Dual, 16-Bit, 1230 MSPS, TxDAC+ Digital-to-Analog Converter

Data Sheet **AD9146**

FEATURES

- Flexible LVDS interface allows byte or nibble 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
- Integrated 2×/4× interpolator/complex modulator allows carrier placement anywhere in the DAC bandwidth
- Gain, dc offset, and phase adjustment for sideband suppression
- Multiple chip synchronization interfaces
- High performance, low noise PLL clock multiplier
- Digital inverse sinc filter
- Low power: 1.2 W at 1.0 GSPS, 800 mW at 500 MSPS, full operating conditions
- 48-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

GENERAL DESCRIPTION

The AD9146 is a dual, 16-bit, high dynamic range digital-to-analog converter (DAC) that provides a sample rate of 1000 MSPS with nominal supplies and 1230 MSPS with increased supplies, permitting multicarrier generation up to the Nyquist frequency.

The AD9146 TxDAC+® 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 3-wire serial port interface provides for 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 AD9146 comes in a 48-lead LFCSP.

PRODUCT HIGHLIGHTS

1. Ultralow noise and intermodulation distortion (IMD) enable high quality synthesis of wideband signals from baseband to high intermediate frequencies (IF).
2. Proprietary DAC output switching technique enhances dynamic performance.
3. Current outputs are easily configured for various single-ended or differential circuit topologies.
4. Compact LVDS digital interface offers reduced width data bus.

COMPANION PRODUCTS

IQ Modulators: ADL5370, ADL537x family
IQ Modulators with PLL and VCO: ADRF6701, ADRF670x family
Clock Drivers: AD9516, AD951x family
Voltage Regulator Design Tool: ADIsimPower
Additional companion products on the AD9146 product page

TYPICAL SIGNAL CHAIN

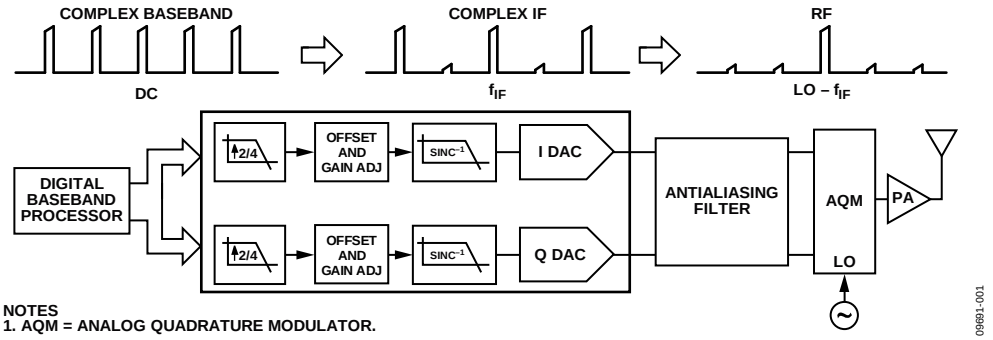

Figure 1.

Rev. A

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

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 www.analog.com
Fax: 781.461.3113 ©2011–2012 Analog Devices, Inc. All rights reserved.

TABLE OF CONTENTS

Features ... 1
Applications ... 1
General Description ... 1
Product Highlights ... 1
Companion Products ... 1
Typical Signal Chain ... 1
Revision History ... 3
Functional Block Diagram ... 4
Specifications ... 5
DC Specifications ... 5
Digital Specifications ... 6
Digital Input Data Timing Specifications ... 6
AC Specifications ... 7
Absolute Maximum Ratings ... 8
Thermal Resistance ... 8
ESD Caution ... 8
Pin Configuration and Function Descriptions ... 9
Typical Performance Characteristics ... 11
Terminology ... 15
Theory of Operation ... 16
Serial Port Operation ... 16
Data Format ... 16
Serial Port Pin Descriptions ... 16
Serial Port Options ... 17
Device Configuration Register Map and Descriptions ... 18
LVDS Input Data Ports ... 28
Byte Interface Mode ... 28
Nibble Interface Mode ... 28
FIFO Operation ... 28
Interface Timing ... 31
Digital Datapath ... 32
Premodulation ... 32
Interpolation Filters ... 32
Datapath Configuration ... 34
Determining Interpolation Filter Modes ... 34
Coarse Modulation Mixing Sequences ... 35
Quadrature Phase Correction ... 35
DC Offset Correction ... 35
Inverse Sinc Filter ... 36
DAC Input Clock Configurations ... 37
Driving the DACCLK and REFCLK Inputs ... 37
Direct Clocking ... 37
Clock Multiplication ... 37
PLL Settings ... 38
Configuring the VCO Tuning Band ... 38
Analog Outputs ... 39
Transmit DAC Operation ... 39
Auxiliary DAC Operation ... 40
Interfacing to Modulators ... 41
Baseband Filter Implementation ... 41
Driving the ADL5375-15 ... 41
Reducing LO Leakage and Unwanted Sidebands ... 42
Device Power Management ... 43
Power Dissipation ... 43
Tx Enable ... 43
Temperature Sensor ... 44
Multichip Synchronization ... 45
Synchronization with Clock Multiplication ... 45
Synchronization with Direct Clocking ... 46
Data Rate Mode Synchronization ... 46
FIFO Rate Mode Synchronization ... 47
Additional Synchronization Features ... 48
Interrupt Request Operation ... 49
Interrupt Service Routine ... 49
Interface Timing Validation ... 50
SED Operation ... 50
SED Example ... 51
Example Start-Up Routine ... 52
Device Configuration ... 52
Derived PLL Settings ... 52
Start-Up Sequence ... 52
Outline Dimensions ... 53
Ordering Guide ... 53

REVISION HISTORY

1/12—Rev. 0 to Rev. A

Change to General Description Section 1
Change to DCI Delay[1:0], Table 11 23
Changes to Interface Timing Section, Figure 37, and Figure 38 31
Changes to SED Operation Section and Table 26 50

4/11—Revision 0: Initial Version

FUNCTIONAL BLOCK DIAGRAM

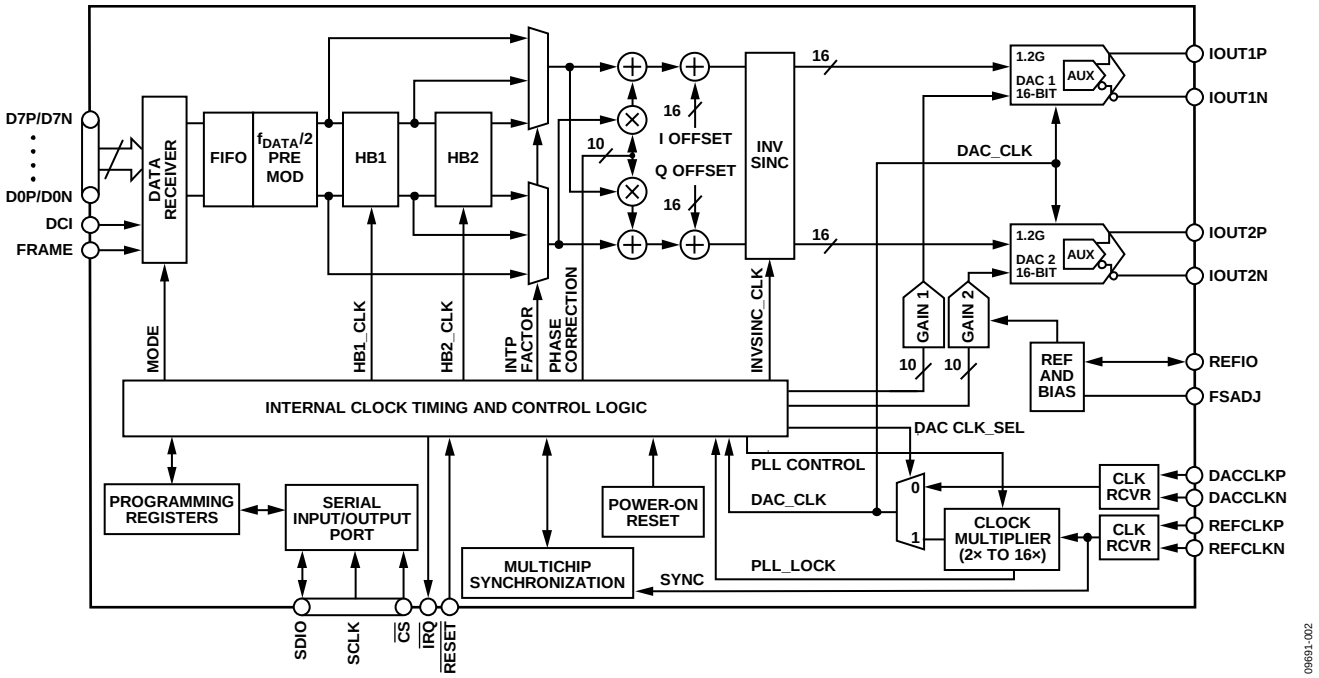

Figure 2.

SPECIFICATIONS

DC SPECIFICATIONS

T_{MIN} to T_{MAX}, AVDD33 = 3.3 V, DVDD18 = 1.8 V, CVDD18 = 1.8 V, I_{FS} = 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
Power Supply Rejection Ratio, AVDD33	−0.3		+0.3	% FSR/V
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
POWER CONSUMPTION				
2× Mode, f_{DAC} = 500 MSPS, IF = 10 MHz				
PLL Off		780		mW
PLL On		864		mW
AVDD33			56	mA
CVDD18			58	mA
DVDD18			343	mA
Power-Down Mode (Register 0x01 = 0xFC)		8.5	19	mW
POWER-UP TIME		260		ms
OPERATING RANGE	−40	+25	+85	°C

[1] Based on a 10 kΩ external resistor between FSADJ and AVSS.

DIGITAL SPECIFICATIONS

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

Table 2.

Parameter	Test Conditions/Comments	Min	Typ	Max	Unit
CMOS INPUT LOGIC LEVEL					
Input V_{IN} Logic High		1.2			V
Input V_{IN} Logic Low				0.6	V
CMOS OUTPUT LOGIC LEVEL					
Output V_{OUT} Logic High		1.4			V
Output V_{OUT} Logic Low				0.4	V
LVDS RECEIVER INPUTS[1]	Applies to data, DCI, and FRAME inputs				
Input Voltage Range, V_{IA} or V_{IB}		825		1575	mV
Input Differential Threshold, V_{IDTH}		−100		+100	mV
Input Differential Hysteresis, V_{IDTHH} to V_{IDTHL}			20		mV
Receiver Differential Input Impedance, R_{IN}		80		120	Ω
LVDS Input Rate	See Table 5				
DAC CLOCK INPUT (DACCLKP, DACCLKN)					
Differential Peak-to-Peak Voltage		100	500	2000	mV
Common-Mode Voltage	Self-biased input, ac-coupled		1.25		V
Maximum Clock Rate		1200			MHz
REFCLK INPUT (REFCLKP, REFCLKN)					
Differential Peak-to-Peak Voltage		100	500	2000	mV
Common-Mode Voltage			1.25		V
REFCLK Frequency					
PLL Mode	1 GHz ≤ f_{VCO} ≤ 2.1 GHz	15.625		525	MHz
SYNC Mode	See the Multichip Synchronization section for conditions	0		525	MHz
SERIAL PORT INTERFACE					
Maximum Clock Rate (SCLK)		40			MHz
Minimum Pulse Width High (t_{PWH})				12.5	ns
Minimum Pulse Width Low (t_{PWL})				12.5	ns
Setup Time, SDIO to SCLK (t_{DS})		2.09			ns
Hold Time, SDIO to SCLK (t_{DH})		0.844			ns
Data Valid, SDIO to SCLK (t_{DV})		2.904			ns
Setup Time, $\overline{CS}$ to SCLK (t_{DCSB})			2.38		ns

[1] LVDS receiver is compliant with the IEEE 1596 reduced range link, unless otherwise noted.

DIGITAL INPUT DATA TIMING SPECIFICATIONS

Table 3.

Parameter	Value	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
Inverse Sinc	20	Cycles

AC SPECIFICATIONS

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

Table 4.

Parameter	Min	Typ	Max	Unit
SPURIOUS-FREE DYNAMIC RANGE (SFDR)				
f_{DAC} = 200 MSPS, f_{OUT} = 50 MHz		70		dBc
f_{DAC} = 400 MSPS, f_{OUT} = 70 MHz		65		dBc
f_{DAC} = 800 MSPS, f_{OUT} = 70 MHz		67		dBc
TWO-TONE INTERMODULATION DISTORTION (IMD)				
f_{DAC} = 600 MSPS, f_{OUT} = 50 MHz		85		dBc
f_{DAC} = 600 MSPS, f_{OUT} = 80 MHz		82		dBc
f_{DAC} = 800 MSPS, f_{OUT} = 60 MHz		83		dBc
f_{DAC} = 800 MSPS, f_{OUT} = 100 MHz		81		dBc
NOISE SPECTRAL DENSITY (NSD), SINGLE-CARRIER W-CDMA				
f_{DAC} = 400 MSPS, f_{OUT} = 80 MHz		−162		dBm/Hz
f_{DAC} = 800 MSPS, f_{OUT} = 80 MHz		−164		dBm/Hz
W-CDMA ADJACENT CHANNEL LEAKAGE RATIO (ACLR), FOUR-CARRIER				
f_{DAC} = 491.52 MSPS, f_{OUT} = 15 MHz		75		dBc
f_{DAC} = 983.04 MSPS, f_{OUT} = 80 MHz		77		dBc
f_{DAC} = 983.04 MSPS, f_{OUT} = 200 MHz		76		dBc
W-CDMA ADJACENT CHANNEL LEAKAGE RATIO (ACLR), SINGLE-CARRIER				
f_{DAC} = 983.04 MSPS, f_{OUT} = 80 MHz		82		dBc
f_{DAC} = 983.04 MSPS, f_{OUT} = 122.88 MHz		80		dBc

Table 5. Maximum Rate (MSPS) with DVDD and CVDD Supply Regulation

Interface Mode	Interpolation Factor	$f_{INTERFACE}$ (MSPS)		f_{HB1} (MSPS)		f_{HB2} (MSPS)		f_{DAC} (MSPS)	
		1.8 V ± 5%	1.9 V ± 5%	1.8 V ± 5%	1.9 V ± 5%	1.8 V ± 5%	1.9 V ± 5%	1.8 V ± 5%	1.9 V ± 5%
Byte (8 Bits)	1×	1200	1230					300	307.5
	2× (HB1)	1200	1230	300	307.5			600	615
	2× (HB2)	1200	1230			300	307.5	600	615
	4×	1200	1230	300	307.5	600	615	1200	1230
Nibble (4 Bits)	1×	1200	1230					150	153.75
	2× (HB1)	1200	1230	150	153.75			300	307.5
	2× (HB2)	1200	1230			150	153.75	300	307.5
	4×	1200	1230	150	153.75	300	307.5	600	615

ABSOLUTE MAXIMUM RATINGS

Table 6.

Parameter	Rating
AVDD33 to AVSS, EPAD, CVSS	−0.3 V to +3.6 V
DVDD18, CVDD18 to AVSS, EPAD, CVSS	−0.3 V to +2.1 V
AVSS to EPAD, CVSS	−0.3 V to +0.3 V
EPAD to AVSS, CVSS	−0.3 V to +0.3 V
CVSS to AVSS, EPAD	−0.3 V to +0.3 V
FSADJ, REFIO, IOUT1P, IOUT1N, IOUT2P, IOUT2N to AVSS	−0.3 V to AVDD33 + 0.3 V
D[7:0]P, D[7:0]N, FRAMEP, FRAMEN, DCIP, DCIN to EPAD	−0.3 V to DVDD18 + 0.3 V
DACCLKP, DACCLKN, REFCLKP, REFCLKN to EPAD	−0.3 V to CVDD18 + 0.3 V
$\overline{\text{RESET}}$, $\overline{\text{IRQ}}$, $\overline{\text{CS}}$, SCLK, SDIO to EPAD	−0.3 V to DVDD18 + 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 pad (EPAD) of the 48-lead LFCSP must be soldered to the ground plane (AVSS). The EPAD provides an electrical, thermal, and mechanical connection to the board.

Typical θ_{JA}, θ_{JB}, and θ_{JC} values are specified for a 4-layer board and an 8-layer board in still air. Airflow increases heat dissipation, effectively reducing θ_{JA} and θ_{JB}.

Table 7. Thermal Resistance

Package[1]	PCB	θ_{JA}	θ_{JB}	θ_{JC}	Unit
48-Lead LFCSP	4-layer	23.2	8.5	8.7	°C/W
48-Lead LFCSP	8-layer	16.4	3.9	7.1	°C/W

[1] EPAD soldered to ground plane.

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

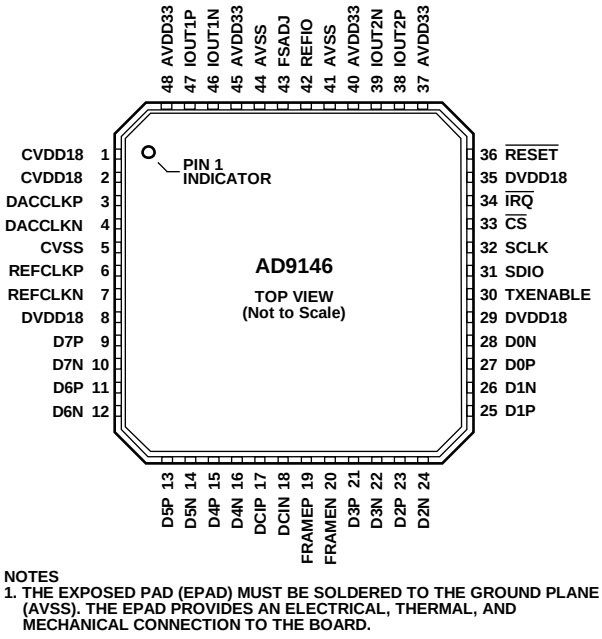

Figure 3. Pin Configuration

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	CVDD18	1.8 V Clock Supply. Supplies clock receivers, clock distribution, and PLL circuitry.
3	DACCLKP	DAC Clock Input, Positive.
4	DACCLKN	DAC Clock Input, Negative.
5	CVSS	Clock Supply Common.
6	REFCLKP	PLL Reference Clock Input, Positive. This pin has a secondary function as a synchronization input.
7	REFCLKN	PLL Reference Clock Input, Negative. This pin has a secondary function as a synchronization input.
8	DVDD18	1.8 V Digital Supply. Supplies power to digital core and digital data ports.
9	D7P	Data Bit 7 (MSB), Positive.
10	D7N	Data Bit 7 (MSB), Negative.
11	D6P	Data Bit 6, Positive.
12	D6N	Data Bit 6, Negative.
13	D5P	Data Bit 5, Positive.
14	D5N	Data Bit 5, Negative.
15	D4P	Data Bit 4, Positive.
16	D4N	Data Bit 4, Negative.
17	DCIP	Data Clock Input, Positive.
18	DCIN	Data Clock Input, Negative.
19	FRAMEP	Frame Input, Positive.
20	FRAMEN	Frame Input, Negative.
21	D3P	Data Bit 3, Positive.
22	D3N	Data Bit 3, Negative.
23	D2P	Data Bit 2, Positive.
24	D2N	Data Bit 2, Negative.
25	D1P	Data Bit 1, Positive.
26	D1N	Data Bit 1, Negative.
27	D0P	Data Bit 0 (LSB), Positive.

