of the clock generation state machine exceeds a threshold. To mitigate the effects of jitter and prevent erroneous resynchronizations, the relative phase can be averaged. The amount of averaging is set by the sync averaging bits (Register 0x10, Bits[2:0]) and can be set from 1 to 128. The higher the number of averages, the more slowly the device recognizes and resynchronizes to a legitimate phase correction. Generally, the averaging should be made as large as possible while still meeting the allotted resynchronization time interval.

The value of the sync phase request bits (Register 0x11, Bits[5:0]) is the state to which the clock generation state machine resets upon initialization. By varying this value, the timing of the internal clocks with respect to the REFCLK signal can be adjusted. Every increment in the value of the sync phase request bits (Register 0x11, Bits[5:0]) advances the internal clocks by one DACCLK period. This offset can be used for two purposes: to skew the outputs of two synchronized DAC outputs in increments of the DACCLK period and to change the relative timing between the DCI input and REFCLK. This may allow for more optimal placement of the DCI sampling point in data rate synchronization mode.

Table 27. Synchronization Setup and Hold Times

Parameter	Min	Max	Unit
t_{SKEW}	$-t_{DACCLK}/2$	$+t_{DACCLK}/2$	ps
t_{SV_SYNC}	100		ps
T_{H_SYNC}	330		ps

INTERRUPT REQUEST OPERATION

The AD9125 provides an interrupt request output signal (on Pin 7, $\overline{IRQ}$) that can be used to notify an external host processor of significant device events. Upon assertion of the interrupt, the device should be queried to determine the precise event that occurred. The $\overline{IRQ}$ pin is an open-drain, active low output. Pull the $\overline{IRQ}$ pin high external to the device. This pin can be tied to the interrupt pins of other devices with open-drain outputs to wire-OR these pins together.

Sixteen event flags provide visibility into the device. These 16 flags are located in the two event flag registers (Register 0x06 and Register 0x07). The behavior of each of the event flags is independently selected in the interrupt enable registers (Register 0x04 and Register 0x05). When the flag interrupt enable is active, the event flag latches and triggers an external interrupt. When the flag interrupt is disabled, the event flag simply monitors the source signal and the external $\overline{IRQ}$ remains inactive.

Figure 86 shows the $\overline{IRQ}$-related circuitry. This diagram shows how event flag signals propagate to the $\overline{IRQ}$ output. The INTERRUPT_ENABLE signal represents one bit from the interrupt enable registers. The EVENT_FLAG_SOURCE signal represents one bit from the event flag registers. The EVENT_FLAG_SOURCE signal represents one of the device signals that can be monitored, such as the PLL_LOCKED signal from the PLL phase detector or the FIFO_WARNING_1 signal from the FIFO controller.

When an interrupt enable bit is set high, the corresponding event flag bit reflects a positively tripped signal (that is, latched on the rising edge of the EVENT_FLAG_SOURCE signal). This signal

also asserts the external $\overline{IRQ}$. When an interrupt enable bit is set low, the event flag bit reflects the current status of the EVENT_FLAG_SOURCE signal, but the event flag has no effect on the external $\overline{IRQ}$.

The latched version of an event flag (the INTERRUPT_SOURCE signal) can be cleared in two ways. The recommended way is by writing 1 to the corresponding event flag bit; however, a hardware or software reset can also clear the INTERRUPT_SOURCE.

INTERRUPT SERVICE ROUTINE

The interrupt request management starts by selecting the set of event flags that require host intervention or monitoring. Those events that require host action should be enabled so that the host is notified when they occur. For events requiring host intervention, upon $\overline{IRQ}$ activation, run the following routine to clear an interrupt request:

1. Read the status of the event flag bits that are being monitored.
2. Set the interrupt enable bit low so that the unlatched EVENT_FLAG_SOURCE signal can be monitored directly.
3. Perform any actions that are required to clear the EVENT_SOURCE_FLAG signal. In many cases, no specific actions are required.
4. Read the event flag to verify that the EVENT_FLAG_SOURCE signal has been cleared.
5. Clear the interrupt by writing 1 to the event flag bit.
6. Set the interrupt enable bits of the events to be monitored.

