CoreadDvices
承芯微电子

CD97D39BG

14 bit, 2.5 GSPS, RF digital to analog converter

Version: Rev 1.0.0 Date: 2025-7-1

Features

- Resolution: 14 bits
- Direct RF synthesis at 2.5 GSPS update rate
  - DC to 1.25 GHz in baseband mode
  - 1.25 GHz to 3.0 GHz in mix mode
- SFDR≥55dBc@fout=950MHz
- Dual-port LVDS data interface
  - Up to 1.25 GSPS operation
  - Source synchronous DDR clocking
- Multichip synchronization capability
- Programmable output current: 8.7 mA

Application

- Broadband communications systems
- Military broadband electronic system
- Instruments and automatic test equipment
- Radar and aviation equipment
- CMTS system equipment

Description

The CD97D39 is a 14-bit, 2.5 GSPS high performance RF digital to-analog converter (DAC) capable of synthesizing wideband signals from dc up to 3.0 GHz. Its DAC core features a quad switch architecture that provides exceptionally low distortion performance with an industry-leading direct RF synthesis capability. This feature enables multicarrier generation up to the Nyquist frequency in baseband mode as well as second and third Nyquist zones in mix mode. The output current can be programmed over the 8.7 mA to 31.7 mA range. The inclusion of on-chip controllers simplifies system integration. A dual-port, source synchronous, LVDS interface simplifies the digital interface with existing FGPA/ASIC technology. On-chip controllers are used to manage external and internal clock domain variations over temperature to ensure reliable data transfer from the host to the DAC core. Multichip synchronization is possible with an on-chip synchronization controller. A serial peripheral interface (SPI) is used for device configuration as well as readback of status registers.

The CBM97D39BG adopts a 0.18μm CMOS process and operates on dual power supplies of 1.8V and 3.3V. It is packaged in PBGA-160.

Contents

Functional Block Diagram

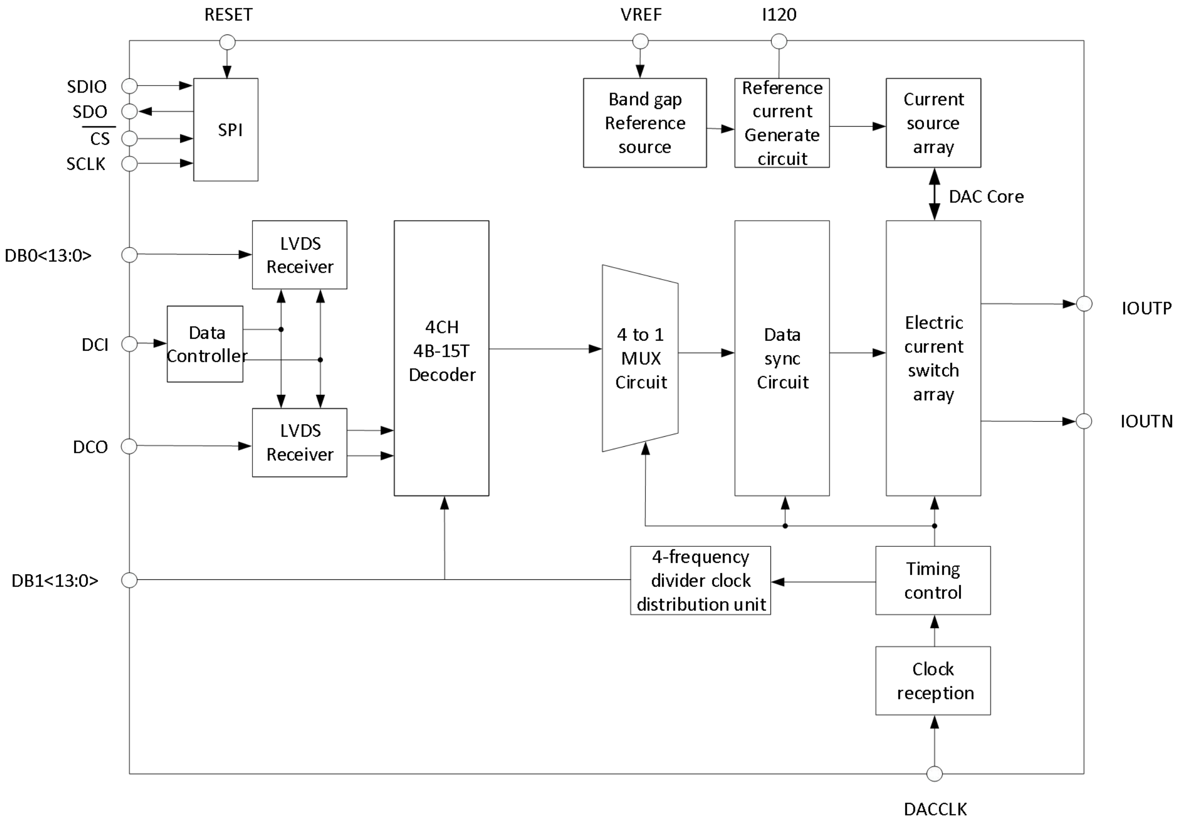


Figure 1. Functional Block Diagram

Product Highlights

1、Ability to synthesize high quality wideband signals with bandwidths of up to 1.25 GHz in the first or second Nyquist zone.

2、A proprietary quad-switch DAC architecture provides exceptional ac linearity performance while enabling mix mode operation.

3、A dual-port, double data rate, LVDS interface supports the maximum conversion rate of 2.5 GSPS.

4、On-chip controllers manage external and internal clock domain skews.

5、A multichip synchronization capability.

6、Programmable differential current output with an 8.7 mA
to 31.7 mA range.

Specifications

Operating conditions: VDDA = VDD33 = 3.3 V，VDDC = VDD = 1.8 V， Temperature: - 40~85℃，f_{CLK}=2500MHz。unless otherwise noted.

Table1

Parameter	Symbol	Conditions	Min	Typ	Max	Unit
Resolution	RES		14			Bits
Reference voltage	V_{REF}		1.15	1.18	1.25	V
Integral Linearity Error	E_L		-3	±2.5	3	LSB
Differential linearity error	E_{DL}		-3	±0.8	3	LSB
Gain error	E_G		-8	3	8	%FSR
Power dissipation	P_D		--	0.95	1.25	W
Spurious free dynamic range	SFDR	fo=100MHz，normal mode	62	70	--	dBc
		fo=350MHz，normal mode	52	59	--	dBc
		fo=550MHz，normal mode	51	56	--	dBc
		fo=950MHz，normal mode	46	57	--	dBc
		fo=1400MHz，normal mode	44	47	--	dBc
		fo=1900MHz，normal mode	50	53	--	dBc
		fo=2400MHz，normal mode	44	45	--	dBc

Absolute Maximum Ratings

Table 5.

Parameter	With Respect To	Rating
VDDA	VSSA	-0.3V to +3.6V
VDD33	VSS	-0.3V to +3.6V
VDD	VSS	-0.3V to +1.98V
VDDC	VSSC	-0.3V to +1.98V
VSSA	VSS	-0.3V to +0.3V

VSSA	VSSC	-0.3V to +0.3V
VSS	VSSC	-0.3V to +0.3V
DACCLK_P,DACCLK_N	VSSC	-0.3V to VDDC+0.18V
DCI,DCO,SYNC_IN,SYNC_OUT	VSS	-0.3V to VDD33+0.3V
LVDS Data Input	VSS	-0.3V to VDD33+0.3V
IOUP,IOUTN	VSSA	-1.0V to VDDA+0.3V
I120,VREF	VSSA	-0.3V to VDDA+0.3V
IRQ,SCLK,SDO,SDIO,RESET	VSS	-0.3V to VDD33+0.3V
Junction Temperature		150℃
Storage Temperature		-65℃ to +150℃

Thermal Resistance

θ_{JA} is specified for the worst-case conditions, that is, a device soldered in a circuit board for surface-mount packages.

Table 6. Thermal Resistance

Package Type	θ JA	θ JC	Unit
160-Ball BGA	31.2	7.0	℃/W[5]

[5] With no airflow movement

Pin Configurations and Function Descriptions

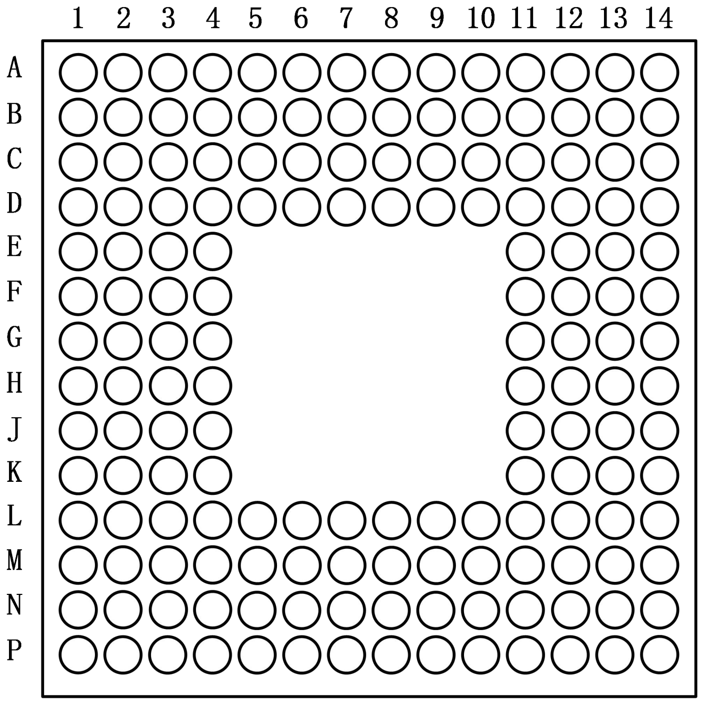


Figure 2. CD97D39 Pin Diagram（Top view)

Pin Function Descriptions

Pin No.	Mnemonic	Description				
		Mode	I/O	Voltage range	Max freq	Others
C1,C2,D1,D2,E1,E2,E3,E4	VDDC	/	/	1.8V	/	/
A1,A2,A3,A4,A5,B1,B2,B3,B4,B5,C4,C5,D4,D5	VSSC	/	/	0V	/	/
A10,A11,B10,B11,C10,C11,D10,D11	VDDA	/	/	3.3V	/	/
A12,A13,B12,B13,C12,C13,D12,D13	VSSA	/	/	0V	/	/
A6,A9,B6,B9,C6,C9,D6,D9,F1,F2,F3,F4,E11,E12,E13,E14,F11,F12	VSSA Shield	/	/	0V	/	/
A14	NC	/	/	/	/	/
A7,B7,C7,D7	IOUTN	Analog	O	3.0V	3.6GHz	/
A8,B8,C8,D8	IOUTP	Analog	O	3.3V, 10~30mA Full range	3.6GHz	/

				output current		
B14	I120	Analog	I/O	≈1.2V	/	/
C14	VREF	Analog	I/O	≈1.2V	/	/
D14	VREF	Analog	O	3.3V，≈10μA Current output	/	/
C3,D3	DACCLK_N/DACCLK_P	Analog	I	1.8V	2.5GHz	/
F13	IRQ	Digital	O	3.3V	/	/
F14	RESET	Digital	I	3.3V	/	/
G13	/CS	Digital	I	3.3V	/	/
G14	SDIO	Digital	I/O	3.3V	/	/
H13	SCLK	Digital	I	3.3V	/	/
H14	SDO	Digital	O	3.3V	/	/
J3,J4,J11,J12	VDD33	/	/	3.3V	/	/
G1,G2,G3,G4,G11,G12	VDD	/	/	1.8V	/	/
H1,H2,H3,H4,H11,H12,K3,K4,K11,K12	VSS	/	/	0V	/	/
J1,J2	SYNC_OUT_P/SYNC_OUT_N	Digital	O	1.8V	625MHz	/
K1,K2	SYNC_IN_P/SYNC_IN_N	Digital	I	1.8V	625MHz	/
J13,J14	DCO_P/DCO_N	Digital	O	1.025V~1.375V	625MHz	/
K13,K14	DCI_P/DCI_N	Digital	I	0.825V~1.575V	625MHz	/
L1,M1	DB1[0]P/DB1[0]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L2,M2	DB1[1]P/DB1[1]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L3,M3	DB1[2]P/DB1[2]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L4,M4	DB1[3]P/DB1[3]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L5,M5	DB1[4]P/DB1[4]N	Digital	I	0.825V~1.575	1.25GSPS	/

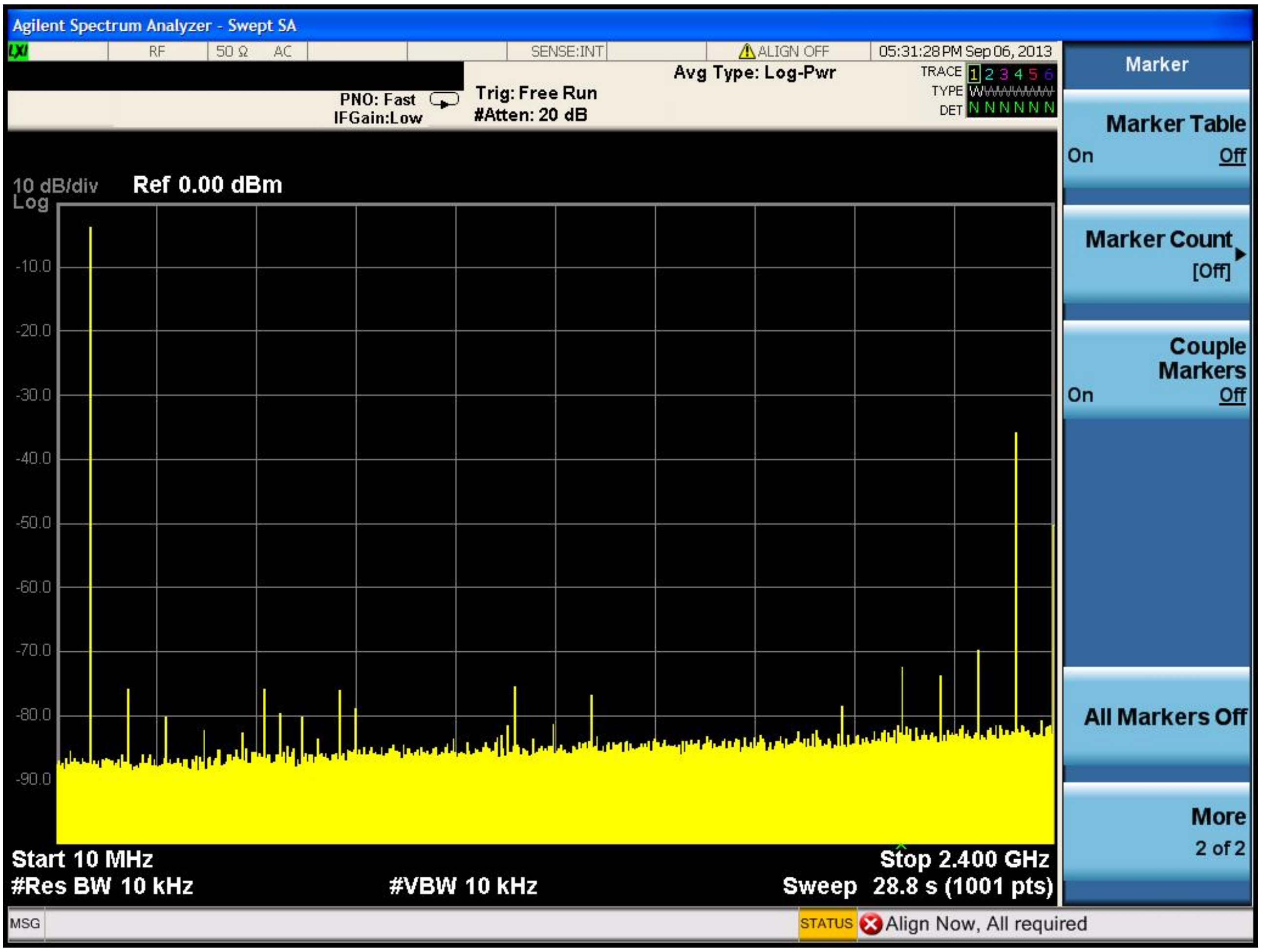
				V		
L6,M6	DB1[5]P/DB1[5]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L7,M7	DB1[6]P/DB1[6]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L8,M8	DB1[7]P/DB1[7]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L9,M9	DB1[8]P/DB1[8]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L10,M10	DB1[9]P/DB1[9]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L11,M11	DB1[10]P/DB1[10]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L12,M12	DB1[11]P/DB1[11]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L13,M13	DB1[12]P/DB1[12]N	Digital	I	0.825V~1.575V	1.25GSPS	/
L14,M14	DB1[13]P/DB1[13]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N1,P1	DB0[0]P/DB0[0]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N2,P2	DB0[1]P/DB0[1]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N3,P3	DB0[2]P/DB0[2]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N4,P4	DB0[3]P/DB0[3]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N5,P5	DB0[4]P/DB0[4]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N6,P6	DB0[5]P/DB0[5]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N7,P7	DB0[6]P/DB0[6]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N8,P8	DB0[7]P/DB0[7]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N9,P9	DB0[8]P/DB0[8]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N10,P10	DB0[9]P/DB0[9]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N11,P11	DB0[10]P/DB0[10]	Digital	I	0.825V~1.575	1.25GSPS	/

	N			V		
N12,P12	DB0[11]P/DB0[11]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N13,P13	DB0[12]P/DB0[12]N	Digital	I	0.825V~1.575V	1.25GSPS	/
N14,P14	DB0[13]P/DB0[13]N	Digital	I	0.825V~1.575V	1.25GSPS	/

Typical Performance Characteristics

AC(Normal Mode)

I_{OUTFS} = 20 mA, nominal supplies, 25°C, unless otherwise noted.



Agilent Spectrum Analyzer - Swept SA

RF 50 Ω AC SENSE:INT ALIGN OFF 05:31:28 PM Sep 06, 2013

Avg Type: Log-Pwr

PNO: Fast IFGain:Low

Trig: Free Run #Atten: 20 dB

TRACE 1 2 3 4 5 6
TYPE W W W W W W
DET N N N N N N

10 dB/div Log Ref 0.00 dBm

Start 10 MHz #Res BW 10 kHz #VBW 10 kHz Stop 2.400 GHz Sweep 28.8 s (1001 pts)