Pin No.	Mnemonic	Description
28	D0N	Data Bit 0 (LSB), Negative.
29	DVDD18	1.8 V Digital Supply. Supplies power to digital core and digital data ports.
30	TXENABLE	Active High Transmit Path Enable (CMOS). A low level on this pin clamps the DAC outputs to midscale.
31	SDIO	Serial Port Data Input/Output (CMOS).
32	SCLK	Serial Port Clock Input (CMOS).
33	$\overline{CS}$	Serial Port Chip Select, Active Low (CMOS).
34	$\overline{IRQ}$	Interrupt Request. Open-drain, active low output. Pull this pin high external to the device.
35	DVDD18	1.8 V Digital Supply. Supplies power to digital core and digital data ports.
36	$\overline{RESET}$	Reset, Active Low (CMOS).
37	AVDD33	3.3 V Analog Supply.
38	IOUT2P	Q DAC Positive Current Output.
39	IOUT2N	Q DAC Negative Current Output.
40	AVDD33	3.3 V Analog Supply.
41	AVSS	Analog Supply Common.
42	REFIO	1.2 V Band Gap Voltage Reference Output. Should be decoupled to AVSS with a 0.1 µF capacitor.
43	FSADJ	Full-Scale Current Output Adjust. Place a 10 kΩ resistor from this pin to AVSS.
44	AVSS	Analog Supply Common.
45	AVDD33	3.3 V Analog Supply.
46	IOUT1N	I DAC Negative Current Output.
47	IOUT1P	I DAC Positive Current Output.
48	AVDD33	3.3 V Analog Supply.
	EPAD	The exposed pad (EPAD) must be soldered to the ground plane (AVSS). The EPAD provides an electrical, thermal, and mechanical connection to the board.

TYPICAL PERFORMANCE CHARACTERISTICS

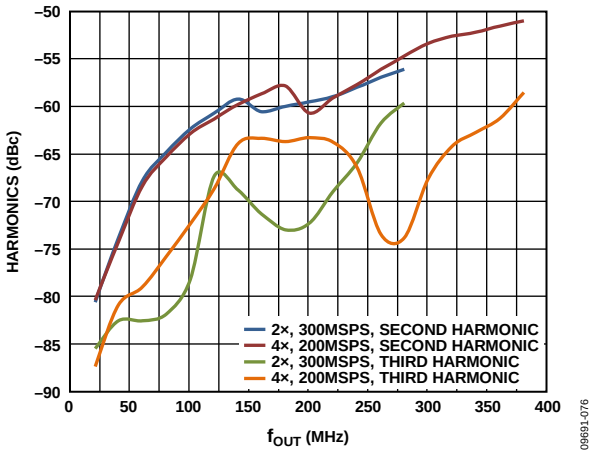

Figure 4. Harmonics vs. f_{OUT} over f_{DATA} and Interpolation, Digital Scale = 0 dBFS, I_{FS} = 20 mA

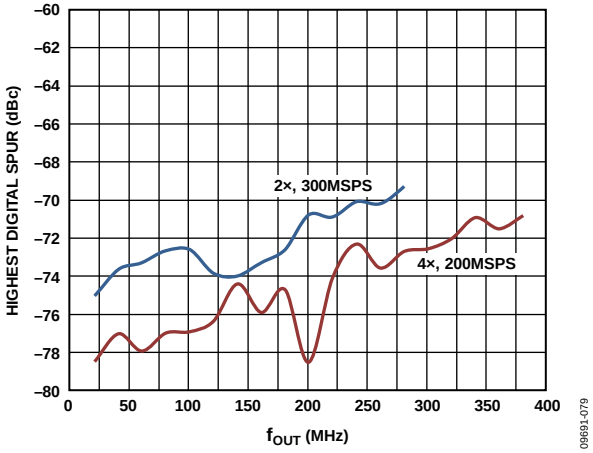

Figure 7. Highest Digital Spur vs. f_{OUT} over f_{DATA} and Interpolation, Digital Scale = 0 dBFS, I_{FS} = 20 mA

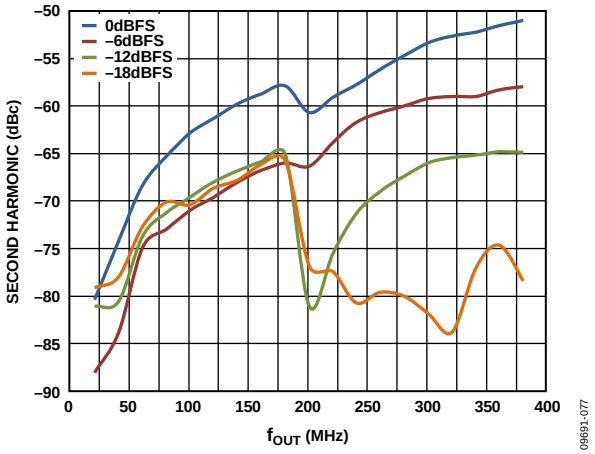

Figure 5. Second Harmonic vs. f_{OUT} over Digital Scale, 4× Interpolation, f_{DATA} = 400 MSPS, I_{FS} = 20 mA

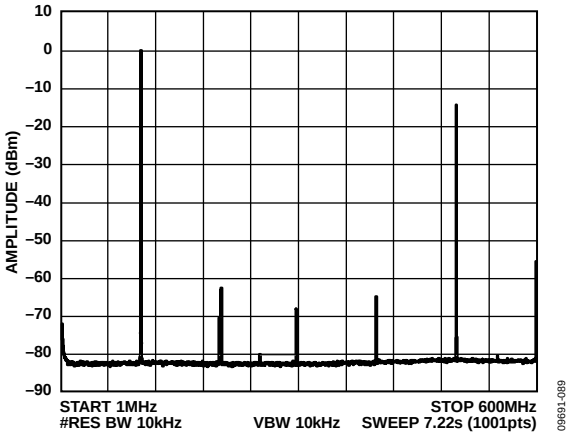

Figure 8. Single-Tone Spectrum, 2× Interpolation, f_{DATA} = 300 MSPS, f_{OUT} = 101 MHz

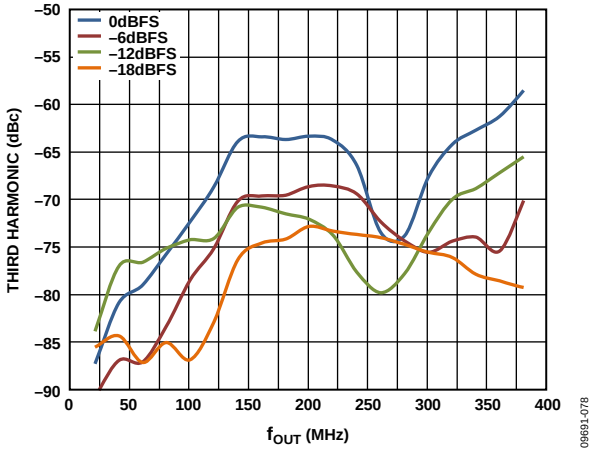

Figure 6. Third Harmonic vs. f_{OUT} over Digital Scale, 4× Interpolation, f_{DATA} = 400 MSPS, I_{FS} = 20 mA

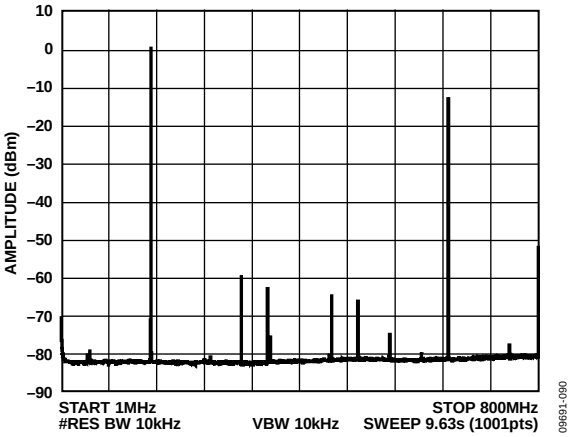

Figure 9. Single-Tone Spectrum, 4× Interpolation, f_{DATA} = 200 MSPS, f_{OUT} = 151 MHz

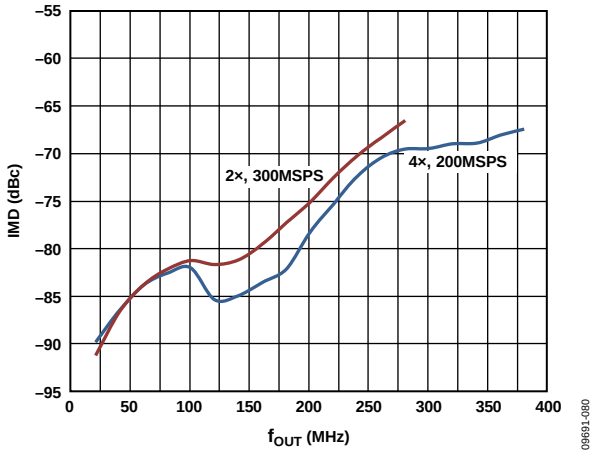

Figure 10. IMD vs. f_{OUT} over f_{DATA} and Interpolation, Digital Scale = 0 dBFS, I_{FS} = 20 mA

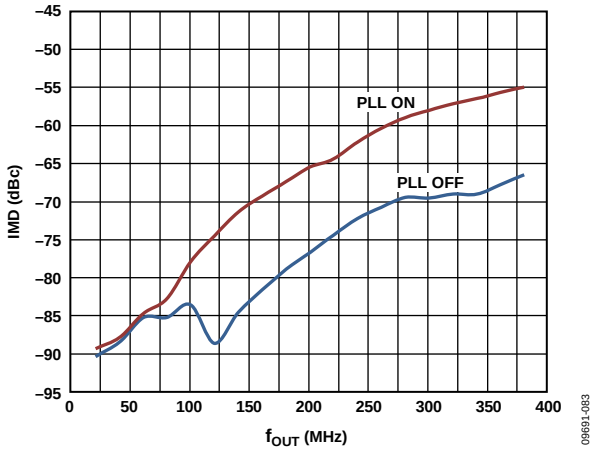

Figure 13. IMD vs. f_{OUT}, 4× Interpolation, f_{DATA} = 200 MSPS, Digital Scale = 0 dBFS, I_{FS} = 20 mA, PLL On and PLL Off

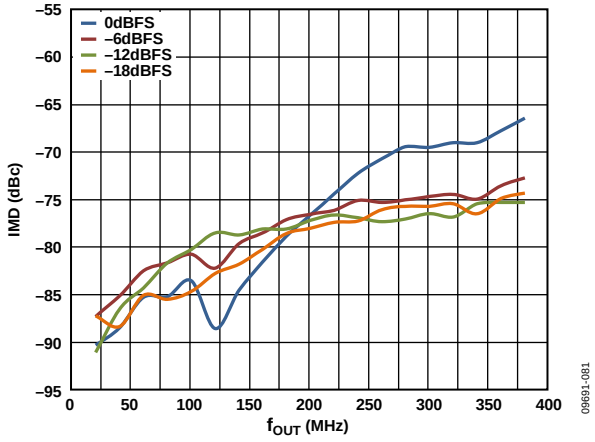

Figure 11. IMD vs. f_{OUT} over Digital Scale, 4× Interpolation, f_{DATA} = 400 MSPS, I_{FS} = 20 mA

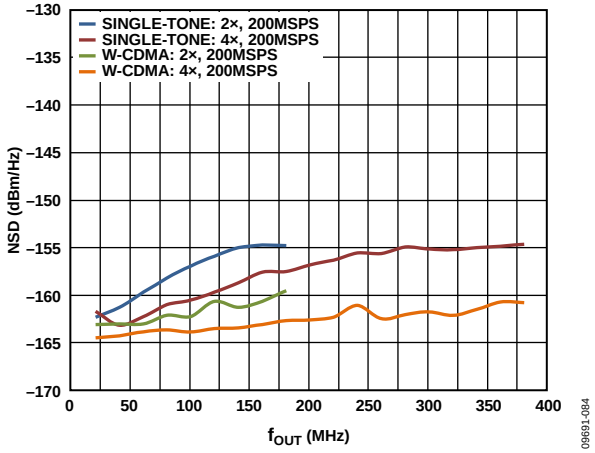

Figure 14. NSD vs. f_{OUT} over Interpolation, Single-Tone and W-CDMA Signals, f_{DATA} = 200 MSPS, Digital Scale = 0 dBFS, I_{FS} = 20 mA, PLL Off

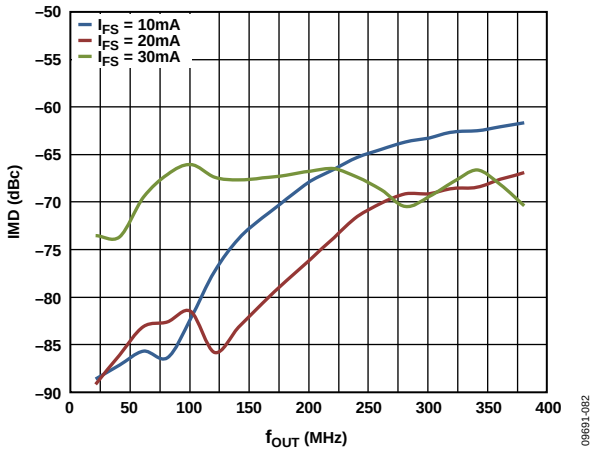

Figure 12. IMD vs. f_{OUT} over Full-Scale Current, 4× Interpolation, f_{DATA} = 400 MSPS, Digital Scale = 0 dBFS

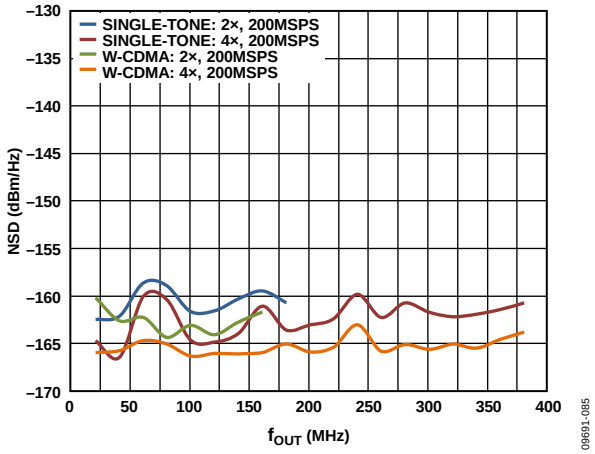

Figure 15. NSD vs. f_{OUT} over Interpolation, Single-Tone and W-CDMA Signals, f_{DATA} = 200 MSPS, Digital Scale = 0 dBFS, I_{FS} = 20 mA, PLL On

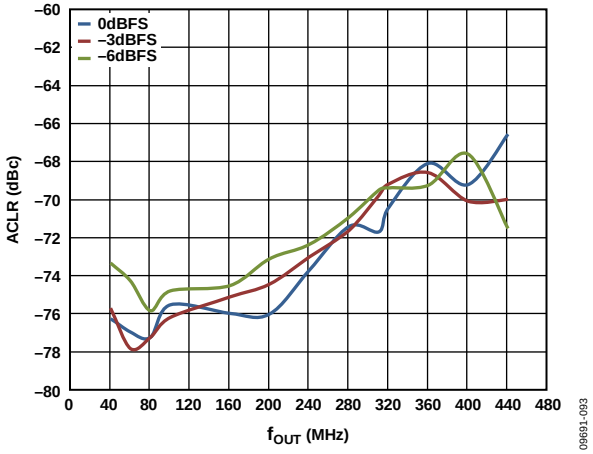

Figure 16. Four-Carrier W-CDMA ACLR vs. f_{OUT} over Digital Scale, Adjacent Channel, 4× Interpolation, f_{DATA} = 245.76 MSPS, PLL Off

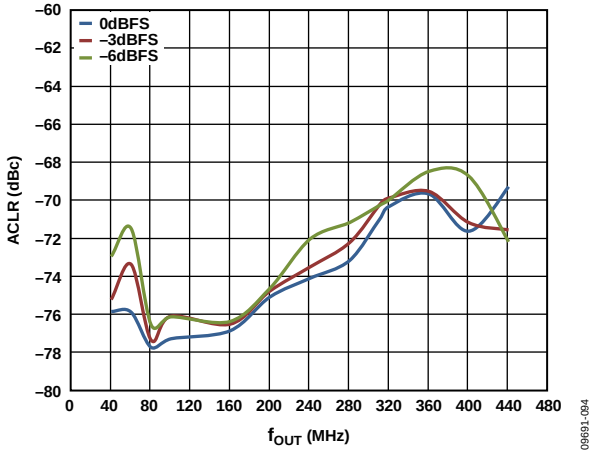

Figure 17. Four-Carrier W-CDMA ACLR vs. f_{OUT} over Digital Scale, First Alternate Channel, 4× Interpolation, f_{DATA} = 245.76 MSPS, PLL Off

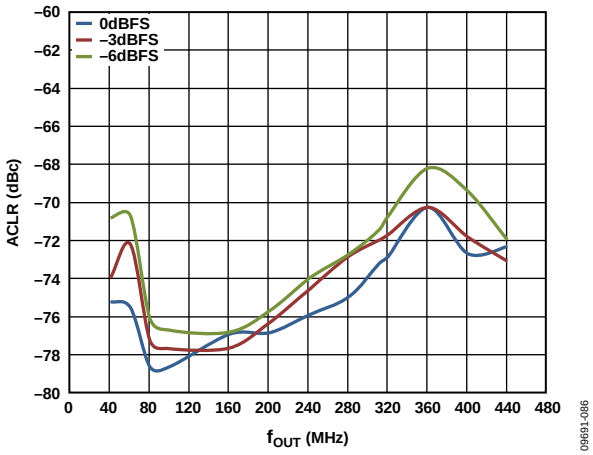

Figure 18. Four-Carrier W-CDMA ACLR vs. f_{OUT} over Digital Scale, Second Alternate Channel, 4× Interpolation, f_{DATA} = 245.76 MSPS, PLL Off

