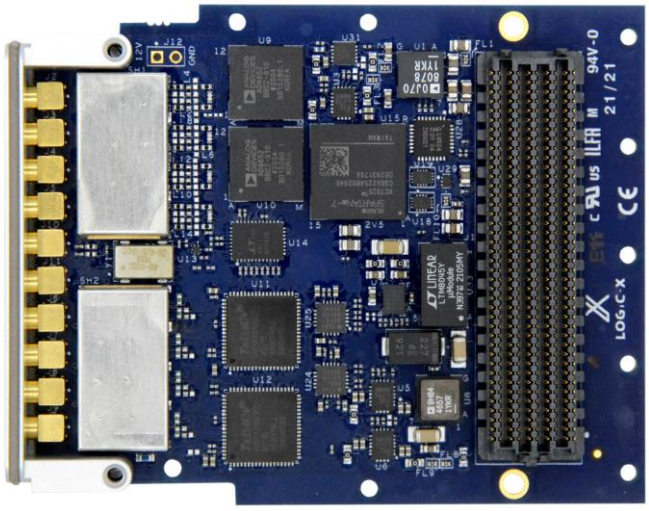
LXD31K4 quad channel MIMO FMC module User manual



Document title	: LXD31K4 quad channel MIMO FMC module User manual
Document type	: DM
Document number	: 005
Document revision	: 1.0
Document classification	: Public

LOGIC-X
THE POWER OF LOGIC

Page is intentionally left blank

Table of Contents

1 General Description

Using off-the-shelf FMC and FPGA products enables system designers to get their products to market faster. The LXD31K4 with its 4 analog input channels and 4 analog output channels will easily fit into, among others, the following systems: MIMO, Wireless communication transceivers, software defined radio, medical equipment and test and instrumentation.

For its analog input the LXD31K4 uses the AD9652 from analog devices that offers 16bits resolution at a sample rate of 310Msps.At the analog output the AD9142A from analog devices can convert 16bit data at a maximum sample rate of 1600 Msps. At the front end the LXD31K4 offers the choice for AC or DC coupled inputs. All the digital data interfaces are LVDS and requires a high pin count FMC connector.
The LXD31K4 allows flexible control on clock source, analog input gain, and offset correction through serial communication busses. Furthermore, the card is equipped with power supply and temperature monitoring and offers several power-down modes to switch off unused functions, reducing system level power.

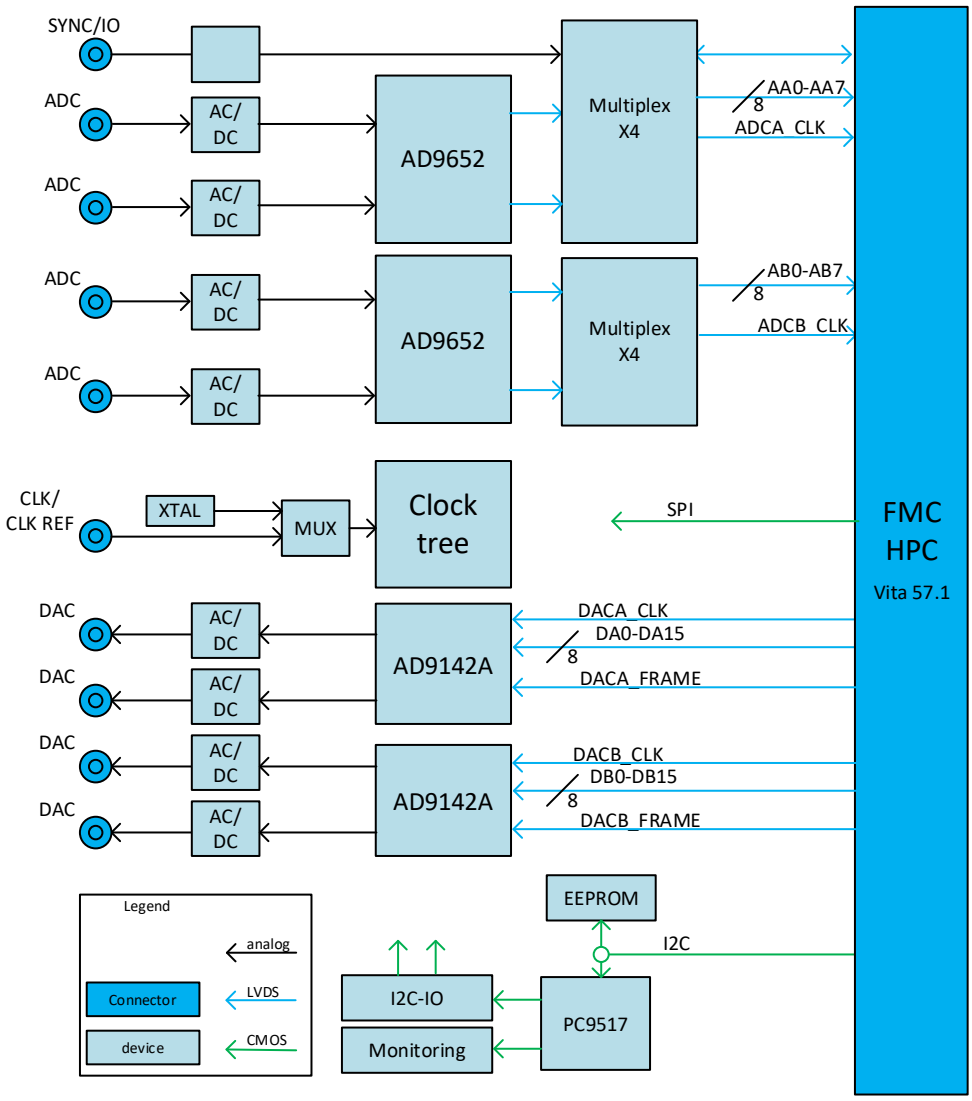


Figure 1: LXD31K4 block diagram

2 Mechanical design

The board design is mechanically compliant to the Vita 57.1 (FMC) specification.

2.1.1 Cooling

Optionally the board is delivered with a heatsink that is violating the maximum component height of 4.7mm. Most FMC carrier products on the market today won't have a conflict. Please contact Logic-X for custom heatsink and conduction cooling solutions.
Without heat sink a forced airflow is required to cool the product

2.2 Front panel design

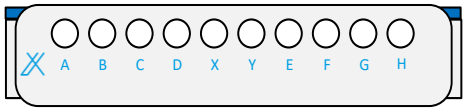


Figure 2: LXD31K4 front panel

Frontpanel marking	Connects to	Frontpanel marking	Connects to
A	DAC channel 0	E	ADC channel 3
B	DAC channel 2	F	ADC channel 1
C	DAC channel 1	G	ADC channel 2
D	DAC channel 3	H	ADC channel 0
X	External trigger	Y	Reference clock

3 FMC

The LXD31K4 is an FMC card that targets a high pin count connector. The ADC data is transferred over two 8 bits wide data busses. Each bus carries 2 multiplexed ADC channels. The DAC data is multiplex onto two 16 bit wide data busses.
The card is a high pin count FMC that requires the LA and HA bus. That means it is compatible with FMC carrier boards that do not implement the HB bus.

3.1 LA, HA and HB connections

The following table shows the signal connections on the FMC connector.
A signal ending with _p or _n indicates the negative and positive signals part of a differential pair. All differential pairs are LVDS signals. The remaining single ended signals are LVCMOS with a voltage rating equal to the power supplied on VADJ.