Note that some of the EVENT_FLAG_SOURCE signals are latched signals. These are cleared by writing to the corresponding event flag bit. Details of each of the event flags can be found in Table 11.
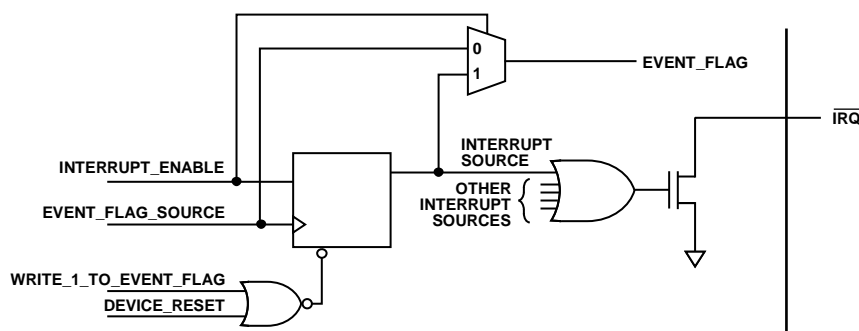

Figure 86. Simplified Schematic of $\overline{IRQ}$ Circuitry

INTERFACE TIMING VALIDATION

The AD9125 provides on-chip sample error detection (SED) circuitry that simplifies verification of the input data interface. The SED compares the input data samples captured at the digital input pins with a set of comparison values, which are loaded into registers through the SPI port. Differences between these values are detected and stored. Options are available for customizing SED test sequencing and error handling.

SED OPERATION

The SED circuitry operates on a data set made up of four 16-bit input words, denoted as I0, Q0, I1, and Q1. To properly align the input samples, the first I and Q data-words (that is, I0 and Q0) are indicated by asserting the FRAME signal for a minimum of two complete input samples.

Figure 87 shows the input timing of the interface in dual-word mode. The FRAME signal can be issued once at the start of the data transmission, or it can be asserted repeatedly at intervals coinciding with the I0 and Q0 data-words.

Figure 87. Timing Diagram for Dual-Word Mode SED Operation

In word mode, the FRAME signal required to align the data samples needs to be extended. The FRAME signal can be issued once at the start of the data transmission, or it can be asserted repeatedly at intervals coinciding with the I0 and Q0 data-words.
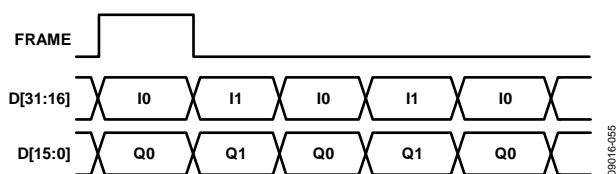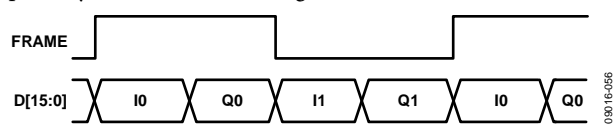

Figure 88. Timing Diagram for Two-Port Mode SED Operation

The SED has three flag bits (Register 0x67, Bit 0, Bit 1, and Bit 5) that indicate the results of the input sample comparisons. The sample error detected bit (Register 0x67, Bit 5) is set when an error is detected and remains set until the bit is cleared. The SED also provides registers that indicate which input data bits experienced errors (Register 0x70 through Register 0x73). These bits are latched and indicate the accumulated errors detected until cleared.