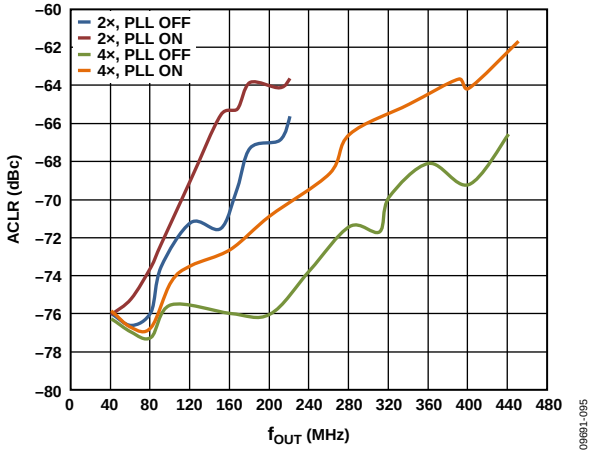

Figure 19. Four-Carrier W-CDMA ACLR vs. f_{OUT} over Interpolation, Adjacent Channel, f_{DATA} = 245.76 MSPS, PLL Off and PLL On

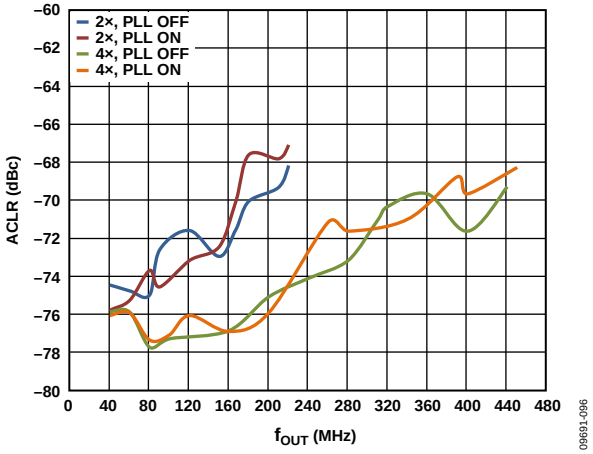

Figure 20. Four-Carrier W-CDMA ACLR vs. f_{OUT} over Interpolation, First Alternate Channel, f_{DATA} = 245.76 MSPS, PLL Off and PLL On

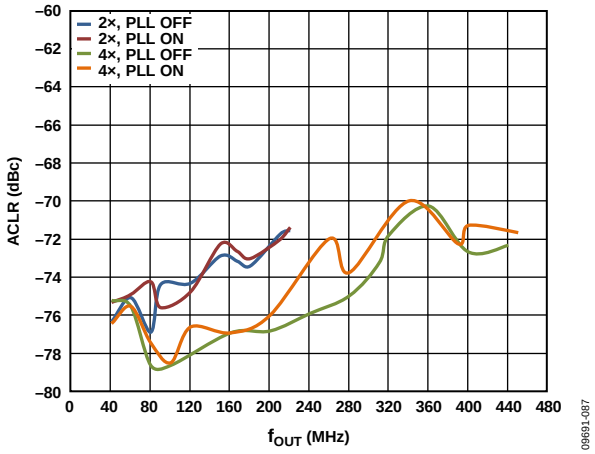

Figure 21. Four-Carrier W-CDMA ACLR vs. f_{OUT} over Interpolation, Second Alternate Channel, f_{DATA} = 245.76 MSPS, PLL Off and PLL On

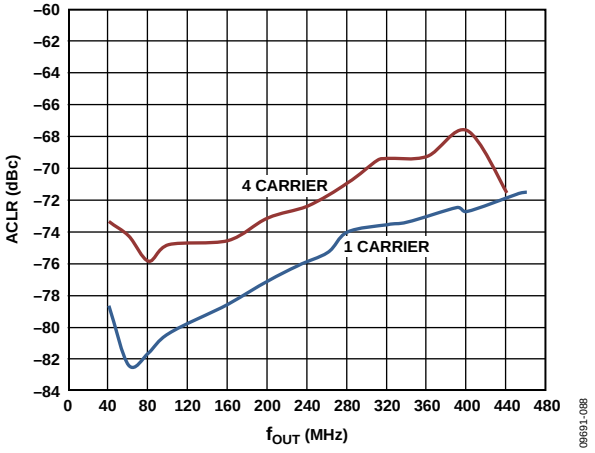

Figure 23. W-CDMA ACLR vs. f_{OUT} over Number of Carriers, Adjacent Channel, 4× Interpolation, f_{DATA} = 245.76 MSPS, Digital Scale = –6 dBFS, PLL Off

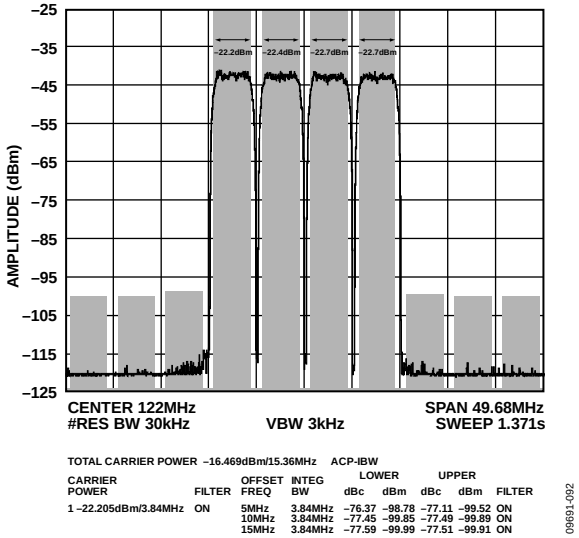

Figure 22. Four-Carrier W-CDMA Performance, 4× Interpolation, f_{DATA} = 245.76 MSPS, Digital Scale = –6 dBFS, f_{OUT} = 122 MHz

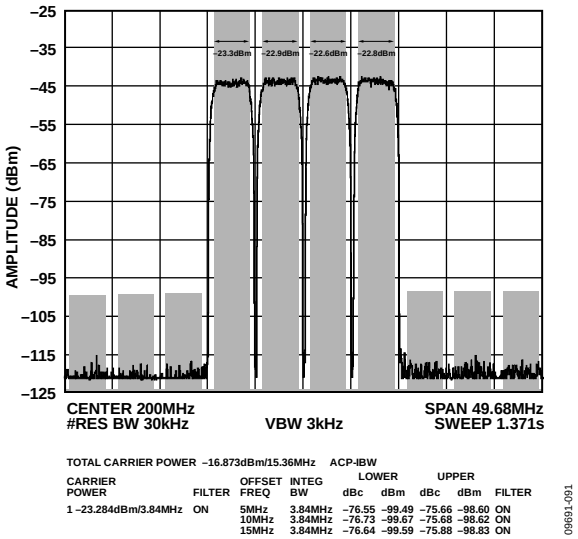

Figure 24. Four-Carrier W-CDMA Performance, 4× Interpolation, f_{DATA} = 245.76 MSPS, Digital Scale = –6 dBFS, f_{OUT} = 200 MHz

TERMINOLOGY

Integral Nonlinearity (INL)
INL is 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
Offset error is the deviation of the output current from the ideal of 0 mA. For IOUT1P, 0 mA output is expected when all inputs are set to 0. For IOUT1N, 0 mA output is expected when all inputs are set to 1.

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

Output Compliance Range
The output compliance range is 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 voltage drift, the drift is reported in ppm per degree Celsius.

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

Settling Time
Settling time is 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)
SFDR is the difference, in decibels, between the peak amplitude of the output signal and the peak spurious signal within the dc to 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 on 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)
ACLR is 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.

THEORY OF OPERATION

High performance, small size, and low power consumption make the AD9146 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 designing single sideband (SSB) transmitters.

The AD9146 offers features that allow simplified synchronization with incoming data and between multiple devices. Auxiliary DACs are also provided on chip. The auxiliary DACs can be used for output dc offset compensation (for LO compensation in SSB transmitters) and for gain matching (for image rejection optimization in SSB transmitters).

SERIAL PORT OPERATION

The serial port is a flexible, synchronous serial communications port that allows easy interfacing to many industry-standard microcontrollers 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 AD9146. Single-byte or multiple-byte transfers are supported, as well as MSB first or LSB first transfer formats.

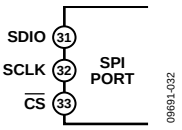

Figure 25. Serial Port Interface Pins

A communication cycle with the AD9146 has two phases. Phase 1 is the instruction cycle (the writing of an instruction byte into the device), coincident with the first eight SCLK rising edges. The instruction byte provides the serial port controller with information regarding the data transfer cycle—Phase 2 of the communication cycle. The Phase 1 instruction byte defines whether the upcoming data transfer is a read or write, along with 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.

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 a 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 communications lines. When the $\overline{CS}$ pin is high, the SDIO pin goes to a high impedance state. During the communication cycle, the $\overline{CS}$ pin should stay low.

Serial Data I/O (SDIO)

The SDIO pin is a bidirectional pin that functions as an input in write mode and as an output in read mode. Data is written into the device on this pin and read from the device on this pin. The configuration of the SDIO pin is controlled by Register 0x00, Bit 7. To enable readback of the register data, this bit must be set to 1.

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 bits 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 high address to 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 bits 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. Subsequent data bytes should follow from low address to high address. In LSB first mode, the serial port internal byte address generator increments for each data byte of the multibyte communication cycle.

If the MSB first mode is active, the serial port controller data address decrements from the data address written toward 0x00 for multibyte I/O operations. If the LSB first mode is active, the serial port controller data address increments from the data address written toward 0x7F for multibyte I/O operations.

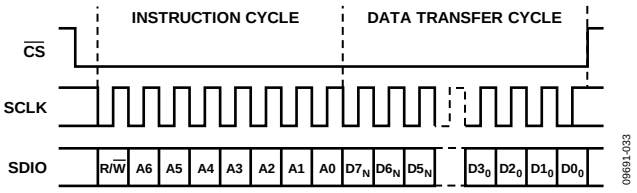

Figure 26. Serial Port Interface Timing, MSB First

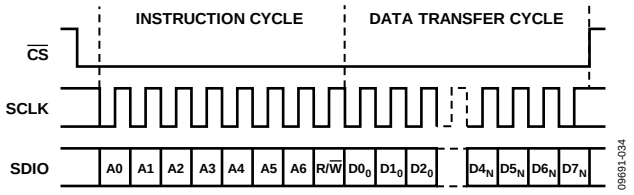

Figure 27. Serial Port Interface Timing, LSB First

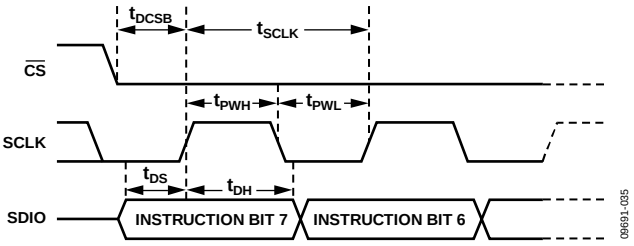

Figure 28. Timing Diagram for Serial Port Register Write

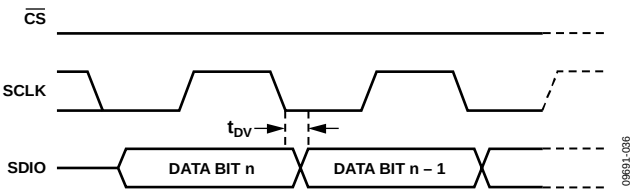

Figure 29. Timing Diagram for Serial Port Register Read

DEVICE CONFIGURATION REGISTER MAP AND DESCRIPTIONS

Table 10. Device Configuration Register Map

Addr (Hex)	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default
0x00	Comm	SDIO	LSB_FIRST	Reset						0x00
0x01	Power control	Power down I DAC	Power down Q DAC	Power down data receiver	Power down aux ADC	Power down aux DACs and reference	Power down clocks			0x10
0x02	Tx enable control		Extended delay length	Enable extended delay	Power down voltage reference	Power down PLL	Power down DACs	Power down FIFO	Power down filters	0x00
0x03	Data format	Binary data format	Q data first	MSB swap				Data Bus Width[1:0]		0x00
0x04	Interrupt enable	Enable PLL lock lost	Enable PLL locked	Enable sync signal lost	Enable sync signal locked			Enable FIFO Warning 1	Enable FIFO Warning 2	0x00
0x05	Interrupt enable	0	0	0	Enable AED compare pass	Enable AED compare fail	Enable SED compare fail	0	0	0x00
0x06	Event flag	PLL lock lost	PLL locked	Sync signal lost	Sync signal locked			FIFO Warning 1	FIFO Warning 2	N/A
0x07	Event flag				AED compare pass	AED compare fail	SED compare fail			N/A
0x08	Clock receiver control	DACCLK duty correction	REFCLK duty correction	DACCLK cross-correction	REFCLK cross-correction	1	1	1	1	0x3F
0x0A	PLL control	PLL enable	PLL manual enable	Manual VCO Band[5:0]						0x40
0x0C	PLL control	PLL Loop Bandwidth[1:0]			PLL Charge Pump Current[4:0]					0xD1
0x0D	PLL control	N2[1:0]			PLL cross-control enable	N0[1:0]		N1[1:0]		0xD9
0x0E	PLL status	PLL locked				VCO Control Voltage[3:0]				N/A
0x0F	PLL status			VCO Band Readback[5:0]						N/A
0x10	Sync control	Sync enable	Data/FIFO rate toggle			Rising edge sync	Sync Averaging[2:0]			0x48
0x11	Sync control			Sync Phase Request[5:0]						0x00
0x12	Sync status	Sync lost	Sync locked							N/A
0x13	Sync status	Sync Phase Readback[7:0] (6.2 format)								N/A
0x15	Data receiver status			LVDS FRAME level high	LVDS FRAME level low	LVDS DCI level high	LVDS DCI level low	LVDS data level high	LVDS data level low	N/A
0x16	DCI delay						Delay bypass	DCI Delay[1:0]		0x00
0x17	FIFO control						FIFO Phase Offset[2:0]			0x04
0x18	FIFO status	FIFO Warning 1	FIFO Warning 2				FIFO soft align ack	FIFO soft align request		N/A
0x19	FIFO status	FIFO Level[7:0]								N/A

Addr (Hex)	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default
0x1B	Datapath control	Bypass premod	Bypass sinc^{-1}	1			Bypass phase comp and dc offset	Select sideband	Send I data to Q data	0xE4
0x1C	HB1 control						HB1[1:0]		Bypass HB1	0x00
0x1D	HB2 control		HB2[5:0]						Bypass HB2	0x00
0x1E	Datapath config	This register must be changed from the default value for proper operation.							1	0x00
0x1F	Chip ID	Chip ID[7:0]								0x08
0x38	I phase adj LSB	I Phase Adj[7:0]								0x00
0x39	I phase adj MSB							I Phase Adj[9:8]		0x00
0x3A	Q phase adj LSB	Q Phase Adj[7:0]								0x00
0x3B	Q phase adj MSB							Q Phase Adj[9:8]		0x00
0x3C	I DAC offset LSB	I DAC Offset[7:0]								0x00
0x3D	I DAC offset MSB	I DAC Offset[15:8]								0x00
0x3E	Q DAC offset LSB	Q DAC Offset[7:0]								0x00
0x3F	Q DAC offset MSB	Q DAC Offset[15:8]								0x00
0x40	I DAC FS adjust	I DAC FS Adj[7:0]								0xF9
0x41	I DAC control	I DAC sleep						I DAC FS Adj[9:8]		0x01
0x42	I aux DAC data	I Aux DAC[7:0]								0x00
0x43	I aux DAC control	I aux DAC sign	I aux DAC current direction	I aux DAC sleep				I Aux DAC[9:8]		0x00
0x44	Q DAC FS adjust	Q DAC FS Adj[7:0]								0xF9
0x45	Q DAC control	Q DAC sleep						Q DAC FS Adj[9:8]		0x01
0x46	Q aux DAC data	Q Aux DAC[7:0]								0x00
0x47	Q aux DAC control	Q aux DAC sign	Q aux DAC current direction	Q aux DAC sleep				Q Aux DAC[9:8]		0x00
0x48	Die temp range control		FS Current[2:0]			Reference Current[2:0]			Capacitor value	0x02
0x49	Die temp LSB	Die Temp[7:0]								N/A
0x4A	Die temp MSB	Die Temp[15:8]								N/A
0x67	SED control	SED compare enable		Sample error detected		Autoclear enable		Compare fail	Compare pass	0x00
0x68	Compare I0 LSBs	Compare Value I_{0LSB}								0xB6
0x69	Compare I0 MSBs	Compare Value I_{0MSB}								0x7A
0x6A	Compare Q0 LSBs	Compare Value Q_{0LSB}								0x45
0x6B	Compare Q0 MSBs	Compare Value Q_{0MSB}								0xEA
0x6C	Compare I1 LSBs	Compare Value I_{1LSB}								0x16
0x6D	Compare I1 MSBs	Compare Value I_{1MSB}								0x1A
0x6E	Compare Q1 LSBs	Compare Value Q_{1LSB}								0xC6
0x6F	Compare Q1 MSBs	Compare Value Q_{1MSB}								0xAA
0x70	SED I LSBs	Errors Detected I_{NLSB}								0x00
0x71	SED I MSBs	Errors Detected I_{NMSB}								0x00
0x72	SED Q LSBs	Errors Detected Q_{NLSB}								0x00
0x73	SED Q MSBs	Errors Detected Q_{NMSB}								0x00
0x7F	Revision			Revision[3:0]						N/A