AV57	Con. Pin	LXD30K00	AV57	Con. Pin	LXD30K00
CLK0_M2C_N	H5	DACA_DCI_N	DP8_M2C_P	B8	n.c.
CLK0_M2C_P	H4	DACA_DCI_P	DP9_C2M_N	B25	n.c.
CLK1_M2C_N	G3	FMC_SPI_SIO (on AC coupled) DACA_FRAME_P (on DC coupled)	DP9_C2M_P	B24	n.c.
CLK1_M2C_P	G2	FMC_SPI_SCLK (on AC coupled) DACA_FRAME_N (on DC coupled)	DP9_M2C_N	B5	n.c.
CLK2_BIDIR_N	K5	BD_8_N	DP9_M2C_P	B4	n.c.
CLK2_BIDIR_P	K4	BD_8_P	GBTCLK0_M2C_N	D5	n.c.
CLK3_BIDIR_N	J3	AD_8_N	GBTCLK0_M2C_P	D4	n.c.
CLK3_BIDIR_P	J2	AD_8_P	GBTCLK1_M2C_N	B21	n.c.
DP0_C2M_N	C3	n.c.	GBTCLK1_M2C_P	B20	n.c.
DP0_C2M_P	C2	n.c.	HA00_N_CC	F5	BD_2_N
DP0_M2C_N	C7	n.c.	HA00_P_CC	F4	BD_2_P
DP0_M2C_P	C6	n.c.	HA01_N_CC	E3	BD_7_N
DP1_C2M_N	A23	n.c.	HA01_P_CC	E2	BD_7_P
DP1_C2M_P	A22	n.c.	HA02_N	K8	DACB_D11_N
DP1_M2C_N	A3	n.c.	HA02_P	K7	DACB_D11_P
DP1_M2C_P	A2	n.c.	HA03_N	J7	DACB_D13_N
DP2_C2M_N	A27	n.c.	HA03_P	J6	DACB_D13_P
DP2_C2M_P	A26	n.c.	HA04_N	F8	DACB_D12_N
DP2_M2C_N	A7	n.c.	HA04_P	F7	DACB_D12_P
DP2_M2C_P	A6	n.c.	HA05_N	E7	DACB_D14_N
DP3_C2M_N	A31	n.c.	HA05_P	E6	DACB_D14_P
DP3_C2M_P	A30	n.c.	HA06_N	K11	DACB_D08_N
DP3_M2C_N	A11	n.c.	HA06_P	K10	DACB_D08_P
DP3_M2C_P	A10	n.c.	HA07_N	J10	DACB_D09_N
DP4_C2M_N	A35	n.c.	HA07_P	J9	DACB_D09_P
DP4_C2M_P	A34	n.c.	HA08_N	F11	DACB_D10_N
DP4_M2C_N	A15	n.c.	HA08_P	F10	DACB_D10_P
DP4_M2C_P	A14	n.c.	HA09_N	E10	DACB_D15_N
DP5_C2M_N	A39	n.c.	HA09_P	E9	DACB_D15_P
DP5_C2M_P	A38	n.c.	HA10_N	K14	DACB_D05_N
DP5_M2C_N	A19	n.c.	HA10_P	K13	DACB_D05_P
DP5_M2C_P	A18	n.c.	HA11_N	J13	DACB_D06_N
DP6_C2M_N	B37	n.c.	HA11_P	J12	DACB_D06_P
DP6_C2M_P	B36	n.c.	HA12_N	F14	DACB_D04_N
DP6_M2C_N	B17	n.c.	HA12_P	F13	DACB_D04_P
DP6_M2C_P	B16	n.c.	HA13_N	E13	DACB_D07_N
DP7_C2M_N	B33	n.c.	HA13_P	E12	DACB_D07_P
DP7_C2M_P	B32	n.c.	HA14_N	J16	DACB_D01_N
DP7_M2C_N	B13	n.c.	HA14_P	J15	DACB_D01_P
DP7_M2C_P	B12	n.c.	HA15_N	F17	DACB_D02_N

DP8_C2M_N	B29	n.c.	HA15_P	F16	DACB_D02_P
DP8_C2M_P	B28	n.c.	HA16_N	E16	DACB_D00_N
DP8_M2C_N	B9	n.c.	HA16_P	E15	DACB_D00_P
HA17_N_CC	K17	DACB_D03_N	HB15_N	J34	n.c.
HA17_P_CC	K16	DACB_D03_P	HB15_P	J33	n.c.
HA18_N	J19	DACB_FRAME_N	HB16_N	F35	n.c.
HA18_P	J18	DACB_FRAME_P	HB16_P	F34	n.c.
HA19_N	F20	DACB_DCI_N	HB17_N_CC	K38	n.c.
HA19_P	F19	DACB_DCI_P	HB17_P_CC	K37	n.c.
HA20_N	E19	BD_5_N	HB18_N	J37	n.c.
HA20_P	E18	BD_5_P	HB18_P	J36	n.c.
HA21_N	K20	BD_1_N	HB19_N	E34	n.c.
HA21_P	K19	BD_1_P	HB19_P	E33	n.c.
HA22_N	J22	BD_4_N	HB20_N	F38	n.c.
HA22_P	J21	BD_4_P	HB20_P	F37	n.c.
HA23_N	K23	BD_3_N	HB21_N	E37	n.c.
HA23_P	K22	BD_3_P	HB21_P	E36	n.c.
HB00_N_CC	K26	n.c.	LA00_N_CC	G7	AB_CLK_N
HB00_P_CC	K25	n.c.	LA00_P_CC	G6	AB_CLK_P
HB01_N	J25	SYNC_C2M_N	LA01_N_CC	D9	BD_0_N
HB01_P	J24	SYNC_C2M_P	LA01_P_CC	D8	BD_0_P
HB02_N	F23	n.c.	LA02_N	H8	FMC_ADCA_SPI_CSN
HB02_P	F22	n.c.	LA02_P	H7	FMC_FPGA0_SPI_CSN
HB03_N	E22	n.c.	LA03_N	G10	BD_6_N
HB03_P	E21	n.c.	LA03_P	G9	BD_6_P
HB04_N	F26	n.c.	LA04_N	H11	FMC_DACA_SPI_CSN
HB04_P	F25	n.c.	LA04_P	H10	FMC_ADCB_SPI_CSN
HB05_N	E25	n.c.	LA05_N	D12	AD_3_N
HB05_P	E24	n.c.	LA05_P	D11	AD_3_P
HB06_N_CC	K29	n.c.	LA06_N	C11	AD_5_N
HB06_P_CC	K28	n.c.	LA06_P	C10	AD_5_P
HB07_N	J28	n.c.	LA07_N	H14	FMC_PLL_SPI_CSN
HB07_P	J27	n.c.	LA07_P	H13	FMC_DACB_SPI_CSN
HB08_N	F29	n.c.	LA08_N	G13	AD_6_N
HB08_P	F28	n.c.	LA08_P	G12	AD_6_P
HB09_N	E28	n.c.	LA09_N	D15	FMC_SPI_SIO2
HB09_P	E27	n.c.	LA09_P	D14	FMC_SPI_WNR
HB10_N	K32	n.c.	LA10_N	C15	AD_0_N
HB10_P	K31	n.c.	LA10_P	C14	AD_0_P
HB11_N	J31	n.c.	LA11_N	H17	DACA_D13_N
HB11_P	J30	n.c.	LA11_P	H16	DACA_D13_P
HB12_N	F32	n.c.	LA12_N	G16	AD_4_N
HB12_P	F31	n.c.	LA12_P	G15	AD_4_P
HB13_N	E31	n.c.	LA13_N	D18	FPGA_CLK_200M_N
HB13_P	E30	n.c.	LA13_P	D17	FPGA_CLK_200M_P

HB14_N	K35	n.c.	LA14_N	C19	AD_7_N
HB14_P	K34	n.c.	LA14_P	C18	AD_7_P
			LA15_N	H20	DACA_FRAME_N (on AC coupled) SPI_ADC_DAC_SDIO (on DC coupled)
			LA15_P	H19	DACA_FRAME_P (on AC coupled) SPI_SCLK (on DC coupled)
LA16_N	G19	AD_2_N	LA26_N	D27	DACA_D06_N
LA16_P	G18	AD_2_P	LA26_P	D26	DACA_D06_P
LA17_N_CC	D21	DACA_D15_P	LA27_N	C27	DACA_D08_N
LA17_P_CC	D20	DACA_D15_N	LA27_P	C26	DACA_D08_P
LA18_N_CC	C23	DACA_D12_N	LA28_N	H32	DACA_D00_P
LA18_P_CC	C22	DACA_D12_P	LA28_P	H31	DACA_D00_N
LA19_N	H23	DACA_D09_N	LA29_N	G31	DACA_D03_N
LA19_P	H22	DACA_D09_P	LA29_P	G30	DACA_D03_P
LA20_N	G22	DACA_D10_N	LA30_N	H35	DACA_D02_N
LA20_P	G21	DACA_D10_P	LA30_P	H34	DACA_D02_P
LA21_N	H26	DACA_D07_N	LA31_N	G34	DACA_D04_N
LA21_P	H25	DACA_D07_P	LA31_P	G33	DACA_D04_P
LA22_N	G25	DACA_D14_P	LA32_N	H38	SYNC_M2C_N
LA22_P	G24	DACA_D14_N	LA32_P	H37	SYNC_M2C_P
LA23_N	D24	DACA_D11_N	LA33_N	G37	AD_1_N
LA23_P	D23	DACA_D11_P	LA33_P	G36	AD_1_P
LA24_N	H29	DACA_D01_P	SCL	C30	SCL
LA24_P	H28	DACA_D01_N	SDA	C31	SDA
LA25_N	G28	DACA_D05_N			
LA25_P	G27	DACA_D05_P			

3.2 Utility connections

The following table shows how the utility connections signals are used on the LXD31K4.

