AN 882: Using ADI AD9217 with Intel® Stratix® 10 Devices

Contents

1. Using ADI AD9217 with Intel Stratix 10 Devices

AD9217 is a single 12-bit, 10.25 gigasample per second (GSPS) radio frequency (RF) analog-to-digital converter (ADC) from Analog Devices Inc. (ADI).

AD9217 features a high speed parallel output interface to support its maximum bandwidth capability. It includes digital data path that can be configured for direct real or down converted to complex intelligence quotient (IQ) data. The interface includes 12 data lines, single data clock operating at one-half of ADC sample rate, and a parity bit.

- **Data outputs** are scrambled to maintain direct current (DC) balance on each data lane. During calibration, the scrambler should remain off.
- A **parity bit** is generated for each 12-bit sample to detect single bit errors that result from channel-to-channel misalignment. The parity bit is passed along simultaneously with the data sample and it is determined prior to being scrambled.
- The **data clock** operates at one-half the data rate. Data and the parity bit operate up to 10 Gbps while the clock operates up to 5 GHz.

AD9217 has a low latency chip-to-chip (C2C) output data formats.

1.1. Hardware Requirements

The hardware setup test requires the following hardware:

- Intel® Stratix® 10 H-Tile Signal Integrity Development Kit
- ADI AD9217 evaluation module (EVM)
- Mini-USB cables
- SMA cables
- Clock source card capable of generating sampling clock frequency

1.2. Hardware Setup

An Intel Stratix 10 GX H-Tile Signal Integrity Development Kit is used with the ADI AD9217 EVM attached to the FPGA Mezzanine Card Plus (FMC+) connector of the development board.

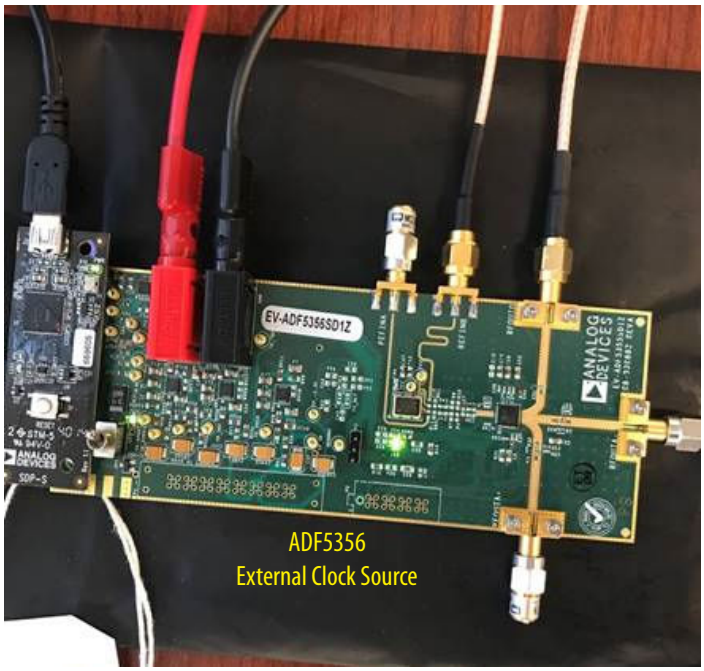
Figure 1. Hardware Setup—10G Design

- The ADC AD9217 derives power from the FMC pins.
- Si5341 clock generator on the Intel Stratix 10 development kit provides reference clock to the ADF5356 clock source and device clock to the field-programmable gate array (FPGA).
- In ADF5356, provide a reference clock to a single-ended SMA input marked as REFINB, while a 5 GHz suppression filter is added. In this setup procedure, VHF-8400+ from Mini-Circuits Inc. is used.

- Attach SMA 50 Ohm terminations to other inputs and outputs and ensure required modifications are done to the ADF5356 board. Refer to Figure 2 on page 5.

  Note: Remove R12 and R27 to apply external reference.