MSG STATUS Align Now, All required

Figure 3. Single-Tone Spectrum at f_{OUT}=91MHz, f_{DAC}=2.4GSPS

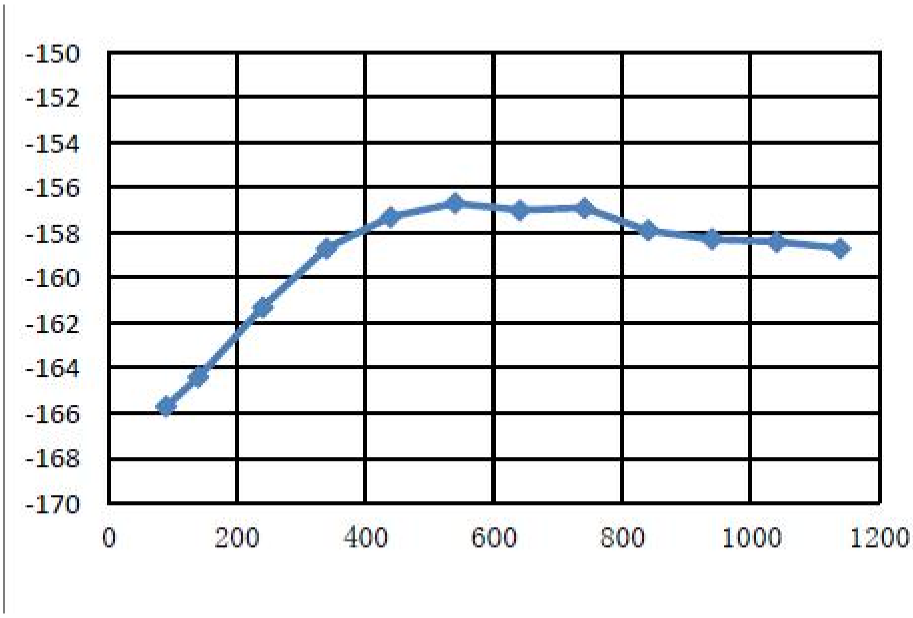


Figure 4. Single-Tone NSD over f_{OUT}

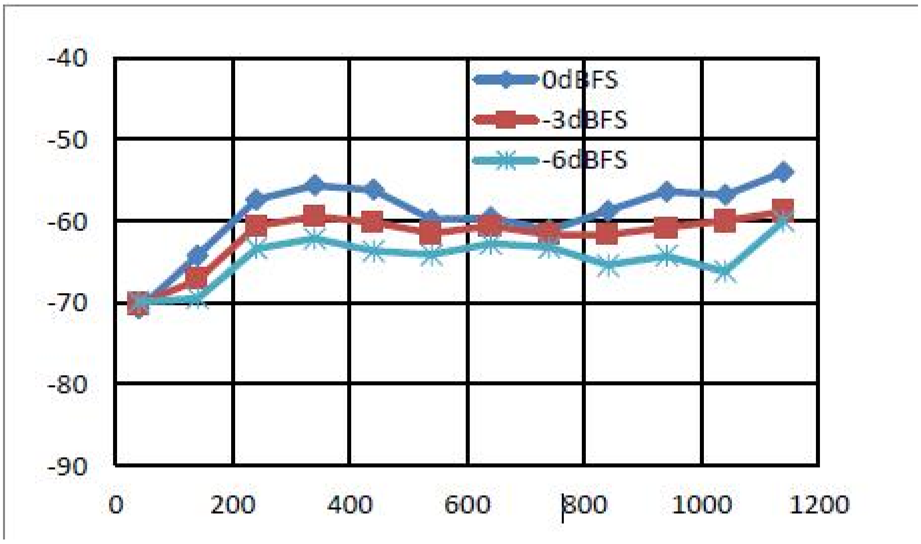


Figure 5. SFDR in different digital power inputs vs.f_{OUT}

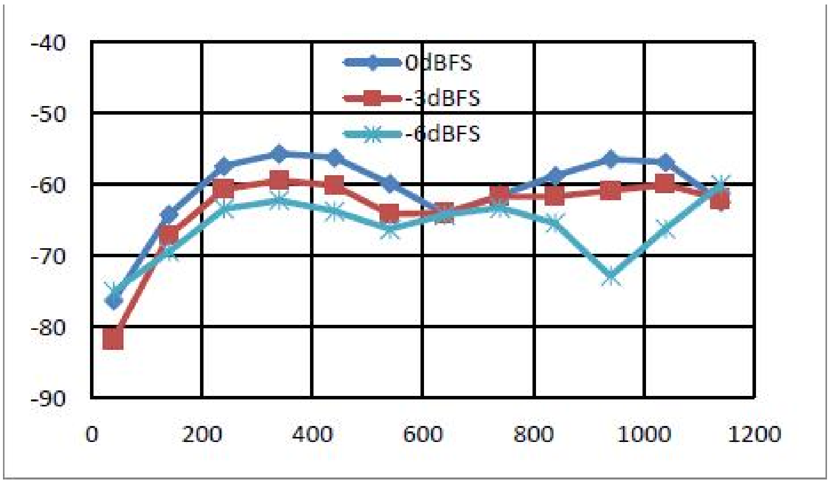


Figure 6.SFDR for Second Harmonic over f_{OUT} vs. Digital Full Scale

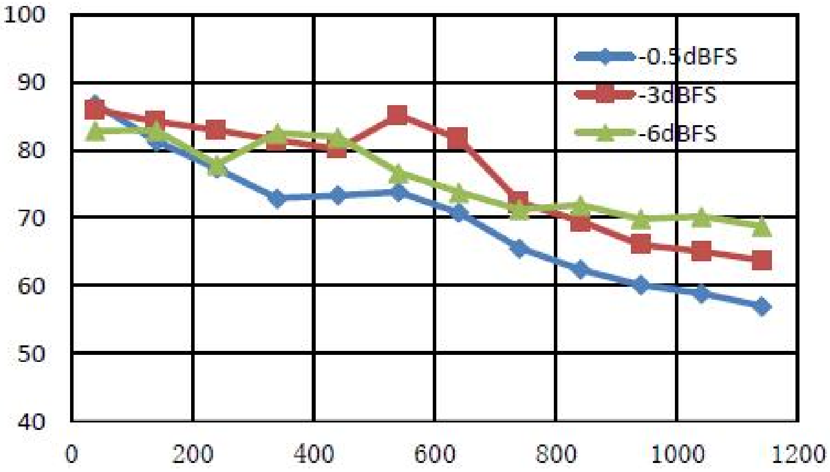


Figure 7.IMD vs. f_{OUT} over Digital Full Scale

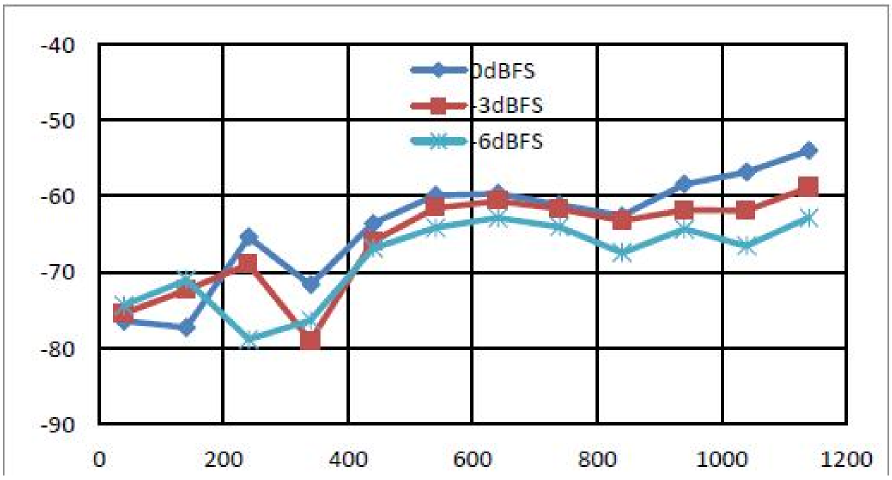


Figure 8.SFDR for Third Harmonic over f_{OUT} vs. Digital Full Scale

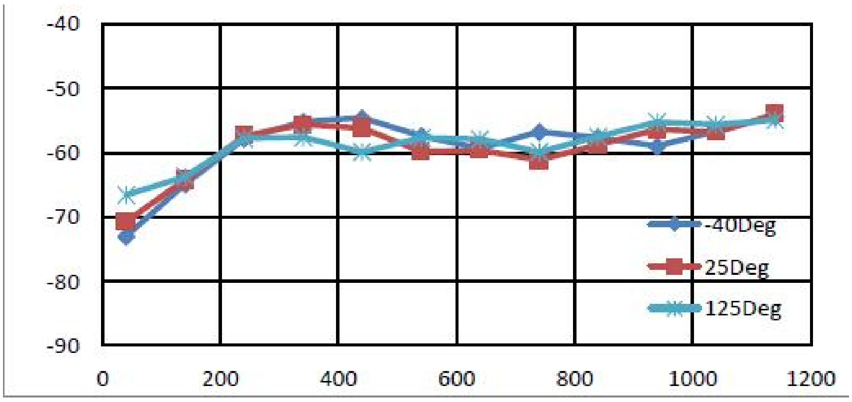


Figure 9. SFDR vs. f_{OUT} over Temperature

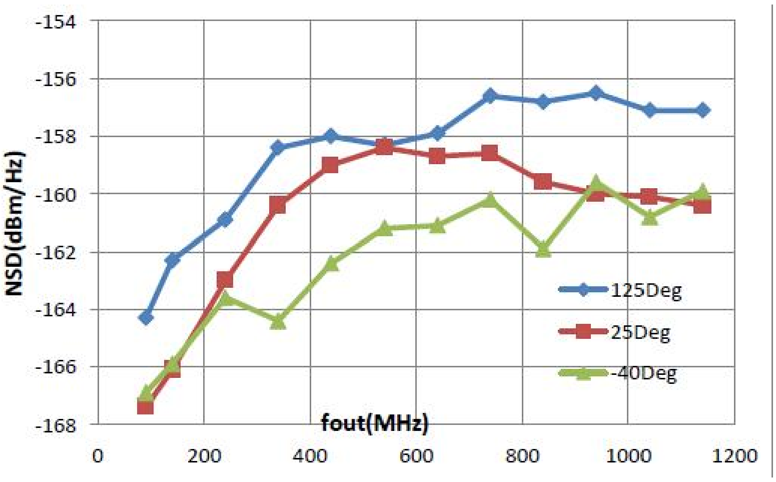


Figure 10. NSD vs. f_{OUT} over Temperature

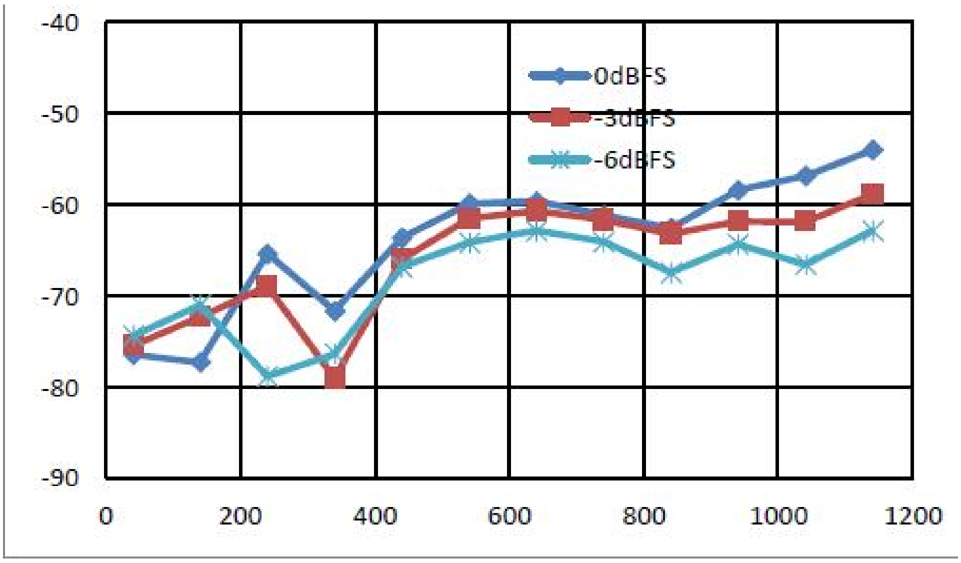


Figure 11.SFDR for Third Harmonic over f_{OUT} vs. Digital Full Scale

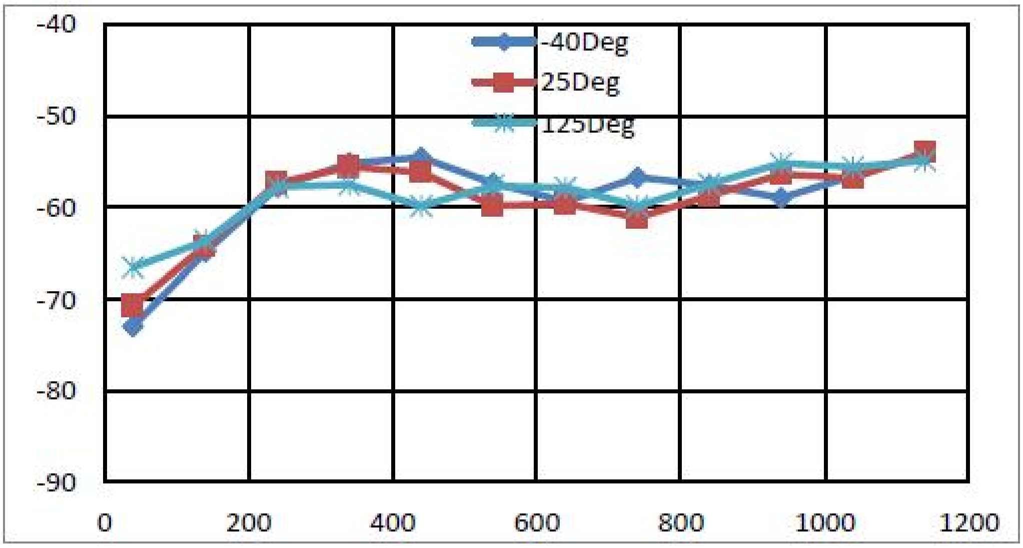


Figure 12.Single-Tone Spectrum SFDR vs. f_{OUT} over Temperature

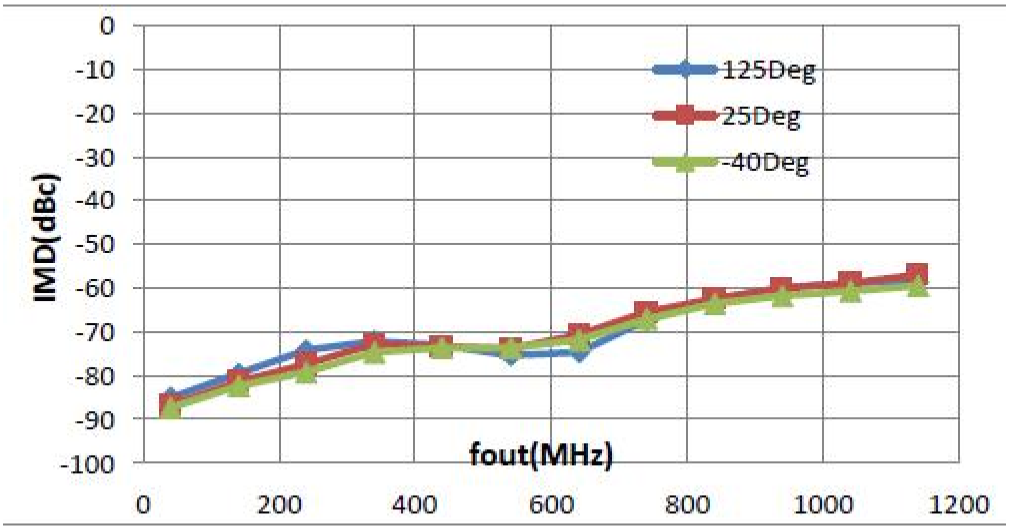
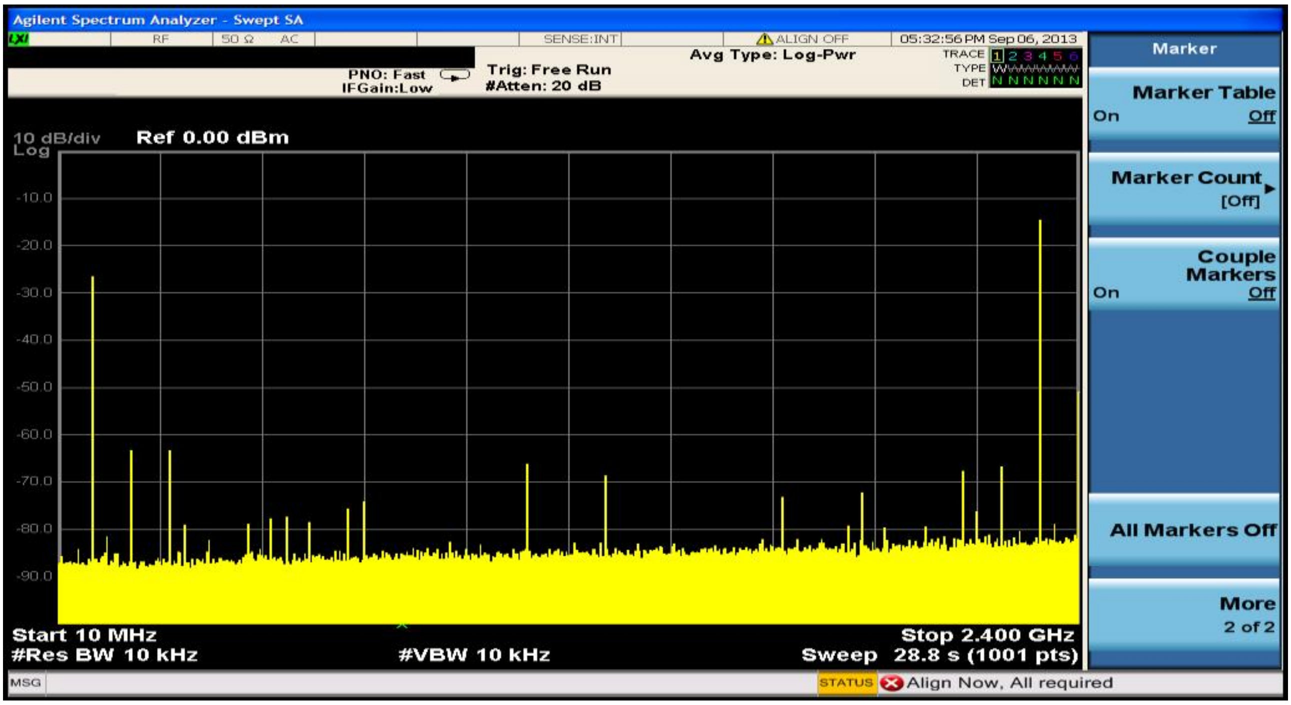


Figure 13.IMD vs. f_{OUT} over Temperature

AC（Mix Mode）

f_{DAC} = 2.4 GSPS, I_{OUTFS} = 20 mA, nominal supplies, 25°C, unless otherwise noted.


Agilent Spectrum Analyzer - Swept SA

RF 50 Ω AC SENSE:INT ALIGN OFF 05:32:56 PM Sep 06, 2013

Avg Type: Log-Pwr

PNO: Fast
IFGain:Low

Trig: Free Run
#Atten: 20 dB

TRACE 1 2 3 4 5 6
TYPE W W W W W W
DET N N N N N N

10 dB/div Ref 0.00 dBm
Log

Start 10 MHz
#Res BW 10 kHz
#VBW 10 kHz
Stop 2.400 GHz
Sweep 28.8 s (1001 pts)

MSG STATUS Align Now, All required

Figure 14.Single-Tone Spectrum at fOUT=2.31GHz, fDAC=2.4GSPS

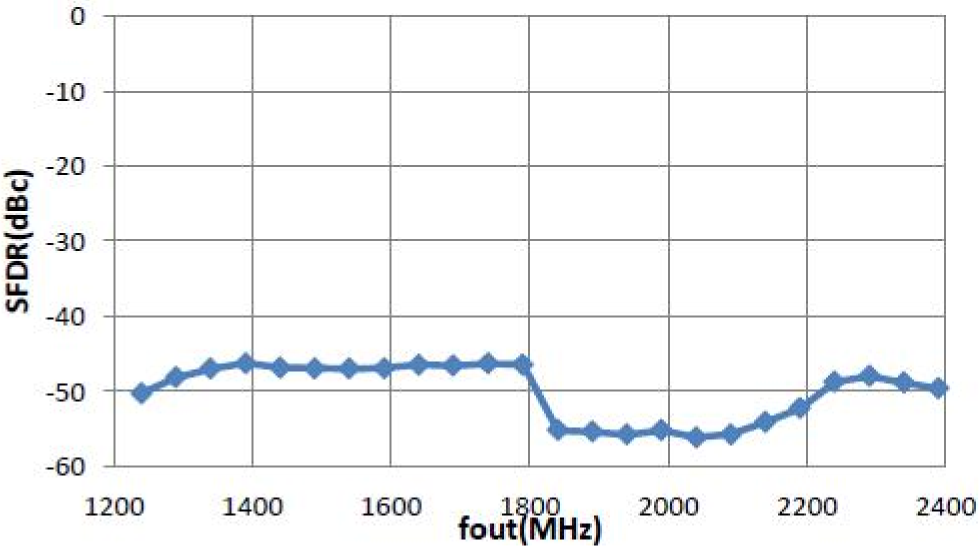
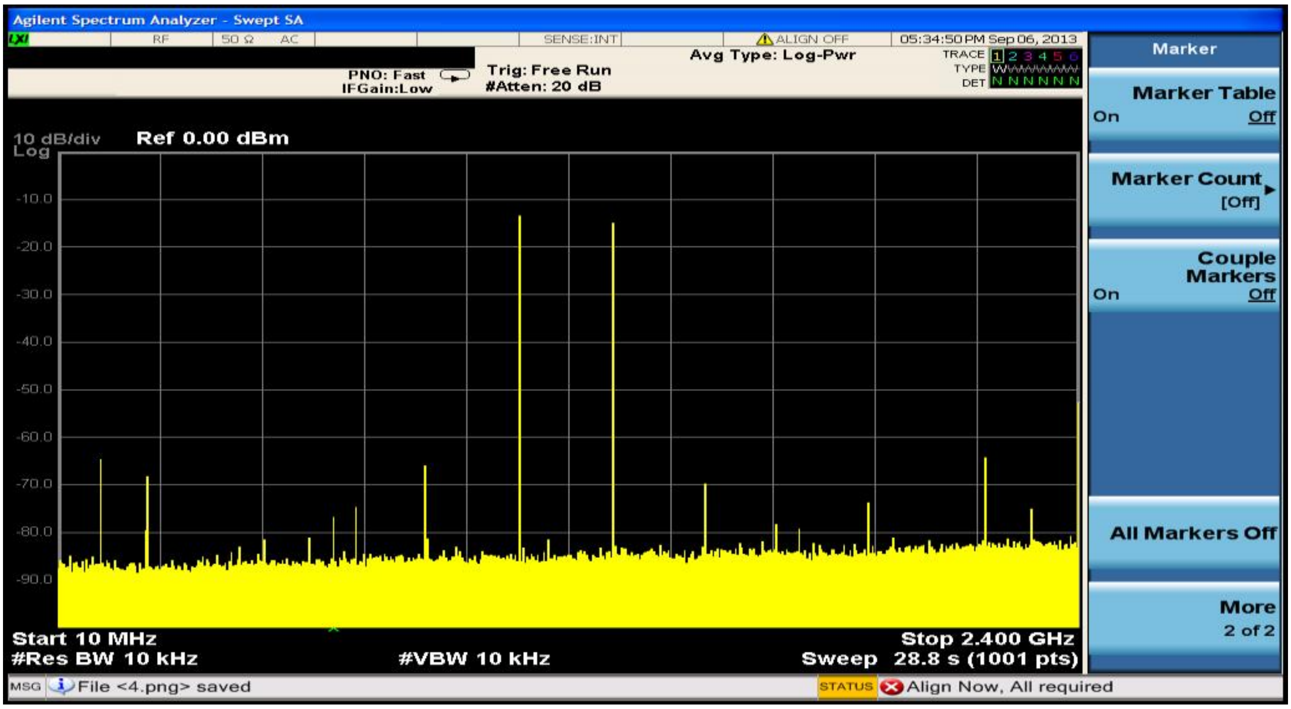