Signal name	Connected to
PRSNT_M2C	Tied to GND
PG_C2M	Not used
TCK	Not used
TDI	See chapter on JTAG
TDO	See chapter on JTAG
SCL	See chapter on I2C
SDA	See chapter on I2C
TMS	Not used
TRST_L	Not used
GA0	See chapter on I2C
GA1	See chapter on I2C
RES0	Not used
PG_M2C	Asserted directly when the local power is OK

3.3 Gigabit transceiver connections

The LXD31K4 does not connect any of the Gigabit transceiver connections or reference clocks.

3.4 I/O Standard Support

Most of the signal connections to the FMC connector on the LXD31K4 are differential LVDS type of signals. The CMOS control signals are passed through level translators that are powered by VADJ on one the FMC side. This ensures proper operation with VADJ voltages between 1.2 and 3.3V. The VREF signal is not used and not connected.

3.5 VIO_B_M2C Support

The LXD31K4 connects VIO_B_M2C directly to VADJ.

3.6 VREF Support

The LXD31K4 leaves VREF_A_M2C and VREF_B_M2C unconnected.

3.7 Main Characteristics

Analog inputs	
Number of channels	4 (A0-A3)
Channel resolution	16-bit
Input voltage range	DC Coupled: +6dBm (max) / 1.25Vp-p DC (50 ohms) AC Coupled: +12dBm (max)/ 1.25Vp-p (50 Ohms)
Input impedance	50Ω
Analog input bandwidth	DC Coupled: DC -200 MHz AC Coupled: 10 MHz to 465MHz
Performance (Fin = 170 MHz)	SFDR = 85 dBc , SNR 73.7 dBFS Typical value provided by Analog devices
Input data rate	80 – 310 Msps
Analog outputs	
Number of channels	4 (D0- D3)
Output voltage range	DC Coupled: +12 dBm (max) / 2.5Vp-p DC into 50 ohms AC Coupled: +0 dBm (max.) / 0.632Vp-p into 50 ohms
Load	50Ω
Performance (Fout = 200 MHz)	SFDR=85 dBc Typical value provided by Analog devices
Analog Bandwidth	DC Coupled 310 Msps, x4 interpolation: DC - 145 MHz AC Coupled: 10 MHz to 145 MHz
Digital Sample Rate	1.6GS/sec maximum data rate for each DAC with interpolation. Each DAC maximum data rate is 575Msps.
Output data Rate	80 – 575 Msps
External Clock/Reference input	
External Reference Input level	External Reference Clock 0.30 to 2.5Vp-p (-7 to +12dBm) recommended range: +10 to +12dBm.
Input impedance	50Ω AC coupled
Input range	Ext Ref 10-750 MHz
External Trigger/Sync input	
Format	LVTTL/LVCMOS Compatible 1.25Vdc Threshold
Input impedance	10kΩ DC coupled
Frequency range	LVTTL & LVCMOS Limited by source
ADC Output	
Output data width	2 * 8bits LVDS @ 320 - 1240MHz DDR
Data Format	Two's Complement / Offset binary
Sampling Frequency Range	80 – 310 Msps
DAC input	
Output data width	2 * 16bits LVDS @ 160– 575 MHz DDR
Data Format	Two's Complement / Offset binary
Sampling Frequency Range	80 – 575 Msps

Table 1 : LXD31K4 Daughter Card Main Characteristics

4 Analog input and output

4.1 AC coupled

At the analog input and output the LXD31K4 uses a balun coupling for single ended to differential conversion. To make maximum use of the ADC capabilities the balun (ETC1-1-13) from MACOM is used. This balun offers a wide bandwidth from 4.5MHz up to 3000 MHz. Typically, the phase matching is within 2 degrees upto 400 MHz bandwidth. In addition, it has a very flat insertion loss. The nominal insertion loss is 0.7dB.
The DAC output uses the TC2-72T+ balun which offers a bandwidth from .4MHz to 700 MHz
The Analog input and output are AC coupled at the connector to limit DC biasing of the transformer.

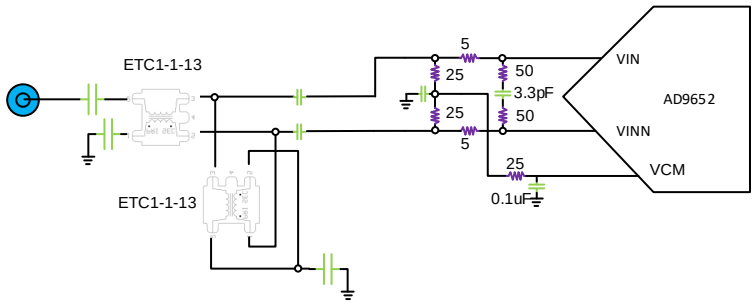


Figure 3: AC- coupled Analog input schematic

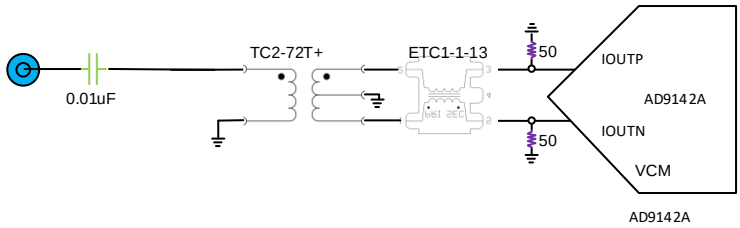


Figure 4: AC-coupled Analog output schematic

4.2 DC coupled

There is also a version of the LXD31K4 that has DC coupled inputs and outputs. The high performance Opamp ADA4927 from analog devices is used to translate between single ended and differential. The part has an excellent harmonic distortion which will not negatively impact the ADC and DAC performances. For each channel there is the option to tune the input and output offset to remove any unwanted offsets.
The DAC output gain is set to 5 to allow for a 12 dBm output that matches the ADC input.
Please note that the ADC0 and ADC2 inputs as well as the DAC0 and DAC2 outputs are inverting by design.