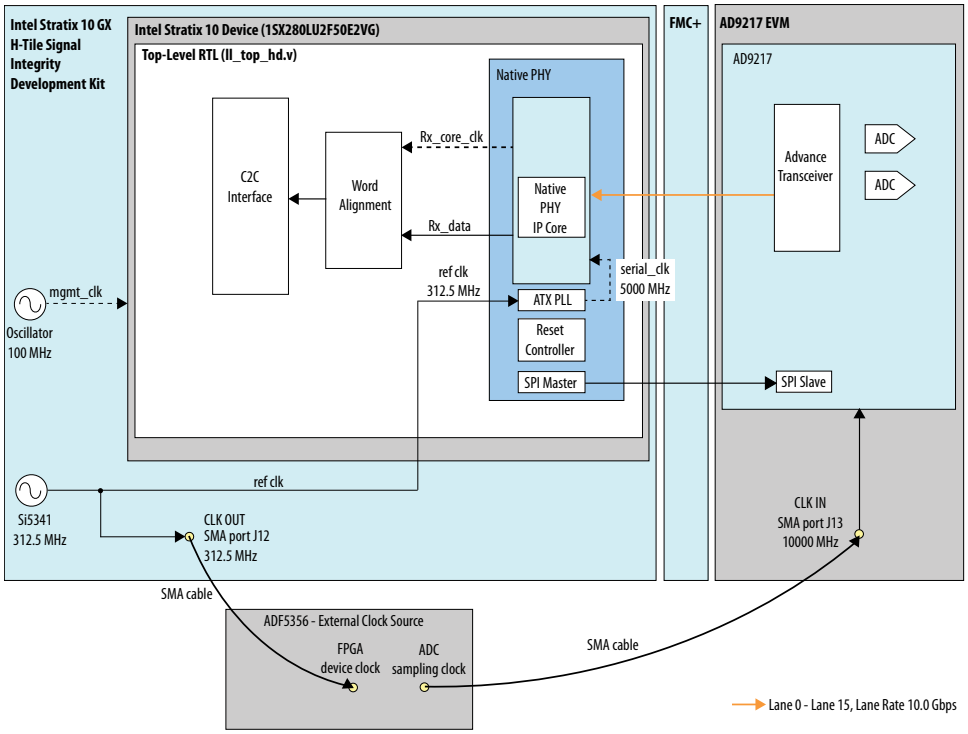
Figure 2. ADF5356 Terminating Unused Input and Output



- Connect RFOUTB of ADF5356 to the clock input J13 of AD9217.
- Connect the mini-USB cable to the system demonstration platform-serial (SDP-S) controller board of ADF5356 and power up the boards.

The following system-level diagram shows how the different modules connect in this design.

Figure 3. System Diagram—10G Design

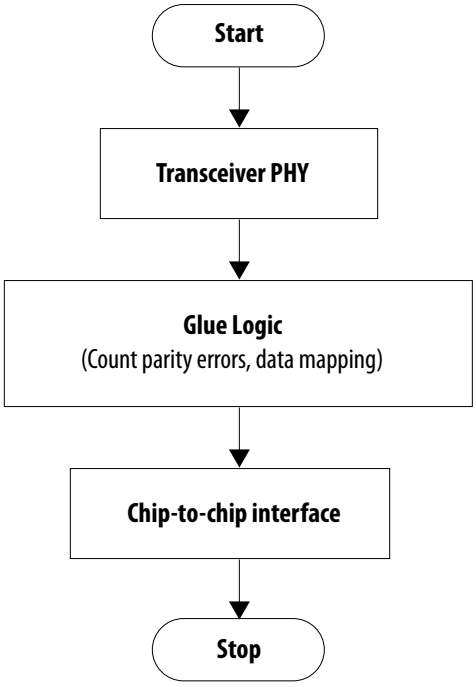


In this setup, the data rate of the native PHY transceiver lanes is 10 Gbps. An external clock source card ADF5356 provides sampling clock of 10000 MHz to the ADC (AD9217) through the SMA cables. The Si5341 oscillator on the development kit board provides reference clock to the FPGA and the external clock source board ADF5356. The reference clock to the FPGA is provided to the ATX PLL, which generates the serial clock to transceiver. SPI is used to access the registers in ADC (AD9217).

1.3. Design Description

1.3.1. Overall Design Flow

Figure 4. Design Flow



1.3.2. Low-Latency C2C Interface

The FPGA program supports the low-latency chip-to-chip (C2C) interface mode.