Table 11. Device Configuration Register Descriptions

Register Name	Address (Hex)	Bits	Name	Description	Default
Comm	0x00	7	SDIO	SDIO pin operation. To enable data readback, set this bit to 1. 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	The device is placed in reset when this bit is written high and remains in reset until the bit is written low.	0
Power Control	0x01	7	Power down I DAC	1 = power down I DAC.	0
		6	Power down Q DAC	1 = power down Q DAC.	0
		5	Power down data receiver	1 = power down the input data receiver.	0
		4	Power down auxiliary ADC	1 = power down the auxiliary ADC for temperature sensor.	1
		3	Power down auxiliary DACs and reference	1 = power down the auxiliary DACs and the voltage reference.	0
		2	Power down clocks	1 = power down the clocks.	0
Tx Enable Control	0x02	6	Extended delay length	Time delay from when the TXENABLE pin is brought high to when the DAC begins transmitting data. See the Tx Enable section for more information. 0 = delay the outputs by 12 to 13 DAC/64 clock edges. 1 = delay the outputs by 19 to 20 DAC/64 clock edges.	0
		5	Enable extended delay	The transmit delay, regardless of whether the extended delay option is selected, has an inherent fixed delay of 10 DAC clock cycles. When the extended delay is disabled, there is a minimum delay time in the outputs of 1 to 2 DAC/64 clock edges from when the TXENABLE pin is brought high. 0 = disable the extended delay option. Delays the outputs by 1 to 2 DAC/64 clock edges. 1 = enable the extended delay option. Delays the outputs based on the setting of Bit 6.	0
		4	Power down voltage reference	0 = no power-down of the internal voltage reference. 1 = power down the internal voltage reference when the TXENABLE pin is held low.	0
		3	Power down PLL	0 = no power-down of the on-chip PLL. 1 = power down the on-chip PLL when the TXENABLE pin is held low.	0
		2	Power down DACs	0 = no power-down of the DAC cores. 1 = power down the DAC cores when the TXENABLE pin is held low.	0
		1	Power down FIFO	0 = no power-down of the FIFO. 1 = power down the FIFO when the TXENABLE pin is held low.	0
		0	Power down filters	0 = no power-down of the interpolation filters. 1 = power down the interpolation filters when the TXENABLE pin is held low.	0

Register Name	Address (Hex)	Bits	Name	Description	Default
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[1:0]	Data receiver interface mode. See the LVDS Input Data Ports section for information about the operation of the different interface modes. 00 = byte mode; 8-bit interface bus width. 01 = byte mode; 8-bit interface bus width. 10 = nibble mode; 4-bit interface bus width. 11 = invalid.	00
Interrupt Enable	0x04	7	Enable PLL lock lost	1 = enable interrupt for PLL lock lost.	0
		6	Enable PLL locked	1 = enable interrupt for PLL locked.	0
		5	Enable sync signal lost	1 = enable interrupt for sync signal lost.	0
		4	Enable sync signal locked	1 = enable interrupt for sync signal locked.	0
		1	Enable FIFO Warning 1	1 = enable interrupt for FIFO Warning 1.	0
		0	Enable FIFO Warning 2	1 = enable interrupt for FIFO Warning 2.	0
	0x05	[7:5]	Set to 0	Set these bits to 0.	000
		4	Enable AED compare pass	1 = enable interrupt for AED comparison pass.	0
		3	Enable AED compare fail	1 = enable interrupt for AED comparison fail.	0
		2	Enable SED compare fail	1 = enable interrupt for SED comparison fail.	0
		[1:0]	Set to 0	Set these bits to 0.	00
Event Flag	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.	N/A
		6	PLL locked	1 = indicates that the PLL has locked to the reference clock input.	N/A
		5	Sync signal lost	1 = indicates that the sync logic, which had been previously locked, has lost alignment. This is a latched signal.	N/A
		4	Sync signal locked	1 = indicates that the sync logic has achieved sync alignment. This is indicated when no phase changes were requested for at least a few full averaging cycles.	N/A
		1	FIFO Warning 1	1 = indicates that the difference between the FIFO read and write pointers is 1.	N/A
		0	FIFO Warning 2	1 = indicates that the difference between the FIFO read and write pointers is 2.	N/A
			Note that all event flags are cleared by writing the respective bit high.		
	0x07	4	AED compare pass	1 = indicates that the SED logic detected a valid input data pattern compared against the preprogrammed expected values. This is a latched signal.	N/A
		3	AED compare fail	1 = indicates that the SED logic detected an invalid input data pattern compared against the preprogrammed expected values. This latched signal is automatically cleared when eight valid I/Q data pairs are received.	N/A
		2	SED compare fail	1 = indicates that the SED logic detected an invalid input data pattern compared against the preprogrammed expected values. This is a latched signal.	N/A
			Note that all event flags are cleared by writing the respective bit high.		

Register Name	Address (Hex)	Bits	Name	Description	Default
Clock Receiver Control	0x08	7	DACCLK duty correction	1 = enable duty cycle correction on the DACCLK input.	0
		6	REFCLK duty correction	1 = enable duty cycle correction on the REFCLK input.	0
		5	DACCLK cross-correction	1 = enable differential crossing correction on the DACCLK input.	1
		4	REFCLK cross-correction	1 = enable differential crossing correction on the REFCLK input.	1
PLL Control	0x0A	7	PLL enable	1 = enable the PLL clock multiplier. The REFCLK input is used as the PLL reference clock signal.	0
		6	PLL manual enable	1 = enable manual selection of the VCO band. The correct VCO band must be determined by the user and written to Bits[5:0].	1
		[5:0]	Manual VCO Band[5:0]	Selects the VCO band to be used.	000000
	0x0C	[7:6]	PLL Loop Bandwidth[1:0]	Selects the PLL loop filter bandwidth. 00 = widest bandwidth. … 11 = narrowest bandwidth.	11
		[4:0]	PLL Charge Pump Current[4:0]	Sets the nominal PLL charge pump current. 00000 = lowest current setting. … 11111 = highest current setting.	10001
	0x0D	[7:6]	N2[1:0]	PLL control clock divider. This divider determines the ratio of the DACCLK frequency to the PLL controller clock frequency. f_{PC_CLK} must always be less than 75 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.	11
		4	PLL cross-control enable	1 = enable PLL cross-point controller.	1
		[3:2]	N0[1:0]	PLL VCO divider. This divider determines the ratio of the VCO frequency to the DACCLK frequency. 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. This divider determines the ratio of the DACCLK frequency to the REFCLK frequency. 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	0x0E	7	PLL locked	1 = the PLL-generated clock is tracking the REFCLK input signal.	N/A
		[3:0]	VCO Control Voltage[3:0]	VCO control voltage readback. See Table 22.	N/A
	0x0F	[5:0]	VCO Band Readback[5:0]	Indicates the VCO band currently selected.	N/A

Register Name	Address (Hex)	Bits	Name	Description	Default
Sync Control	0x10	7	Sync enable	1 = enable the synchronization logic.	0
		6	Data/FIFO rate toggle	0 = operate the synchronization at the FIFO reset rate. 1 = operate 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 in determining the sync phase. 000 = 1. 001 = 2. 010 = 4. 011 = 8. 100 = 16. 101 = 32. 110 = 64. 111 = 128.	000
	0x11	[5:0]	Sync Phase Request[5:0]	This register sets the requested clock phase offset after sync. The offset unit is in DACCLK cycles. This register 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.	000000
Sync Status	0x12	7	Sync lost	1 = synchronization was attained but has been lost.	N/A
		6	Sync locked	1 = synchronization has been attained.	N/A
	0x13	[7:0]	Sync Phase Readback[7:0]	Indicates the averaged sync phase offset (6.2 format). If this value differs from the Sync Phase Request[5:0] value in Register 0x11, a sync timing error has occurred. For more information, see the Sync Status Bits section. 00000000 = 0.0. 00000001 = 0.25. … 11111110 = 63.50. 11111111 = 63.75.	N/A
Data Receiver Status	0x15	5	LVDS FRAME level high	One or both LVDS FRAME input signals have exceeded 1.7 V.	N/A
		4	LVDS FRAME level low	One or both LVDS FRAME input signals have crossed below 0.7 V.	N/A
		3	LVDS DCI level high	One or both LVDS DCI input signals have exceeded 1.7 V.	N/A
		2	LVDS DCI level low	One or both LVDS DCI input signals have crossed below 0.7 V.	N/A
		1	LVDS data level high	One or more LVDS Dx input signals have exceeded 1.7 V.	N/A
		0	LVDS data level low	One or more LVDS Dx input signals have crossed below 0.7 V.	N/A
DCI Delay	0x16	2	Delay bypass	0 = enable the on-chip DCI delay feature. Set the delay using Bits[1:0]. 1 = bypass the on-chip DCI delay feature.	0
		[1:0]	DCI Delay[1:0]	These bits control the delay applied to the DCI signal. The DCI delay affects the sampling interval of the DCI with respect to the Dx inputs. See Table 14. 00 = 165 ps delay of DCI signal. 01 = 375 ps delay of DCI signal. 10 = 615 ps delay of DCI signal. 11 = 720 ps delay of DCI signal.	00

Register Name	Address (Hex)	Bits	Name	Description	Default
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 (100). 000 = 0. 001 = 1. … 111 = 7.	100
FIFO Status	0x18	7	FIFO Warning 1	1 = FIFO read and write pointers are within ±1.	N/A
		6	FIFO Warning 2	1 = FIFO read and write pointers are within ±2.	N/A
		2	FIFO soft align acknowledge	1 = FIFO read and write pointers are aligned after a serial port initiated FIFO reset.	N/A
		1	FIFO soft align request	1 = request FIFO read and write pointer alignment via the serial port.	0
	0x19	[7:0]	FIFO Level[7:0]	Thermometer encoded measure of the FIFO level.	N/A
Datapath Control	0x1B	7	Bypass premod	1 = bypass the $f_S/2$ premodulator.	1
		6	Bypass $sinc^{-1}$	1 = bypass the inverse sinc filter.	1
		5	Set to 1	Set this bit to 1 for proper operation.	1
		2	Bypass phase compensation and dc offset	1 = bypass phase compensation and dc offset.	1
		1	Select sideband	0 = the modulator outputs the high-side image. 1 = the modulator outputs the low-side image. The image is spectrally inverted compared to the input data.	0
		0	Send I data to Q data	1 = ignore Q data from the interface and disable the clocks to the Q datapath. Send I data to both the I and Q DACs.	0
HB1 Control	0x1C	[2:1]	HB1[1:0]	Modulation mode for I Side Half-Band Filter 1. 00 = input signal not modulated; filter pass band is from −0.4 to +0.4 of f_{IN1}. 01 = input signal not modulated; filter pass band is from 0.1 to 0.9 of f_{IN1}. 10 = input signal modulated by f_{IN1}; filter pass band is from 0.6 to 1.4 of f_{IN1}. 11 = input signal modulated by f_{IN1}; filter pass band is from 1.1 to 1.9 of f_{IN1}.	00
		0	Bypass HB1	1 = bypass the first-stage interpolation filter.	0
HB2 Control	0x1D	[6:1]	HB2[5:0]	Modulation mode for I Side Half-Band Filter 2. 000000 = input signal not modulated; filter pass band is from −0.25 to +0.25 of f_{IN2}. 001001 = input signal not modulated; filter pass band is from 0.0 to 0.5 of f_{IN2}. 010010 = input signal not modulated; filter pass band is from 0.25 to 0.75 of f_{IN2}. 011011 = input signal not modulated; filter pass band is from 0.5 to 1.0 of f_{IN2}. 100100 = input signal modulated by f_{IN2}; filter pass band is from 0.75 to 1.25 of f_{IN2}. 101101 = input signal modulated by f_{IN2}; filter pass band is from 1.0 to 1.5 of f_{IN2}. 110110 = input signal modulated by f_{IN2}; filter pass band is from 1.25 to 1.75 of f_{IN2}. 111111 = input signal modulated by f_{IN2}; filter pass band is from 1.5 to 2.0 of f_{IN2}.	000000
		0	Bypass HB2	1 = bypass the second-stage interpolation filter.	0

Register Name	Address (Hex)	Bits	Name	Description	Default
Datapath Config	0x1E	0	Set to 1	Set this bit to 1 for proper operation. (The default value must be changed from 0 to 1.)	0
Chip ID	0x1F	[7:0]	Chip ID[7:0]	This register identifies the device as an AD9146.	00001000
I Phase Adj LSB	0x38	[7:0]	I Phase Adj[7:0]	See Register 0x39.	00000000
I Phase Adj MSB	0x39	[1:0]	I Phase Adj[9:8]	I Phase Adj[9:0] is used to insert a phase offset between the I and Q datapaths. This offset can be used to correct for phase imbalance in a quadrature modulator. See the Quadrature Phase Correction section for more information.	00
Q Phase Adj LSB	0x3A	[7:0]	Q Phase Adj[7:0]	See Register 0x3B.	00000000
Q Phase Adj MSB	0x3B	[1:0]	Q Phase Adj[9:8]	Q Phase Adj[9:0] is used to insert a phase offset between the I and Q datapaths. This offset can be used to correct for phase imbalance in a quadrature modulator. See the Quadrature Phase Correction section for more information.	00
I DAC Offset LSB	0x3C	[7:0]	I DAC Offset[7:0]	See Register 0x3D.	00000000
I DAC Offset MSB	0x3D	[7:0]	I DAC Offset[15:8]	I DAC Offset[15:0] is a value that is added directly to the samples written to the I DAC.	00000000
Q DAC Offset LSB	0x3E	[7:0]	Q DAC Offset[7:0]	See Register 0x3F.	00000000
Q DAC Offset MSB	0x3F	[7:0]	Q DAC Offset[15:8]	Q DAC Offset[15:0] is a value that is added directly to the samples written to the Q DAC.	00000000
I DAC FS Adjust	0x40	[7:0]	I DAC FS Adj[7:0]	See Register 0x41, Bits[1:0].	11111001
I DAC Control	0x41	7	I DAC sleep	1 = puts the I DAC into sleep mode (fast wake-up mode).	0
		[1:0]	I DAC FS Adj[9:8]	I DAC FS Adj[9:0] sets the full-scale current of the I DAC. The full-scale current can be adjusted from 8.64 mA to 31.68 mA in step sizes of approximately 22.5 µA. 0x000 = 8.64 mA. … 0x200 = 20.16 mA. … 0x3FF = 31.68 mA.	01
I Aux DAC Data	0x42	[7:0]	I Aux DAC[7:0]	See Register 0x43, Bits[1:0].	00000000
I Aux DAC Control	0x43	7	I aux DAC sign	0 = the I auxiliary DAC sign is positive, and the current is directed to the IOUT1P pin (Pin 47). 1 = the I auxiliary DAC sign is negative, and the current is directed to the IOUT1N pin (Pin 46).	0
		6	I aux DAC current direction	0 = the I auxiliary DAC sources current. 1 = the I auxiliary DAC sinks current.	0
		5	I aux DAC sleep	1 = puts the I auxiliary DAC into sleep mode.	0
		[1:0]	I Aux DAC[9:8]	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.	00
Q DAC FS Adjust	0x44	[7:0]	Q DAC FS Adj[7:0]	See Register 0x45, Bits[1:0].	11111001

Register Name	Address (Hex)	Bits	Name	Description	Default
Q DAC Control	0x45	7	Q DAC sleep	1 = puts the Q DAC into sleep mode (fast wake-up mode).	0
		[1:0]	Q DAC FS Adj[9:8]	Q DAC FS Adj[9:0] sets the full-scale current of the Q DAC. The full-scale current can be adjusted from 8.64 mA to 31.68 mA in step sizes of approximately 22.5 µA. 0x000 = 8.64 mA. … 0x200 = 20.16 mA. … 0x3FF = 31.68 mA.	01
Q Aux DAC Data	0x46	[7:0]	Q Aux DAC[7:0]	See Register 0x47, Bits[1:0].	00000000
Q Aux DAC Control	0x47	7	Q aux DAC sign	0 = the Q auxiliary DAC sign is positive, and the current is directed to the IOUT2P pin (Pin 38). 1 = the Q auxiliary DAC sign is negative, and the current is directed to the IOUT2N pin (Pin 39).	0
		6	Q aux DAC current direction	0 = the Q auxiliary DAC sources current. 1 = the Q auxiliary DAC sinks current.	0
		5	Q aux DAC sleep	1 = puts the Q auxiliary DAC into sleep mode.	0
		[1:0]	Q Aux DAC[9:8]	Q 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.	00
Die Temp Range Control	0x48	[6:4]	FS Current[2:0]	Auxiliary ADC full-scale current. 000 = lowest current. … 111 = highest current.	000
		[3:1]	Reference Current[2:0]	Auxiliary ADC reference current. 000 = lowest current. … 111 = highest current.	001
		0	Capacitor value	Auxiliary ADC internal capacitor value. 0 = 5 pF. 1 = 10 pF.	0
Die Temp LSB	0x49	[7:0]	Die Temp[7:0]	See Register 0x4A.	N/A
Die Temp MSB	0x4A	[7:0]	Die Temp[15:8]	Die Temp[15:0] indicates the approximate die temperature. For more information, see the Temperature Sensor section.	N/A
SED Control	0x67	7	SED compare enable	1 = enable 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 was detected. The bit remains set until cleared. Any write to this register clears this bit to 0.	0
		3	Autoclear enable	1 = enable autoclear mode. This activates Bit 1 and Bit 0 of this register and causes Register 0x70 through Register 0x73 to be autocleared when eight consecutive sample data sets are received error free.	0
		1	Compare fail	1 = indicates an error was detected. This bit remains set until it is autocleared by the reception of eight consecutive error-free comparisons or is cleared by a write to this register.	0
		0	Compare pass	1 = indicates that the last sample comparison was error free.	0

Register Name	Address (Hex)	Bits	Name	Description	Default
Compare I0 LSBs	0x68	[7:0]	Compare Value I_{0LSB}	Compare Value I_{0LSB} is the byte that is compared with the I_{0LSB} input sample captured at the input interface.	10110110
Compare I0 MSBs	0x69	[7:0]	Compare Value I_{0MSB}	Compare Value I_{0MSB} is the byte that is compared with the I_{0MSB} input sample captured at the input interface.	01111010
Compare Q0 LSBs	0x6A	[7:0]	Compare Value Q_{0LSB}	Compare Value Q_{0LSB} is the byte that is compared with the Q_{0LSB} input sample captured at the input interface.	01000101
Compare Q0 MSBs	0x6B	[7:0]	Compare Value Q_{0MSB}	Compare Value Q_{0MSB} is the byte that is compared with the Q_{0MSB} input sample captured at the input interface.	11101010
Compare I1 LSBs	0x6C	[7:0]	Compare Value I_{1LSB}	Compare Value I_{1LSB} is the byte that is compared with the I_{1LSB} input sample captured at the input interface.	00010110
Compare I1 MSBs	0x6D	[7:0]	Compare Value I_{1MSB}	Compare Value I_{1MSB} is the byte that is compared with the I_{1MSB} input sample captured at the input interface.	00011010
Compare Q1 LSBs	0x6E	[7:0]	Compare Value Q_{1LSB}	Compare Value Q_{1LSB} is the byte that is compared with the Q_{1LSB} input sample captured at the input interface.	11000110
Compare Q1 MSBs	0x6F	[7:0]	Compare Value Q_{1MSB}	Compare Value Q_{1MSB} is the byte that is compared with the Q_{1MSB} input sample captured at the input interface.	10101010
SED I LSBs	0x70	[7:0]	Errors Detected I_{NLSB}	Errors Detected I_{NLSB} indicates which bits were received in error.	00000000
SED I MSBs	0x71	[7:0]	Errors Detected I_{NMSB}	Errors Detected I_{NMSB} indicates which bits were received in error.	00000000
SED Q LSBs	0x72	[7:0]	Errors Detected Q_{NLSB}	Errors Detected Q_{NLSB} indicates which bits were received in error.	00000000
SED Q MSBs	0x73	[7:0]	Errors Detected Q_{NMSB}	Errors Detected Q_{NMSB} indicates which bits were received in error.	00000000
Revision	0x7F	[5:2]	Revision[3:0]	This value corresponds to the die revision number. 0011 = Die Revision 1.	N/A

LVDS INPUT DATA PORTS

The AD9146 has one LVDS data port that receives data for both the I and Q transmit paths. The device can accept data in byte and nibble formats. In byte and nibble modes, the data is sent over 8-bit and 4-bit LVDS data buses, respectively. The pin assignments of the bus in each mode are shown in Table 12.