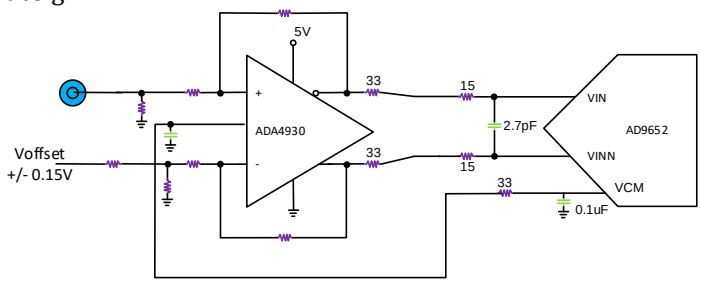


Figure 5: Differential analog input schematic

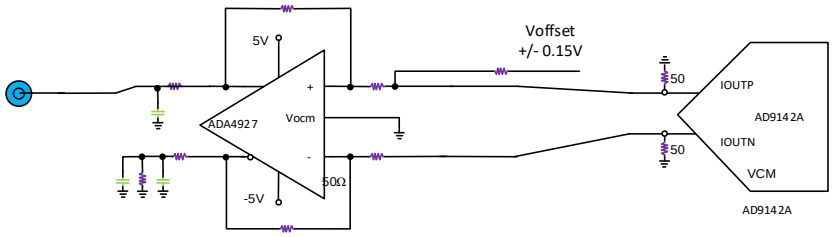


Figure 6: Differential analog output schematic

4.3 Input frequency response

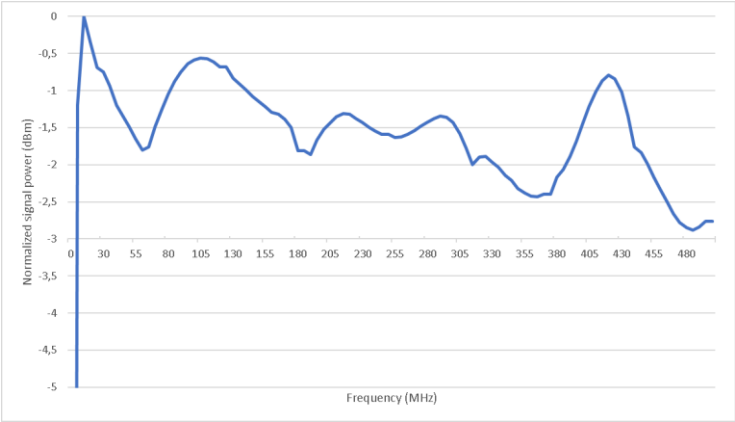


Figure 7:ADC input bandwidth AC coupled

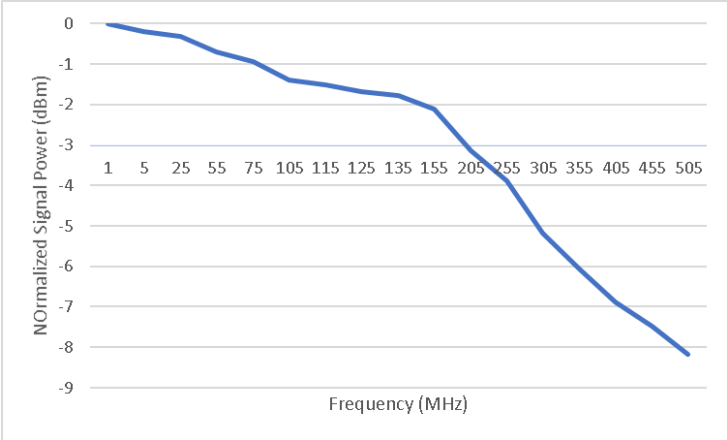


Figure 8:ADC input bandwidth DC coupled

4.4 Output frequency response

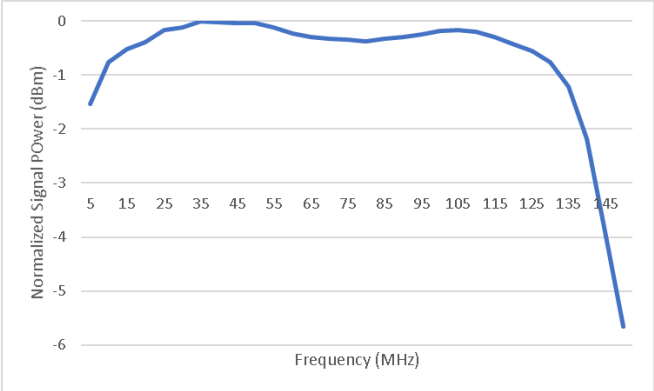


Figure 9: output power AC coupled

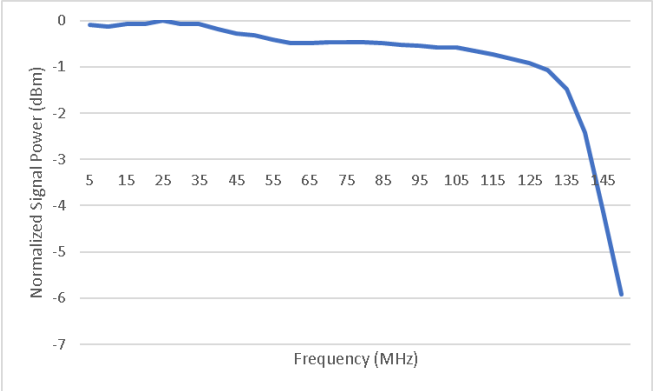


Figure 10: output power DC coupled

5 I2C Interface

The FMC identification EEPROM is connected directly to the FMC I2C signals and is active with only the 3V3_AUX.
The other devices on the chain are the voltage monitoring devices and an I2C IO expander (PCAL6524HEAZ). The IO expander has 24 bits that can individually be set as input or output. Also, each bit can be configured to latch a state change and generate an interrupt if required. The IO expander signals are used in order to drive the different static control signals and to read back status signals.
The image below depicts the I2C schematic and the device addresses.

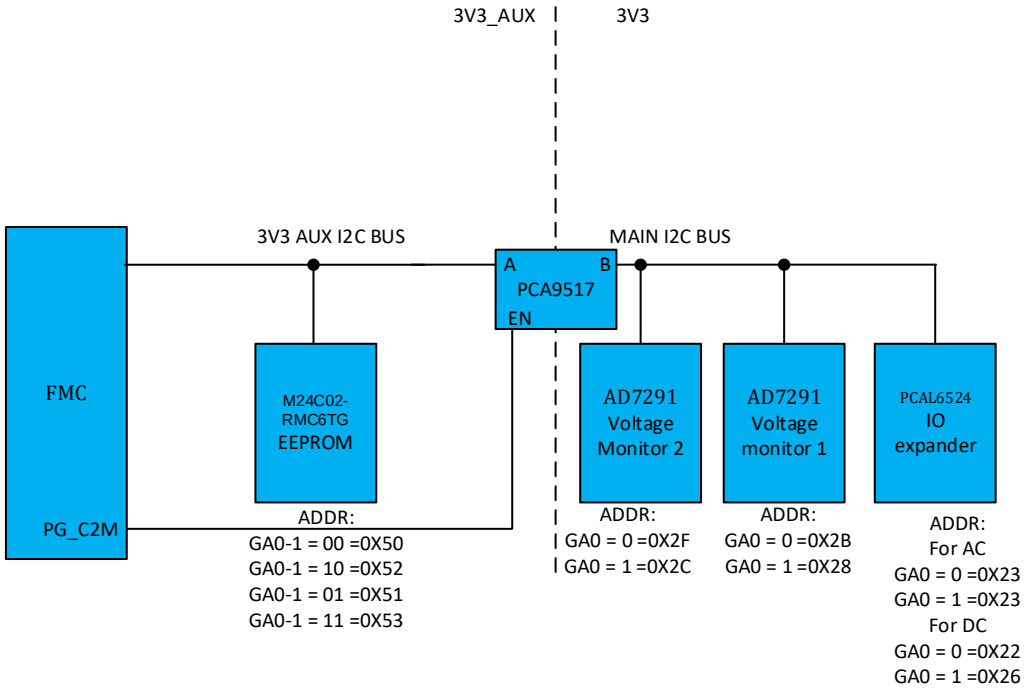


Figure 11:I2C schematic

IO pin	Direction	Signal name	Connected to	Function
P0_0	O	CLK_REF_MUX	Reference clock multiplexer	Select external reference (0) or local (1) reference clock for the PLL.
P0_1	O	RSVD	RSVD	reserved
P0_2	-	RSVD	RSVD	reserved
P0_3	-	RSVD	RSVD	reserved
P0_4	-	RSVD	RSVD	reserved
P0_5	-	RSVD	RSVD	reserved
P0_6	-	RSVD	RSVD	reserved
P0_7	-	RSVD	RSVD	reserved
P1_0	O	3V3_REF_EN	Connects to the Reference clock power supply.	Disable (0) the reference clock when not used to avoid interference from this clock.
P1_1	O	VCC_5VA_EN	5VA regulator and minus 5V for the DC coupled board.	Setting this pin to 1 enables the 5VA regulators
P1_2	O	VCC_ANALOG_EN	Connects to the 3V3A and 1V8A regulators	Setting this pin to 1 enables the 1V8A and 3V3A power supplies.
P1_3	O	VCC_FPGFA_EN	Connects to the 1V0, 2V5 and 1V8D regulators	Setting this pin to 1 powers up the entire FPGA voltage rails
P1_4	O	VCC_SWITCH_EN	Connects to the first stage switching regulators.	Setting this bit to 1 enables the switching regulators that enable the intermediate 3V5, 5V2 and 2V0 voltages.
P1_5	-	RSVD	RSVD	reserved
P1_6	-	RSVD	RSVD	reserved
P1_7	-	RSVD	RSVD	reserved
P2_0	I	PG_REF	Reference Clock power good	REF clock power is good
P2_1	I	PG_5VA	5VA power good	5VA power is good
P2_2	I	PG_3V3A	3V3A power good	3V3A power is good
P2_3	I	PG_1V8A	1V8A power good	1V8A power is good
P2_4	I	PG_FPGA	1V0, 2V5 and 1V8D power good	FPGA power is good
P2_5	I	PG_SWITCH	The first stage switching power supplies	If asserted high if both switching regulators are OK
P2_6	-	RSVD	RSVD	reserved
P2_7	-	RSVD	RSVD	reserved

6 SPI Interface

The PLL, ADC, DAC and FPGA devices connect to the same SPI bus. They can be individually selected by asserting their respective CSN signal. The SCLK signal is buffered and level translated in a way that each device receives its own clock. The SIO signal is also buffered and level translated. A dedicated direction control signal should be provided by the FMC connector to set the direction of the buffer correctly.