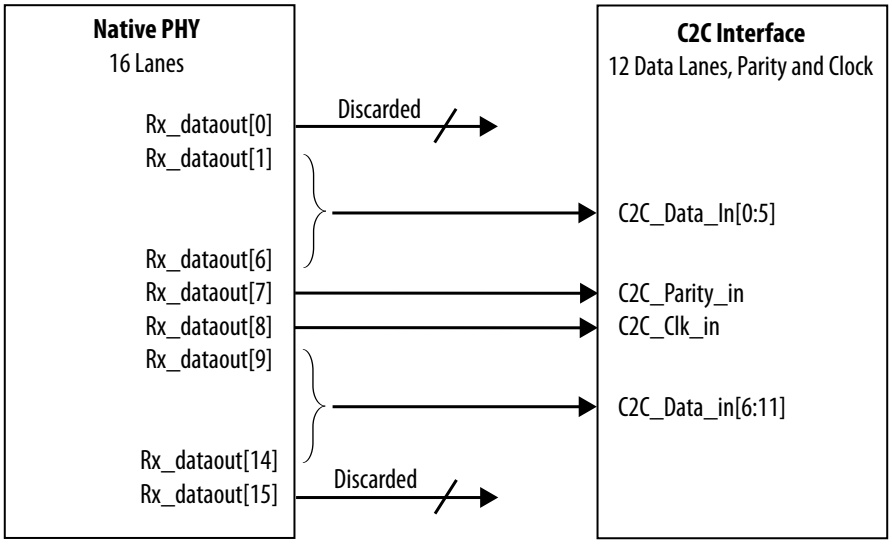
- In C2C mode, the FPGA controls the C2C pattern select and scramble fields of the ADC to implement the C2C alignment sequence.
- The FPGA de-asserts scrambling during alignment and enables scrambling after alignment is complete.
- FPGA supports inverted or non-inverted parity.
- Before running C2C mode, the ADC and FPGA C2C alignment pattern values must be set the same.

1.3.2.1. Lane Mapping in C2C

In the native PHY Transceiver, there are 16 lanes. 16 lanes transceiver is mapped to the C2C interface with 12 data lanes, 1 parity, and a clock.

- The lane 0 and lane 15 from the transceiver were discarded.
- The lanes 1 to 6 and lanes 9 to 14 are mapped to the 12 C2C data lanes.
- Lane 7 and lane 8 of the native PHY Transceiver is mapped to the parity and clock of the C2C interface.

Figure 5. Lane Mapping



1.3.2.2. C2C Design Flow

The following local parameters are involved in the C2C interface:

- WORD_WIDTH = 16
- WORDS_PER_CYCLE = DATA_WIDTH / WORD_WIDTH
- DELAY_CYCLES = ((2 * ALIGN_CNT_WIDTH) + 2 * WORDS_PER_CYCLE – 2) / (WORDS_PER_CYCLE)

1.3.2.3. Calibration Operation Sequence

Table 1. Calibration Sequence

Step	Action	Device	Details
1	Data capture start asserted	FPGA	FPGA initiates system startup procedure.
2	FPGA sets DUT scrambling off & sets pattern to clock/data alignment pattern (default is 0xAAAA)	FPGA	FPGA initiates data and clock alignment.
		AD9217	0x4B1 = 0; scrambler off. 0x4A0[3:0] = 1; Select pattern 1 (Reg 0x481 – 0x482).
	FPGA transceivers configured	FPGA	Write appropriate SPI bits.
	FPGA transceivers clock data recovery locks	FPGA	Read appropriate SPI bits to verify bit alignment.
3	FPGA sets DUT pattern to channel word alignment pattern (default is 0xFF00)	FPGA	FPGA initiates word alignment.
		AD9217	0x4A0[3:0] = 2; Select pattern 2 (Reg 0x483 – 0x484).
	FPGA transceiver comma detect used to align to word alignment pattern	FPGA	Read appropriate SPI bits to verify word alignment.
4	FPGA sets DUT pattern to channel alignment pattern (default is 0xB496)	FPGA	FPGA initiates channel alignment.