Table 12. Data Bit Pair Assignments for Data Input Modes

Mode	MSB to LSB
Byte	D7, D6, D5, D4, D3, D2, D1, D0
Nibble[1]	D5, D4, D3, D2

[1] In nibble mode, the unused pins can be left floating.

The data is accompanied by DCI and FRAME signals. The DCI signal is a reference bit that is used to generate a double data rate (DDR) clock. The FRAME signal is required for controlling to which DAC the data is sent. All of the interface signals can be time aligned, so there is a maximum skew requirement on the bus. In some cases, it is best to delay the DCI signal for optimum timing.

BYTE INTERFACE MODE

In byte mode, the DCI signal is a reference bit used to generate the data sampling clock and should be time aligned with the data. The most significant byte of the data should correspond to DCI high, and the least significant byte of the data should correspond to DCI low. 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. The complete timing diagram is shown in Figure 30.

NIBBLE INTERFACE MODE

In nibble mode, the DCI signal is a reference bit used to generate the data sampling clock and should be time aligned with the data. 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. All four nibbles must be written to the device for proper operation. For 12-bit resolution devices, the data in the fourth nibble acts as a placeholder for the data framing structure. The complete timing diagram is shown in Figure 31.

FIFO OPERATION

The AD9146 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 the DAC, such as the clock-to-data variation of an FPGA or ASIC, which significantly increases the timing budget of the interface.

Figure 32 shows the block diagram of the datapath through the FIFO. The data is latched into the device, is formatted, and is then 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 incremented every time data is read into the datapath from the FIFO. The FIFO pointers are incremented at the data rate (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. An overflow or empty condition of the FIFO occurs when the write pointer and read pointer point to the same FIFO location. This simultaneous access of data leads to unreliable data transfer through the FIFO and must be avoided.

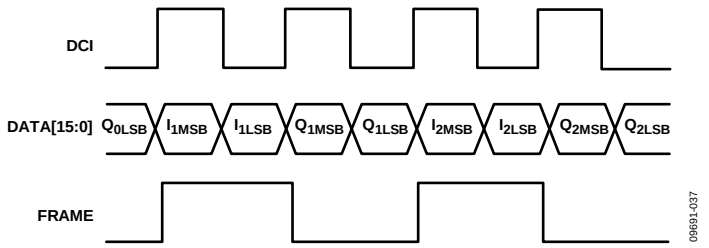

Figure 30. Timing Diagram for Byte Mode

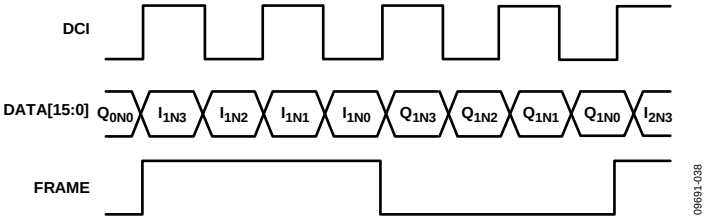

Figure 31. Timing Diagram for Nibble Mode

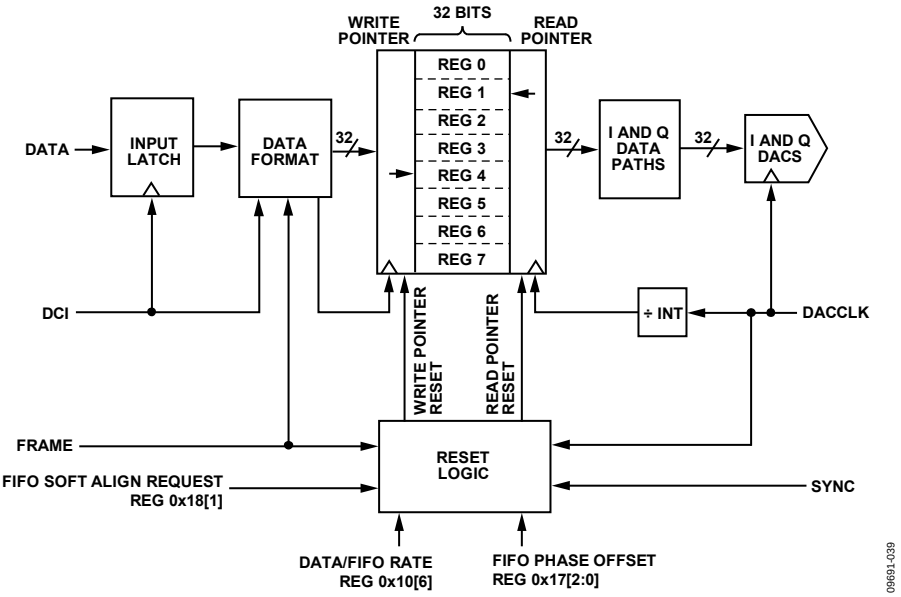

Figure 32. Block Diagram of FIFO

Nominally, data is written to and read from the FIFO at the same rate. This keeps the FIFO depth constant. If data is written to the FIFO faster than data is read out, the FIFO depth increases. If data is read out of the FIFO faster than data is written to it, the FIFO depth decreases. For optimum timing margin, the FIFO depth should be maintained near half full (a difference of 4 between the write pointer and read pointer values). The FIFO depth represents the FIFO pipeline delay and is part of the overall latency of the AD9146.

Resetting the FIFO

When the AD9146 is powered on, the FIFO depth is unknown. To avoid a concurrent read and write to the same FIFO address and to ensure a fixed pipeline delay, it is important to reset 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 reset: 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 it is configured for data rate mode synchronization, the FRAME strobe initiates a relative FIFO reset. The reference point of the relative reset is the position of the read pointer.
- When FIFO mode synchronization is chosen, the FRAME strobe initiates an absolute FIFO reset.

For more information about the synchronization function, see the Multichip Synchronization section.

A summary of the synchronization modes and the types of FIFO reset used is provided in Table 13.

Table 13. Summary of FIFO Resets

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

For a FRAME dependent FIFO reset to occur, an extended FRAME pulse must be sent to the part for proper operation. The extended FRAME pulse must be asserted high for an entire I and Q DAC data sample load. This corresponds to four data clock samples in byte mode and eight data clock samples in nibble mode (see Figure 33 and Figure 34, respectively).

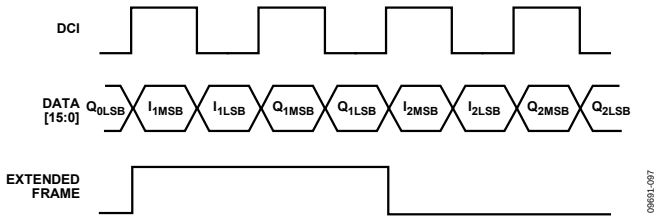

Figure 33. Timing Diagram for Extended Frame Pulse (Byte Mode)

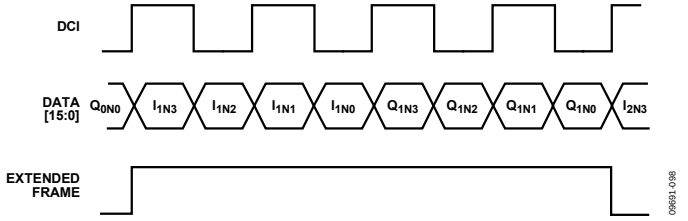

Figure 34. Timing Diagram for Extended Frame Pulse (Nibble Mode)

Serial Port Initiated FIFO Reset

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 back. When the write to this register is complete, the FIFO data level is initialized. When the initialization is triggered, the next time that the read pointer becomes 0, the write pointer is set to the value of the FIFO start level variable (Register 0x17, Bits[2:0]) upon initialization. By default, this value is 4, but it can be programmed to a value from 0 to 7.

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

1. Program Register 0x17 to 0x05.
2. Request FIFO level reset by setting Register 0x18, Bit 1, to 1.
3. Verify that the part acknowledges the request by ensuring that Register 0x18, Bit 2, is set to 1.
4. Remove the request by setting Register 0x18, Bit 1, to 0.
5. Verify that the part drops the acknowledge signal by ensuring that Register 0x18, Bit 2, is set to 0.
6. Read back Register 0x19 to verify that the pointer spacing is set to 3 (0x07) or 4 (0x0F).
7. If the readback of Register 0x19 shows a pointer spacing of 2 (0x03), increment Register 0x17 to a spacing of 0x06 and repeat Step 2 through Step 5. Read back Register 0x19 again to verify that the pointer spacing is now set to 3 (0x07).
8. If the readback of Register 0x19 shows a pointer spacing of 5 (0x1F) after Step 6, decrement Register 0x17 to a spacing of 0x04 and repeat Step 2 through Step 5. Read back Register 0x19 again to verify that the pointer spacing is now set to 4 (0x0F).

FRAME Initiated Relative FIFO Reset

The primary function of the FRAME input is to indicate to which DAC the input data is written. Another function of the FRAME input is to initialize the FIFO data level value. This is done by asserting the FRAME signal high for at least the time interval required to load complete data to the I and Q DACs. This corresponds to four DCI periods in byte mode and eight DCI periods in nibble mode.

To initiate a relative FIFO reset with the FRAME signal, the device must be configured in data rate mode (Register 0x10, Bit 6 = 1). 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 that the read pointer becomes 0 (see Figure 35).

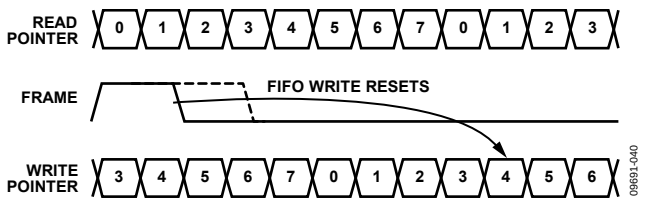

Figure 35. FRAME Input vs. Write Pointer Value, Data Rate Mode

FRAME Initiated Absolute FIFO Reset

In FIFO rate synchronization mode, the write pointer of the FIFO is reset in an absolute manner. The synchronization signal aligns the internal clocks on the part to a common reference clock so that the pipeline delay in the digital circuit stays the same during power cycles. The synchronization signal is sampled by the DAC clock in the AD9146. The edge of the DAC clock used to sample the synchronization 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 required to load complete data to the I and Q DACs. The FIFO write pointer is reset to the value of the FIFO Phase Offset[2:0] bits in Register 0x17. FIFO rate synchronization is selected by setting Bit 6 of Register 0x10 to 0.

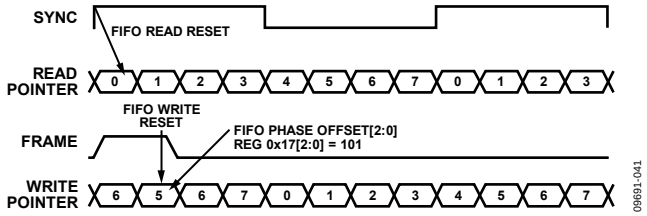

Figure 36. FRAME Input vs. Write Pointer Value, FIFO Rate Mode

Monitoring the FIFO Status

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

The FIFO data level can be read from Register 0x19 at any time. The serial port reported FIFO data level is denoted as a 7-bit thermometer code (Base 1 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.

Note that, depending on the timing relationship between the DCI and the main DACCLK, the FIFO level value can be off by a ±1 count; that is, the readback of Register 0x19 can be 00011111 in the case of a +1 count and 00000111 in the case of a −1 count. Therefore, it is important to keep the difference between the read and write pointers to a value of at least 2.

INTERFACE TIMING

The timing diagram for the digital interface port is shown in Figure 37. When Register 0x16, Bits[2:0] are set to 000, the sampling point of the data bus nominally occurs 165 ps after each edge of the DCI signal and has an uncertainty of ±285 ps, as illustrated by the data valid window shown in Figure 37. The data and FRAME signals must be valid throughout this window. The data and FRAME signals may change at any time between data valid windows.

The setup (t_S) and hold (t_H) times, with respect to the edges, are shown in Figure 37. The minimum setup and hold times are shown in Table 14.

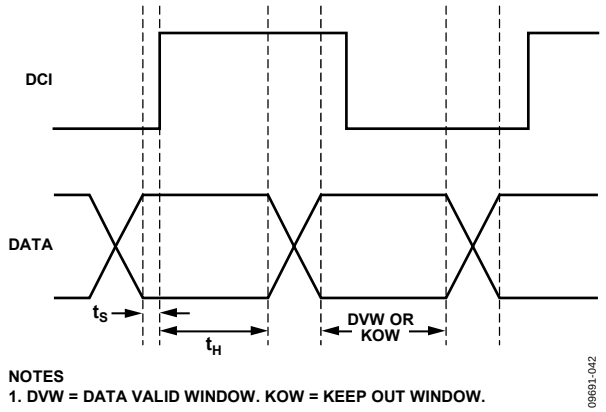

Figure 37. Timing Diagram for Input Data Port

Table 14. Data to DCI Setup and Hold Times

DCI Delay Register 0x16, Bits[1:0]	Minimum Setup Time, t_S (ns)	Minimum Hold Time, t_H (ns)	Sampling Interval (ns)
00	0.12	0.45	0.57
01	−0.01	0.74	0.73
10	−0.2	1.03	0.83
11	−0.28	1.16	0.88

Bypass DCI Delay Mode

An additional option for the timing of the data, DCI, and FRAME signals requires the DCI to be delayed by 90° ahead of the data and FRAME signals. In bypass DCI delay mode, the DCI signal is placed in the optimal data valid window outside the part, and the delay circuitry inside the part is bypassed. This mode provides a smaller sampling window that allows for a wider range of placement area for correct sampling edges. The bypass DCI delay mode is enabled by setting Bit 2 in Register 0x16 to 1. The sampling point of the data bus nominally occurs 90 ps before each edge of the DCI signal and has an uncertainty of ±180 ps, as illustrated by the sampling interval shown in Figure 38. The resulting setup and hold times for this mode are as follows:

- Minimum setup time (t_S): 0.27 ns
- Minimum hold time (t_H): 0.09 ns
- Sampling interval: 0.36 ns

Figure 38 shows the timing for the bypass DCI delay mode.

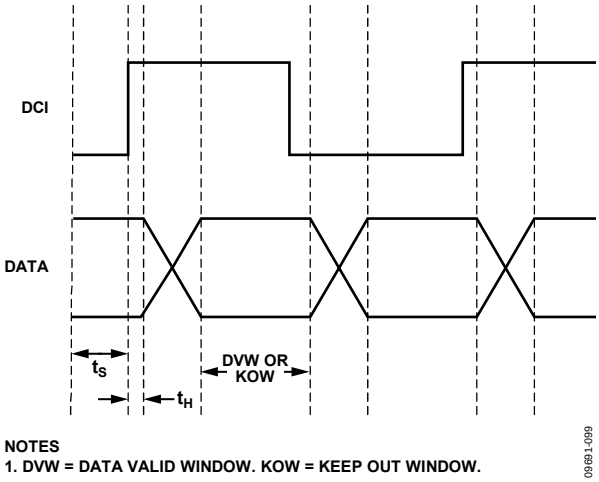

Figure 38. Timing Diagram for Input Data Port (Bypass DCI Delay Mode)

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

DIGITAL DATAPATH

The block diagram in Figure 39 shows the functionality of the digital datapath. The digital processing includes a premodulation block, two half-band (HB) interpolation filters, phase and offset adjustment blocks, and an inverse sinc filter.

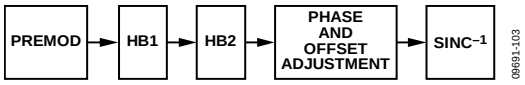

Figure 39. Block Diagram of Digital Datapath

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 digital datapath can also be used to process an input data stream representing two independent real data streams, but the functionality is somewhat restricted. The premodulation block and any of the nonshifted interpolation filter modes can be used for an input data stream representing two independent real data streams. See the Coarse Modulation Mixing Sequences section for more information.

PREMODULATION

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

INTERPOLATION FILTERS

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

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

- Bandwidth of HB1 = 0.8 × f_{IN1}
- Bandwidth of HB2 = 0.5 × f_{IN2}

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 described 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 filter has eight modes of operation.

Half-Band Filter 1 (HB1)

HB1 has four modes of operation, as shown in Figure 40. 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.

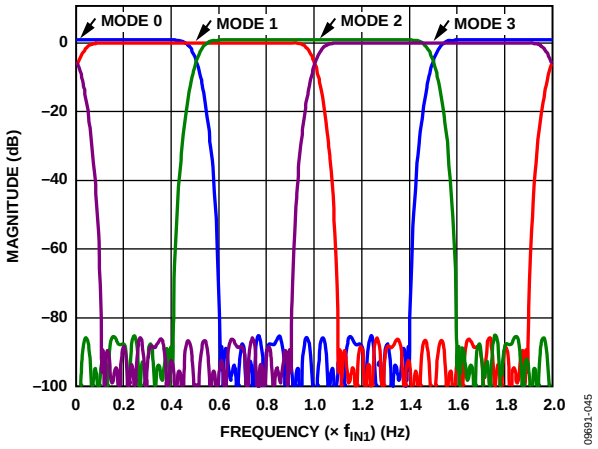

Figure 40. HB1 Filter Modes

As shown in Figure 40, the center frequency in each mode is offset by one-half 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 operating in Mode 0 and Mode 2, the I and Q paths operate independently and no mixing of the data between channels occurs. When operating 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 to be complex. Table 15 summarizes the HB1 modes.