Figure 15. SFDR in Mix Mode vs. f_{OUT} at 2.4 GSPS



Agilent Spectrum Analyzer - Swept SA

RF | 50 Ω | AC | SENSE:INT | ALIGN OFF | 05:34:50 PM Sep 06, 2013

PNO: Fast
IFGain:Low

Trig: Free Run
#Atten: 20 dB

Avg Type: Log-Pwr

TRACE 1 2 3 4 5 6
TYPE WWWWWW
DET N N N N N N

10 dB/div Log Ref 0.00 dBm

Start 10 MHz
#Res BW 10 kHz
#VBW 10 kHz
Stop 2.400 GHz
Sweep 28.8 s (1001 pts)

MSG File <4.png> saved

STATUS Align Now, All required

Figure 16. Single-Tone Spectrum in Mix Mode at f_{OUT} = 1.31 GHz, f_{DAC} = 2.4 GSPS

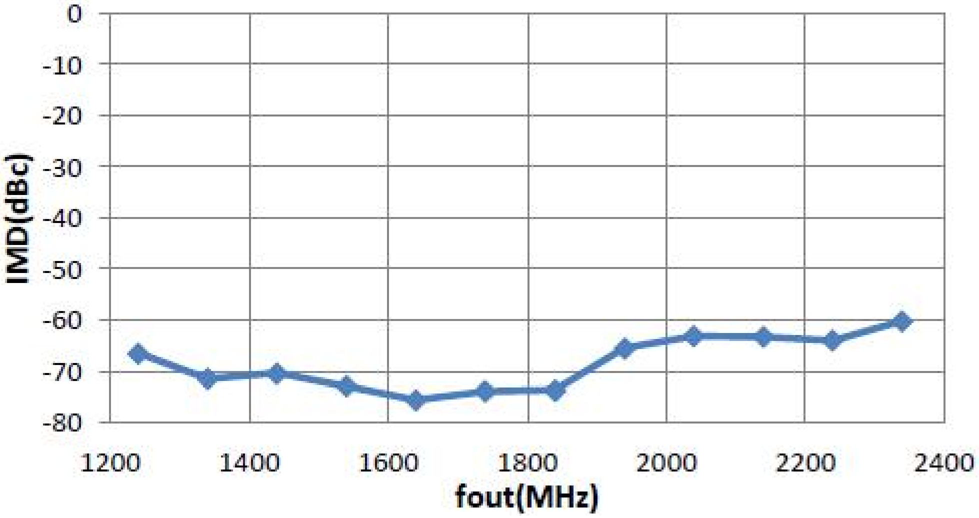


Figure 17. IMD in Mix Mode vs. f_{OUT} at 2.4 GSPS

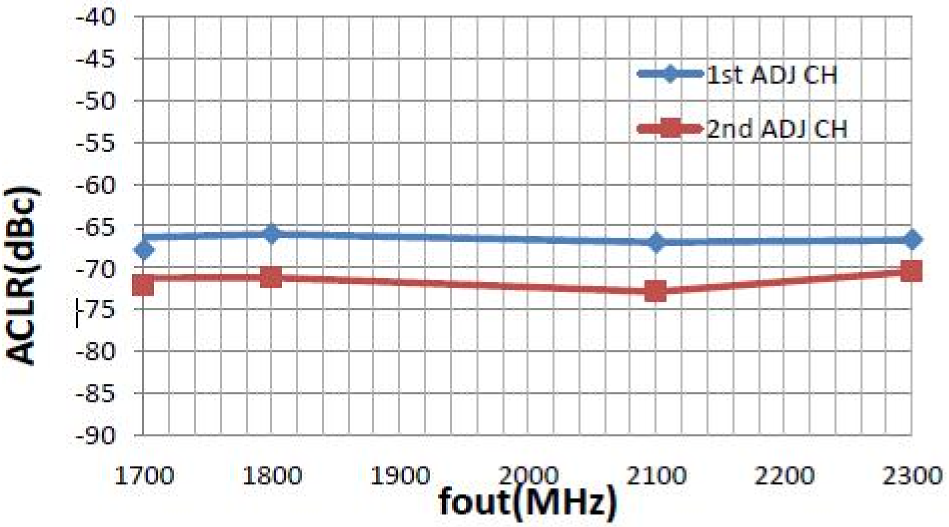


Figure 18. Single-Carrier WCDMA ACLR vs. fOUT at 2457.6 MSPS

Serial Port Interface(SPI)Register

The CD97D39 contains a set of programmable registers described in Table 8 that are used to configure and monitor various internal parameters. Note the following points when programming the CD97D39 SPI registers:

Reset

The CD97D39 contains the function of power on reset. In the process of use, users are recommended to perform a hardware or software reset operation after power on. Software reset is realized through register 0x00. For detailed operation of software reset, please refer to the detailed description of this register in Table 8. Hardware reset is realized by adding a high-level pulse with a minimum pulse width of 40ns to RESET pin (pin F14). When not in use, the hardware reset pin RESET needs to be connected to the GND.

SPI Operation

The SPI of CD97D39 consists of serial port clock signal (SCLK), serial port enable signal (/CS), serial port data input/output signal (SDIO) and serial port data output signal (SDO). The serial port is compatible with 3-wire and 4-wire modes, and the interface level is 3.3V. After power on and reset, the default state is 4-wire mode. SDIO can be set by_ DIR (register 0x00, bit 7) is "1" to realize 3-wire SPI interface. At this time, SDIO is a two-way data line, while SDO is in high impedance state. The maximum frequency of SCLK is 20MHz.

SPI Instruction format

Table 7： SPI Instruction format

MSB LSB

17	16	15	14	13	12	11	10
R/W	A6	A5	A4	A3	A2	A1	A0

Each read and write operation of SPI is accompanied by an 8-bit instruction header. The highest bit is the R/W identification bit. '1' corresponds to the read operation, '0' corresponds to the write operation, and bits 6 to 0 describe the register address during data transmission. During read and write operations, 8-bit data is closely followed by the instruction header. For write operation, the register is valid immediately after each transmitted byte is written to the last bit. Chip selection signal (/CS) can be pulled to high level after every 8-bit sequence (except the last byte) to stop bus operation/ When CS is low, serial transmission continues. Stopping bus operation at a non byte boundary will reset the SPI port.

CD97D39 can support two data transmission modes: MSB first and LSB first. Its data format is shown in Figure 19 and can be transmitted through SDIO_ DIR bit (register 0x00 bit 7). After power on and reset, it defaults to MSB priority data format.

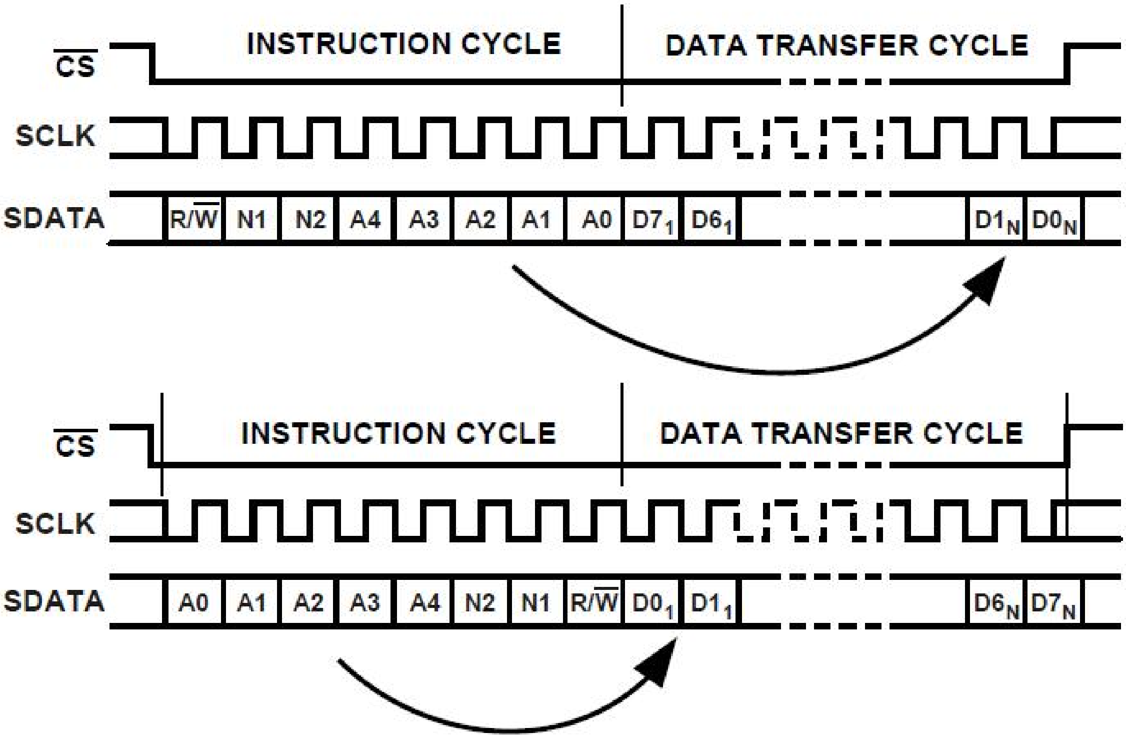


Figure 19. SPI Serial Port Timing, MSB Priority (above) and LSB Priority (below)

The timing of 3-wire write operation of SPI port is shown in Figure 20, the timing of 3-wire read operation is shown in Figure 21, and the timing of 4-wire read and write operation is shown in Figure22.

Figure 20. 3-wire SPI Write Operation Sequence

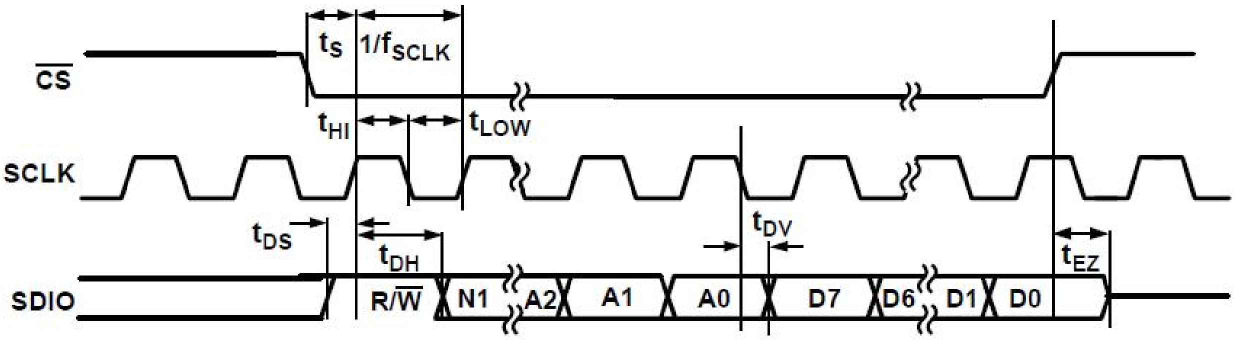


Figure 21. Timing sequence of 3-wire SPI read operation

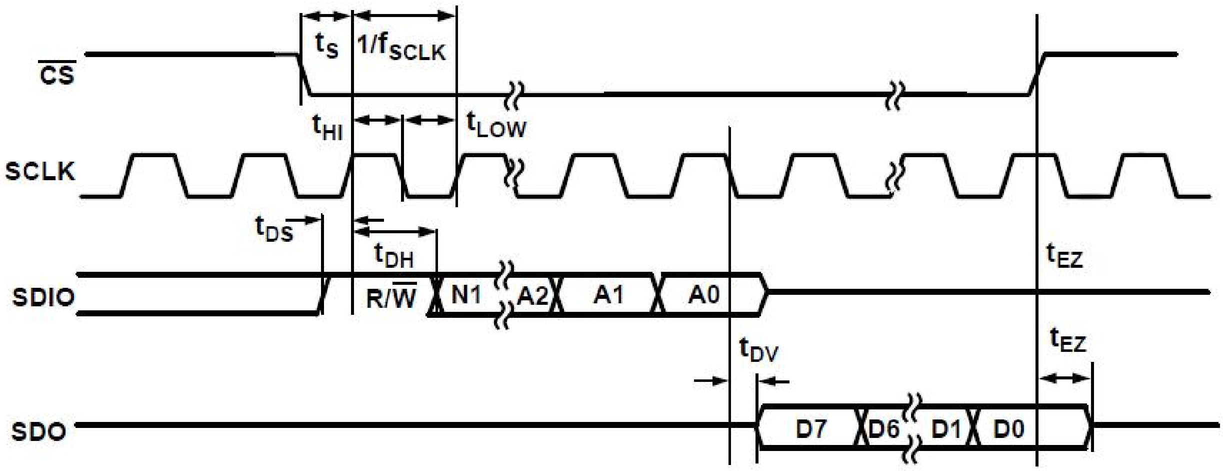


Figure 22. Timing sequence of 4-wire SPI read operation

SPI Register

Table 8：Register Description（N/A=Not Applicable)

Name	Hex Addr	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Default
Mode	00	SDIO_DIR	LSB/MSB	Reset	N/A	N/A	N/A	N/A	N/A	0x00
Power-Down	01	N/A	N/A	LVDS_DRVR_PD	LVDS_RCVR_PD	N/A	N/A	CLK_RCVR_PD	DAC_BIAS_PD	0x00
CNT_CLK_DIS	02	N/A	N/A	N/A	N/A	CLKGEN_PD	N/A	REC_CNT_CLK	MU_CNT_CLK	0x03
IRQ_EN	03	N/A	N/A	SYNC_LST_EN	SYNC_LCK_EN	MU_LST_EN	MU_LCK_EN	RCV_LST_EN	RCV_LCK_EN	0x00
IRQ_REQ	04	N/A	N/A	SYNC_LST_IRQ	SYNC_LCK_IRQ	MU_LST_IRQ	MU_LCK_IRQ	RCVLST_IRQ	RCVLCK_IRQ	0x00
RSVD	05	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
FSC_1	06	FSC[7]	FSC[6]	FSC[5]	FSC[4]	FSC[3]	FSC[2]	FSC[1]	FSC[0]	0x00
FSC_2	07	Sleep	N/A	N/A	N/A	N/A	N/A	FSC[9]	FSC[8]	0x02
DEC_CNT	08	N/A	N/A	N/A	N/A	N/A	N/A	DAC_DEC[1]	DAC_DEC[0]	0x00
RSVD	09	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
LVDS_CNT	0A	LVDS_REG[7]	LVDS_REG[6]	N/A	N/A	HNDOFF_CHK_RST	N/A	LVDS_Bias[1]	LVDS_Bias[0]	0x00

DIG_STAT	0B	HNDOFF_ Fall[3]	HNDOFF_ Fall[2]	HNDOFF_ Fall[1]	HNDOFF_ Fall[0]	HNDOFF_ Rise[3]	HNDOFF_ Rise[2]	HNDOFF_ Rise[1]	HNDOFF_ Rise[0]	RNDM
LVDS_STAT1	0C	SUP/HLD_ Edge1	N/A	DCI_PHS3	DCI_PHS1	DCI_PRE_ PH2	DCI_PRE_ PH0	DCI_PST_ PH2	DCI_PST_ PH0	RNDM
LVDS_STAT2	0D	SUP/HLD_ SYNC	SUP/HLD_ Edge0	SYNC_SAMP1	SYNC_SAMP0	LVDS1_HI	LVDS1_LO	LVDS0_HI	LVDS0_LO	RNDM/0
RSVD	0E	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
RSVD	0F	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
LVDS_REC_CNT1	10	SYNC_FLG_RST	SYNC_LOOP_ON	SYNC_MST/SLV	SYNC_CNT_ENA	N/A	RCVR_FLG_RST	RCVR_LOOP_ON	RCVR_CNT_ENA	0x42
LVDS_REC_CNT2	11	SMP_DEL[1]	SMP_DEL[0]	FINE_DEL_MID[3]	FINE_DEL_MID[2]	FINE_DEL_MID[1]	FINE_DEL_MID[0]	RCVR_GAIN[1]	RCVR_GAIN[0]	0xDD
LVDS_REC_CNT3	12	SMP_DEL[9]	SMP_DEL[8]	SMP_DEL[7]	SMP_DEL[6]	SMP_DEL[5]	SMP_DEL[4]	SMP_DEL[3]	SMP_DEL[2]	0x29
LVDS_REC_CNT4	13	DCI_DEL[3]	DCI_DEL[2]	DCI_DEL[1]	DCI_DEL[0]	FINE_DEL_SKW[3]	FINE_DEL_SKW[2]	FINE_DEL_SKW[1]	FINE_DEL_SKW[0]	0x71
LVDS_REC_CNT5	14	CLKDIVPH[1]	CLKDIVPH[0]	DCI_DEL[9]	DCI_DEL[8]	DCI_DEL[7]	DCI_DEL[6]	DCI_DEL[5]	DCI_DEL[4]	0x0A
LVDS_REC_CNT6	15	SYNC_GAIN[1]	SYNC_GAIN[0]	SYNCOUT_ PH[1]	SYNCOUT_PH[0]	LCKTHR[3]	LCKTHR[2]	LCKTHR[1]	LCKTHR[0]	0x42
LVDS_REC_CNT7	16	N/A	SYNCO_DEL[6]	SYNCO_DEL[5]	SYNCO_DEL[4]	SYNCO_DEL[3]	SYNCO_DEL[2]	SYNCO_DEL[1]	SYNCO_DEL[0]	0x00
LVDS_REC_CNT8	17	SYNCSH_ DEL[0]	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0x00
LVDS_REC_CNT9	18	SYNCSH_ DEL[8]	SYNCSH_ DEL[7]	SYNCSH_ DEL[6]	SYNCSH_ DEL[5]	SYNCSH_ DEL[4]	SYNCSH_ DEL[3]	SYNCSH_ DEL[2]	SYNCSH_ DEL[1]	0x00
LVDS_REC_STAT1	19	SMP_DEL[1]	SMP_DEL[0]	N/A	N/A	SMP_FINE_DEL[3]	SMP_FINE_DEL[2]	SMP_FINE_DEL[1]	SMP_FINE_DEL[0]	0xC7
LVDS_REC_STAT2	1A	SMP_DEL[9]	SMP_DEL[8]	SMP_DEL[7]	SMP_DEL[6]	SMP_DEL[5]	SMP_DEL[4]	SMP_DEL[3]	SMP_DEL[2]	0x29
LVDS_REC_STAT3	1B	DCI_DEL[1]	DCI_DEL[0]	N/A	N/A	SYNCOUT PH[1]	SYNCOUT PH[0]	CLKDIVPH[1]	CLKDIVPH[0]	0xC0
LVDS_REC_STAT4	1C	DCI_DEL[9]	DCI_DEL[8]	DCI_DEL[7]	DCI_DEL[6]	DCI_DEL[5]	DCI_DEL[4]	DCI_DEL[3]	DCI_DEL[2]	0x29