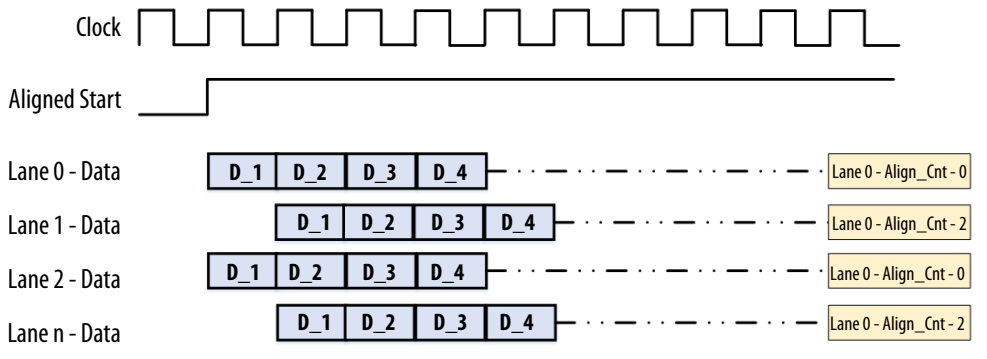
continued...

Step	Action	Device	Details
		AD9217	0x4A0[3:0] = 3; Select pattern 2 (Reg 0x485 – 0x486).
	FPGA RTL de-skew channels	FPGA	Read appropriate SPI bits to verify channel alignment.
5	FPGA sets DUT pattern to ADC data	FPGA	Set mux for ADC data transmission.
		AD9217	0x4A0[3:0] = 0; Selects ADC Data.
	DUT sends break word then ADC Data	AD9217	Automatically sends break word (0x3333) once then ADC data; 0x4A0[3:0] = 0.
	FPGA optionally sets DUT scrambling on after break word seen	FPGA	Sets scrambling on (recommended).
		AD9217	0x4B1[0] = 1; Scrambler on.
6	FPGA starts capturing ADC data	FPGA	FPGA monitors parity bit.

1.3.2.4. Calibration Operation Sequence Descriptions

- Data capture start is given by the FPGA to initiate the calibration operation sequence.
- During bit/clock alignment phase, the FPGA sets the alignment pattern and disables the scrambler. The same is done in ADC (AD9217) by configuring the registers `0x4B1` and `0x4A0` through SPI. Once the bit alignment is done in the FPGA, the word alignment phase starts.
- During the word alignment phase, the FPGA sets the alignment pattern. In ADC (AD9217), configure the register `0x4A0` with appropriate bits to select word alignment pattern. Once the word alignment is done, the channel alignment starts in the FPGA.
- During the channel alignment phase, the FPGA sets the alignment pattern. In ADC (AD9217), configure the register `0x4A0` with appropriate bits to select the channel alignment pattern. During this phase, the lane de-skew is taken care in the C2C module.