Table 15. HB1 Filter Modes

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 41 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 and not by the pass-band flatness. Table 16 shows the pass-band flatness and the stop-band rejection supported by the HB1 filter at different bandwidths.

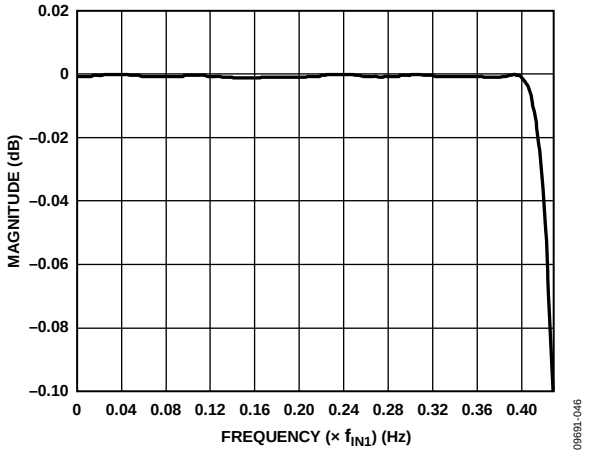

Figure 41. Pass-Band Detail of HB1

Table 16. HB1 Pass-Band and Stop-Band Performance by Bandwidth

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	0.0076	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 42 and Figure 43. 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.

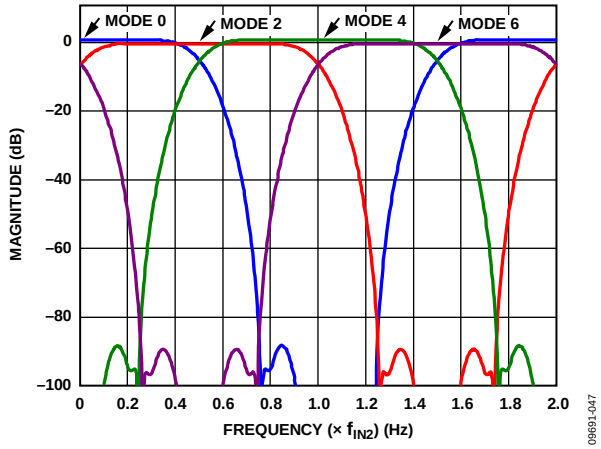

Figure 42. HB2, Even Filter Modes

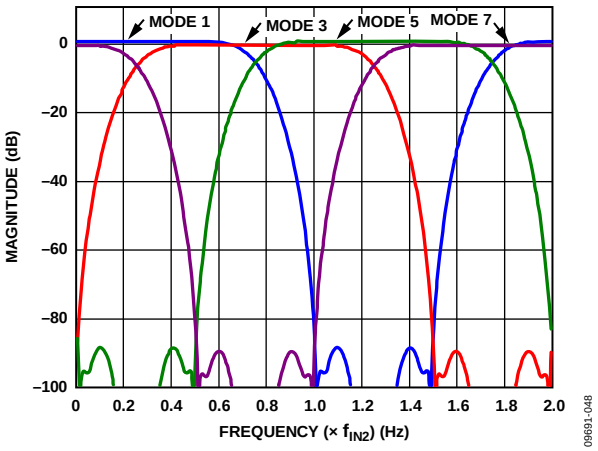

Figure 43. HB2, Odd Filter Modes

As shown in Figure 42 and Figure 43, the center frequency in each mode is offset by one-fourth 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 operating in Mode 0 and Mode 4, the I and Q paths operate independently and no mixing of the data between channels occurs. When operating 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 to be complex. Table 17 summarizes the HB2 modes.

Table 17. HB2 Filter Modes

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	$3f_{IN}/2$	f_{IN}	Complex
7	$7f_{IN}/4$	f_{IN}	Complex

Figure 44 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 and not by the pass-band flatness. Table 18 shows the pass-band flatness and stop-band rejection supported by the HB2 filter at different bandwidths.

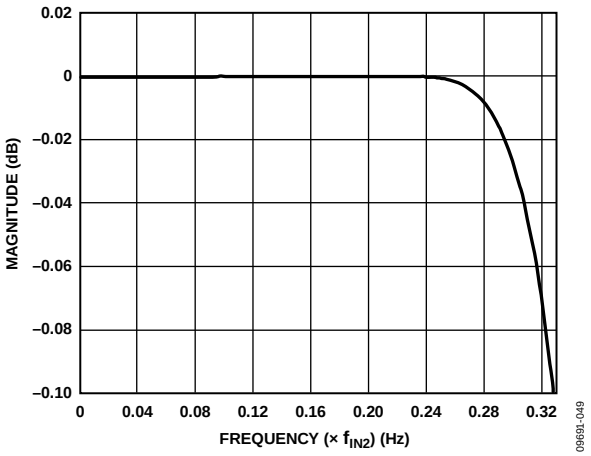

Figure 44. Pass-Band Detail of HB2

Table 18. HB2 Pass-Band and Stop-Band Performance by Bandwidth

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	0.0089	60
60	0.0287	50
64.8	0.1877	40

DATAPATH CONFIGURATION

Configuring the AD9146 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.

DETERMINING INTERPOLATION FILTER MODES

Table 19 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 parameters are known and put in terms of the input data rate (f_{DATA}), the filter configuration that comes closest to matching is selected from Table 19.

Table 19. Recommended Interpolation Filter Modes (Register 0x1C and Register 0x1D)

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

[1] When HB1 Mode 1 or Mode 3 is used, enabling premodulation provides an additional frequency translation of the input signal by $f_{DATA}/2$, which centers a baseband input signal in the filter pass band.

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 the mixing functions shown in Table 20.

Table 20. 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, …
$3f_S/4$	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 $r = \frac{\sqrt{2}}{2}$

As shown in Table 20, the mixing functions of most of the modes cross-couple samples between the I and Q channels. The I and Q channels operate independently only 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.

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 I Phase Adj[9:0] (Register 0x38 and Register 0x39) 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 I Phase Adj[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.

Q Phase Adj[9:0] (Register 0x3A and Register 0x3B) works in a similar fashion. When Q Phase Adj[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 Q Phase Adj[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 45 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 45 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.

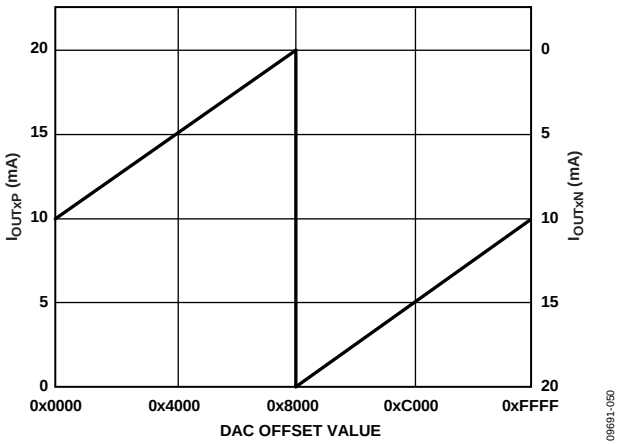

Figure 45. DAC Output Currents vs. DAC Offset Value

INVERSE SINC FILTER

The inverse sinc ($sinc^{-1}$) filter is a nine-tap FIR filter. The composite response of the $sinc^{-1}$ filter and the sin(x)/x response of the DAC is shown in Figure 46. The composite response has a pass-band ripple of less than ±0.05 dB 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 shown have an intrinsic insertion loss of about 3.2 dB. Figure 46 shows the composite frequency response.

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

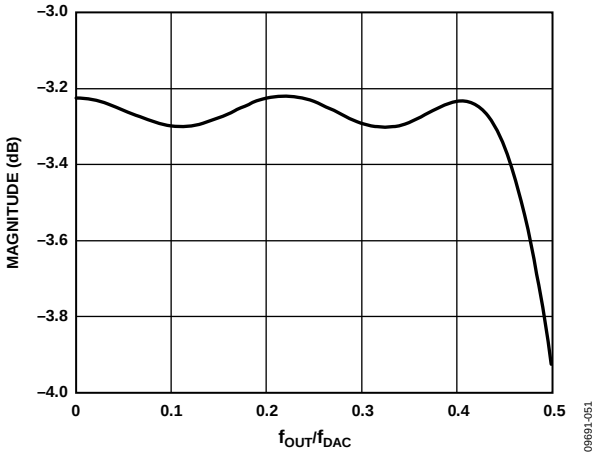

Figure 46. Sample Composite Responses of the $Sinc^{-1}$ Filter with sin(x)/x Roll-Off

DAC INPUT CLOCK CONFIGURATIONS

The AD9146 DAC sampling clock (DACCLK) can be sourced directly or by clock multiplying. Clock multiplying uses the on-chip phase-locked loop (PLL), which 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 then 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 eliminates the need to generate and distribute the high speed DACCLK.

The second mode bypasses the clock multiplier circuitry and allows the 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 differential DACCLK and REFCLK inputs share similar clock receiver input circuitry. Figure 47 shows a simplified circuit diagram of the inputs. 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 direct 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.

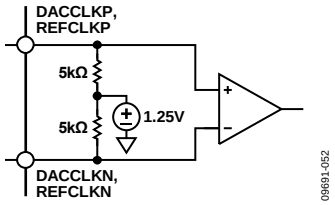

Figure 47. Clock Receiver Input Simplified Equivalent Circuit

The minimum input drive level to either of the clock inputs is 100 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 have 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 CLK 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 sampling 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 are in Register 0x08 (see Table 11).

CLOCK MULTIPLICATION

The on-chip PLL clock multiplication circuit can be used to generate the DAC sampling 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 sampling clock from the lower rate REFCLK input. The functional diagram of the clock multiplier is shown in Figure 48.

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

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

The DAC sampling 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.

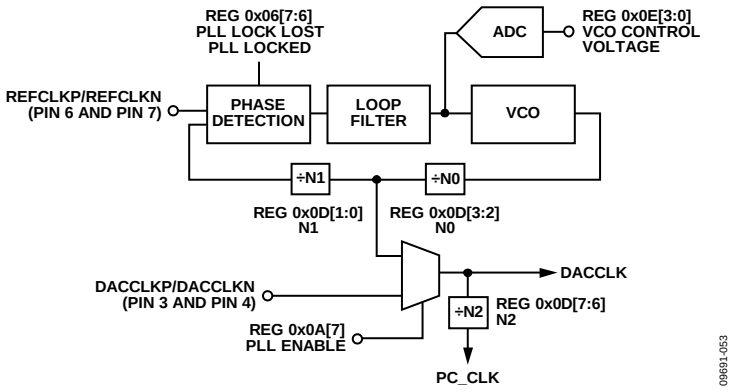

Figure 48. PLL Clock Multiplication Circuit

PLL SETTINGS

Three settings for the PLL circuitry should be programmed to their nominal values. The PLL values shown in Table 21 are the recommended settings for these parameters.

Table 21. PLL Settings

PLL Control Register	Register Address	Bits	Optimal Setting
PLL Loop Bandwidth[1:0]	0x0C	[7:6]	11
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 49. 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.

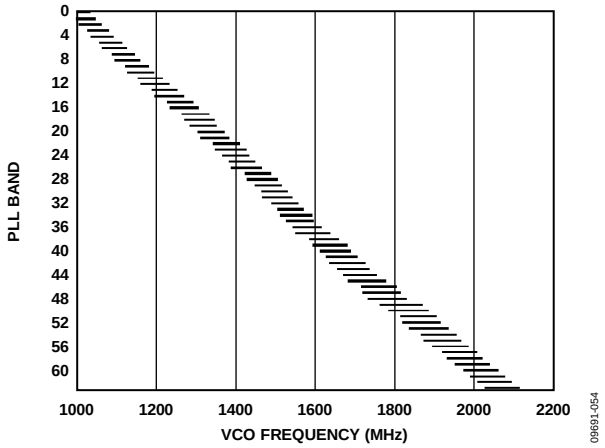

Figure 49. PLL Lock Range over Temperature for a Typical Device

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 starting the PLL in manual mode, then placing the PLL in auto band select mode. This is done by setting Register 0x0A to a value of 0xCF, then to a value of 0xA0. When these values are 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.)

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. In manual mode, the VCO band is set directly with the value written to the manual VCO band bits (Register 0x0A, Bits[5:0]). To properly select the VCO band, follow these steps:

1. Put the device in manual band select mode by setting Register 0x0A, Bit 6 = 1.
2. Sweep the VCO band over a range of bands that results in the PLL being locked.
3. For each band, verify that the PLL is locked and read the PLL using the VCO control voltage bits (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, 1001 or 1000 (see Table 22). The resulting VCO band should be the optimal setting for the device. Write this value to the manual VCO band bits (Register 0x0A, Bits[5:0]).
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 22 shows how to interpret the PLL VCO control voltage value (Register 0x0E, Bits[3:0]).

Table 22. VCO Control Voltage Range Indications

VCO Control Voltage (Register 0x0E, Bits[3:0])	Indication
1111 1110	Move to 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 lower VCO band

ANALOG OUTPUTS

TRANSMIT DAC OPERATION

Figure 50 shows a simplified block diagram of the transmit path DACs. The DAC core consists of a current source array, a switch core, digital control logic, and full-scale output current control. The DAC full-scale output current (I_{FS}) 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.

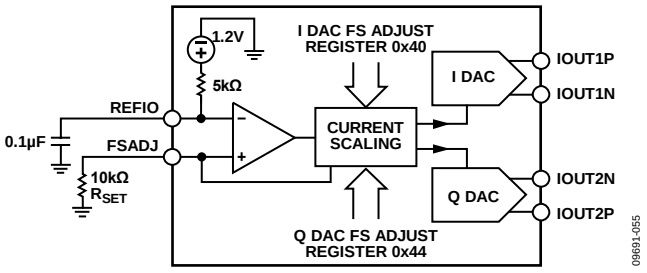

Figure 50. Simplified Block Diagram of DAC Core

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, decouple the REFIO pin to AVSS with a 0.1 µF capacitor. Use the internal reference only 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, the internal reference can be overdriven by applying an external reference (from 1.10 V to 1.30 V) to the REFIO pin.

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, is as follows:

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

For the 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 8.64 mA to 31.68 mA by setting the DAC gain parameter, as shown in Figure 51.

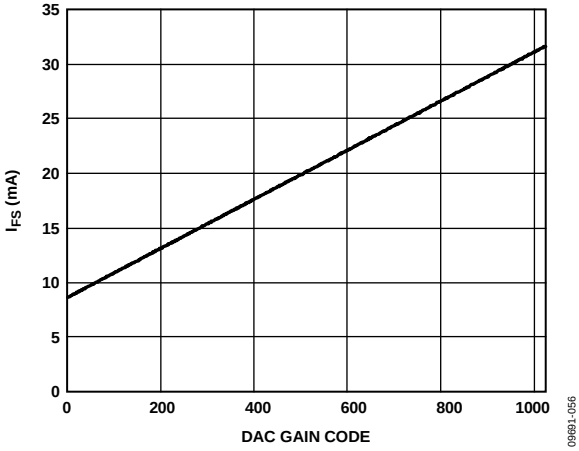

Figure 51. DAC Full-Scale Current vs. DAC Gain Code

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

$$I_{OUTxP} = \left[\frac{DACCODE}{2^N}\right] \times I_{FS} \quad (1)$$

$$I_{OUTxN} = I_{FS} - I_{OUTxP} \quad (2)$$

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

Transmit DAC Output Configurations

The optimum noise and distortion performance of the AD9146 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 52 shows the most basic transmit 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 × R_O. Figure 53 illustrates the output voltage waveforms.

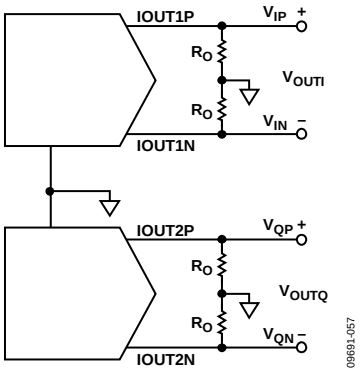

Figure 52. Basic Transmit DAC Output Circuit

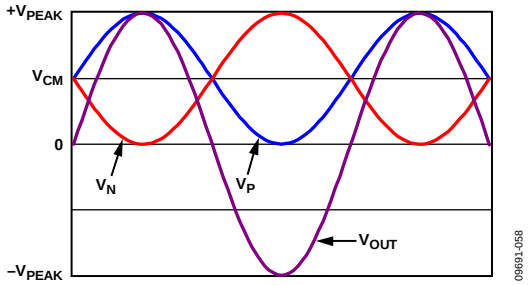

Figure 53. Output Voltage Waveforms

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_{FS}, and the common-mode level of the output. Figure 54 shows the IMD performance vs. the output common-mode voltage at different full-scale currents.

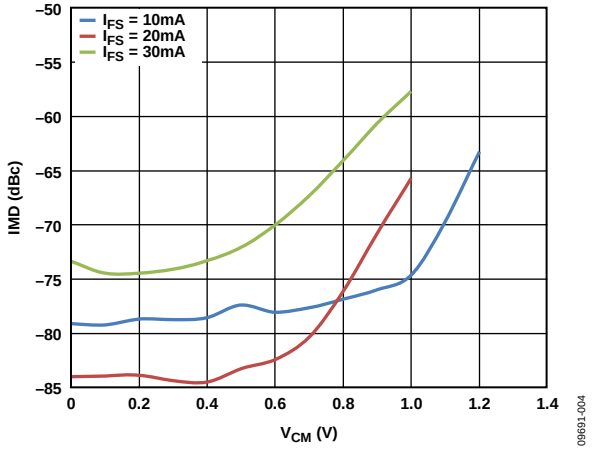

Figure 54. IMD vs. Output Common-Mode Voltage (f_{OUT} = 161 MHz, R_{LOAD} = 50 Ω Differential, I_{FS} = 10 mA, 20 mA, and 30 mA)

AUXILIARY DAC OPERATION

The AD9146 has two auxiliary DACs: one associated with the I path and one 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 positive (P) or negative (N) output of the associated transmit DAC. The auxiliary DAC structure is shown in Figure 55.

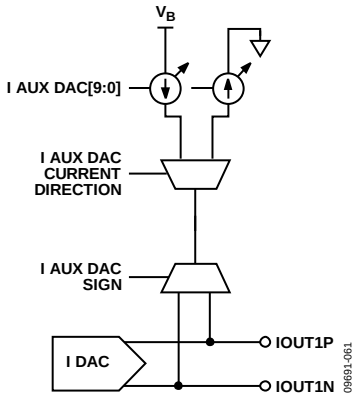

Figure 55. Auxiliary DAC Structure

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

INTERFACING TO MODULATORS

The AD9146 interfaces to the ADL537x family of modulators with a minimal number of components. An example of the recommended interface circuitry is shown in Figure 56.