On the DC coupled board there is also an LTC2666 octal DAC device that is used to change the DC-offset of the analog inputs and outputs. This device is controlled through the local FPGA. Please refer to chapter 9.2 for details on communicating with the LTC2666 device over SPI.

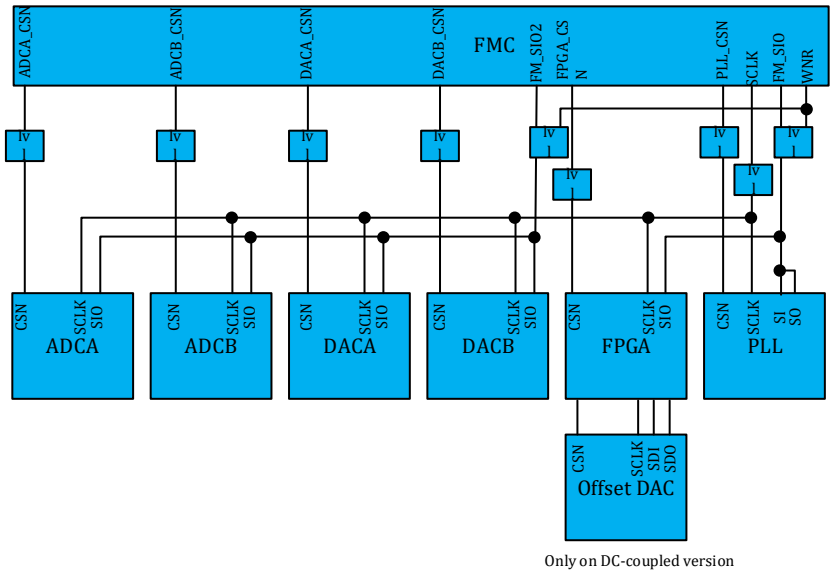


Figure 12: SPI architecture

The maximum SPI configuration clock frequency is 10 MHz.

7 External IO and synchronisation

On the front panel there is one MMCX that can be used for external trigger/synchronization or as general-purpose IO. The signal is buffered and connected to the local spartan 7 device. Also connected to the spartan7 are the SYNC ports of the two DAC devices and the SYNC port of the PLL. The spartan 7 includes a cross bar that allows the user to select any of 4 options for each sync output. The options are:

- EXT_IO
- SYNC_C2M
- Logic 0
- Programmable pulse based on SPI command.

The EXT_IO signal has the following 4 options

- INPUT
- SYNC_C2M
- Logic 0
- Programmable pulse based on SPI command.

In case the EXT_IO is set to INPUT the spartan 7 will float the EXT_IO signal. In the other cases it will drive a logic 0 or the value of the SYNC_C2M or a programmable pulse to the EXT_IO pin. All selections are done through the spartan 7 register map that are accessible through SPI.

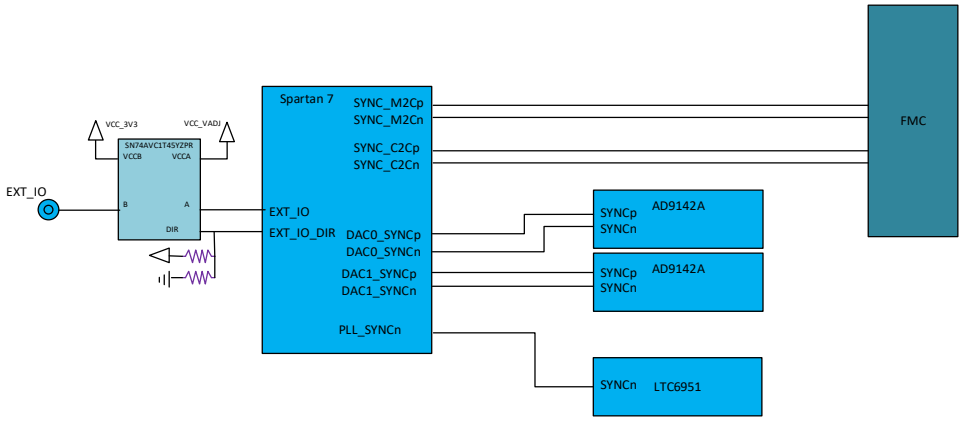


Figure 13: External IO and synchronisation schematic

8 Clock tree

Providing a very clean and stable clock is important for ADC and DAC devices. Therefore, special care is taken on the LXD31K4 to generate a very clean low jitter and low phase noise clock that is distributed to the ADC and DAC. The next image shows the LXD31K4 clock tree architecture.

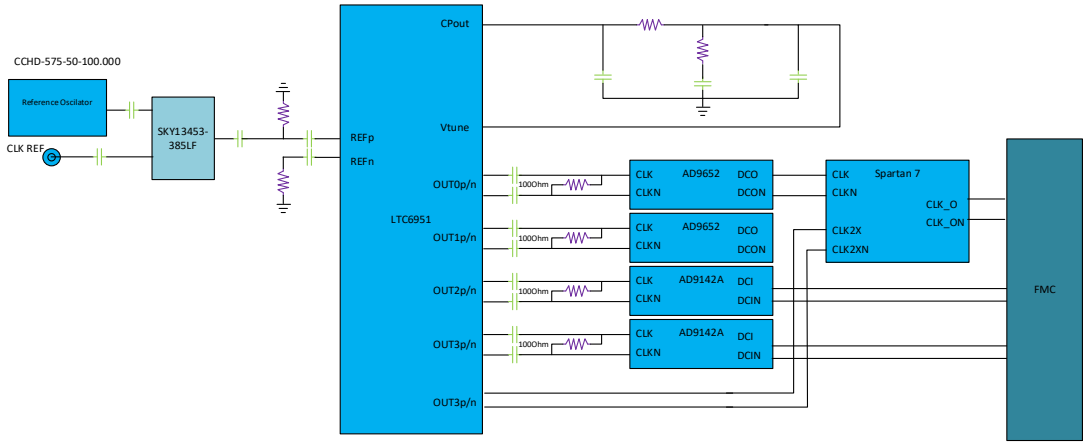


Figure 14: LXD31K4 clock tree architecture

8.1 Local clock generator

For the local clock a the LTC6951 device is used. This is a low-noise, wideband PLL with integrated VCO that supports a frequency range from 1.95 MHz to 2.5 GHz. The device supports integer-N modes. The device supports easy synchronization of multiple LXD31K4 boards to the same reference clock.

8.1.1 Local reference clock

For maximum stability, a low phase noise reference oscillator from Crystek is used (CCHD-575-50-100.000). This oscillator has a phase noise performance that outperforms the PLL.

8.1.2 External reference clock

A wide band RF switch (SKY13453-385LF) with low insertion loss is used to select between the local reference oscillator and a user provided reference clock. The reference clock can be provided on the MMCX input connector.

8.1.3 PLL loop filter design

With the design tool from linear technologies the loopfilter components are determined to work with a 100 MHz reference and 300 MHz ADC clock and 600 MHz DAC clock. The next image shows the loop filter design and the required part values.