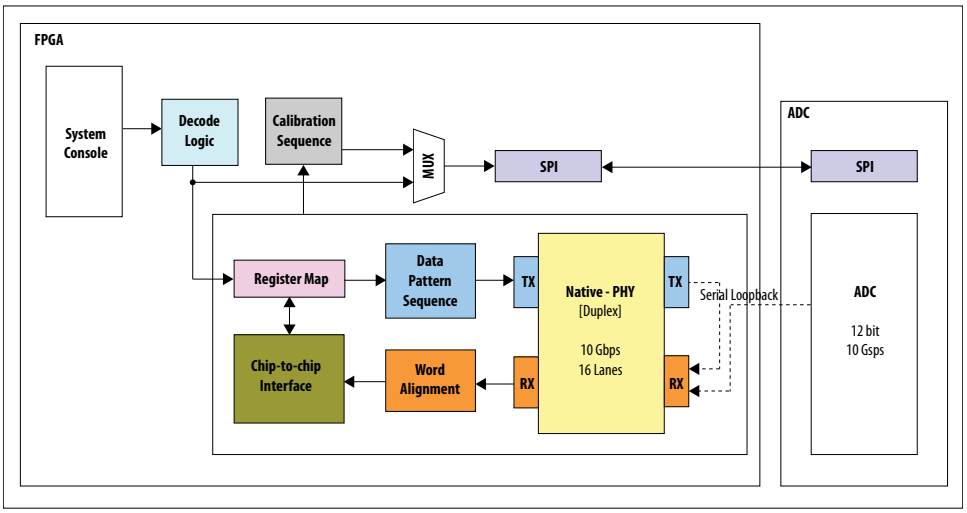
Figure 6. Channel Alignment in C2C



- In Figure 6 on page 9, Lane_0 and Lane_2 are aligned and its channel align count is 0. And in Lane_1 and Lane_n, there is a 32-bit delay in data. Therefore, the channel align count is 2.
- Based on the channel align count, the channels are aligned properly.
- After the lane/channel alignment is done and when the break pattern is detected, the scrambler is enabled in both FPGA and AD9217 and the ADC starts the data transmission.
- The FPGA starts capturing the ADC data and monitors the parity bit.

1.4. Functional Description

Figure 7. System Architecture



The design has two major blocks:

- Native PHY
- Chip-to-chip (C2C) interface

The native PHY supports data rate of 5 Gbps. The C2C interface combines controls from multiple PHYs, takes care of channel/word alignment, maps physical lane to logical lane, and converts lane data to sample data.

1.5. Parameterization

1.5.1. Design Parameters

- NO OF TRANSCEIVERS = 16
- NUM_PHY = 1
- NUM_LINK = 1
- C2C_NUM_DATA_LANE = 12
- C2C_ALIGN_CNT_WIDTH = 2

1.5.2. Derived Parameters

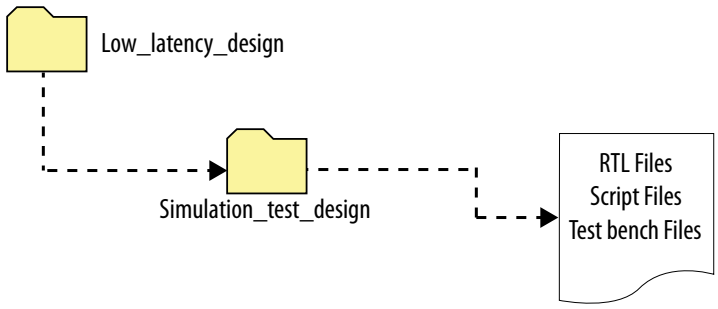
- Lane rate = 10 Gbps
- Number of lanes = 16
- ADC sampling clock = 10 GHz
- ADC samples = 10 GSps
- Sample bits = 12
- Bit rate = 10 Gsps * 12 = 120 Gbps
- *Note:* There are 12 C2C data lanes. Hence, lane rate = 120 Gbps/12 = 10 Gbps.

1.5.3. Clocks

- Master_clk = 100 MHz
- Mgt_Refclk = 312.5 MHz (Actual reference clock to generate serial clock)
- Rx_Phy Clock = (Lane rate/32) = 10 Gbps/32 = 312.5 MHz
- Serial Clock = (Lane rate/2) = 10 Gbps/2 = 5000 MHz

1.6. Directory Structure

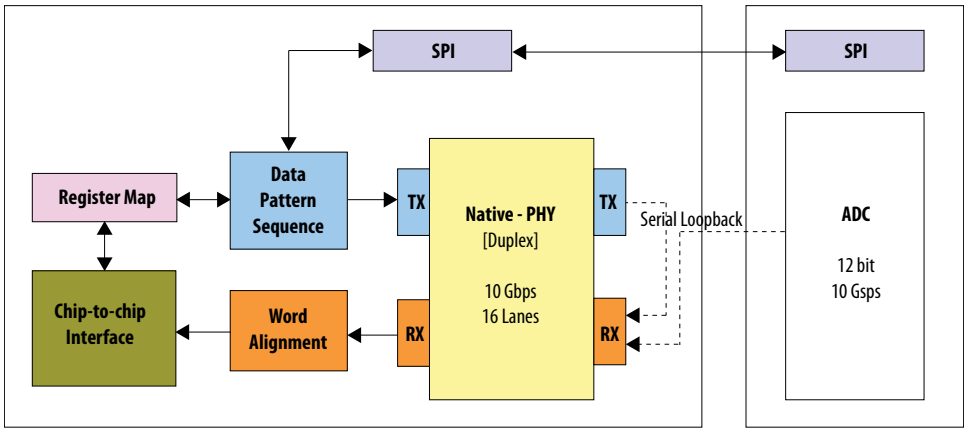
Figure 8. Directory Structure for the AD9217 Low Latency 10G Design



Simulation test design includes RTL files, testbench files, and script files to perform functional simulation and latency measurement.