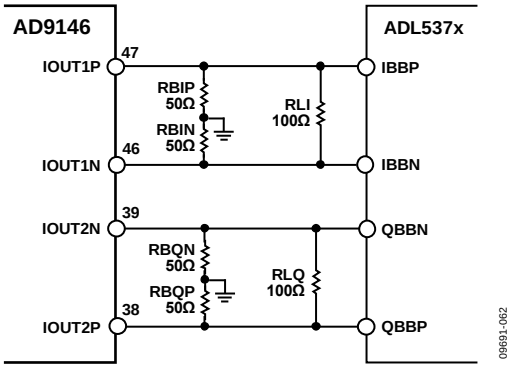

Figure 56. Typical Interface Circuitry Between the AD9146 and the ADL537x Family of Modulators

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 (one-half the full-scale current). Therefore, a single 50 Ω resistor to ground from each of the DAC outputs 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

$$V_{SIGNAL} = I_{FS} \times \frac{(2 \times R_B \times R_L)}{(2 \times R_B + R_L)}$$

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 establishes the input and output impedances for the filter.

Figure 58 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 requires a 1500 mV dc bias and, therefore, requires a slightly more complex interface than most other Analog Devices modulators. It is necessary to level-shift the DAC output from a 500 mV dc bias to the 1500 mV dc bias required by the ADL5375-15. Level-shifting can be achieved with a purely passive network, as shown in Figure 57. In this network, the dc bias of the DAC remains at 500 mV, whereas the input to the ADL5375-15 is 1500 mV. This passive, level-shifting network introduces approximately 2 dB of loss in the ac signal.

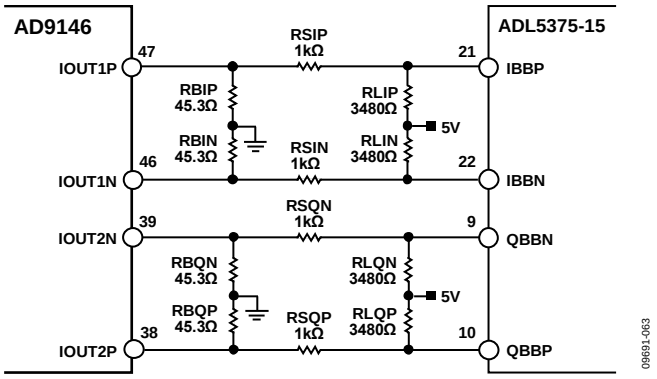

Figure 57. Passive, Level-Shifting Network for Biasing the ADL5375-15

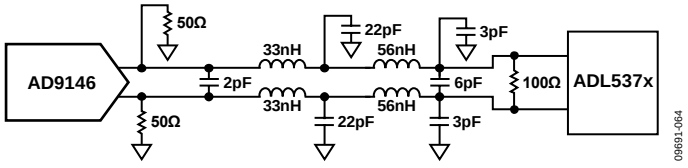

Figure 58. DAC Modulator Interface with Fifth-Order, Low-Pass Filter

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. 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/Q phase adjust registers (Register 0x38 through Register 0x3B) and the DAC FS adjust registers (Register 0x40 and Register 0x44) can be used to calibrate the I and Q transmit paths to optimize sideband suppression.

DEVICE POWER MANAGEMENT

POWER DISSIPATION

The AD9146 has three supply rails: AVDD33, 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 54 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 affects 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 34 mA.

The DVDD18 supply powers all of the digital signal processing blocks of the device. The serial port I/O pins, the $\overline{\text{RESET}}$ pin, and the $\overline{\text{IRQ}}$ pin are also supplied from the DVDD18 power supply. 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 59 through Figure 61 show the power dissipation of the AD9146 under a variety of operating conditions. All of the graphs were taken with data being supplied to both the I and Q DACs. The power consumption of the device does not vary significantly with changes in the coarse modulation mode selected or with the analog output frequency. Figure 59 shows the total power dissipation. Figure 60 and Figure 61 show the power dissipation of the DVDD18 and CVDD18 supplies.

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

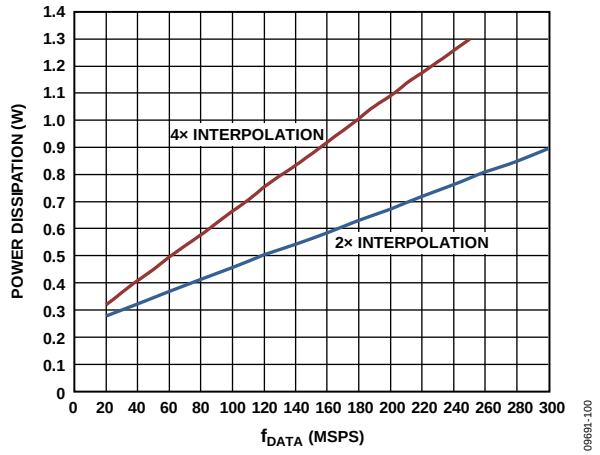

Figure 59. Total Power Dissipation vs. f_{DATA} Without PLL and Inverse Sinc

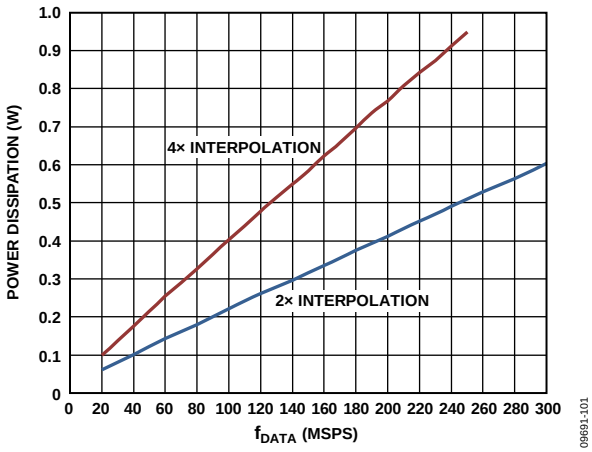

Figure 60. DVDD18 Power Dissipation vs. f_{DATA} Without Inverse Sinc

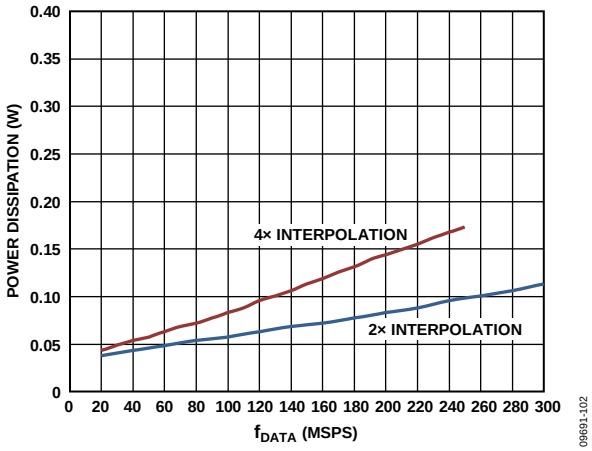

Figure 61. CVDD18 Power Dissipation vs. f_{DATA} with PLL Disabled

Tx ENABLE

The Tx enable feature provides additional power management techniques that can be implemented in system applications. The TXENABLE pin, when taken to a logic low, stops the transmission of data from the part and clamps the outputs to midscale. In addition, various portions of the DAC can be powered down while the pin is held low, depending on the power saving requirements of the system and the amount of wake-up time required when the pin is brought high.

Register 0x02 contains the bit controls to power down these individual blocks: DAC cores, FIFO, interpolation filters, PLL, and the internal reference. Depending on the power-down bits selected, the necessary wake-up time and reprogramming of the DAC may vary.

The Tx enable feature also allows for an extended delay from when the TXENABLE pin is brought high to when the DAC outputs begin transmitting the data present in the FIFO and datapath. Two different delay lengths are available. These delays allow the part to be set up properly during the delay time without transmitting false data and to begin receiving correct data after the datapath is flushed. The amount of delay time to be allotted for various wake-up times depends on the delay setting used, as well as which portions of the DAC are powered down and need to be reinitialized.

Table 23 lists the minimum wait time required for the DAC to begin transmitting again after the TXENABLE pin is brought high. Regardless of the delay setting, there is an inherent fixed delay of 10 DAC clock cycles for all the options listed in Table 23 before the DAC begins transmitting. Additionally, because the Tx enable logic is timed from a divided-down rate of the DAC clock—specifically, DAC/64—the number of edges that the part waits for before allowing data to be transmitted from the DAC can vary. Because the synchronization between the DAC/64 clock and the Tx enable logic trigger is unknown, the number of DAC/64 clock edges that must be waited for before the outputs are released can vary by up to one cycle.

Table 23. Wake-Up Time for Various Tx Enable Delay Settings

Register 0x02	Number of DAC/64 Edges to Wait[1]	Additional DAC Edges to Wait	Minimum Wait Time[2]
No extended delay (0x00)	1	10	360.82 ns
Extended Delay 0 (0x20)	12	10	4.18 µs
Extended Delay 1 (0x60)	19	10	6.611 µs

[1] Values may vary by up to one DAC/64 cycle for the amount of wake-up time of each delay setting.
[2] Values based on 737.28 MHz DAC rate condition; uses (number of DAC/64 + 10 DAC clocks) for calculation.

For timing purposes and to ensure that incorrect data is flushed, the minimum wake-up time must be considered. This constraint determines how soon the datapath must begin to be flushed. Depending on which portions of the DAC are powered down using the Tx enable feature, the amount of time required to start setting up the part and flushing the datapaths must be adjusted. An appropriate delay setting is required to accommodate the earliest possible wake-up time needed for flushing before the outputs are enabled.

In addition to the delays listed in Table 23, specific wake-up times for individual powered-down portions of the AD9146 must be accounted for during the preparation time.

The following example provides a typical configuration that uses the Tx enable feature to power down the interpolation filters. This example provides guidelines for how to determine the amount of wake-up time to design in a system.

- f_{DATA} = 184.32 MHz
- f_{DAC} = 737.28 MHz
- Interpolation = 4×
- Inverse sinc on
- Tx enable filter power-down option selected
- Datapath flush time = 175 DAC clocks
- t_{DAC} = 1.36 ns
- $t_{DPFLUSH}$ = 238 ns

The minimum wake-up time with no delay setting is 360.82 ns (see Table 23). In this example, the time required to flush the datapath is only 238 ns. Therefore, if datapath flushing is done simultaneous to the TXENABLE pin being brought high, there is enough time for the flush to complete before the minimum possible time that the outputs can begin transmitting. For each individual case, the amount of time needed to flush the datapath must be accounted for when calculating the minimum time after which the DACs can begin transmitting data.

The TXENABLE pin must be held high while the part is being powered up. After the part is powered up, the pin can be brought low to clamp the outputs, when desired. Note that the pin cannot be held low during power-up because the circuit logic is transition sensitive and the part must see a falling edge before it clamps the outputs.

TEMPERATURE SENSOR

The AD9146 has a band gap temperature sensor for monitoring the temperature change of the AD9146. The temperature must be calibrated against a known temperature to remove the part-to-part variation on the band gap circuit used to sense the temperature. The DACCLK must be running at a minimum of 100 MHz to obtain a reliable temperature measurement.

To monitor temperature change, the user must take a reading at a known ambient temperature for a single-point calibration of each AD9146 device.

$$Tx = T_{REF} + 7.7 \times (Code_x - Code_ref)/1000 + 1$$

where:
Code_x is the readback code at the unknown temperature, *Tx*.
Code_ref is the readback code at the calibrated temperature, T_{REF}.

To use the temperature sensor, it must be enabled by setting Register 0x01, Bit 4, to 0. In addition, to obtain accurate readings, the die temperature 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 beamforming, 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 machines 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 fixed and known relationship between the clock generation state machine and the data being read from the FIFO and a particular clock edge of the system clock. The AD9146 has provisions for enabling multiple devices to be synchronized to each other or to a system clock.

The AD9146 supports synchronization in two different modes: data rate mode and FIFO rate mode. In data rate mode, the input data rate represents the lowest synchronized clock rate. 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 the setup and hold time windows for DCI changes relative to the DACCLK or REFCLK input. When the synchronization state machine is on 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 setup and hold time windows repeating at the input data rate.

The method chosen for providing the DAC sampling clock directly affects 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. The following sections describe the synchronization methods for enabling both clocking modes and querying the status of the synchronization logic.

The full synchronization methods described are used to align multiple dual DACs within one DACCLK cycle. To achieve synchronization within one DACCLK cycle, both the REFCLK and FRAME signals are required to perform back-end and front-end alignment. If synchronization does not need to be this accurate, other options can be used. In data rate mode or in FIFO rate mode, using soft alignment of the FIFO for multiple DACs synchronizes the DAC outputs within two data clock cycles (see the Serial Port Initiated FIFO Reset section). For more information about synchronization, see the AN-1093 Application Note, "Synchronization of Multiple AD9122 TxDAC+ Converters."

SYNCHRONIZATION WITH CLOCK MULTIPLICATION

When using the clock multiplier to generate the DAC sample rate clock, the REFCLK input signal acts as both 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 the devices that need to be synchronized. Skew between the REFCLK signals of the different devices shows up directly as a timing mismatch at the DAC outputs.

Because two clocks are shared on the same signal, an appropriate frequency must be chosen for the synchronization and REFCLK signals. The FRAME and DCI signals can be created in the FPGA along with the data. A circuit diagram of a typical configuration is shown in Figure 62.

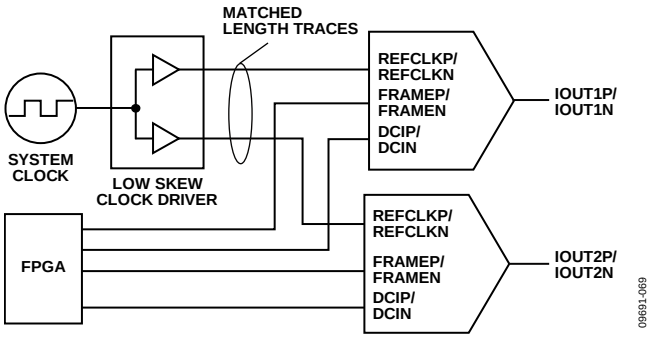

Figure 62. Typical Circuit Diagram for Synchronizing Devices

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 the devices, and that the PLL of each device is phase locked to it. The following procedure must be carried out on each individual device.

Procedure for Synchronization When Using the PLL

In the initialization of the AD9146, all the clock signals (DACCLK, DCI, FRAME, synchronization, and REFCLK) must be present and stable before the synchronization feature is turned on. Configure the AD9146 for data rate, periodic synchronization by writing 0xC8 to the sync control register (Register 0x10). Additional synchronization options are available (see the Additional Synchronization Features section).

Read the sync status register (Register 0x12) to 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 and known phase relative to the synchronization signal.

Reset the FIFO by strobing the FRAME signal high for the time interval required to write two complete input data words. Resetting the FIFO ensures that the correct data is being read from the FIFO.

This completes the synchronization procedure; all devices should now be synchronized.

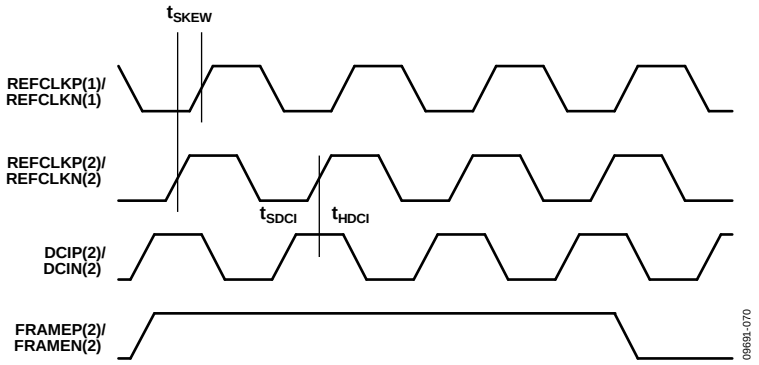

Figure 63. Timing Diagram Required for Synchronizing Devices

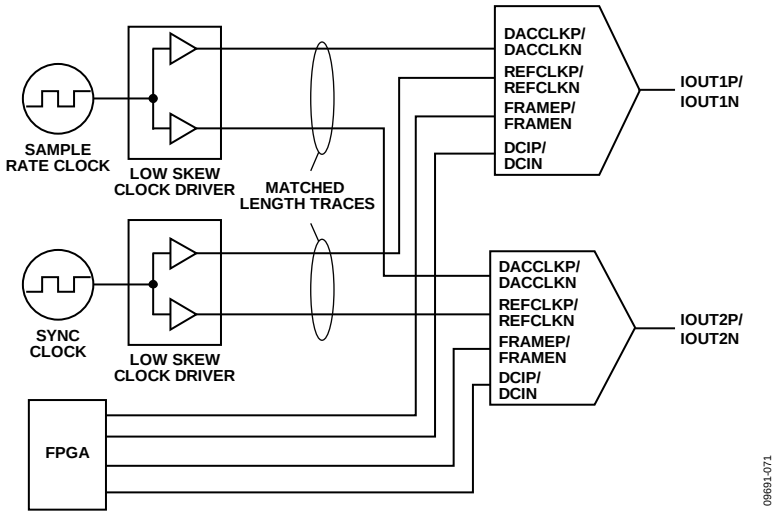

Figure 64. Typical Circuit Diagram for Synchronizing Devices to a System Clock

To maintain synchronization, the skew between the REFCLK signals of the devices must be less than t_{SKEW} ns. 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 63.

Figure 63 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. (This equation is valid only when the PLL is used because only data rate mode is available with the PLL on.)

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

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.

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 the devices being synchronized. If the devices need to be synchronized to a master clock, use the master clock directly for generating the REFCLK input (see Figure 64).

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 the devices. The following procedure must be carried out on each individual device.

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

Configure the AD9146 for data rate, periodic synchronization by writing 0xC8 to the sync control register (Register 0x10). Additional synchronization options are available (see the Additional Synchronization Features section).

Read 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 synchronization signal.

Reset the FIFO by strobing the FRAME signal high for two complete DCI periods. Resetting the FIFO ensures that the correct data is being read from the FIFO of each of the devices simultaneously.

This completes the synchronization procedure; all devices should now be synchronized.

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.

- DCIP/DCIN and D[7:0]P/D[7:0]N must meet the setup and hold times with respect to the rising edge of DACCLK.
- REFCLK must also meet the setup and hold times with respect to the rising edge of DACCLK.