The autoclear mode has two effects: it activates the compare fail bit and the compare pass bit (Register 0x67, Bit 1 and Bit 0) and

changes the behavior of Register 0x70 through Register 0x73. The compare pass bit is set if the last comparison indicated that the sample was error free. The compare fail bit is set if an error is detected. The compare fail bit is automatically cleared by the reception of eight consecutive error-free comparisons. When autoclear mode is enabled, Register 0x70 through Register 0x73 accumulate errors as previously described but reset to all 0s after eight consecutive error-free sample comparisons are made.

The sample error, compare pass, and compare fail flags can be configured to trigger an $\overline{IRQ}$ when active, if desired. This is done by enabling the appropriate bits in the Event Flag 2 register (Register 0x07). Table 28 shows a progression of the input sample comparison results and the corresponding states of the error flags.

SED EXAMPLE

Normal Operation

The following example illustrates the SED configuration for continuously monitoring the input data and assertion of an $\overline{IRQ}$ when a single error is detected.

1. Write to the following registers to enable the SED and load the comparison values.
   Register 0x67: 0x80
   Register 0x68: I0[7:0]
   Register 0x69: I0[15:8]
   Register 0x6A: Q0[7:0]
   Register 0x6B: Q0[15:8]
   Register 0x6C: I1[7:0]
   Register 0x6D: I1[15:8]
   Register 0x6E: Q1[7:0]
   Register 0x6F: Q1[15:8]
   Comparison values can be chosen arbitrarily; however, choosing values that require frequent bit toggling provides the most robust test.
2. Enable the SED error detect flag to assert the $\overline{IRQ}$ pin by writing 0x04 to Register 0x05.
3. Begin transmitting the input data pattern.

If $\overline{IRQ}$ is asserted, read Register 0x67 and Register 0x70 through Register 0x73 to verify that a SED error was detected and to determine which input bits were in error. The bits in Register 0x70 through Register 0x73 are latched; therefore, the bits indicate any errors that occurred on those bits throughout the test, not just the errors that caused the error detected flag to be set.

Table 28. Progression of Comparison Outcomes and the Resulting SED Register Values

Compare Results (Pass/Fail)	P	F	F	F	P	P	P	P	P	P	P	P	P	F	P	F
Register 0x67, Bit 5 (Sample Error Detected)	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Register 0x67, Bit 1 (Compare Fail)	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
Register 0x67, Bit 0 (Compare Pass)	1	0	0	0	1	1	1	1	1	1	1	1	1	0	1	0
Register 0x70 to Register 0x73 (Errors Detected x_BITS[15:0])	Z[1]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	N[2]	Z[1]	N[2]	N[2]	N[2]

[1] Z = all 0s.
[2] N = nonzero.

EXAMPLE START-UP ROUTINE

There are certain sequences that should be followed to ensure reliable startup of the AD9125.

The example start-up routine assumes the following device configuration:

- f_{DATA} = 122.88 MSPS
- Interpolation = 4×, using HB1 = 10 and HB2 = 010010
- Input data = baseband data
- f_{OUT} = 140 MHz
- f_{REFCLK} = 122.88 MHz
- PLL = enabled
- Fine NCO = enabled
- Inverse SINC filter = enabled
- Synchronization = enabled

The following PLL settings can be derived from the device configuration:

- $f_{DACCLK} = f_{DATA} \times Interpolation$ = 491.52 MHz
- $f_{VCO} = 4 \times f_{DACCLK}$ = 1966.08 MHz (1 GHz < f_{VCO} < 2.1 GHz)
- $N1 = f_{DACCLK}/f_{REFCLK}$ = 4
- $N2 = f_{VCO}/f_{DACCLK}$ = 4

The following NCO settings can be derived from the device configuration:

- $f_{NCO} = 2 \times f_{DATA}$
- $f_{CARRIER} = f_{OUT} - f_{MODHB1}$ = 140 − 122.88 = 17.12 MHz
- $FTW = 17.12/(2 \times 122.88) \times 2^{32}$ = 0x11D55555

Start-Up Sequence

The following procedure sets the power clock and register write sequencing for reliable device start-up:

1. Power up the device (no specific power supply sequence is required).
2. Apply stable REFCLK input signal.
3. Apply stable DCI input signal.
4. Issue a hardware reset (optional).