1.7. Simulation

Figure 9. Simulation Environment



The data pattern sequence block generates the calibration sequence data, which is serial loopbacked and is provided as an input to the PHY receiver (RX).

1.7.1. Procedure

To run the functional simulation using VCS simulation tool, follow these steps:

1. In a Terminal window, open the `simulation_test_design` file, which is present in the design folder.
2. Change the working directory to `simulation_test_design/ll_nphy_tb`.
3. In the command line, type `sh run_vcs.sh` to invoke the VCS Discovery Visual Environment (DVE).
4. In the VCS DVE, select **File ➤ Load Session** from the main menu and select `simulation_test_design/load_wave.tcl`.
5. Click **Load** to load the session.
6. Run the simulation for 0.5 ms.
7. When the simulation is completed, observe the output displayed. The following diagrams show the simulation waveforms of the native PHY status output, data pattern generation and C2C states, and latency measurements for 10G design.

Figure 10. Native PHY Status Output

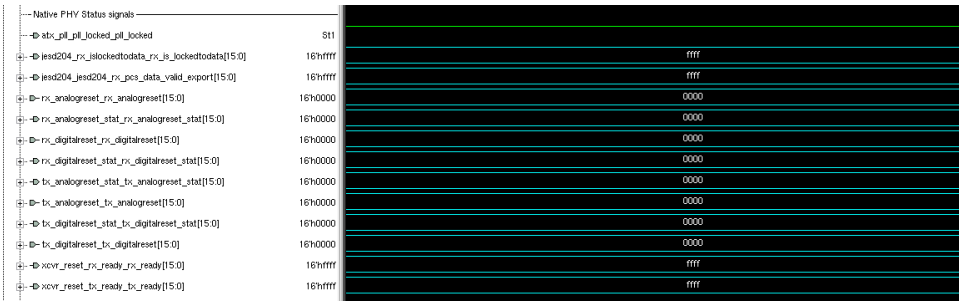


Figure 11. Data Pattern Generation and C2C States

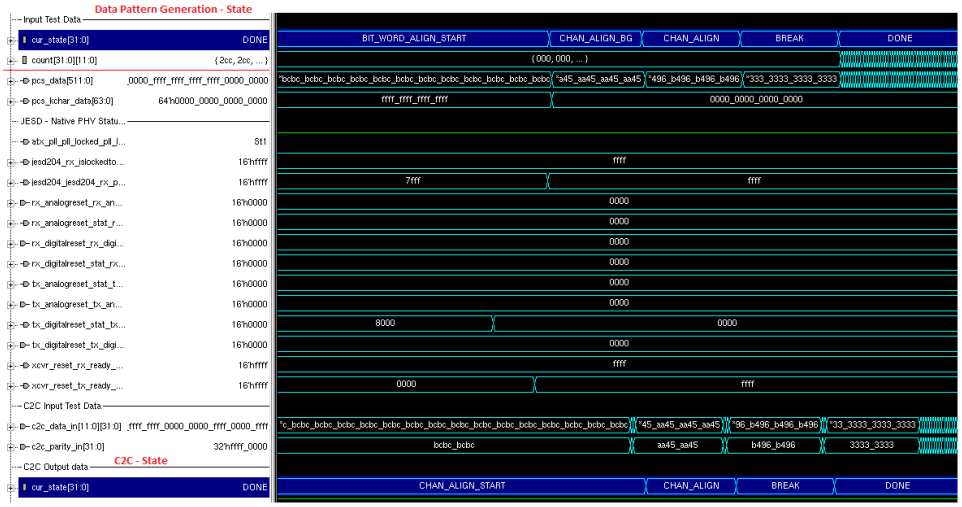


Figure 12. Data at C2C Output

This simulation waveform has the data at C2C output. Incremental data is provided in each lane and it is observed at C2C output.