LVDS_REC_STAT5	1D	FINE_DEL_ PST[3]	FINE_DEL_ PST[2]	FINE_DEL_ PST[1]	FINE_DEL_ PST[0]	FINE_DEL_ PRE[3]	FINE_DEL_ PRE[2]	FINE_DEL_ PRE[1]	FINE_DEL_ PRE[0]	0x86
LVDS_REC_STAT6	1E	N/A	SYNCO_DEL[6]	SYNCO_DEL[5]	SYNCO_DEL[4]	SYNCO_DEL[3]	SYNCO_DEL[2]	SYNCO_DEL[1]	SYNCO_DEL[0]	0x00
LVDS_REC_STAT7	1F	SYNCSH_ DEL[0]	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0x00
LVDS_REC_STAT8	20	SYNCSH_ DEL[8]	SYNCSH_ DEL[7]	SYNCSH_ DEL[6]	SYNCSH_ DEL[5]	SYNCSH_ DEL[4]	SYNCSH_ DEL[3]	SYNCSH_ DEL[2]	SYNCSH_ DEL[1]	0x00
LVDS_REC_STAT9	21	SYNC_TRK_ON	SYNC_INIT_ON	SYNC_LST_LCK	SYNC_LCK	RCVR_TRK_ON	RCVR_FE_ON	RCVR_LST	RCVR_LCK	0x00
CROSS_CNT1	22	N/A	N/A	N/A	DUTY_EN	N/A	N/A	N/A	N/A	0x00
CROSS_CNT2	23	CROSS_CTRL[5]	N/A	N/A	CROSS_CTRL[4]	CROSS_CTRL[3]	CROSS_CTRL[2]	CROSS_CTRL[1]	CROSS_CTRL[0]	0x00
PHS_ DET	24	N/A	N/A	CMP_BST	PHS_DET AUTO_EN	Bias[3]	Bias[2]	Bias[1]	Bias[0]	0x00
MU_ DUTY	25	MU_DUTYAUTO_EN	POS/NEG	ADJ[5]	ADJ[4]	ADJ[3]	ADJ[2]	ADJ[1]	ADJ[0]	0x00
MU_ CNT1	26	N/A	Slope	Mode[1]	Mode[0]	Read	Gain[1]	Gain[0]	Enable	0x42
MU_ CNT2	27	MUDEL[0]	SRCH_MODE [1]	SRCH_MODE[0]	SET_PHS[4]	SET_PHS[3]	SET_PHS[2]	SET_PHS[1]	SETPHS[0]	0x40
MU_ CNT3	28	MUDEL[8]	MUDEL[7]	MUDEL[6]	MUDEL[5]	MUDEL[4]	MUDEL[3]	MUDEL[2]	MUDEL[1]	0x00
MU_ CNT4	29	SEARCH_TOL	Retry	CONTRST	Guard[4]	Guard[3]	Guard[2]	Guard[1]	Guard[0]	0x0B
MU_ STAT1	2A	N/A	N/A	N/A	N/A	N/A	N/A	MU_LOST	MU_LKD	0x00
ANA_STAT	2B	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
RSVD	2C	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
ANA_ CNT1	32	HDRM[7]	HDRM[6]	HDRM[5]	HDRM[4]	HDRM[3]	HDRM[2]	HDRM[1]	HDRM[0]	0xCA
ANA_ CNT2	33	N/A	N/A	N/A	N/A	N/A	N/A	MSEL[1]	MSEL[0]	0x03
RSVD	34	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
PART ID	35	ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]	0x20

SPI Port Configuration and Software Reset

Table 9: SPI Port Configuration and Software Reset Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x00	SDIO_DIR	7	R/W	0	
	LSB/MSB	6	R/W	0	
	Reset	5	R/W	0	

Power-Down LVDS Interface and DAC

Table 10: Power-Down LVDS Interface and DAC Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x01	LVDS_DRVR_PD	5	R/W	0	Power-down of the LVDS drivers/receivers and DAC. 0 = enable, 1 = disable.
	LVDS_RCVR_PD	4	R/W	0	
	CLK_PCVR_PD	1	R/W	0	
	DAC_BIAS_PD	0	R/W	0	

Controller Clock Disable

Table 11: Controller Clock Disable Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x02	CLKGEN_PD	3	R/W	0	Internal CLK distribution enable: 0 = enable, 1 = disable.
	REC_CNT_CLK	1	R/W	1	LVDS receiver (REC_CNT_CLK) and mu controller clock disable (MU_CNT_CLK). 0 = disable, 1 = enable.
	MU_CNT_CLK	0	R/W	1	

Interrupt Request (IRQ) Enable/Status

Table 12: Interrupt Request (IRQ) Enable/Status Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments

0x03	SYNC_LST_EN	5	W	0	This register enables the sync, mu, and LVDS Rx controllers to update their corresponding IRQ status bits in Register 0x04, which defines whether the controller is locked (LCK) or unlocked (LST).0 = disable (resets the status bit). 1 = enable.
	SYNC_LCK_EN	4	W	0	
	MU_LST_EN	3	W	0	
	MU_LCK_EN	2	W	0	
	RCV_LST_EN	1	W	0	
	RCV_LCK_EN	0	W	0	
0x04	SYNC_LST_IRQ	5	R	0	This register indicates the status of the controllers. For LCK_IQR bits: 0 = lost locked, 1 = locked. For LST_IQR bits: 0 = not lost locked, 1 = unlocked. Note that, if the controller IRQ is serviced, the relevant bits in Register 0x03 should be reset by writing 0, followed by another write of 1 to enable.
	SYNC_LCK_IRQ	4	R	0	
	MU_LST_IRQ	3	R	0	
	MU_LCK_IRQ	2	R	0	
	RCV_LST_IRQ	1	R	0	
	RCV_LCK_IRQ	0	R	0	

DAC FULL-Scale Current Setting (I_{OUTFS}) and Sleep

Table 13：DAC Full-Scale Current Setting (I_{OUTFS}) and Sleep Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x06	FSC_1	[7:0]	R/W	0	Sets the DAC I_{OUTFS} current between 8 mA and 31 mA (default = 20 mA). I_{OUTFS} = 0.0226 × FSC[9:0] + 8.58, where FSC = 0 to 1023.
0x07	FSC_2	[1:0]	R/W	0x02	
	Sleep	7	R/W		0 = enable DAC output, 1 = disable DAC output (sleep).

DAC Quad-Switch Mode Of Operation

Table 14：DAC Quad-Switch Mode of Operation Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x08	DAC-DEC	[1:0]	R/W	0x00	0x00 = normal baseband mode. 0x01 = return-to-zero mode. 0x02 = mix mode.

DCI Phase Alignment Status

Table 15: DCI Phase Alignment Status Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x0C	DCI_PRE_PHO	2	R	0	0 = DCI rising edge is after the PRE delayed version of the Phase 0 sampling edge. 1 = DCI rising edge is before the PRE delayed version of the Phase 0 sampling edge.
	DCI_PST_PHO	0	R	0	0 = DCI rising edge is after the POST delayed version of the Phase 0 sampling edge. 1 = DCI rising edge is before the POST delayed version of the Phase 0 sampling edge.

SYNC_IN Phase Alignment Status

Table 16: SYNC_IN Phase Alignment Status Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x0D	SYNC_IN_PH90	5	R	0	0 = SYNCIN rising edge is after Phase 90 sampling edge. 1 = SYNCIN rising edge is before Phase 90 sampling edge.
	SYNC_IN_PHO	4	R	0	0 = SYNCIN rising edge is after Phase 0 sampling edge. 1 = SYNCIN rising edge is before Phase 0 sampling edge.

Data Receive Controller Configuration

Table 17: Data Receiver Controller Configuration Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x10	SYNC_FLG_RST	7	W	0	Sync controller flag reset. Write 1 followed by 0 to reset flags.
	SYNC_LOOP_ON	6	R/W	1	0 = disable, 1 = enable. Enable for master only. When enabled, sync controller generates an IRQ

					when master falls out of lock and automatically begins search/track routine.
	SYNC__MST/SLV	5	R/W	0	Sync controller configuration. 0 = slave, 1 = master.
	SYNC_CNT_ENA	4	R/W	0	Sync controller enable. 0 = disable, 1 = enable
	RCVR_FLG_RST	2	W	0	Data receiver controller flag reset. Write 1 followed by 0 to reset flags.
	RCVR_LOOP_ON	1	R/W	1	0 = disable, 1 = enable. When enabled, the data receiver controller generates an IRQ; it falls out of lock and automatically begins a search/track routine.
	RCVR_CNT_ENA	0	R/W	0	Data receiver controller enabled. 0 = disable, 1 = enable.

Data Receive Controller_Data Sample Delay Value

Table 18: Data Receiver Controller_Data Sample Delay Value Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x11	SMP_DEL[1:0]	[7:6]	R/W	11	Controller enabled: the 10-bit value (with a maximum of 332) represents the start value for the delay line used by the state machine to sample data. Leave at the default setting of 167, which represents the midpoint of the delay line. Controller disabled: the value sets the actual value of the delay line.
0x12	SMP_DEL[9:2]	[7:0]	R/W	0X25	

Data and SYNC Receiver Controller_DCI Delay Value/Window and Phase Rotation

Table 19: Data and Sync Receiver Controller_DCI Delay Value/Window and Phase Rotation Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x13	DCI_DEL[3:0]	[7:4]	R/W	0111	Refer to the DCI_DEL description in Register 0x14.
	FINE_DEL_SKEW	[3:0]	R/W	0001	A 4-bit value sets the difference (that is, window) for the DCI PRE and POST sampling clocks. Leave at the default value of 1 for a narrow window.

0x14	CLKDIVPH[1:0]	[7:6]	R/W	00	Relative phase of internal divide-by-4 circuit. This feature allows phase rotation in 90° increments (that is, 1 count) to extend Rx controllers locking range for clock rates between 0.8 GSPS to 1.6 GSPS (only valid with sync controller disabled).
	DCI_DEL[9:4]	[5:0]	R/W	001010	Controller enabled: the 10-bit value (with a maximum of 332) represents the start value for the delay line used by the state machine to sample the DCI input. Leave at the default setting of 167, which represents the midpoint of the delay line. Controller disabled: the value sets the actual value of the delay line.
0x15	SYNC GAIN[1:0]	[7:6]	R/W	00	Sets the sync tracking gain (optimal value is 1).
	SYNCOUT_PH [1:0]	[5:4]	R/W	00	Readback of the present SYNC_OUT phase selection.
	LCKTHR[3:0]	[3:0]	R/W	0000	Sets the difference between the sample and DCI delays to lock (optimal value is 2)
0x16	SYNCO_DEL[6:0]	[6:0]	R/W	0x00	Sets the sync output delay value when the synch controller is disabled; otherwise, is the read status of the sync output delay value when sync is enabled.
0x17	SYNCO_DEL[0]	[7]	R/W	0x00	Sets the sync setup and hold delay value when the synch controller is disabled; otherwise, is the read status of sync setup and hold value when sync is enabled.
0x18	SYNCO_DEL[8:1]	[7:0]	R/W	0x00	Sets the sync setup and hold delay value when the synch controller is disabled; otherwise, is the read status of sync setup and hold value when sync is enabled.

Data Receiver Controller_Delay Line Status and SYNC Controller SYNC_Out Status

Table 20： Data Receiver Controller_Delay Line Status and Sync Controller SYNC_OUT Status Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x19	SMP_DEL[1:0]	[7:6]	R	00	The actual value of the DCI and data delay lines

0x1A	SMP_DEL[9:2]	[7:0]	R	0x00	determined by the data receiver controller (when enabled) after the state machine completes its search and enters track mode. Note that these values should be equal. SYNCOUT_PH provides phase status (0/90/180/270) of phase select mux, while CLKDIVPH provides phase status of data receiver controller (Register 0x14).
0x1B	SYNCOUT_PH [1:0]	3:2	R	00	
	CLKDIV PH[1:0]	1:0	R	00	
	DCI_DEL[1:0]	[7:6]	R	00	
0x1C	DCI_DEL[9:2]	[7:0]	R	0x00	

SYNC and Data Receiver Controller Lock/Tracking Status

Table 21: Sync and Data Receiver Controller Lock/Tracking Status Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x21	SYNC_TRK_ON	7	R	0	SYNC_TRK_ON and RCVR_TRK_ON: 0 = tracking not established. 1 = tracking established. SYNC_LCK and RCVR_LCK: 0 = controller is not locked. 1 = controller is locked. SYNC_LST and RCVR_LST: 0 = lock has not been lost. 1 = lock has been lost at some point.
	SYNC_LST	5	R	0	
	SYNC_LCK	4	R	0	
	RCVR_TRK_ON	3	R	0	
	RCVR_LST	1	R	0	
	RCVR_LCK	0	R	0	

CLK Input Common Mode

Table 22: CLK Input Common Mode Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x22	DUTY_EN	4	R/W	0	1=clock duty cycle control loop working (recommended) 0=clock duty cycle control loop does not work
0x23	CROSS_AUTO_B	7	R/W	0	1=Clock intersection control works when no signal is detected 0=clock intersection control works only when

					signal is detected (recommended)
	CROSS_ON	4	R/W	0	1=clock intersection control loop working (recommended) 0=clock intersection control loop does not work
	CROSS_DIR	3	R/W	0	1=lower the clock cross point (recommended) 0=clock cross point raised
	CROSS_OFFSET	[2:0]	R/W	000	111=maximum adjustment range of clock intersection (recommended) 000=minimum adjustment amplitude of clock intersection

MU Controller Configuration and Status

Table 23: Mu Controller Configuration and Status Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x24	CMP_BST	5	R/W	0	Phase detector enable and boost bias bits. Note that both bits should always be set to 1 to enable these functions.
	PHS_DET AUTO_EN	4	R/W	0	
0x25	MU_DUTY AUTO_EN	7	R/W	0	Mu controller duty cycle enable. Note that this bit should always be set to 1 to enable.
0x26	Slope	6	R/W	1	Mu controller phase slope lock. 0 = negative slope, 1 = positive slope. Refer to Table 28 for optimum setting.
	Mode[1:0]	[5:4]	R/W	00	Sets the mu controller mode of operation.00 = search and track (recommended).01 = search only. 10 = track.
	Read	3	R/W	0	Set to 1 to read the current value of the mu delay line in.
	Gain[1:0]	[2:1]	R/W	01	Sets the mu controller tracking gain. Recommended to leave at the default 01 setting.
	Enable	0	R/W	0	1 = enable the mu controller. 0 = disable the mu controller.
0x27	MUDEL[0]	7	R/W	0	The LSB of the 9-bit MUDEL setting.
	SRCH_MODE[1:0]	[6:5]	R/W	0	Sets the direction in which the mu controllersearches (from its initial MUDEL setting) for the optimum mu delay line setting that corresponds to the desired phase/slope setting

					(that is, SET_PHS and slope). 00 = down. 01 = up. 10 = down/up (recommended)
	SET_PHS[4:0]	[4:0]	R/W	0	Sets the target phase that the mu controller locks to with a maximum setting of 16. Refer to Table 28 for optimum setting.
0x28	MUDEL[8:1]	[7:0]	W	0x00	With enable (Bit 0, Register 0x26) set to 0, this 9-bit value represents the value that the mu delay is set to. Note that the maximum value is 432. With enable set to 1, this value represents the mu delay value at which the controller begins its search. Setting this value to the delay line midpoint of 216 is recommended.
			R	0x00	When read (Bit 3, Register 0x26) is set to 1, the value read back is equal to the value written into the register when enable = 0 or the value that the mu controller locks to when enable = 1.
0x29	SEARCH_TOL	7	R/W	0	0 = not exact (can find a phase within two values of the desired phase). 1 = finds the exact phase that is targeted (optimal setting).
	Retry	6	R/W	0	0 = stop the search if the correct value is not found. 1 = retry the search if the correct value is not found.
	CONTRST	5	R/W	0	Controls whether the controller resets or continues when it does not find the desired phase. 0 = continue (optimal setting). 1 = reset.
	Guard[4:0]	4	R/W	01011	Sets a guard band from the beginning and end of the mu delay line which the mu controller does not enter into unless it does not find a valid phase outside the guard band (optimal value is Decimal 11 or 0x0B).
0x2A	MU_LST	1	R	0	0 = mu controller has not lost lock. 1 = mu controller has lost lock.
	MU_LKD	0	R	0	0 = mu controller is not locked. 1= mu controller is locked.

Part ID

Table 24. Part ID Register

Address (Hex)	Name	Bit	R/W	Default Setting	Comments
0x35	PART_ID	[7:0]	R	0x20	Part ID number.

Theory of Operation

LVDS Data Port Interface

The CD97D39 supports input data rates from 1.6 GSPS to 2.5 GSPS using dual LVDS data ports. The interface is source synchronous and double data rate (DDR) where the host provides an embedded data clock input (DCI) at f_{DAC} /4 with its rising and falling edges aligned with the data transitions. The data format is offset binary; however, twos complement format can be realized by reversing the polarity of the MSB differential trace. As shown in Figure 23, the host feeds the CD97D39 with deinterleaved input data into two 14-bit LVDS data ports (DB0 and DB1) at ½ the DAC clock rate (that is, f_{DAC} /2). The CD97D39 internal data receiver controller then generates a phase shifted version of DCI to register the input data on both the rising and falling edges.

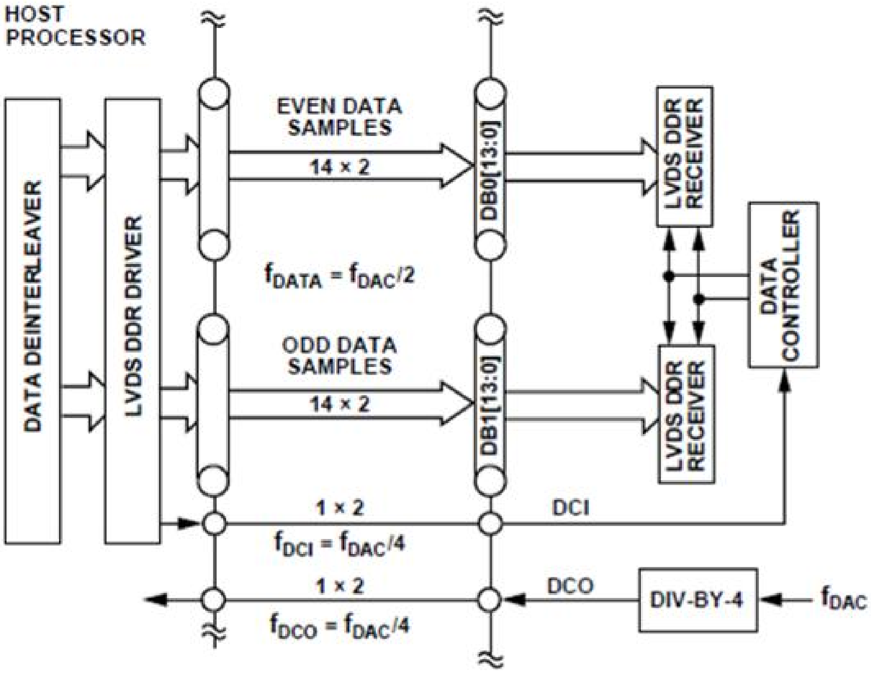


Figure 23.Recommended Digital Interface Between the CD97D39 and Host Processor

As shown in Figure 24, the DCI clocks edges must be coincident with the data bit transitions with minimum skew, jitter, and inter symbol interference. To ensure coincident transitions with the data bits, the DCI signal should be implemented as an additional data line with an alternating (010101…) bit sequence from the same output drivers used for the data. Maximizing the opening of the eye in both the DCI and data signals improves the reliability of the data port interface. Differential controlled impedance traces of equal length (that is, delay) should also be used between the host processor and CD97D39 input to limit bit-to-bit skew.

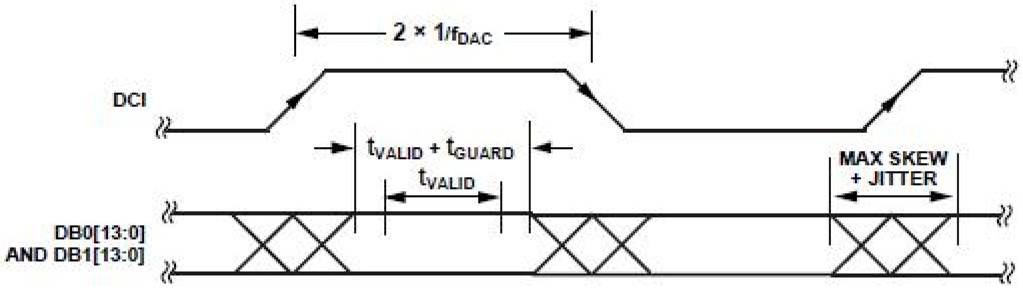


Figure 24. LVDS Data Port Timing Requirements

The maximum allowable skew and jitter out of the host processor with respect to the DCI clock edge on each LVDS port is calculated as ：

MaxSkew + Jitter= Period(ns) −ValidWindow(ps) −Guard

= 800 ps −344 ps −100 ps

= 356 ps

where ValidWindow(ps) is represented by tVALID and Guard is represented by tGUARD in Figure 24. The minimum specified LVDS valid window is 344 ps, and a guard band of 100 ps is recommended. Therefore, at the maximum operating frequency of 2.5 GSPS, the maximum allowable FPGA and PCB bit skew plus jitter is equal to 356 ps.

For synchronous operation, the CD97D39 provides a data clock output, DCO, to the host at the same rate as DCI (that is, $f_{DAC}/4$)

to maintain the lowest skew variation between these clock domains. Because the DCO signal is generated from a separate clock divider, its phase relationship relative to the $f_{DAC}/4$ clocks used by the data receiver controller varies upon each power-up. Applications sensitive to this phase ambiguity (resulting in a ±2 DACCLK pipeline variation) should consider using the sync controller. The host processor has a worst-case skew between DCO and DCI that is both implementation and process dependent. This worst-case skew can also vary an additional 30% over temperature and supply corners. The delay line within the data receiver controller can track a ±1.5 ns skew variation after initial lock. While it is possible for the host to have an internal PLL that generates a synchronous f_{DAC} /4 from which the DCI signal is derived, digital implementations that result in the shortest propagation delays result in the lowest skew variation.