As a result, the device configuration register write sequence is

```
0x00 → 0x20 /* Issue software reset */
0x00 → 0x00

0x0C → 0xD1 /* Start PLL */
0x0D → 0xD9
0x0A → 0xC0
0x0A → 0x80
        /* ??Verify PLL is locked?? */
Read 0x0E, expect Bit 7 = 1, Bit 6 = 0
Read 0x06, expect 0x5C

0x10 → 0x48 /* Choose data rate mode */
0x17 → 0x04 /* Issue software FIFO reset */
0x18 → 0x02
0x18 → 0x00
        /* ??Verify FIFO reset?? */
Read 0x18, expect 0x05
Read 0x19, expect 0x07

0x1B → 0x84 /* Configure interpolation
filters */
0x1C → 0x04
0x1D → 0x24

0x1E → 0x01 /* Configure NCO */
0x30 → 0x55
0x31 → 0x55
0x32 → 0xD5
0x33 → 0x11
0x36 → 0x01 /* Update frequency tuning
word */
0x36 → 0x00
```
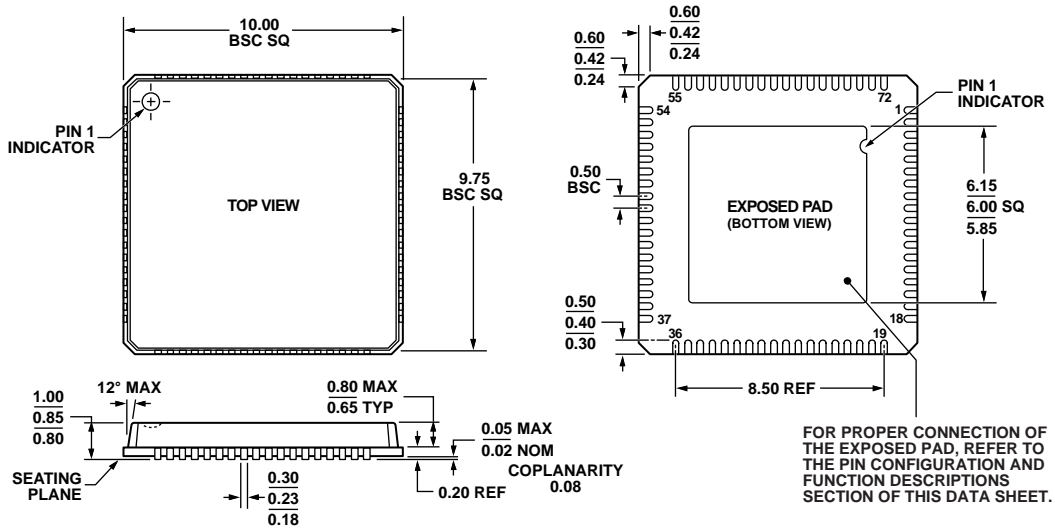
OUTLINE DIMENSIONS

COMPLIANT TO JEDEC STANDARDS MO-220-VNND-4

Figure 89. 72-Lead Lead Frame Chip Scale Package [LFCSP_VQ]
10 mm × 10 mm Body, Very Thin Quad
(CP-72-7)
Dimensions shown in millimeters

ORDERING GUIDE

Model[1]	Temperature Range	Package Description	Package Option
AD9125BCPZ	−40°C to +85°C	72-lead LFCSP_VQ	CP-72-7
AD9125BCPZRL	−40°C to +85°C	72-lead LFCSP_VQ	CP-72-7
AD9125-M5372-EBZ		Evaluation Board Connected to ADL5372 Modulator	
AD9125-M5375-EBZ		Evaluation Board Connected to ADL5375 Modulator	

[1] Z = RoHS Compliant Part.

NOTES

**ANALOG
DEVICES**

www.analog.com