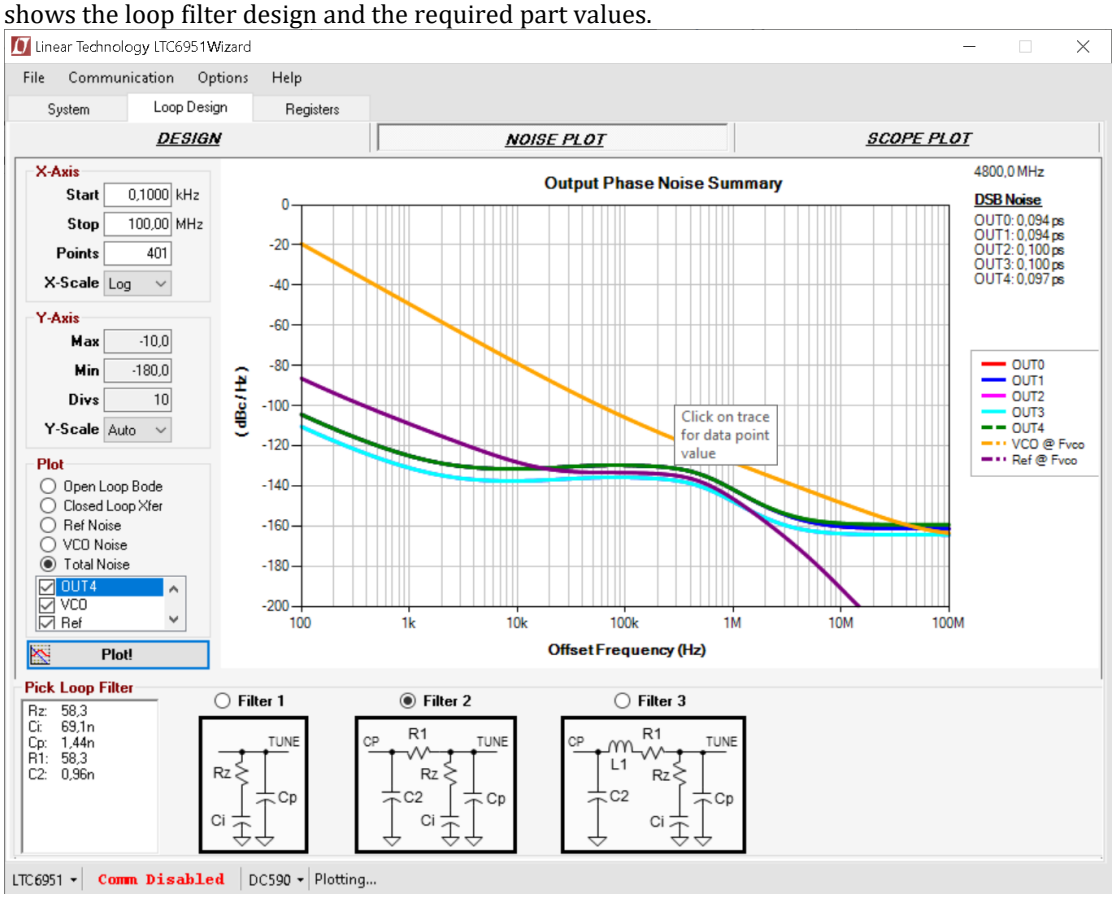


Figure 15: PLL loop filter design

8.2 External clock

The board only supports an external reference clock. Multi board synchronization is achieved by providing a common reference clock and an external trigger to SYNC the PLL and DAC dividers.

9 Multiplexing, Sync and external IO

In order to reduce the required number of signals towards the FMC connector the ADC data channels will be multiplexed using a Spartan 7 FPGA. Besides the multiplexing there is also the trigger and sync logic implemented in the FPGA. The following image shows the block diagram of the functionality implemented in the FPGA.

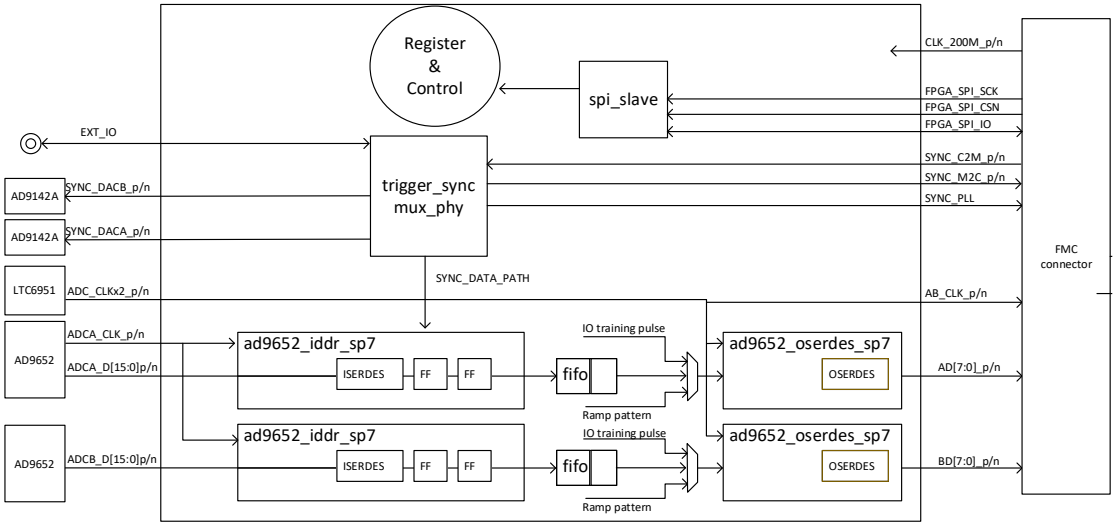


Figure 16: ADC data multiplexer top level diagram

9.1 Data format multiplexed data stream

The data stream between the LXD31K4 and the FMC connector is depicted in the following figure.

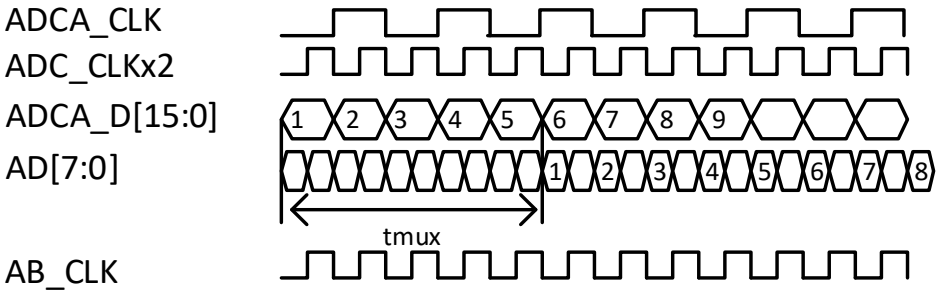


Figure 17: ADC data to FMC data multiplexing.

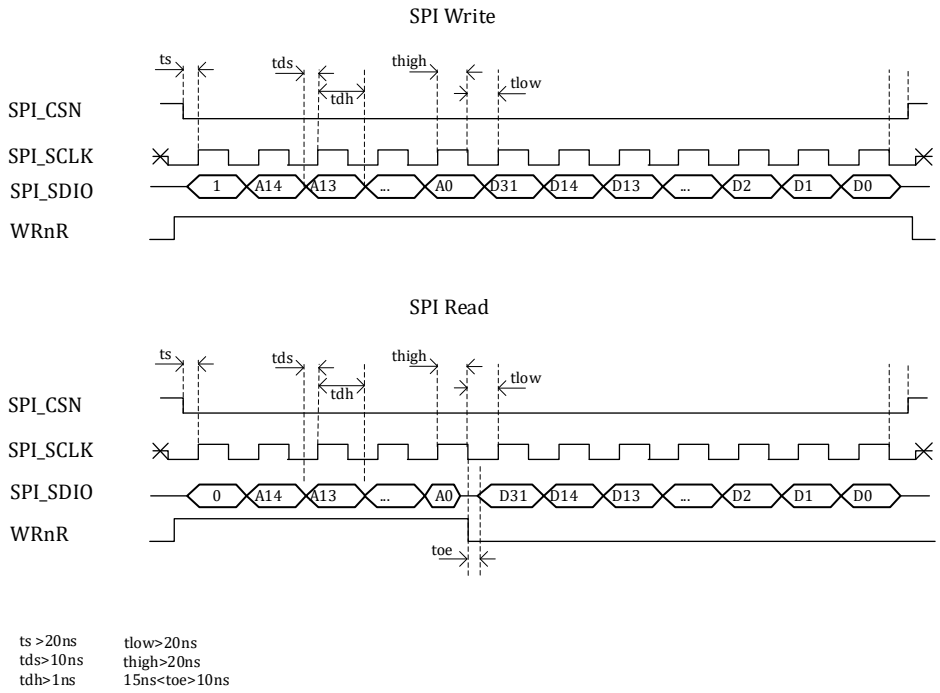
The time tmux has a constant value of TBD after a SYNC_DATA_PATH is provided.
The data rate on the data bus going to the FMC connector is twice as fast as the data rate on the data bus from the ADC device. That means the data rate is 4 times the ADC sample rate. With a 310 Msps the FMC data rate will be 1240 Msps.
In order to have a reliable data transfer it is advised to implement a training logic inside the carrier board FPGA. To facilitate the training, it is possible to select a training pulse for the FMC data bus. This pattern is a repeating pulse with the value of 0XFF followed by 10 times 0x00. The pattern is shown in the following table.