The data receiver controller is used to ensure proper data hand-off between the host and CD97D39 internal digital clock domains. The circuit shown in Figure 25 functions as a delay lock loop in which a 90° phase shifted version of the DCI clock input is used to sample the input data into the DDR receiver registers. This ensures that the sampling instance occurs in the middle of the data pattern eyes (assuming matched DCI and DBx[13:0] delays). Note that, because the DCI delay and sample delay clocks are derived from the divide-by-4 circuitry, this 90° phase relationship holds as long as the delay settings (that is, DCI_DEL, SMP_DEL) are also matched.

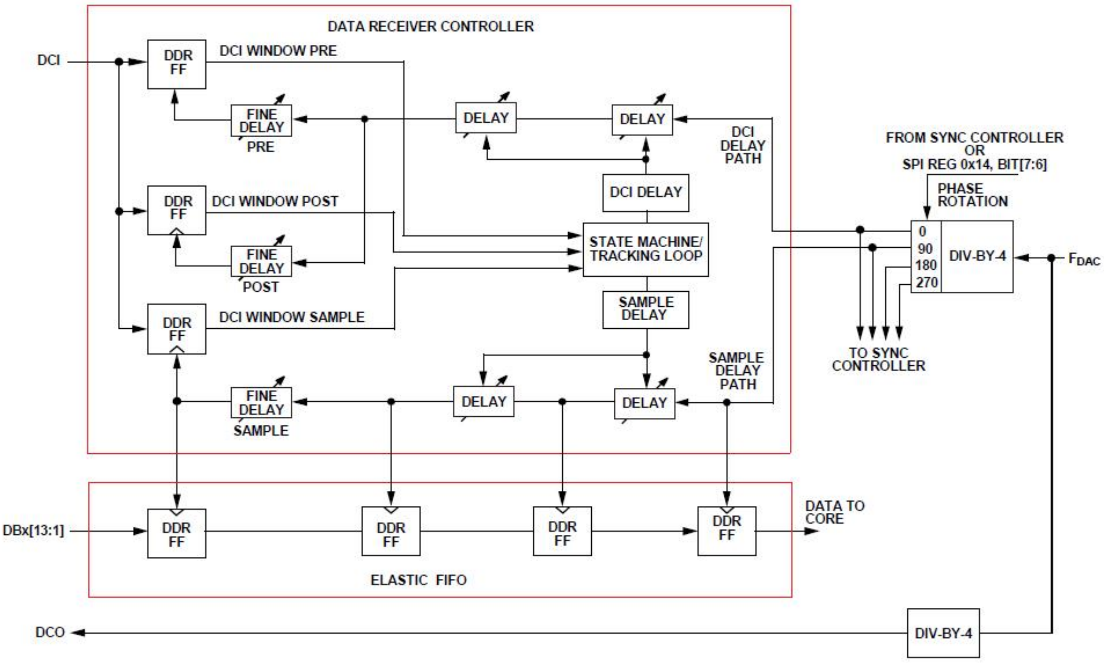


Figure 25. Top Level Diagram of the Data Receiver Controller

Once this data has been successively sampled into the first set of registers, an elastic FIFO is used to transfer the data into the CD97D39 clock domain. To continuously track any phase variation between the two clock domains, the data receiver controller should always be enabled and placed into track mode (Register 0x10, Bit 1 and Bit 0). Tracking mode operates continuously in the background to track delay variations between the host and CD97D39 clock domains. It does so by ensuring that the DCI signal is sampled within a very narrow window defined by two internally generated clocks (that is, PRE and PST), as shown in Figure 26.

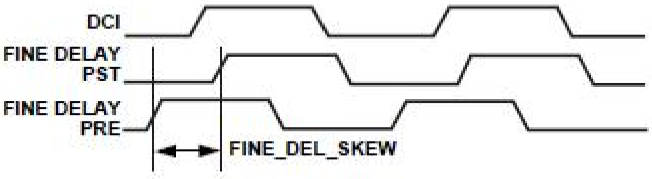


Figure 26. Pre- and Post-Delay Sampling Diagram

Proper sampling of the DCI signal can also be confirmed by monitoring the status of DCI_PRE_PH0 (Register 0x0C, Bit 2) and DCI_PST_PH0 (Register 0x0C, Bit 0). If the delay settings are correct, the state of DCI_ PRE_PH0 should be 0, and the state of DCI_PST_PH0 should be 1. Note that the states of these status bits may toggle occasionally due to cycle-to cycle jitter exceeding the window width. However, the controller averages these status bits over multiple clock cycles to ensure that the DCI signal falls within a programmable window.

The skew or window width (FINE_DEL_SKEW) is set via Register 0x13, Bits[3:0], with a maximum skew of approximately 180 ps and resolution of 12 ps. It is recommended that the skew be set to 36 ps (that is, Register 0x13 = 0x72) during initialization. The skew setting also affects the speed of the controller loop, with tighter skew settings corresponding to longer response time.

Data Receiver Controller Initialization Description

The data controller should be initialized and placed into track mode as the second step in the SPI boot sequence. The following steps are recommended for the initialization of the data receiver controller:

1. Set FINE_DEL_SKEW to 2 for a larger DCI sampling window (Register 0x13 = 0x72). Note that the default DCI_DEL and SMP_DEL settings of 167 are optimum.
2. Disable the controller before enabling (that is, Register 0x10 = 0x00).
3. Enable the Rx controller in two steps: Register 0x10 = 0x02 followed by Register 0x10 = 0x03.
4. Wait 135K clock cycles.
5. Read back Register 0x21 and confirm that it is equal to 0x05 to ensure that the DLL loop is locked and tracking.
6. Include this step for operation <1.6 GSPS. Read back the DCI_DEL value to determine whether the value falls within a user-defined tracking guard band. If it does not, rotate CLKDIVPH by 1 (Register 0x14, Bits[7:6] and go back to Step 2.

Once the controller is enabled during the initial SPI boot process (see Table 28 and Table 292), the controller enters a search mode where it seeks to find the closest rising edge of the DCI clock (relative to a delayed version of an internal f_{DAC} /4 clock) by simultaneously adjusting the delays in the clocks used to register the DCI and data inputs. A state machine searches above and below the initial DCI_DEL value. The state machine first searches for the first rising edge above the DCI_DEL and then searches for the first rising edge below the DCI_DEL value. The state machine selects the closest rising edge and then enters track mode. It is recommended that the default midscale delay setting (that is, Decimal 167) for the DCI_DEL and SMP_DEL bits be kept to ensure that the selected edge remains closest to the delay line midpoint, thus providing the greatest range for tracking timing variations and preventing the controller from falling out of lock. The adjustable delay span for these internal clocks (that is, DCI and sample delay) is nominally 4 ns. The 10-bit delay value is user programmable from the decimal equivalent code (0 to 384) with approximately 12 ps/LSB resolution via the DCI_DEL and SMP_DEL registers (via Register 0x11 thru Register 0x14). When the controller is enabled, it overwrites these registers with the delay value it converges upon. The minimum difference between this delay value and the minimum/maximum values (that is, 0 and 334) represents the guard band for tracking. Therefore, if the controller initially converges upon a DCI_DEL and SMP_DEL value between 80 and 304, the controller has a guard band of at least 80 code (approximately 1 ns) to track phase variations between the clock domains. Upon initialization of the CD97D39, a certain period of time is required for the data receiver controller to lock onto the DCI clock signal. Note that, due to its dependency on the mu controller and synchronization controller (optional), the data receiver controller should be enabled only after these other controllers have been enabled and established locked. All of the internal controllers operate at submultiples of the DAC update rate. The number of f_{DAC} clock cycles required to lock onto the DCI clock is dependent on whether the synchronization controller is enabled as shown in Table 25.

Table 25. Typical/Worst-Case Lock Times for LVDS Controller (Relative to $1/f_{DAC}$)

Synchronization Controller	Typical	Worst Case
Off	70K	135K
Slave（Slave mode)	70K	135K
Master（Main mode)	300K	560K

During the SPI initialization process, the user has the option of polling Register 0x21 (Bit 0, Bit 1, and Bit 3) to determine if the data receiver controller is locked, has lost lock, or has entered into track mode before completing the boot sequence. Alternatively, the appropriate IRQ bit (Register 0x03 and Register 0x04) can be enabled such that an IRQ output signal is generated upon the controller establishing lock (see the Interrupt Requests section). The data receiver controller can also be configured to generate an interrupt request (IRQ) upon losing lock. Losing lock can be caused by excessive jitter on the DCI input signal, disruption of the main DAC clock input, or loss of a power supply rail. To service the interrupt, the host can poll the RCVR_LCK bit (Register 0x21, Bit 0) to determine the current state of the controller. If this bit is cleared, the search/track procedure can be restarted by setting the RCVR_LOOP_ON bit in Register 0x10, Bit 1. After waiting the required lock time, the host can poll the RCVR_LCK bit to see if it has been set. Before leaving the interrupt routine, the RCVR_FLG_RST bit (Register 0x10, Bit 2) should be reset by writing a high followed by a low.

Data Receiver Operation at Lower Clock Rates

At clock rates below 1.6 GSPS, it is recommended to include provisions to rotate the CLKDIVPH setting in the SPI boot process. As previously mentioned, the delay line can be varied over a nominal 4 ns window. If the minimum specified clock rate of 800 MSPS is considered, a DCI clock rate of 200 MSPS corresponds to a 5 ns period, thus exceeding the delay line length. Therefore, it becomes possible that the initial startup phase from the divide-by-4 circuit (and DCO output) is such that the data receiver controller can never establish initial lock upon power up. If the clock rate is increased to 1600 MSPS (that is, DCI clock period of 2.5 ns), the controller always finds at least two DCI clock edges, therefore, establishing lock. However, should the DCI edges fall symmetrically (equal distance) from the initial DCI_DEL midscale setting, a guard band of ±0.75 ns (that is, (4.0 − 2.5)/2)results. Rotating the CLKDIVPH can result in an improvement in this case by skewing one of the DCI edges toward the DCI_DEL midscale value. Rotating the CLKDIVPH phase provides a means of adjusting the delay in course steps of f_{DAC} /4. For example, in the 800 MSPS and 1600 MSPS cases described above, rotating the CLKDIVPH setting by 1 corresponds to a delay shift of 5 ns and 2.5 ns, respectively. By adding an additional step in the SPI initialization routine for the data receiver controller, it becomes possible to increase the effective range of the delay line to ensure a DCI_DEL value that falls within a reasonable guard band.

LVDS Driver and Receiver Input

The CD97D39 features a LVDS-compatible driver and receivers. The LVDS driver output used for the DCO and SYNC_OUT signal includes an equivalent 200 Ω source resistor that limits its nominal output voltage swing to ±200 mV when driving a 100 Ω load. The DCO output driver can be powered down via Register 0x01, Bit 5. An equivalent circuit is shown in Figure 27.

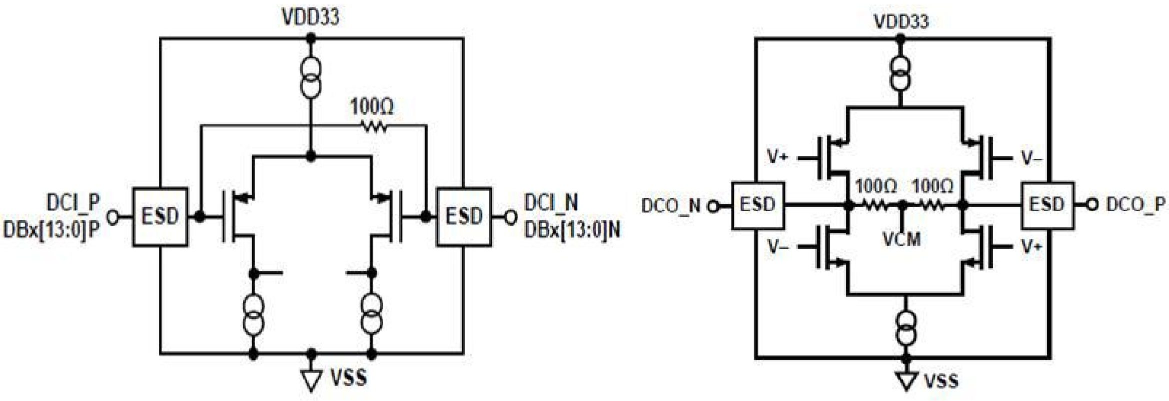


Figure 27. Equivalent LVDS Input and Output

The LVDS receivers include 100 Ω termination resistors, as shown in Figure 27. These receivers meet the IEEE-1596.3-1996 reduced swing specification (with the exception of input hysteresis, which cannot be guaranteed over all process corners).

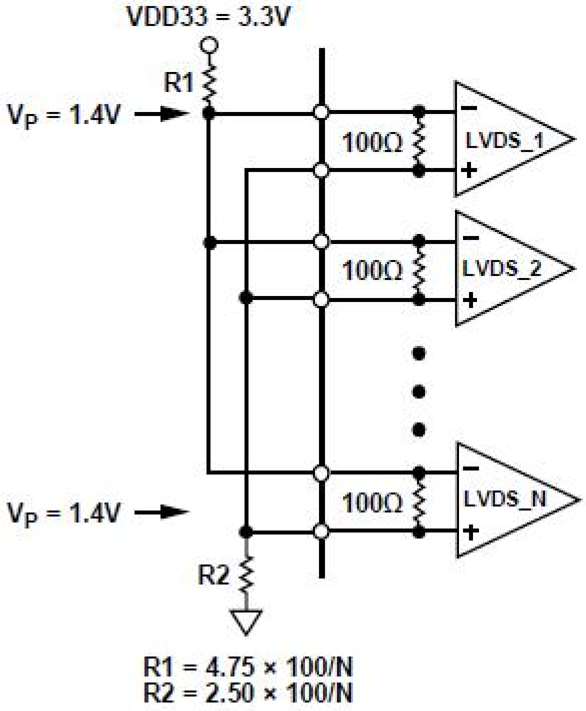


Figure 28. Resistor Network to Bias Unused LVDS Data Inputs

MU Controller

A delay lock loop (DLL) is used to optimize the timing between the internal digital and analog domains of the CD97D39 such that data is successfully transferred into the DAC core at rates of up to 2.5 GSPS. As shown in Figure 29, the DAC clock is split into an analog and a digital path with the critical analog path leading to the DAC core (for minimum jitter degradation) and the digital path leading to a programmable delay line. Note that the output of this delay line serves as the master internal digital clock from which all other internal and external digital clocks are derived. The amount of delay added to this path is under the control of the mu controller, which optimizes the timing between these two clock domains and continuously tracks any variation (once in track mode) to ensure proper data hand-off.

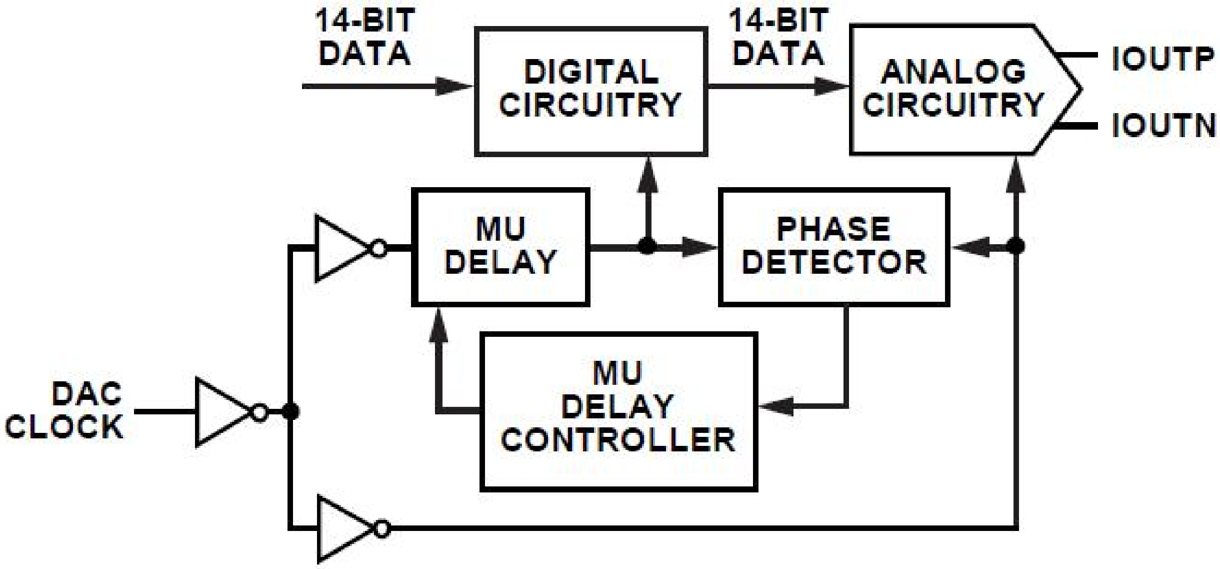


Figure 29. Mu Delay Controller Block Diagram

The mu controller adjusts the timing relationship between the digital and analog domains via a tapped digital delay line having a nominal total delay of 864 ps. The delay value is programmable to a 9-bit resolution (that is, 0 to 432 decimal) via the MUDEL register, resulting in a nominal resolution of 2 ps/LSB. Because a time delay maps to a phase offset for a fixed clock frequency, the control loop essentially compares the phase relationship between the two clock domains and adjusts the phase (that is, via a tapped delay line) of the digital clock such that it is at the desired fixed phase offset (SET_PHS) from the critical analog clock. Figure 30 maps the typical mu phase characteristic at 2.4 GSPS vs. the 9-bit digital delay setting (MUDEL). The mu phase scaling is such that a value of 16 corresponds to 180 degrees. The critical keep-out window between the digital and analog domains occurs at a value of 0 (but can extend out to 2 depending on the clock rate). The target mu phase (and slope) is selected to provide optimum ac performance while ensuring that the mu controller for any device can establish and maintain lock. For example, while a slope and phase setting of −6 is considered optimum for operation between 1.6 GSPS and 2.5 GSPS, other values are required below 1.6 GSPS.

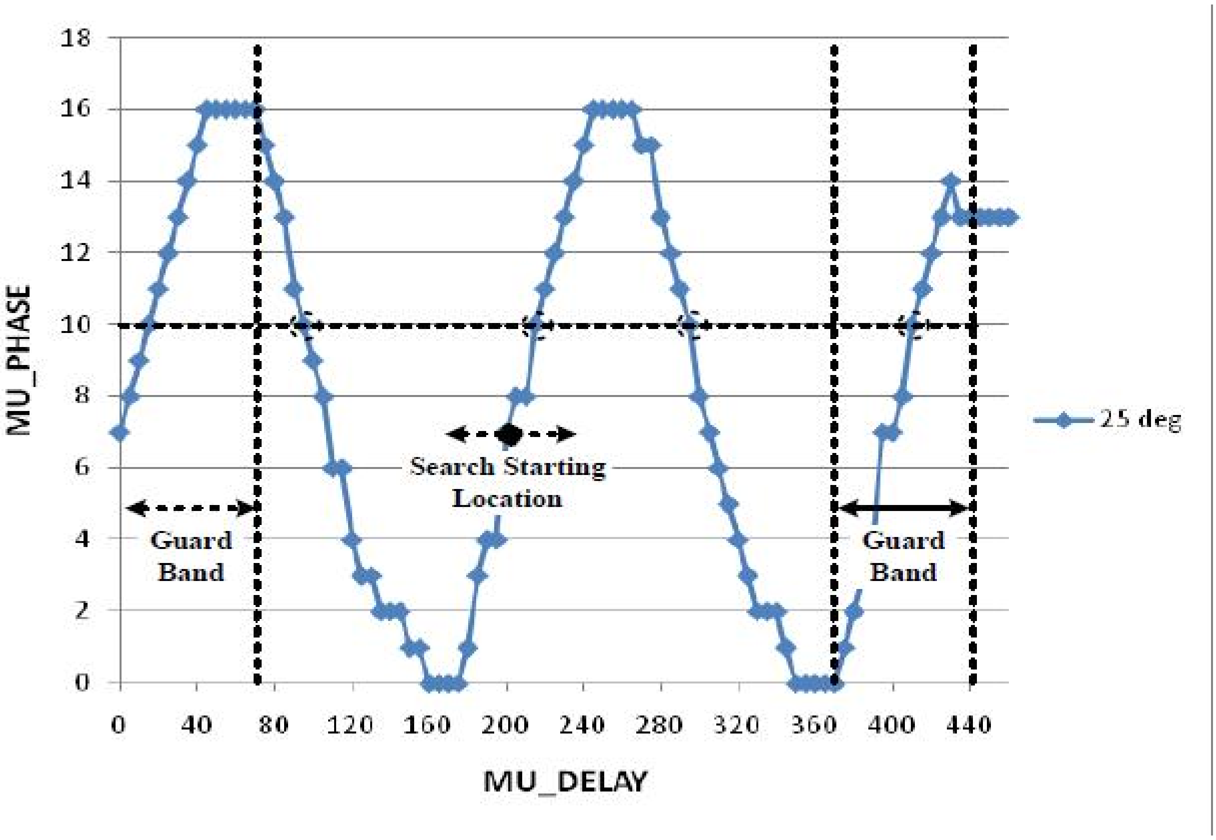


Figure 30. MU phase characteristic vs. Delay

The mu phase characteristics can vary significantly among devices due to gm variations in the digital delay line that are sensitive to process skews (along with temperature and supply). As a result, careful selection of the target phase location is required such that the mu controller can converge upon this phase location for all devices. Figure 31 shows that mu phase characteristics of three devices at 25°C from slow, nominal, and fast skew lots at 800MSPS. Note that a −6 mu phase setting does not map to any delay line tap setting for the fast process skew case; therefore, another target mu phase is recommended at this clock rate. Table 28 provides a list of recommended mu phase/slope settings over the specified clock range of the CD97D39 based on the considerations previously described. These values should be used to ensure robust operation of the mu controller.