When these conditions are met, the outputs of the DACs are updated within one DAC clock cycle of each other. The timing requirements of the input signals are shown in Figure 65.

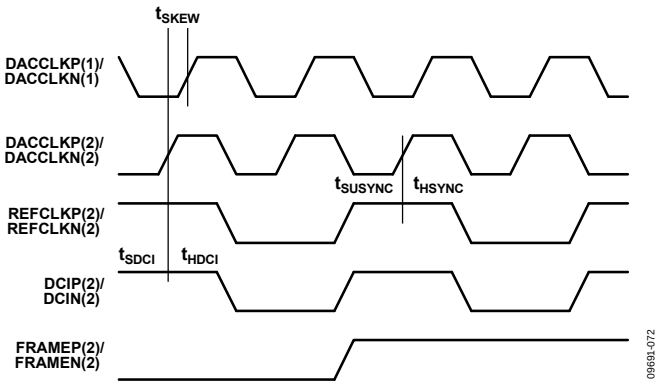

Figure 65. Data Rate Synchronization Signal Timing Requirements, 2× Interpolation

Figure 65 shows the synchronization signal timing with 2× interpolation; therefore, $f_{DCI} = ½ \times f_{CLK}$. 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 non-negative integer.

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

When synchronization is used in data rate mode, the timing constraint between the DCI and DACCLK must be met according to Table 24. In data rate mode, the allowed phase drift between the DCI and DACCLK is limited to one DCI period. The DCI to DACCLK timing restriction is required to prevent corruption of the data transfer when the FIFO is constantly reset. The required timing between the DCI and DACCLK is shown in Figure 66.

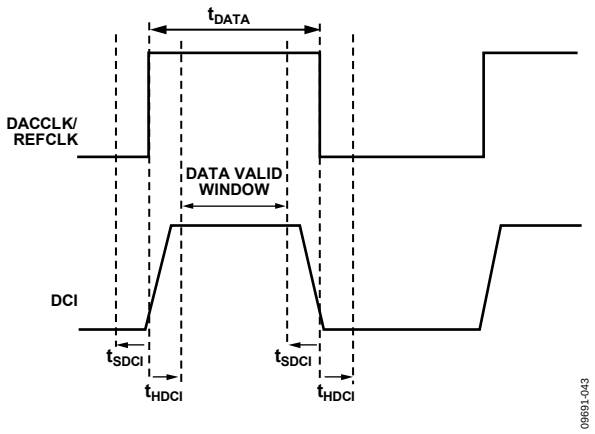

Figure 66. Timing Diagram for Input Data Port (Data Rate Mode)

Table 24. DCI to DACCLK Setup and Hold Times

DCI Delay Register 0x16, Bits[1:0]	Minimum Setup Time, t_{SDCI} (ns)	Minimum Hold Time, t_{HDCI} (ns)	Sampling Interval (ns)
00	−0.07	0.82	0.75
01	−0.24	1.13	0.89
10	−0.39	1.40	1.01
11	−0.49	1.55	1.06

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 DACCLK and REFCLK signals are applied to all the devices. The procedure must be carried out on each individual device.

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

Configure the AD9146 for FIFO rate, periodic synchronization by writing 0x88 to the sync control register (Register 0x10). Additional synchronization options are available (see the Additional Synchronization Features section).

Read 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 synchronization signal.

Reset the FIFO by strobing the FRAME signal high for two complete DCI periods. Resetting the FIFO ensures that the correct data is being read from the FIFO of each of the devices simultaneously.

This completes the synchronization procedure; all devices should now be synchronized.

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.

- DCIP/DCIN and D[7:0]P/D[7:0]N must meet the setup and hold times with respect to the rising edge of DACCLK.
- REFCLK must also meet the setup and hold times with respect to the rising edge of DACCLK.

When these conditions are met, the outputs of the DACs are updated within one DAC clock cycle of each other. The timing requirements of the input signals are shown in Figure 67.

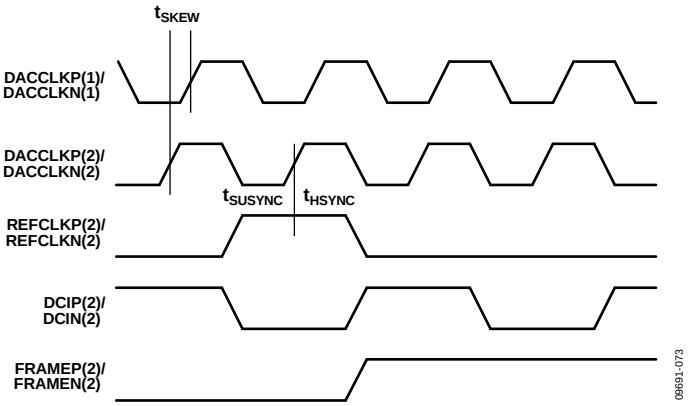

Figure 67. FIFO Rate Synchronization Signal Timing Requirements, 2× Interpolation

Figure 67 shows the synchronization signal timing with 2× interpolation; therefore, $f_{DCI} = ½ \times f_{CLK}$. The REFCLK input is shown to be equal to the FIFO rate. The maximum frequency at which the device can be resynchronized in FIFO rate mode can be expressed as

$$f_{SYNC_I} = f_{DATA}/(8 \times 2^N)$$

where N is any non-negative integer.

ADDITIONAL SYNCHRONIZATION FEATURES

Table 25 shows the required timing between the DACCLK and the synchronization clock when synchronization is used. This timing restriction applies to both data rate mode and FIFO rate mode.

Table 25. Synchronization Setup and Hold Times

Parameter	Min	Max	Unit
t_{SKEW}	$-t_{DACCLK}/2$	$+t_{DACCLK}/2$	ps
t_{SUSYNC}	100		ps
t_{HSYNC}	330		ps

One-Time Synchronization

When implementing the full multichip synchronization feature (with the REFCLK and FRAME signals aligned within one DACCLK cycle), the user may experience difficulty meeting the DACCLK to synchronization clock timing. In this case, a one-time synchronization method can be used. Before implementing the one-time synchronization, make sure that the synchronization signal is locked by checking both the sync signal locked and the sync signal lost flags (Bit 4 and Bit 5 in Register 0x06). It is also important that synchronization not be enabled before stable REFCLK signals are present from the FPGA or ASIC. For more information and a detailed flowchart of the one-time synchronization feature, see the AN-1093 Application Note, "Synchronization of Multiple AD9122 TxDAC+ Converters."

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.

Alignment takes from (11 + averaging) × 64 to (11 + averaging) × 128 DACCLK cycles. The sync locked bit can also trigger an $\overline{IRQ}$, as described in the Interrupt Request Operation section.

When the sync lost bit (Register 0x12, Bit 7) is set, it indicates that a previously synchronized device has lost alignment. This bit is latched and remains set until cleared by overwriting the register. This bit can also trigger an $\overline{IRQ}$, 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 synchronization 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 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. Note that, if the average synchronization sampling result is in approximately the middle of the probability curve, the synchronization engine can be unstable, resulting in corrupted output.

The value of the Sync Phase Request[5:0] 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 of the Sync Phase Request[5:0] value advances the internal clocks by one DACCLK cycle. This offset can be used for two purposes: to skew the outputs of two synchronized DAC outputs in increments of the DACCLK cycle, and to change the relative timing between the DAC output and the sync input (REFCLK). This may allow for a more optimal placement of the DCI sampling point in data rate synchronization mode.

INTERRUPT REQUEST OPERATION

The AD9146 provides an interrupt request output signal on Pin 34 ($\overline{IRQ}$) 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.

The event flags provide visibility into the device. These flags are located in the two event flag registers, Register 0x06 and Register 0x07. The behavior of each event flag 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 monitors the source signal, but the $\overline{IRQ}$ pin remains inactive.

Figure 68 shows the $\overline{IRQ}$-related circuitry and how the event flag signals propagate to the $\overline{IRQ}$ output. The INTERRUPT_ ENABLE signal represents one bit from the interrupt enable register. The EVENT_FLAG_SOURCE signal represents one bit from the event flag register. 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 version of the EVENT_FLAG_ SOURCE signal; that is, the event flag bit is latched on the rising edge of the EVENT_FLAG_SOURCE signal. This signal also asserts the external $\overline{IRQ}$ pin.

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

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. A hardware or software reset also clears the INTERRUPT_SOURCE signal.

INTERRUPT SERVICE ROUTINE

Interrupt request management starts by selecting the set of event flags that require host intervention or monitoring. The 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 may be required to clear the EVENT_FLAG_SOURCE. In many cases, no specific actions may be required.
4. Read the event flag to verify that the actions taken have cleared the EVENT_FLAG_SOURCE.
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 EVENT_FLAG_SOURCE signals are latched signals. These signals are cleared by writing to the corresponding event flag bit. For more information about each event flag, see Register 0x06 and Register 0x07 in Table 11.

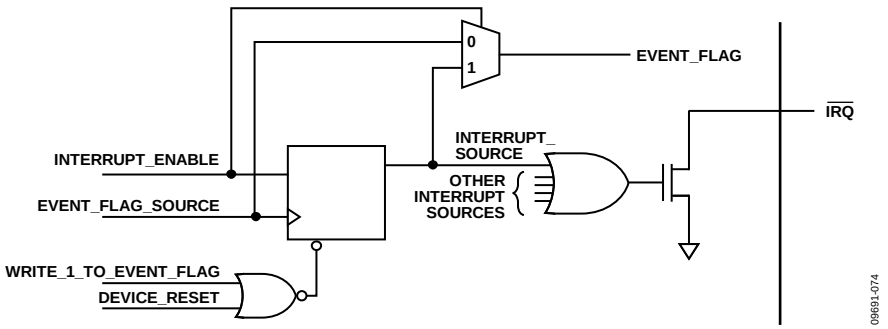

Figure 68. Simplified Schematic of $\overline{IRQ}$ Circuitry

INTERFACE TIMING VALIDATION

The AD9146 provides on-chip sample error detection (SED) circuitry that simplifies verification of the input data interface. The SED circuitry compares the input data samples captured at the digital input pins with a set of comparison values. The comparison values are loaded into registers through the SPI port. Differences between the captured values and the comparison 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 divided into eight 8-bit input words, denoted as I0, Q0, I1, and Q1. To properly align the input samples, the first I data-word (that is, I0) is indicated by asserting FRAME for at least one complete input sample.

Figure 69 shows the input timing of the interface in byte 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.

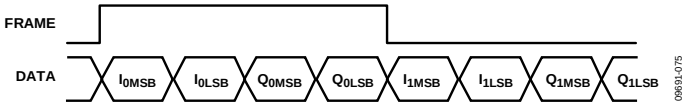

Figure 69. Timing Diagram of Extended FRAME Signal Required to Align Input Data for SED

The SED has three flag bits (Register 0x67, Bit 5, Bit 1, and Bit 0) 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 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.

Autosample error detection (AED) is an autoclear function in the SED. 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 eight sample set comparisons are error free. The compare fail bit is set if an error is detected. The compare fail bit is not activated until the part has received eight error-free sample set comparisons, that is, the pass bit has gone high at least once. Once enabled, 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 are reset to all 0s after eight consecutive error-free sample set comparisons are made.

If desired, the sample error detected, compare pass, and compare fail flags can be configured to trigger the $\overline{\text{IRQ}}$ pin when active. This is done by enabling the appropriate bits in the event flag register (Register 0x07).

Table 26 shows a progression of the input sample comparison results and the corresponding states of the error flags.

Table 26. Progression of Input Sample Comparison Results 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	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Register 0x67, Bit 0 (Compare Pass)	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Register 0x70 to Register 0x73 (Errors Detected Bits)	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.

SED EXAMPLE

Normal Operation

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

1. Load the following comparison values. (Comparison values can be chosen arbitrarily; however, choosing values that require frequent bit toggling provides the most robust test.)
   Register 0x68: I_{0LSB}
   Register 0x69: I_{0MSB}
   Register 0x6A: Q_{0LSB}
   Register 0x6B: Q_{0MSB}
   Register 0x6C: I_{1LSB}
   Register 0x6D: I_{1MSB}
   Register 0x6E: Q_{1LSB}
   Register 0x6F: Q_{1MSB}
2. Enable the SED error detect flag to assert the $\overline{\text{IRQ}}$ pin. (Set Register 0x05 to 0x04.)
3. Begin transmitting the input data pattern.
4. Write to Register 0x67 to enable the SED. (Set Register 0x67 to 0x80.)
5. Clear the SED errors in Register 0x67 and Register 0x07. When the SED is first turned on, the FRAME signal may be detected immediately; therefore, the SED failure bit may be asserted due to the unknown initial FRAME status. For this reason, the SED compare fail status bit must be cleared at least once immediately after enabling the SED.

If $\overline{\text{IRQ}}$ 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 only the errors that caused the error detected flag to be set).

Enabling the alignment of the I0 sample as described in the SED Operation section requires the use of the FRAME signal. The timing diagrams for byte and nibble modes are the same as during normal operation and are shown in Figure 33 and Figure 34, respectively.

EXAMPLE START-UP ROUTINE

To ensure reliable start-up of the AD9146, certain sequences should be followed. This section shows an example start-up routine. This example uses the configuration described in the Device Configuration section.

DEVICE CONFIGURATION

The following device configuration is used for this example.

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

DERIVED PLL SETTINGS

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 GHz)
- $N1 = f_{DACCLK}/f_{REFCLK} = 4$
- $N2 = f_{VCO}/f_{DACCLK} = 4$

START-UP SEQUENCE

The following sequence configures the power clock and register write sequencing for reliable device start-up.

```
Power up Device (no specific power supply
sequence is required)
Apply stable REFCLK input signal.
Apply stable DCI input signal.

Device Configuration Register Write Sequence:
0x00 → 0x20  /* Issue Software Reset */
0x00 → 0x80  /* Enable 3-wire SPI */
0x1E → 0x01

/* Start PLL */
0x0C → 0xE1
0x0D → 0xD9
0x0A → 0xCF
0x0A → 0xA0

/* Verify PLL is Locked */
Read 0x0E   /* Expect bit 7 = 0, bit 6 = 1 */
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 → 0xA4  /* Enable Inverse Sinc */

/* Configure Interpolation Filters */
0x1C → 0x04
0x1D → 0x24
```

OUTLINE DIMENSIONS

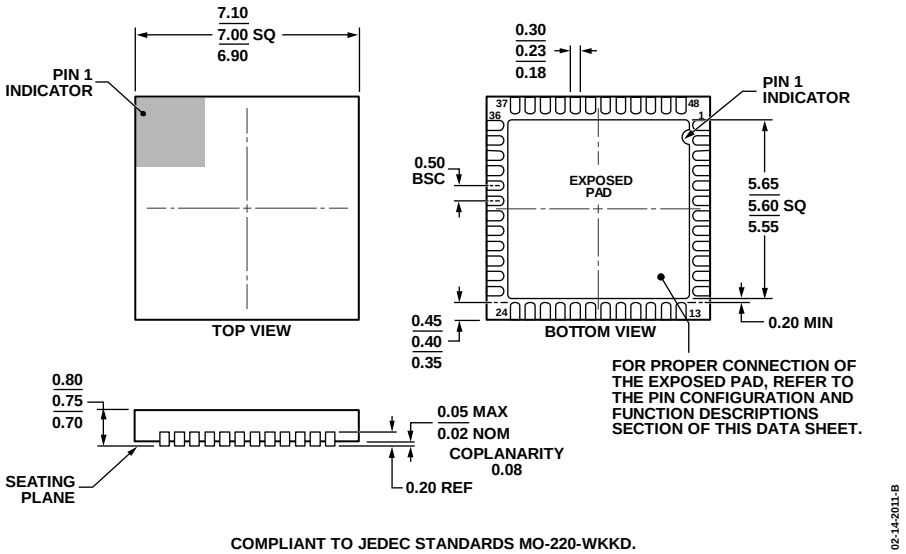

Figure 70. 48-Lead Lead Frame Chip Scale Package [LFCSP_WQ]
7 mm × 7 mm Body, Very Very Thin Quad
(CP-48-13)
Dimensions shown in millimeters

ORDERING GUIDE

Model[1]	Temperature Range	Package Description	Package Option
AD9146BCPZ	−40°C to +85°C	48-lead LFCSP_WQ	CP-48-13
AD9146BCPZRL	−40°C to +85°C	48-lead LFCSP_WQ	CP-48-13
AD9146-M5375-EBZ		Evaluation Board Connected to ADL5375 Modulator	

[1] Z = RoHS Compliant Part.

NOTES

NOTES

NOTES

©2011–2012 Analog Devices, Inc. All rights reserved. Trademarks and registered trademarks are the property of their respective owners.
D09691-0-1/12(A)

OCEAN CHIPS
Океан Электроники
Поставка электронных компонентов

Компания «Океан Электроники» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Поставка оригинальных импортных электронных компонентов напрямую с производств Америки, Европы и Азии, а так же с крупнейших складов мира;
- Широкая линейка поставок активных и пассивных импортных электронных компонентов (более 30 млн. наименований);
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Помощь Конструкторского Отдела и консультации квалифицированных инженеров;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Поставка электронных компонентов под контролем ВП;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- При необходимости вся продукция военного и аэрокосмического назначения проходит испытания и сертификацию в лаборатории (по согласованию с заказчиком);
- Поставка специализированных компонентов военного и аэрокосмического уровня качества (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Actel, Aeroflex, Peregrine, VPT, Syfer, Eurofarad, Texas Instruments, MS Kennedy, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Компания «Океан Электроники» является официальным дистрибьютором и эксклюзивным представителем в России одного из крупнейших производителей разъемов военного и аэрокосмического назначения «**JONHON**», а так же официальным дистрибьютором и эксклюзивным представителем в России производителя высокотехнологичных и надежных решений для передачи СВЧ сигналов «**FORSTAR**».

AVIC JONHON

«JONHON» (основан в 1970 г.)

Разъемы специального, военного и аэрокосмического назначения:

(Применяются в военной, авиационной, аэрокосмической, морской, железнодорожной, горно- и нефтедобывающей отраслях промышленности)

«FORSTAR» (основан в 1998 г.)

ВЧ соединители, коаксиальные кабели, кабельные сборки и микроволновые компоненты:

(Применяются в телекоммуникациях гражданского и специального назначения, в средствах связи, РЛС, а так же военной, авиационной и аэрокосмической отраслях промышленности).

FORSTAR®

Телефон: 8 (812) 309-75-97 (многоканальный)
Факс: 8 (812) 320-03-32
Электронная почта: ocean@oceanchips.ru
Web: http://oceanchips.ru/
Адрес: 198099, г. Санкт-Петербург, ул. Калинина, д. 2, корп. 4, лит. А