Pattern index	DATA[7:0]	Pattern index		DATA[7:0]
1	0xFF	6		0x00
2	0x00	7		0x00
3	0x00	8		0x00
4	0x00	9		0x00
5	0x00	10		0x00
		11		0x00

The receiving FPGA can implement logic to increment the input delay per IO pin and detect the edges and find the ideal sample window. The board support package includes the training logic and is delivered as open VHDL.

9.2 FPGA SPI interface

The serial port interface allows the user to read status information and configure the multiplexer data source and configure the trigger and sync logic.
Three pins define the SPI of the FPGA the SPI_SCLK pin, the SPI_SDIO pin, and the SPI_CSN pin. The SCLK (serial clock) pin synchronizes the read and write data presented from/to the FPGA. The SDIO (serial data input/output) pin is a dual-purpose pin that allows data to be sent and read from the internal FPGA memory map registers. The CSN (chip select not) pin is an active low control that enables or disables the read and write cycles. The following figure shows the timing diagram of the SPI interface.



ts >20ns	tlow>20ns
tds>10ns	thigh>20ns
tdh>1ns	15ns<toe>10ns

Figure 18: FPGA SPI interface timing

All transactions start with a 16-bit word. The first bit of the serial data indicates whether a read (1) or write (0) command is issued. This allows the serial data input/output (SDIO) pin to change direction from an input to an output.
For the DC coupled boards the next 3 bits (A14-A12) select the target for the SPI transaction
000 -> target is FPGA
101 -> target is LTC2666 offset DAC.
The remaining bits are used as address for the FPGA or for the LTC2666. During write commands the first 16 bits should be followed with a 32 bits data word. During reads the FPGA will output a 32 bits data word.
The SPI register map is described in the next table.

Table 2: FPGA SPI register map

Address	Name	Description
0x0050	MUX command	Bit 2:IO reset = 1 causes a synchronization of the input and output logic of the MUX. Bit 3: sim reset = 1 causes the ramp counter to reset to 0 Bit 4: FIFO reset = 1 causes the FIFOs to reset to align the A en B data streams. All bits are self clrearing
0x0054	MUX control	Bit 3:0:AD data_select 0= ADCA data, 1= 16 bits ramp, 2=training pulse, 3=flash between 0xFF and 0x00 Bit 7:4:BD data_select 0= ADCB data, 1= 16 bits ramp, 2=training pulse, 3=flash between 0xFF and 0x00
0x0070	SYNC command	Bit 0: Software trigger =1 will generate a software trigger to those outputs that are listening to the software trigger.
0x0074	SYNC control	Bit [1:0]:dacA_sync source 0=external IO, 1=sync_c2m, 2=software trigger, 3 = 0 Bit [3:2]:dacB_sync source 0=external IO, 1=sync_c2m, 2=software trigger, 3 = 0 Bit [5:4]:pll_sync source 0=external IO, 1=sync_c2m, 2=software trigger, 3 = 0 Bit [9:8]:SYNC_M2C source 0=external IO, 1=sync_c2m, 2=software trigger, 3 = 0 Bit [11:10]:extio_source (only if external IO is set to output) 0=toggle 0/1, 1=sync_c2m, 2=software trigger Bit [12]:external IO direction 0= input, 1 = output
0x0078	SYNC trigger duration	Bit[31:0] trigger duration in CLK_200M counts. Set the amount of clock cycles the trigger is pulsed high after receiving a software trigger command.

10 Power Supply

An FMC board receives several voltages from the FMC connector. These are 12V, 3V3, VADJ and 3V3_AUX. The LXD31K4 uses all these supplies. The following table shows the typical power consumption and current draw of the LXD31K4 on the different voltage rails.

Table 3: Power consumption AC coupled version

Voltage Rail	Typical Current (A)	Typical Power (Watts)
VADJ	0.49	0,4 @ 1.8V
3P3V	2.88	9.5
12P0V	0.23	2.8
3P3VAUX (Operating) 3P3VAUX (Standby)	0.010 0.000001	0.033 0.0000033

Table 4: Power consumption DC coupled version

Voltage Rail	Typical Current (A)	Typical Power (Watts)
VADJ	0.21	0,4 @ 1.8V
3P3V	2.88	9.5
12P0V	0.53	6.4
3P3VAUX (Operating) 3P3VAUX (Standby)	0.010 0.000001	0.033 0.0000033

10.1 Power supply control

Through the I2C IO expander the carrier board FPGA can independently enable the power for the FPGA, Analog powers, and reference clocks. By default all the powers are disabled at power-up. Appendix A describes how to powerup the FMC.

11 Health monitoring

11.1 Requirements and handling instructions

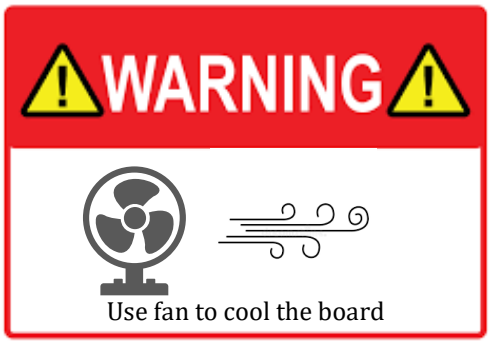
- The LXD31K4 must be installed on an FMC carrier board compliant to the VITA 57.1 standard.
- Do not flex the board
- Observe ESD precautions when handling the board to prevent electrostatic discharges.
- Do not install the LXD31K4 while the FMC carrier board is powered up.

11.2 Temperature

Operating temperature
- -0°C to +70°C (Commercial)
- -40°C to +85°C (Industrial)

Storage temperature:
- -40°C to +120°C

11.3 Cooling



The air flow provided by the fans of the chassis the LXD31K4 is enclosed in will dissipate the heat generated by the onboard components. A minimum airflow of 300 LFM is recommended.
For standalone operations (such as on a Xilinx development kit), it is highly recommended to blow air across the FMC to ensure that the temperature of the devices is within the allowed range. Warranty does not cover boards on which the maximum allowed temperature has been exceeded. Optionally, a heatsink with or without fans can be ordered for the LXD31K4. Contact sales for this option.

11.4 Monitoring

The LXD31K4 uses two AD7291 '8-channel I²C 12-bit SAR ADCs with temperature monitor' devices for monitoring power supply voltages. The device can be programmed and read through the I²C bus at the address defined in chapter 5. The following steps are required to control the AD7291 devices.

1) At power-up, the firmware should write a '1' to the Reset bit in the Command Register to initialize the part to a known state.

2) All measured values must be multiplied by a constant to convert to the actual analog level, formulas are included in the associated tables and text.
3) Continuously operating the I²C bus might interfere with the A/D conversion process resulting in signal distortion. It is recommended to program the minimum and maximum thresholds in the monitoring device and only read from the device when the interrupt line is asserted

11.5 Monitor 1

The first device is configured to monitor the voltages shown in the table below. To convert the ADC reading to voltage multiply the ADC value by the scaling factor listed in the table.

Parameter:	Voltage	Formula
Channel 0	GND	Measured value*(2.5/4096.0)*1
Channel 1	VCCA_2V0	Measured value*(2.5/4096.0)*1
Channel 2	GND	Measured value*(2.5/4096.0)*1
Channel 3	VCCA_5VA	Measured value*(2.5/4096.0)*3
Channel 4	VCCD_3V5	Measured value*(2.5/4096.0)*2
Channel 5	VCC_3V3	Measured value*(2.5/4096.0)*2
Channel 6	VCC_VADJ	Measured value*(2.5/4096.0)*2
Channel 7	VCC_2V5	Measured value*(2.5/4096.0)*2
Temperature	Internal temperature monitor	Measured value/4.0

Table 5: first AD7291 Voltage Parameters

11.6 Monitor 2

The second device is configured to monitor the voltages are shown in the table below. To convert the ADC reading to a voltage or temperature multiply the ADC value by the scaling factor listed in the table.