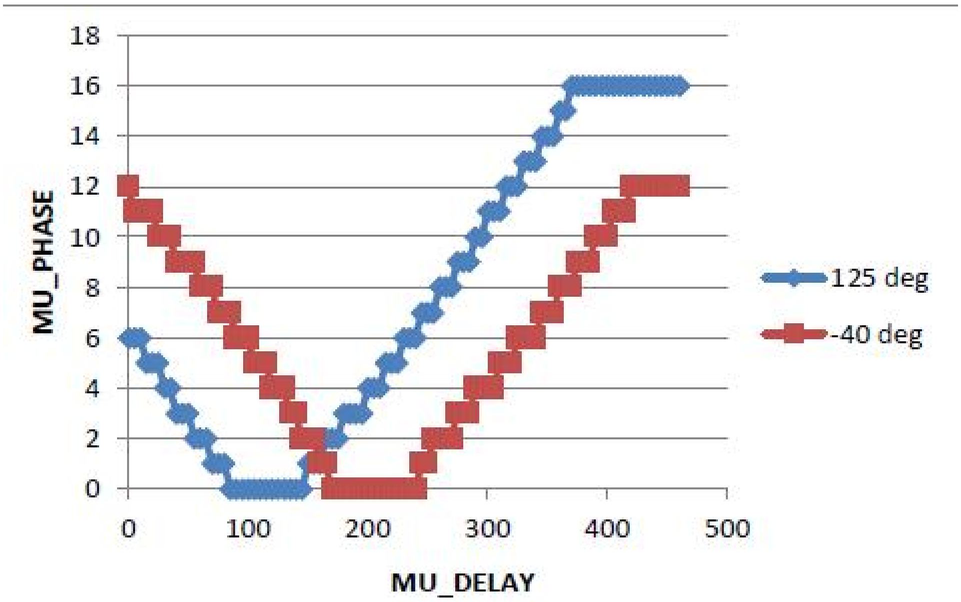


Figure 31. MU phase characteristic vs. Delay vs. Temperature

Table 26 provides a list of recommended mu phase/slope settings over the specified clock range of the CD97D39 based on the considerations previously described. These values should be used to ensure robust operation of the mu controller.

Table 26. Recommended Target Mu Phase Settings vs. Clock Rate

Clock Rate（GSPS）	Slope	MU Phase
0.8	+	5
0.9	+	11
1.0	-	10
1.1	-	4
1.2	-	5
1.3	-	4
1.4	+	6
1.5	+	8
1.6 to 2.4	-	10

After the mu controller completes its search and establishes lock on the target mu phase, it attempts to maintain a constant timing relationship between the two clock domains over the specified temperature and supply range. If the mu controller requests a mu delay setting that exceeds the tapped delay line range (that is, <0 or >432), the mu controller can lose lock, causing possible system disruption (that is, can generate IRQ or restart the search). To avoid this scenario, symmetrical guard bands are recommended at each end of the mu delay range. The guard band scaling is such that one LSB of Guard[4:0] (Register 0x29) corresponds to eight LSBs of MUDEL (Register 0x28). The recommended guard band setting of 11 (that is, Register 0x29 = 0xCB) corresponds to 88 LSBs, thus providing sufficient margin.

MU Controller Initialization Description

The mu controller must be initialized and placed into track mode as a first step in the SPI boot sequence. The following steps are required for initialization of the mu controller. Note that the CD97D39 data sheet specifications and characterization data are based on the following mu controller settings:

1. Turn on the phase detector with boost (Register 0x24 = 0x30).
2. Enable the mu delay controller duty-cycle correction circuitry and specify the recommended slope for phase. (that is, Register 0x25 = 0x80 corresponds to a negative slope).
3. Specify search/track mode with a recommended target phase, SET_PHS, of 6 (for example) and an initial MUDEL[8:0] setting of 216 (Register 0x27 = 0x46 and Register 0x28 = 0x6C).
4. Set search tolerance to exact and retry if the search fails its initial attempt. Also, set the guard band to the recommended setting of 11 (Register 0x29 = 0xCB).
5. Set the mu controller tracking gain to the recommended setting and enable the mu controller state machine (Register 0x26 = 0x03).

Upon completion of the last step, the mu controller begins a search algorithm that starts with an initial delay setting specified by the MUDEL register (that is, 216, which corresponds to the midpoint of the delay line). The initial search algorithm works by sweeping through different mu delay values in an alternating manner until the desired phase (that is, a SET_PHS of 4) is exactly measured. When the desired phase is measured, the slope of the phase measurement is then calculated and compared against the specified slope (slope = negative). If everything matches, the search algorithm is finished. If not, the search continues in both directions until an exact match can be found or a programmable guard band is reached in one of the directions. When the guard band is reached, the search still continues but only in the opposite direction. If the desired phase is not found before the guard band is reached in the second direction, the search changes back to the alternating mode and continues looking within the guard band. The typical locking time for the mu controller is approximately 180k DAC cycles (at 2 GSPS ~ 75 μs). The search fails if the mu delay controller reaches the endpoints. The mu controller can be configured to retry (Register 0x29, Bit 6) the search or stop. For applications that have a microcontroller, the preferred approach is to poll the MU_LKD status bit (Register 0x2A, Bit 0) after the typical locking time has expired. This method allows the system controller to check the status of other system parameters (that is, power supplies and clock source) before reattempting the search (by writing 0x03 to Register 0x26). For applications that do not have polling capabilities, the mu controller state machine should be reconfigured to restart the search in hopes that the system' s condition that did not cause locking on the first attempt has disappeared. Once the mu delay value is found that exactly matches the desired mu phase setting and slope (for example, 6 with a negative. slope), the mu controller goes into track mode. In this mode, the mu controller makes slight adjustments to the delay value to track any variations between the two clock paths due to temperature, time, and supply variations. Two status bits, MU_LKD (Register 0x2A, Bit 0) and MU_LST (Register 0x2A, Bit 1) are available to the user to signal the existing status control loop. If the current phase is more than four steps away from the desired phase, the MU_LKD bit is cleared, and if the lock acquired was previously set, the MU_LST bit is set. Should the phase deviation return to within three steps, the MU_LKD bit is set again while the MU_LST is cleared. Note that this sort of event may occur if the main clock input (that is, DACCLK) is disrupted or the mu controller exceeds the tapped delay line range (that is, <0 or >432).If lock is lost, the mu controller has the option of remaining in the tracking loop or resetting and starting the search again via the CONTRST bit (Register 0x29, Bit 5). Continued tracking is the preferred state because it is the least disruptive to a system in which the CD97D39 temporarily loses lock. The user can poll the mu delay and phase value by first setting the read bit high (Register 0x26, Bit 3). Once the read bit is set, the MUDEL[8:0]bits and the SET_PHS[4:0] bits (Register 0x27 and Register 0x28) that the controller is currently using can be read.

Interrupt Requests

The CD97D39 can provide the host processor with an interrupt request output signal (IRQ) that indicates that one or more of the CD97D39 internal controllers have achieved lock or lost lock. These controllers include the mu, data receiver, and synchronization controllers. The host can then poll the IRQ status register (Register 0x04) to determine which controller has lost lock. The IRQ output signal is an active high output signal available on Pin F13. If used, its output should be connected via a 10 kΩ pull-up resistor to VDD33. Each IRQ is enabled by setting the enable bits in Register 0x03, which purposely has the same bit mapping as the IRQ status bits in Register 0x04. Note that these IRQ status bits are set only when the controller transitions from a false to true state. Therefore, it is possible for the x_LCK_IRQ and x_LST_IRQ status bits to be set when a controller temporarily loses lock but is able to reestablish lock before the IRQ is serviced by the host. In this case, the host should validate the present status of the suspect controller by reading back its current status bits, which are available in Register 0x21 and/or Register 0x2A. Based on the status of these bits, the host can take appropriate action, if required, to reestablish lock. To clear an IRQ after servicing, it is necessary to reset relevant bits in Register 0x03 by writing 0 followed by another write of 1 to reenable. A detailed diagram of the interrupt circuitry is shown in Figure 32.

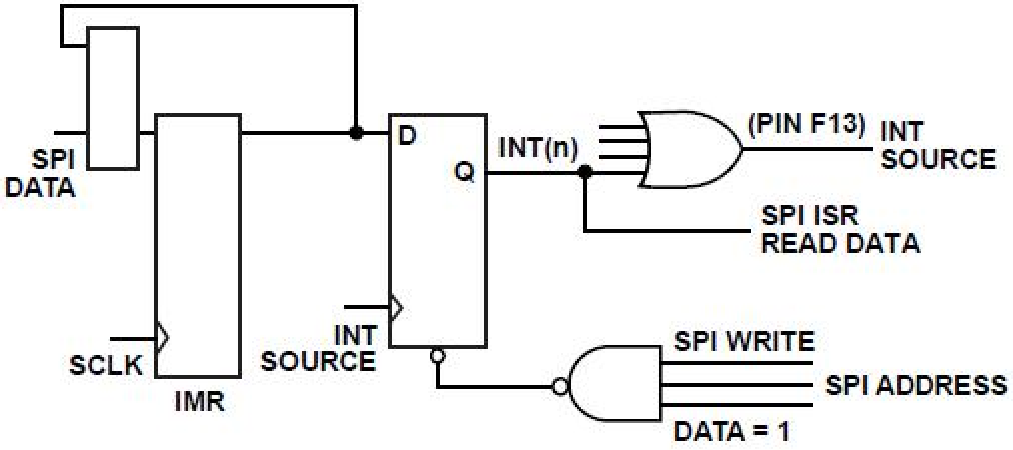


Figure 32. Interrupt Request Circuitry

It is also possible to use the IRQ during the CD97D39 initialization phase after power-up to determine when the mu and data receiver controllers have achieved lock. For example, before enabling the mu controller, the MU_LCK_EN bit (Register 0x03, Bit 2) can be set and the IRQ output signal monitored to determine when lock has been established before continuing in a similar manner with the data receiver controllers. Note that the relevant LCK bit should be cleared before continuing to the next controller. After all controllers are locked, the lost lock enable bits (that is, x_LST_EN) should be set.

Table 27. Interrupt Request Registers

Address (Hex)	Bit	Description
0x03	5	SYNC_LST_EN
	4	SYNC_LCK_EN
	3	MU_LST_EN
	2	MU_LCK_EN
	1	RCV_LST_EN

	0	RCV_LCK_EN
0x04	5	SYNC_LST_IRQ
	4	SYNC_LCK_IRQ
	3	MU_LST_IRQ
	2	MU_LCK_IRQ
	1	RCV_LST_IRQ
	0	RCV_LCK_IRQ
0x21	7	SYNC_TRK_ON
	5	SYNC_LST
	4	SYNC_LCK
	3	RCVR_TRK_ON
	1	RCVR_LST
	0	RCVR_LCK
0x2A	1	MU_LST
	0	MU_LKD

Multiple Device Synchronization

Synchronization of multiple CD97D39 devices requires all of the devices to have matching pipeline delays. This implies the DAC outputs are time aligned to the same phase when all devices are fed with the same data pattern at the same instance of time. The main contributor to phase ambiguity between devices is from the divide-by-4 circuitry that drives the Rx data path and data controller (see Figure 34). This phase ambiguity can result in a ±2 sample offset between any two devices. Because the state of this internal divider is unknown at power-up, a synchronization method that phase aligns the digital paths of multiple CD97D39 devices is required to ensure matching pipeline delays. Figure 33 shows a top-level diagram of multiple CD97D39 devices synchronized to each other with sample alignment of the different data streams within the FPGA (or among multiple FPGAs) being assumed. A common RF clock source is distributed to each of the CD97D39 devices via a dual clock buffer (such as the CLK IC) with matched PCB trace lengths to each device to ensure matched propagation delays. One CD97D39 is designated as the master providing a SYNC_OUT reference clock (equal to f_{DAC} /4) to itself as well as the other CD97D39 slave device’ s SYNC_IN input. LVDS fanout buffers with matched output delays are again used to distribute the SYNC_OUT and DCO signals of the master to the slave devices and FPGAs, respectively, thus ensuring tight time alignment. Note, in the case of a single FPGA implementation (that is, I/Q application), the DCO of the master can drive the FPGA directly. After synchronization, the internal divide-by-4 circuitry has equal phases that drive their respective LVDS controllers. Note, the mu and data receiver controller of both devices must be configured for the same SPI register settings (that is, SET_PHS and DCI_DEL) upon SPI initialization such that controllers converge to similar delays. To validate that delays are roughly matched, the user can read back the delays of both devices (that is, MUDEL and DCI_DEL) to determine if they are in an acceptable window that accounts for slight mismatches between different devices’ delay lines.

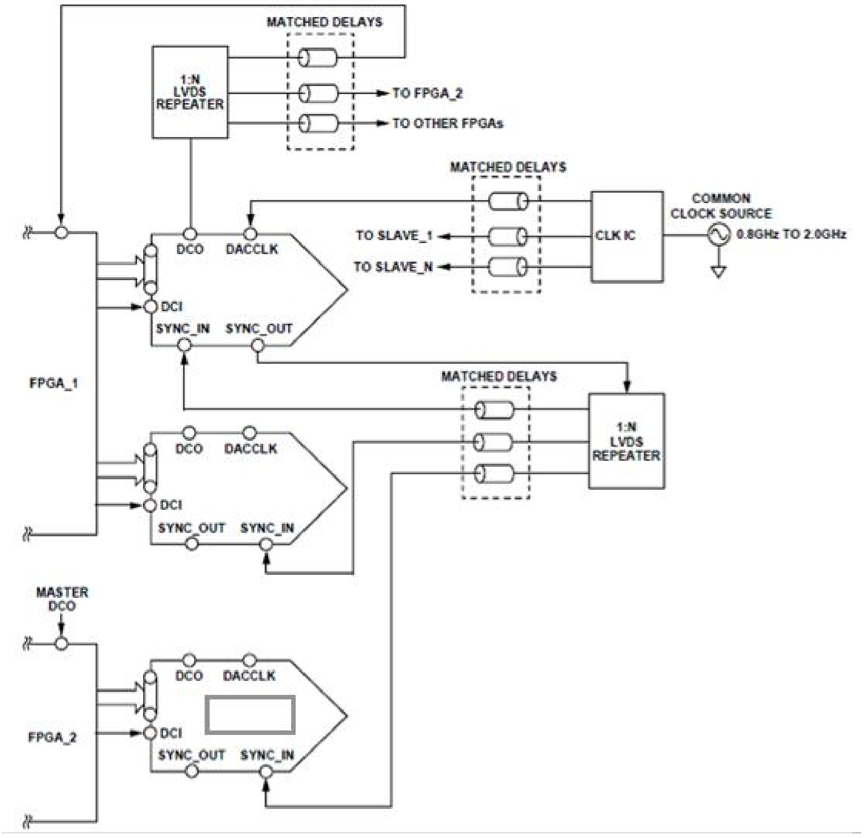


Figure 33. Functional Block Diagram of Two CD97D39 Devices Synchronized

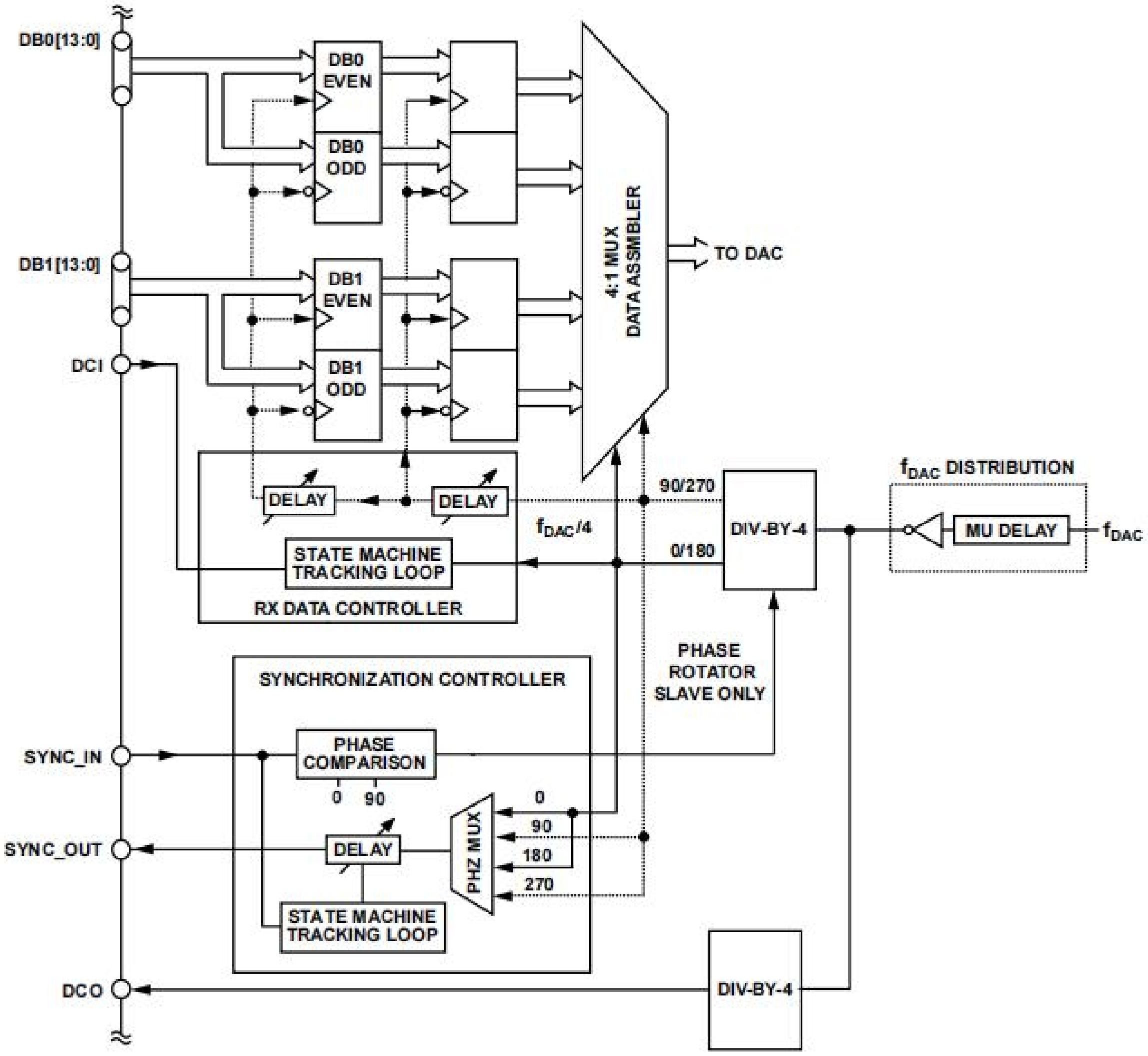


Figure 34. Top Level Block Diagram of Synchronization Circuitry and Controller

Figure 34 shows a top-level diagram of the synchronization controller (bottom) and how it interfaces to other digital functional blocks within the CD97D39. Note the following observations of this top level diagram:

- Synchronization between multiple devices is achieved by rotating the divide-by-4 phases of the slave devices such that they align with the master.
- For the slave devices, the sync controller compares the phase alignment of the master’s SYNC_IN reference signal with the initial 0o /90o outputs of the divide-by-4 and then rotates the divide-by-4 phase until the SYNC_IN signal falls between these phases.
- A reference signal common to all devices is required for synchronization. The master device generates this signal by providing a SYNC_OUT signal which is then distributed to all the devices (including itself with tight time alignment) as a SYNC_IN signal.
- Because the SYNC_IN signal has a defined relationship between the divide-by-4 phase of the

master, the slave devices can now align their respective divide-by-4 phases to the SYNC_IN phase thus ensuring phase alignment among all devices.

- It is not possible to manually rotate the divide-by-4 phases of the data path with the sync controller enabled. This can be a problem at lower clock rates were one may desire to rotate the divide-by-4 phase to ensure locking of the data receiver controller and/or achieve a more optimum DCI_DEL value.
- The DCO output signal is generated from a separate divide-by-4 circuit, and therefore, has a random phase upon each startup. For this reason, the DCO of the master should be distributed to all the FPGAs.

SYNC Controller Initialization Description

The sync controller of the master is enabled by writing 0x70 to Register 0x10. Once enabled, a state machine automatically adjusts the output delay of its SYNC_OUT signal such that the fed back reference SYNC_IN signal is centered between the 0° and 90° output phases associated with its divide-by-4 circuit. Note that the coarse delay is performed by shifting phases via PHZ MUX while the fine delay that centers (and tracks) variation is done by a variable delay line. The variable delay line tap size is 12 ps. Once SYNC_IN is centered, the controller enters tracking mode such that SYNC_IN remains centered despite possible system variations in temperature and/or supply. Centering the SYNC_IN signal on the master device ensures that the SYNC_IN signals of the slave devices also remain centered between their respective divide-by-4 phases; therefore, providing the greatest margin to absorb non ideal timing skews. The following status bits are available in Register 0x21 indicating lock, lost-lock, and tracking: SYNC_LCK, SYNC_LST and SYNC_TRK_ON. The sync controller of the slave is enabled by writing 0x50 to Register 0x10. Once enabled, the state machine compares the reference SYNC_IN signal to the 0°/90° phase outputs of the divide-by-4 phase settings. If the SYNC_IN signal does not fall between these phases, the state machine of the slave rotates the divide-by-4 phase setting until it does. To validate that phase alignment has been achieved, the SYNC_IN_PH90 and SYNC_IN_PH0 status bits should read 1 and 0, respectively (that is, Register 0x0D, Bits[5:4]). Note that the DCO and SYNC_OUT outputs of the slave can be disabled via Register 0x01, Bit 5.

Synchronization Limitations

Ensuring consistent synchronization over production lots in systems containing two or more CD97D39 devices becomes increasingly more challenging at the higher update rates because the timing offset between adjacent phases of the divide-by-4 output clock is equal to $1/f_{DAC}$. For example, a DAC update of 2 GSPS corresponds to a 500 ps period. If the SYNC_IN signal of an ideal master device is positioned in the center of its divide-by-4 0° and 90° phase outputs, only ±250 ps of timing margin exists for the slave devices. Thisideal margin is actually reduced by quadrature phase errors in the divide-by-4 circuit of the master as well as its ability to position the SYNC_IN exactly in the center of the 0° and 90° output phases.

The timing margin is further eroded by the following sources:

- Master-to-slave device(s) mismatch in the propagation delays in the mu delay clock path and SYNC_IN. Note that these mismatches can be up to 100 ps between devices that are at opposite extremes of the process corners.
- Quadrature phase errors in the divide-by-4 outputs of the slave. These sources of timing skews become more significant as the DACCLK period is decreased (that is, clock rate is increased), leaving less margin for timing skews external to the master-to slave device(s). Special consideration to PCB layout and selection of clock distribution ICs are required to ensure minimum skew between the distributed DACCLK and SYNC_IN signals. Note that timing skews can quickly accumulate considering that the propagation delay on an FR4 PCB is on the order of 170 ps/inch, and that output-to-output skews on each clock distribution IC

can be as high as 25 ps.

The problem becomes more pronounced in multi board synchronization where clock signals (that is, DACCLK, SYNC_OUT, and DCO) are distributed over a back plane to multiple PCBs. Data alignment among the various data sources is required when driven by phase aligned DCO signals that are a buffered version of the master’ s DCO. However, these data sources (FPGAs) also have process, supply voltage, and temperature sensitivities (PVTs) that can cause misalignment among their respective DCI outputs. Adding to this dilemma is that it also possible for the data receiver controller of different CD97D39 devices to converge on different delay settings due to PVT variations of the delay line (even if DCI inputs are exactly aligned). This can result in a four sample pipeline mismatch between devices if the difference in absolute delays exceeds a period of $4/f_{DAC}$. Recall that the controller searches up/down for its first valid edge from its initial start value (that is, DCI_DEL and SMP_DEL). While the initial start values between devices should be made the same, different absolute time delays due to PVT can cause devices to converge on different edges of DCI above or below this initial start value. As a result, confirm that DCI_DEL values between multiple devices are matched sufficiently such that the absolute differences between the readback DCI_DEL values do not exceed a data period (that is, $4/f_{DAC}$). If the difference exceeds a data period, modify the DCI_DEL (and SMP_DEL) setting of the slave device so that its start point is roughly ½ the difference between the master and slave readback values.

Analog Interface

Analog Modes of Operation

As shown in Figure 35, the CD97D39 uses a four switch architecture. In the half clock cycle, only one pair of switches is enabled, so it is necessary to use each pair of switches on the alternate clock edge. The main advantage of the four switch architecture is that it can eliminate the code type related glitches in the output signals of traditional dual switch DAC architecture.

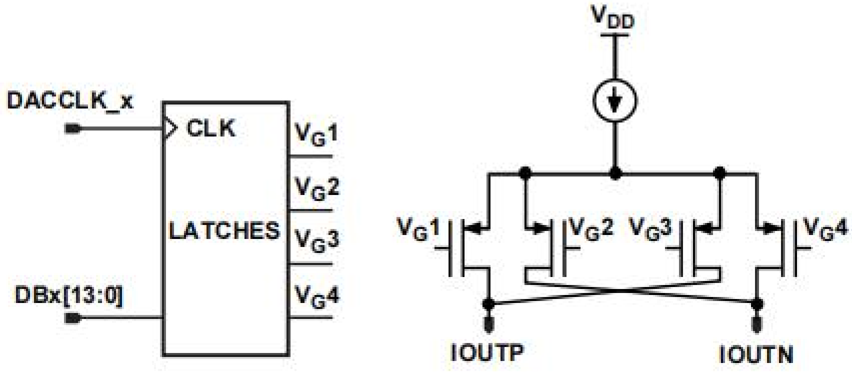


Figure 35. Four switch analog output architecture of CD97D39

In the double on/off architecture, when switching occurs and D1 and D2 are in different states, burrs will occur. However, if D1 and D2 are in the same state, switching will not cause burrs. This code related glitch will increase the distortion of DAC. In the four switch architecture, no matter what kind of code is input, there are always two switches switching at half a clock cycle, so code related glitches will not be generated, but large 2 * f_{DAC} CLK clock spurious will be generated.

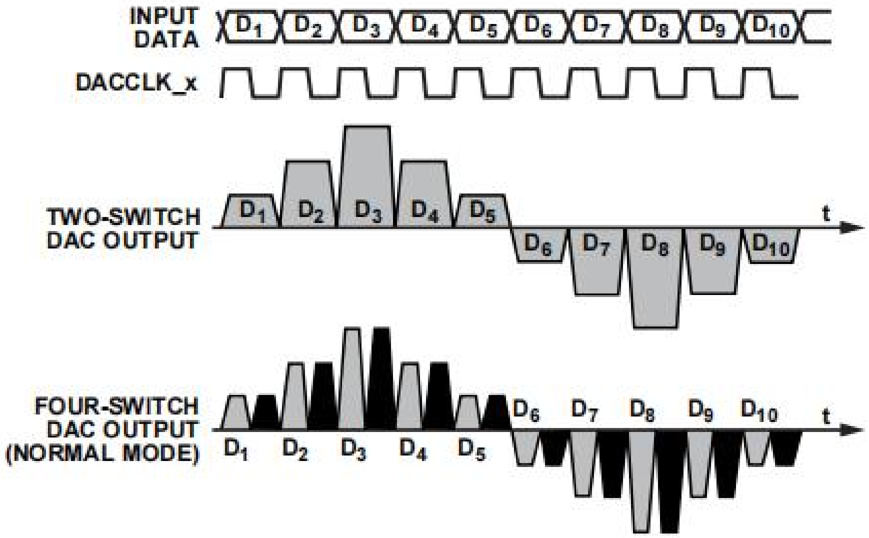

INPUT DATA
D1 D2 D3 D4 D5 D6 D7 D8 D9 D10
DACCLK_x
TWO-SWITCH DAC OUTPUT
t
FOUR-SWITCH DAC OUTPUT (NORMAL MODE)

Figure 36. Output Waveform of Double Switch and Four Switch DAC

Another feature of 4-switch DAC is that the core of DAC can work in three modes: normal mode, mix mode (Mix mode) and return to zero mode (RZ mode). The working mode of CD97D39 can be set through bit [1:0] of SPI register 0x08. After power on and reset, it defaults to normal mode. When mixing mode is used, the device effectively clipping the output signal at DAC clock rate. Its function is to reduce the power of the fundamental frequency signal, thereby improving the output power of the mirror signal. The return to zero (RZ) mode is basically similar to the mixing mode, except that the middle sample is replaced by the midpoint value output by the DAC. Refer to Figure 37 for output waveforms of mixing mode and zeroing mode.

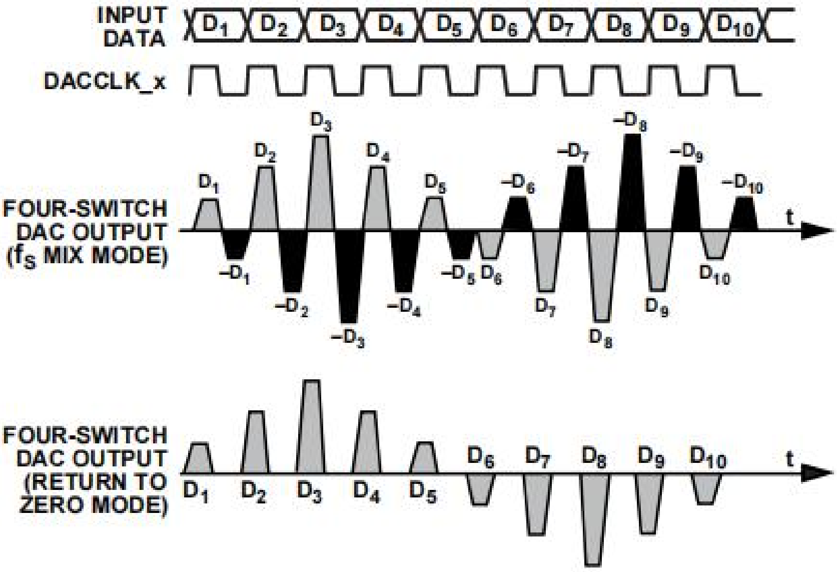


Figure 37. Output Waveform of DAC in Mixing Mode and Reset to Zero Mode

Figure 38 shows the DAC waveforms for both the mix mode and the RZ mode. Note that the disadvantage of the RZ mode is the 6 dB loss of power to the load because the DAC is only functioning for ½ the DAC update period. This ability to change modes provides the user the flexibility to place a carrier anywhere in the first three Nyquist zones, depending on the operating mode selected. Switching between the analog modes reshapes the sinc roll-off inherent at the DAC output. The maximum amplitude in all three Nyquist zones is impacted by this sinc roll-off, depending on where the carrier is placed (see Figure 38). As a practical matter, the usable bandwidth in the third Nyquist zone becomes limited at higher DAC clock rates (that is, >2 GSPS) when the output bandwidth of DAC core and the interface network (that is, balun) contributes to additional roll-off.

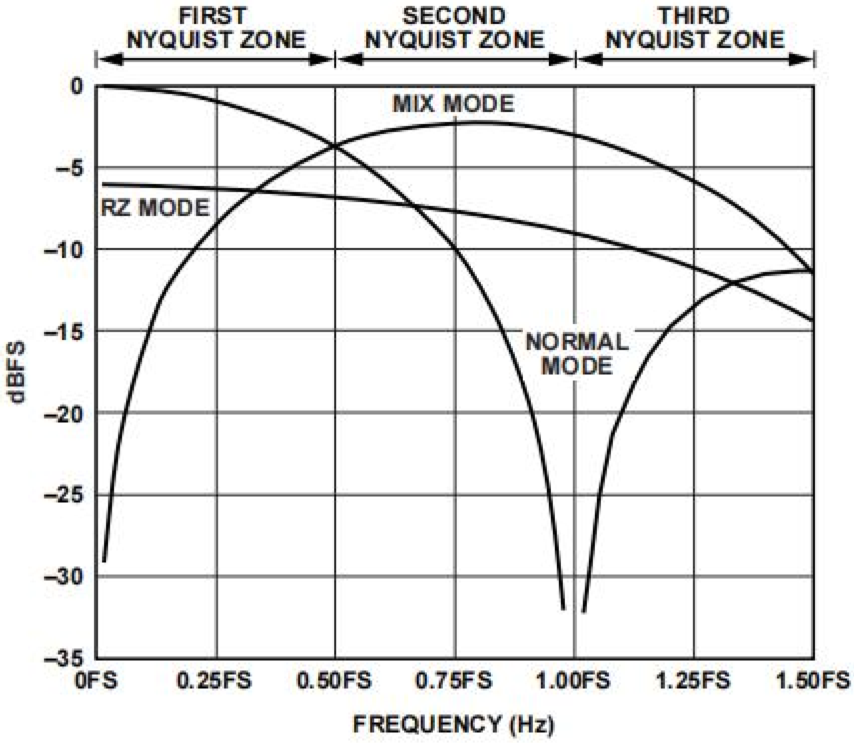


Figure 38. Sinc Roll-Off for Each Analog Operating Mode

Clock Input

The quality of clock signal and the appropriate level will directly affect the AC output characteristics of DAC. The phase noise and spurious characteristics of the selected clock source should meet the requirements of the target application. The phase noise and spurious at the specified frequency offset of the clock source will be directly converted to the output signal. It can be proved that when the impact of DAC clock path on clock performance can be ignored without timing, the relationship between the phase noise characteristics of the sine wave output after DAC reconstruction and the clock source is 20xlog10 (f_{OUT}/fCLK). The CD97D39 is internally integrated with a high-performance clock receiver. When the external input clock power is as low as 0dBm, it can still achieve good DAC output performance. Figure 39 shows a typical DAC clock input reference circuit, which can ensure good DAC performance even when the clock input is as low as 0dBm.

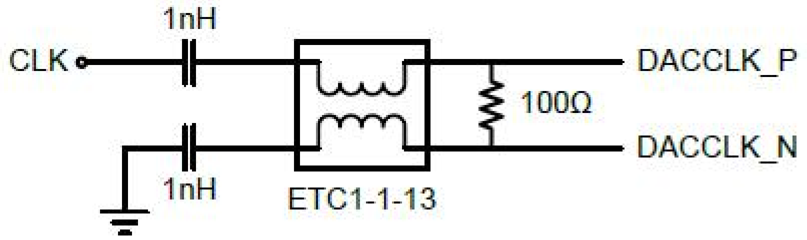


Figure 39. CD97D39 CLK Input Ref circuit

Voltage Reference

The CD97D39 output current is set by a combination of digital control bits and the I120 reference current, as shown in Figure 40. The reference current is obtained by forcing the band gap voltage across an external 10 kΩ resistor from I120 (Pin B14) to ground. The 1.2 V nominal band gap voltage (VREF) generates a 120 μA reference current in the 10 kΩ resistor.

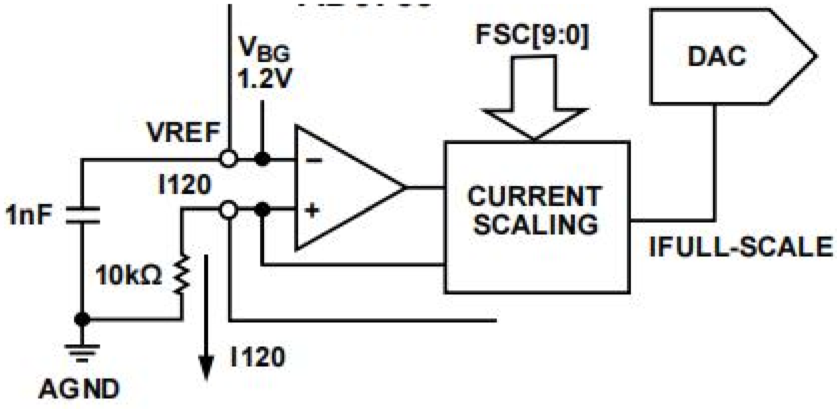


Figure 40. Voltage Reference Circuit

Note the following constraints when configuring the voltage reference circuit:

- Both the 10 kΩ resistor and 1 nF bypass capacitor are required for proper operation.
- Adjusting the output full-scale current, IOUTFS, of the DAC from its default setting of 20 mA should be performed digitally.
- The CD97D39 is not a multiplying DAC. Modulating the reference current, I120, with an ac signal is not supported.
- The band gap voltage appearing at VREF (Pin C14) must be buffered for use with external circuitry because its output impedance is approximately 5 kΩ.
- An external reference can be used to overdrive the internal reference by connecting it to

VREF (Pin C14).

I_{OUTFS} can be adjusted digitally over 8.7 mA to 31.7 mA by using FSC[9:0] (Register 0x06 and Register 0x07).

The following equation relates I_{OUTFS} to the FSC[9:0] register, which can be set from 0 to 1023:

$$I_{OUTFS} = 22.6 \times FSC[9:0]/1000 + 8.7 \quad (1)$$

Note that a default value of 0x200 generates 20 mA full scale, which is used for most of the characterization presented in this data sheet (unless noted otherwise).

Analog Outputs

Equivalent DAC Output and Transfer Function

The CD97D39 provides complementary current outputs, IOUTP and IOUTN, that source current into an external ground reference load. Figure 41 shows an equivalent output circuit for the DAC. Note that, compared to most current output DACs of this type, the CD97D39 outputs exhibit a slight offset current (that is, $I_{OUTFS}/16$), and the peak differential ac current is slightly below $I_{OUTFS}/2$ (that is, $15/32 \times I_{OUTFS}$).

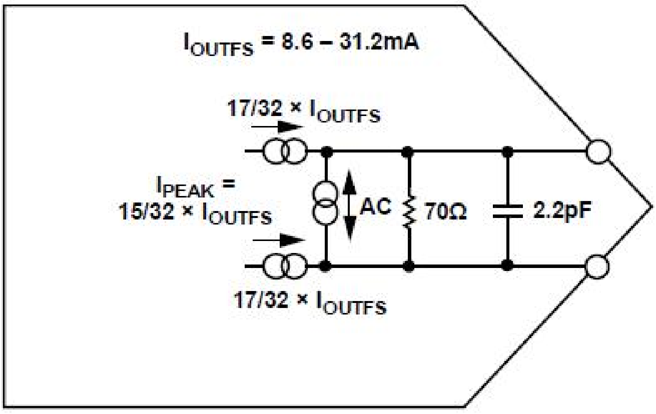


Figure 41. Equivalent DAC Output Circuit

As shown in Figure 41, the DAC output can be modeled as a pair of dc current sources that source a current of 17/32 × IOUTFS to each output. A differential ac current source, IPEAK, is used to model the signal-dependent nature of the DAC output. The polarity and signal dependency of this ac current source are related to the digital code by the following equations:

$$F(Code) = (DACCODE - 8192)/8192 \quad (2)$$

$$-1 \le F(Code) < 1 \quad (3)$$

where DACCODE = 0 to 16,383 (decimal).

Because I_{PEAK} can swing ±(15/32) × I_{OUTFS}, the output currents measured at IOUTP and IOUTN can span from I_{OUTFS}/16 to I_{OUTFS}. However, because the ac signal-dependent current component is complementary, the sum of the two outputs is always constant (that is, IOUTP + IOUTN = (34/32) × I_{OUTFS}). The code-dependent current measured at the IOUTP (and IOUTN) output is as follows:

$$IOUTP = 17/32 \times IOUTFS + 15/32 \times IOUTFS \times F(Code) \tag{4}$$

$$IOUTN = 17/32 \times IOUTFS - 15/32 \times IOUTFS \times F(Code) \tag{5}$$

Figure 42 shows the IOUTP vs. DACCODE transfer function when IOUTFS is set to 19.65 mA.

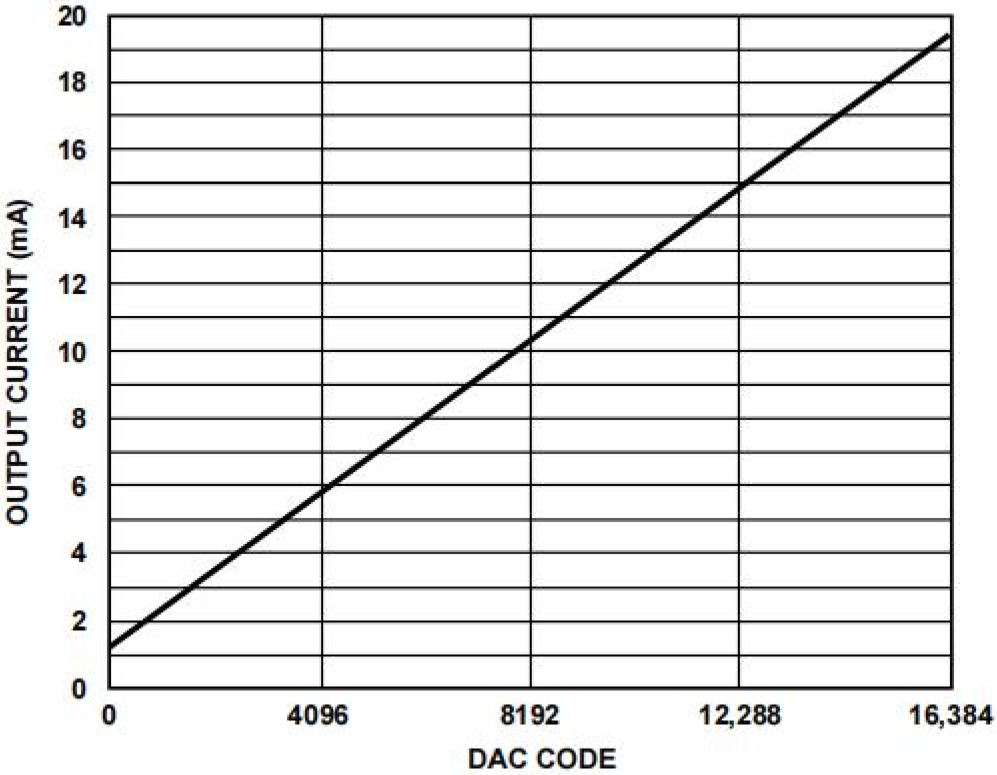


Figure 42. Gain Curve for FSC[9:0] = 512, DAC Offset = 1.228 mA

Peak DAC Output Power Capability

The maximum peak power capability of a differential current output DAC is dependent on its peak differential ac current, IPEAK,and the equivalent load resistance it sees. Because the CD97D39 includes a differential 70 Ω resistance, it is best to use a doubly terminated external output network similar to what is shown in Figure 43. In this case, the equivalent load seen by the ac current source of the DAC is 25 Ω.