Parameter:	Voltage	Formula
Channel 0	VCC_1V0	Measured value*(2.5/4096.0)*1
Channel 1	VCC_1V8A	Measured value*(2.5/4096.0)*1
Channel 2	VCCD_12V	Measured value*(2.5/4096.0)*6
Channel 3	VCCO_3V3	Measured value*(2.5/4096.0)*2
Channel 4	VCC_3V3A	Measured value*(2.5/4096.0)*2
Channel 5	VCC_1V8D	Measured value*(2.5/4096.0)*1
Channel 6	VCC_3V3_REF	Measured value*(2.5/4096.0)*2
Channel 7	VCC_5V3	Measured value*(2.5/4096.0)*3
Temperature	Temp Mon2	Measured value/4.0

Table 6: second AD7291 Voltage Parameters

12 Safety

This module presents no hazard to the user.

13 EMC

This module is designed to operate within an enclosed host system built to provide EMC shielding. Operation within the EU EMC guidelines is not guaranteed unless it is installed within an adequate host system. This module is protected from damage by fast voltage transients originating from outside the host system which may be introduced through the system.

14 Board support package

Logic-X has a board support package targeting the KCU105 and VCU118 that includes VHDL based Vivado IP blocks that have an AXI lite configuration interface and AXI stream data interfaces. The following IP is included:

- the input IO deserialization and IODELAY training
- output IO serialization
- Configurable spi master to target all the SPI devices on the LXD31K4
- I2C master
- Data capture IP
- Data playback/waveform generation IP
- TCP/IP offload engine to AXI lite and AXI stream input and output.

In addition to the FPGA IP it comes with a windows based software application that takes care of all the initialisation of the different devices on the LXD31K4. The software communicates with the carrier board FPGA over TCP/IP. After the initialisation the software will configure the wave form generator to playback several different frequencies out over the DAC outputs. At the same time 32 KB snapshots are recorded on the ADC inputs using the Data capture IP. The received data is then stored in a file for further analysis.

On request Logic-X can port the design to other Xilinx FPGA platforms.

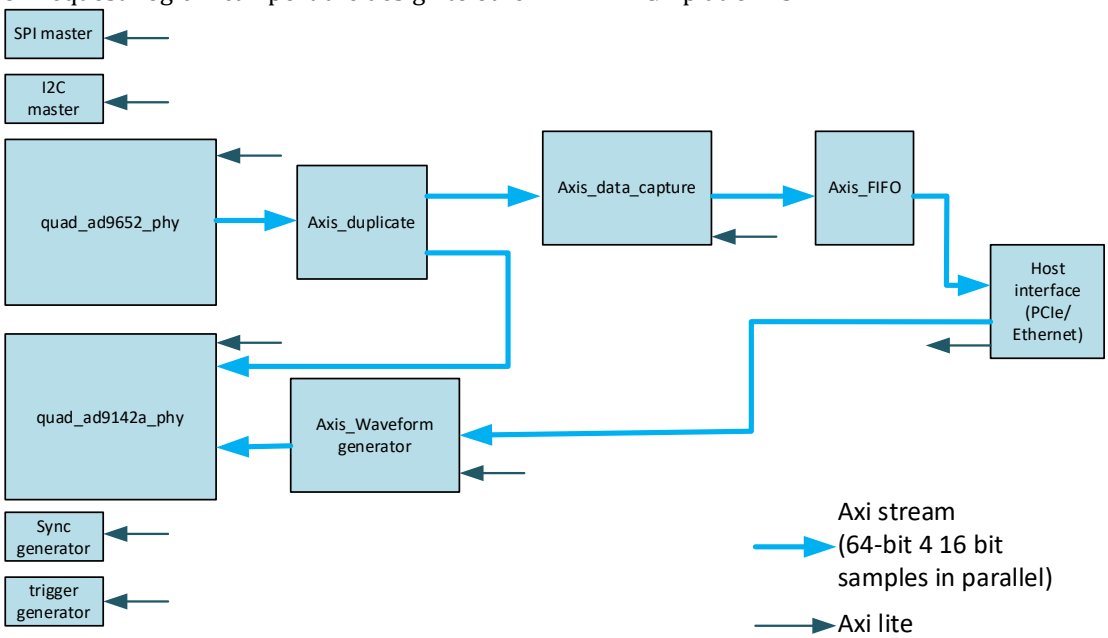


Figure 19: Reference design FPGA top level

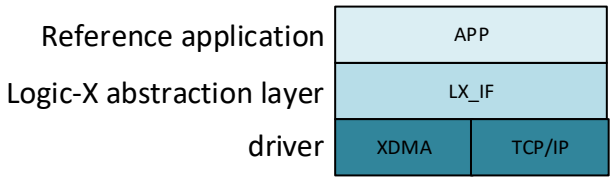


Figure 20:Reference application software layers

Appendix A: Enable LXD31K4 powers

Through the I2C IO expander the carrier board FPGA should enable the power of the LXD31K4. By default, all the powers are disabled at power-up.
To power up the FMC board the following steps should be taken:

- Write 0x01 to IOexpander P0 output register.
  - Select local reference clock.
- Write 0x00 to IO expander P1 output register.
  - all power disabled.
- Write 0x0 to IO expander P2 output register.
  - Set all pins to 0.
- Write 0x00 to IOexpander P0 control to set all outputs.
- Write 0x00 to IOexpander P1 control to set all outputs.
- Write 0x3F to IOexpander P2 control to set [5..0] to inputs and [7..6] to outputs.
- Write 0x10 to IO expander P1 output register.
  - Enable Switching regulators.
- Write 0x1F to IO expander P1 output register.
  - Enable all powers.

Appendix B: PLL configuration

Below is a table with the register values to write to the PLL that shows configures the PLL to 250 Mhz towards the ADCs, 1000 MHz towards the DAC and 500 MHz towards the FPGA. The PLL will use the local 100 MHz reference clock. For this configuration to work the reference clock mux should be set to Local. Also, reference power supply should be enabled. The registers should be written in the order of the index.

Table 7: PLL configuration table.

Index	SPI address:	Data to write
1	0x02	0x02
2	WAIT 1 ms	
3	0x03	0x38
4	0x04	0xb3
5	0x05	0x04
6	0x06	0x32
7	0x07	0x07
8	0x08	0x20
9	0x09	0x83
10	0x0a	0x00
11	0x0b	0x83
12	0x0c	0x00
13	0x0d	0x81
14	0x0e	0x00
15	0x0f	0x81
16	0x10	0x00
17	0x11	0x82
18	0x12	0x00

Appendix C: ADC configuration

Below is a table with the typical initialization sequence to configure the ADC for binary data output at 250 MHz.
Before running this sequence there should be a valid clock for the ADC.

Table 8: ADC configuration table.

Index	SPI address:	Data to write
1	0x00	0x20
2	0x20	0x01
3	0x14	0x00
4	0xFF	0x01

Appendix D: DAC configuration

Below is a table with the typical initialization sequence to configure the DAC for times 4 interpolation and binary data input at 250 MHz. Before running this sequence there should be a valid clock for the ADC.

Table 9: ADC configuration table.

Index	SPI address:	Data to write
1	0x00	0x3A
2	0x20	0x01
3	0x5E	0x00
4	0x5F	0x60
5	0x0A	0x00
6	0x28	0x02
7	0x26	0x80
8	0x25	0x01
9	0x01	0x00

Appendix E: FPGA IO training

Below is a table with the typical initialization sequence to train the FPGA IOs. The LXD31K4 BSP includes VHDL that takes care of the IO training.

Table 10: ADC configuration table.

Index	Action	Interface
1	Make sure the ADC and PLL are configured	
2	Reset the FPGA IO on the LXD31K4	SPI
3	Reset the FPGA FIFOs on the LXD31K4	SPI
4	Set the data pattern for AD an AB to training pules	SPI
5	Reset the carrier board FPGA IO delays and input deserialisers	
6	Optionaly increment the IO delay to a start position	
7	Start the IODELAY training in the carrier board FPGA	
8	Verify if IO delay training completed successfully.	
9	Switch the AD and BD data source to ramp	SPI
10	Verify if you receive a proper ramp	
11	Set AD and BD data source to ADC data	SPI

Appendix F: Revision history

Document Revision	Changes			Quality Approval	Date
R1.0	First release			EBa	August 31 2021