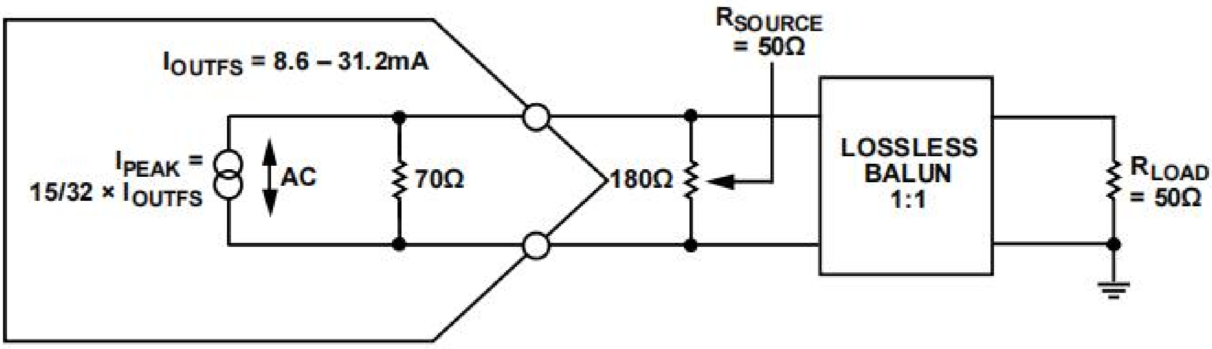


Figure 43. Equivalent Circuit for Determining Maximum Peak Power to a 50 Ω Load

If the CD97D39 is programmed for I_{OUTFS} = 20 mA, its peak ac current is 9.375 mA and its peak power delivered to the equivalent load is 2.2 mW (that is, P = I2 R). Because the source and load resistance seen by the 1:1 balun are equal, this power is shared equally; therefore, the output load receives 1.1 mW or 0.4 dBm. To calculate the rms power delivered to the load, the following must be considered:

- Peak-to-rms of the digital waveform
- Any digital backoff from digital full scale
- The DAC’s sinc response and nonideal losses in external network

For example, a reconstructed sine wave with no digital backoff ideally measures −2.6 dBm because it has a peak-to-rms ratio of 3 dB. If a typical balun loss of 0.4 dBm is included, −3 dBm of actual power can be expected in the region where the sinc response of the DAC has negligible influence. Increasing the output power is best accomplished by increasing IOUTFS, although any degradation in linearity performance must be considered acceptable for the target application.

Output Stage Configuration

The CD97D39 is intended to serve high dynamic range applications that require wide signal reconstruction bandwidth (that is, DOCSIS CMTS) and/or high IF/RF signal generation. Optimum ac performance can only be realized if the DAC output is configured for differential (that is, balanced) operation with its output common-mode voltage biased to analog ground. The output network used to interface to the DAC should provide a near 0Ω dc bias path to analog ground. Any imbalance in the output impedance between the IOUTP and IOUTN pins results in asymmetrical signal swings that degrade the distortion performance (mostly even order) and noise performance. Component selection and layout are critical in realizing the performance potential of the CD97D39.

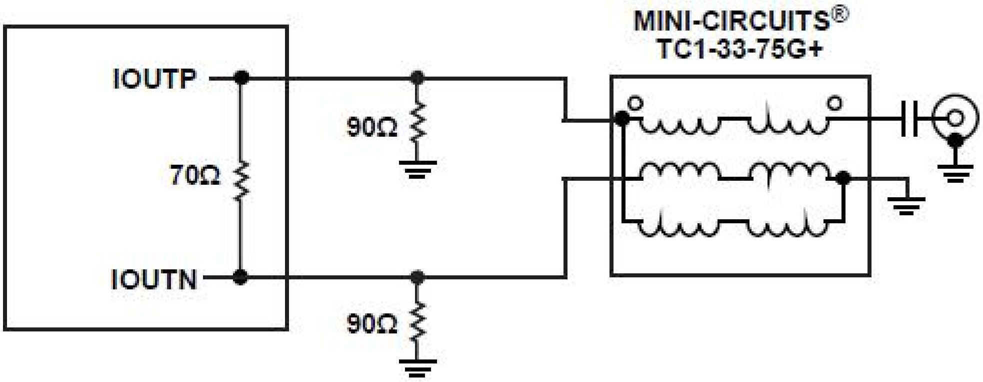


Figure 44. Recommended Balun for Wideband Applications with Upper Bandwidths of up to 2.2 GHz

Most applications requiring balanced-to-unbalanced conversion can take advantage of the Ruthroff 1:1 balun configuration shown in Figure 44. This configuration provides excellent amplitude/phase balance over a wide frequency range while providing a 0 Ω dc bias path to each DAC output. Also, its design provides exceptional bandwidth and can be considered for applications requiring signal reconstruction of up to 2.2 GHz. The characterization plots shown in this data sheet are based on the CD97D39 evaluation board, which uses this configuration.

Figure 45 shows an interface that can be considered when interfacing the DAC output to a self-biased differential gain block. The inductors shown serve as RF chokes (L) that provide the dc bias path to analog ground. The value of the inductor, along with the dc blocking capacitors (C), determines the lower cut off frequency of the composite pass-band response. An RF balun should also be considered before the RF differential gain stage and any filtering to ensure symmetrical common-mode impedance seen by the DAC output while suppressing any common-mode noise, harmonics, and clock spurs prior to amplification.

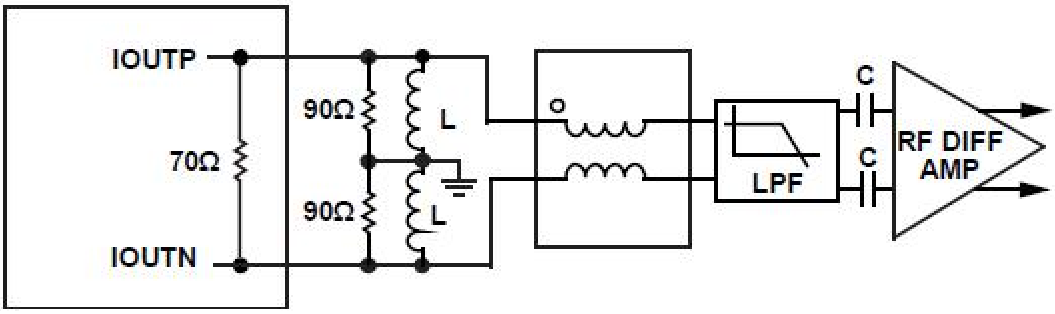


Figure 45. Interfacing the DAC Output to the Self-Biased Differential Gain Stage

For applications operating the CD97D39in mix mode with output frequencies extending beyond

2.2 GHz, the circuits shown in Figure 46 should be considered. The circuit in Figure 46 uses a wideband balun with a configuration similar to the one shown in Figure 45 to provide a dc bias path for the DAC outputs.

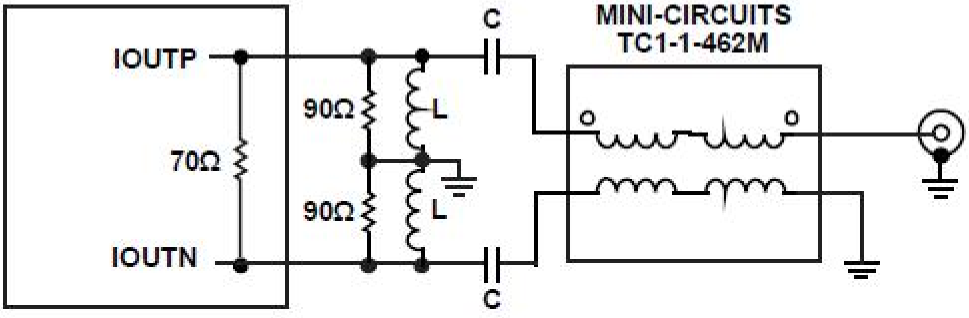


Figure 46. Recommended Mix Mode Configuration Offering Extended RF

Bandwidth Using a TC1-1-43A+ Balun

Recommended Start-Up Sequence

Upon power-up of the CD97D39, a host processor is required to initialize and configure the CD97D39 via its SPI port. Figure 47 shows a flowchart of the sequential steps required, while Table 28 and Table 29 provides more detail on the SPI register write/read operations required to implement the flowchart steps. Note the following:

- A software reset is optional because the CD97D39 has both an internal POR circuit and a RESET pin.
- The SYNC controller is optional because it is only required to synchronize two or more devices. If synchronization is required, validate that DCI_DEL values between devices are sufficiently matched.
- The mu controller must be first enabled (and in track mode) before the data receiver controller is enabled because the DCO output signal is derived from this circuitry.
- A wait period is related to f_{DATA} periods.
- Limit the number of attempts to lock the controllers to three; locks typically occur on the first attempt.
- Hardware or software interrupts can be used to monitor the status of the controllers.

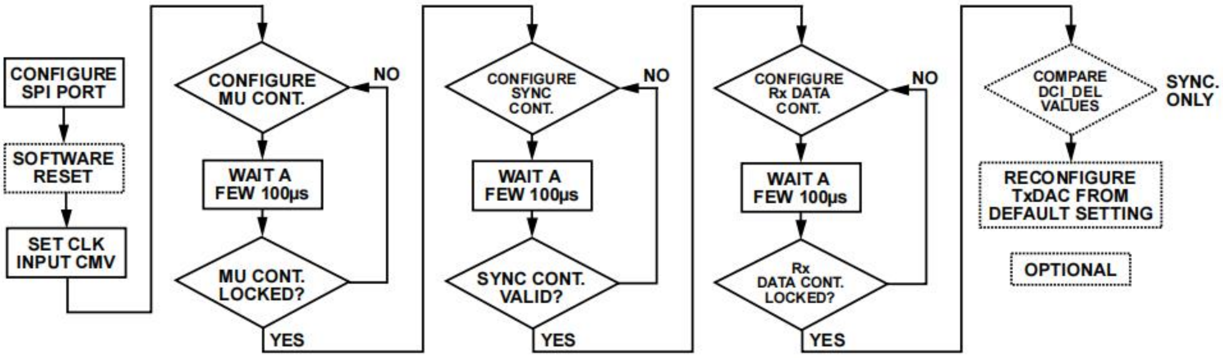


Figure 47. Flowchart for Initialization and Configuration of the CD97D39

Table 28. Recommended SPI Initialization with SYNC Controller Disabled

Step	Address (Hex)	Write Value	Comments
1	0x00	0x00	Configure for the 4-wire SPI mode with MSB. Note that Bits[7:5] must be mirrored onto Bits[2:0] because the MSB/LSB format can be unknown at power-up.
2	0x00	0x20	Software reset to default SPI values.
3	0x00	0x00	Clear the reset bit.
4	0x22	0x0F	Set the common-mode voltage of DACCLK_P and DACCLK_N inputs.
5	0x23	0x0F	
6	0x24	0x30	Configure the mu controller. Refer to Table 26 for recommended target mu slope and phase settings vs. clock rate.
7	0x25	0x80	
8	0x27	0x4C	
9	0x28	0x6C	
10	0x29	0xCB	
11	0x26	0x02	
12	0x26	0x03	Enable the mu controller search and track mode.

13	Not applicable	Not applicable	Wait for 160k × $1/f_{DATA}$ cycles.
14	0x2A	Not applicable	Read back Register 0x2A and confirm that it is equal to 0x01 to ensure that the DLL loop is locked. If it is not locked, proceed to Step 10 and repeat. Limit attempts to three before breaking out of the loop and reporting a mu lock failure.
15	Not applicable	Not applicable	Ensure that the CD97D39 is fed with DCI clock input from the data source.
16	0x13	0x72	Set FINE_DEL_SKEW to 2.
17	0x10	0x00	Disable the data Rx controller before enabling it.
18	0x10	0x02	Enable the data Rx controller for loop and IRQ.
19	0x10	0x03	Enable the data Rx controller for search and track mode.
20	Not applicable	Not applicable	Wait for 135 K × $1/f_{DATA}$ cycles.
21	0x21	--	Read back Register 0x21 and confirm that it is equal to 0x09 to ensur that the DLL loop is locked and tracking. If it is not locked and trackin advance the CLKDIVPH[1:0] phase in Register 0x14, Bit[7:6] before proceeding to Step 17 to repeat attempt. Limit attempts to three befo breaking out of the loop and reporting an Rx data lock failure.
22	0x06，0x07	0x00，0x02	Optional: modify the DAC IOUTFS setting (the default is 20 mA).
23	0x08	0x00	Optional: modify the DAC operation mode (the default is normal mode).

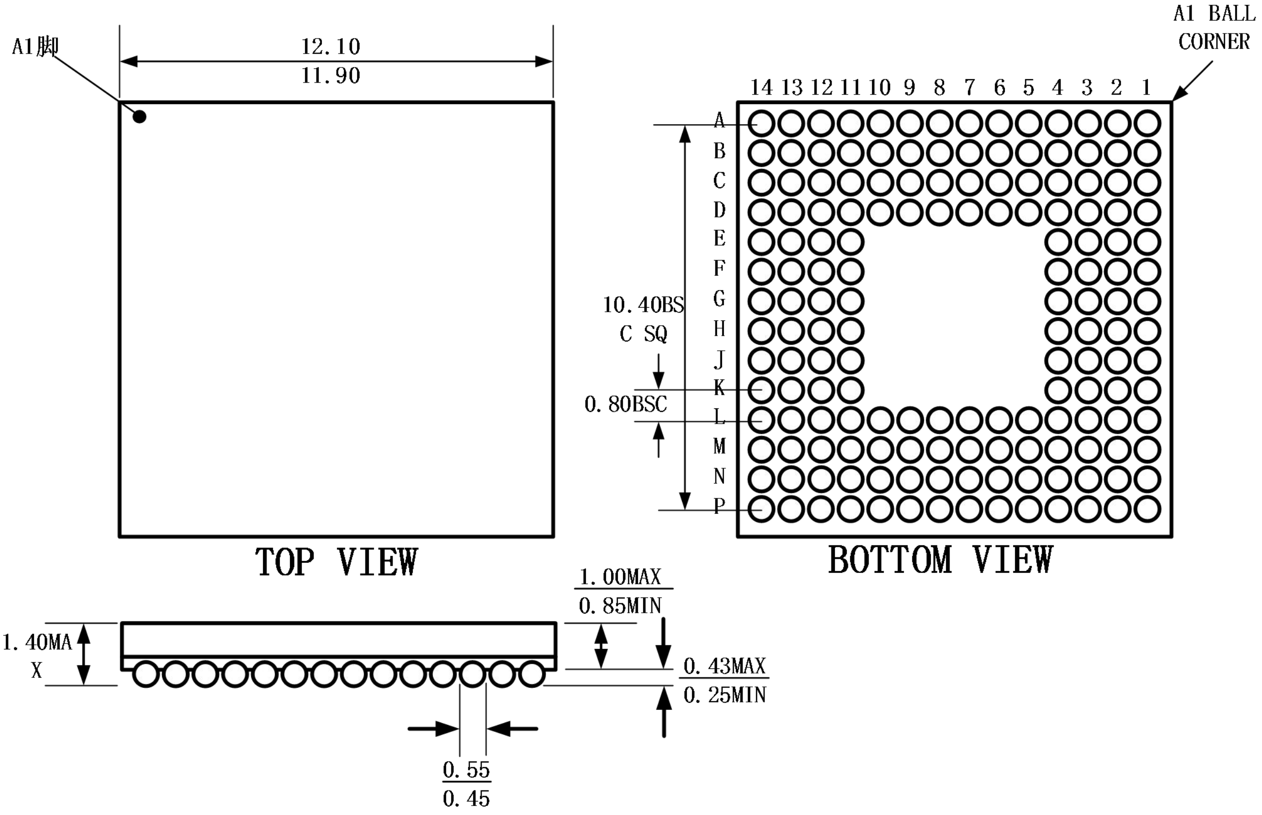
Table 29. Recommended SPI Initialization with SYNC Controller Enabled

Step	Address (Hex)	Write Value	Comments
1	0x00	0x00	Configure for the 4-wire SPI mode with MSB. Note that Bits[7:5] must be mirrored onto Bits[2:0] because the MSB/LSB format can be unknown at power-up.

2	0x00	0x20	Software reset to default SPI values.
3	0x00	0x00	Clear the reset bit.
4	0x22	0x0F	Set the common-mode voltage for the input differential clocks DACCLK_P and DACCLK_N.
5	0x23	0x0F	
6	0x24	0x30	Configure the mu controller. Refer to Table 28 for recommended target mu slope and phase settings vs. clock rate.
7	0x25	0x80	
8	0x27	0x4C	
9	0x28	0x6C	
10	0x29	0xCB	
11	0x26	0x02	
12	0x26	0x03	Enable the mu controller search and track mode.
13	Not applicable	Not applicable	Enable the mu controller search and track mode.
14	0x2A	Not applicable	Read back Register 0x2A and confirm that it is equal to 0x01 to ensure that the DLL loop is locked. If it is not locked, proceed to Step 10 and repeat. Limit attempts to three before breaking out of the loop and reporting a mu lock failure.
15	0x15	0x42	Configure sync controller.
16	0x10	0x00	Disable sync controller before enabling it.
17	0x10	0x60 or 0x40	Enable sync controller for loop and IRQ. 0x60 = master mode. 0x40 = slave mode.
18	0x10	0x70or 0x50	Enable sync controller: 0x70 = master mode. 0x50 = slave mode.
19	Not applicable	Not applicable	Wait for 160k × 1/f_{DATA} for DLL to lock.
20	0x21	--	Read back Register 0x21 to confirm proper operation: 0x90 = master mode. 0x00 = slave mode.

			If not, proceed to Step 15 and repeat. Limit to three attempts before breaking out of loop and reporting sync lock failure.
21	0x0D	--	Read back Register 0x0D and confirm Bits[5:4] = 10. If not, proceed to Step 2 and repeat. Limit to three attempts before breaking out of loop and reporting sync lock failure.
22	Not applicable	Not applicable	Ensure that the CD97D39 is fed with DCI clock input from the data source.
23	0x13	0x72	Set FINE_DEL_SKEW to 2.
24	0x10	0x00	Disable the data Rx controller before enabling it. 0x70 = master mode. 0x50 = slave mode.
25	0x10	0x02	Enable the data Rx controller for loop and IRQ. 0x72 = master mode. 0x52 = slave mode.
26	0x10	0x03	Enable the data Rx controller for search and track mode. 0x73 = master mode. 0x53 = slave mode.
27	--	--	Wait for 135k × 1/f_{DATA} cycles.
28	0x21	--	Read back Register 0x21 and confirm that it is equal to 0x09 to ensure that the DLL loop is locked and tracking. If it is not locked and tracking, proceed to Step 16 and repeat. Limit attempts to three before breaking out of the loop and reporting an Rx data lock failure.
29	Not applicable	Not applicable	Readback DCI_DEL value in Register 0x13 and Register 0x14 for master and slave. If slave devices are not within 40 codes of each other, re-specify target DCI_DEL value to be average between master and readback DCI_DEL value.
30	0x06，0x07	0x00，0x02	Optional: modify the DAC IOUTFS setting (the default is 20 mA).
31	0x08	0x00	Optional: modify the DAC operation mode (the default is normal mode).

Outline Dimensions



Dimensions shown in millimeters

Figure 48. 160-Ball Chip Scale Package Ball Grid Array

Package/Ordering Information

MODEL	TEMPERATURE	PACKAGE DESCRIPTION	PACKAGE OPTION
CD97D39AG	-40℃ ~ 85℃	PBGA-160	Tray, 189
CD97D39BG	-40℃ ~ 85℃	PBGA-160	Tray, 189

Revision Log

Version	Revision date	Change content	Reason for Change	Modified by	Reviewed By	Note
V1.0	2025.7.1	Initial version	Regular update	WW	LYL	