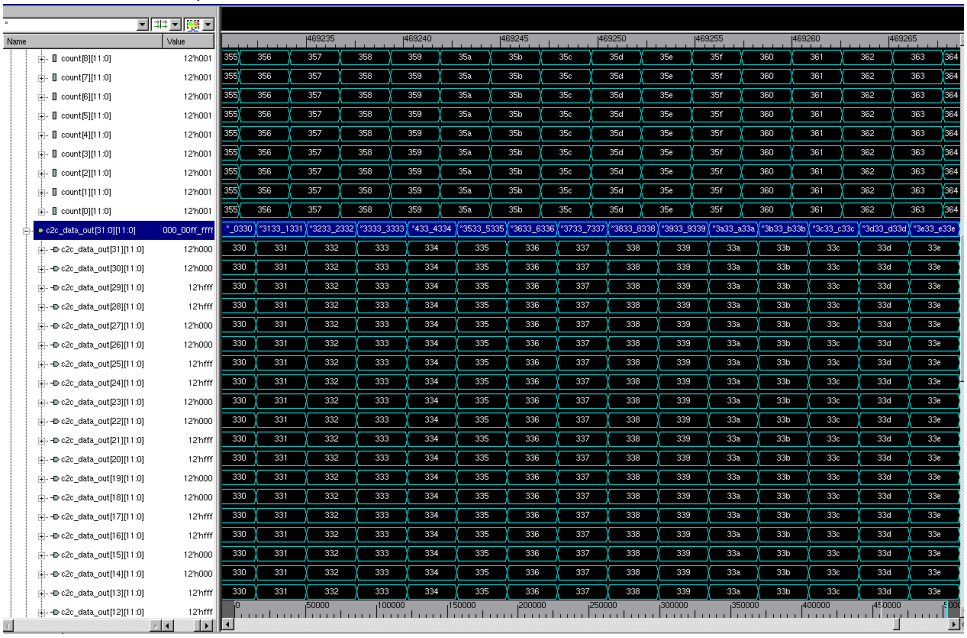


Figure 13. Latency Measurement from PHY Transmitter to C2C Output

This simulation waveform shows the latency measured from the data injected at the PHY transmitter to C2C output. The data 0x001 is a reference for latency measurement, which is highlighted below.

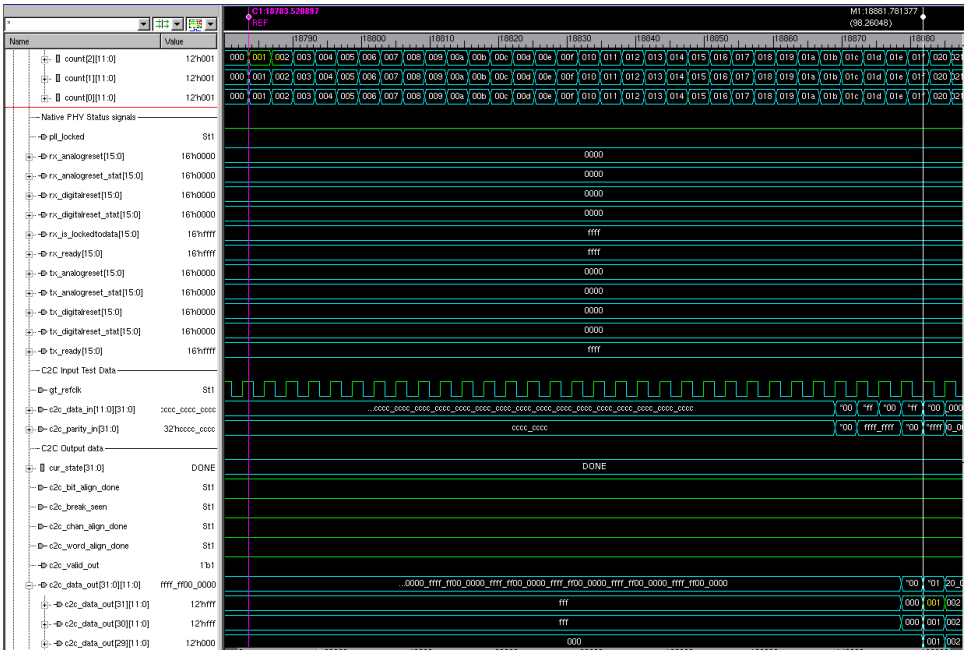
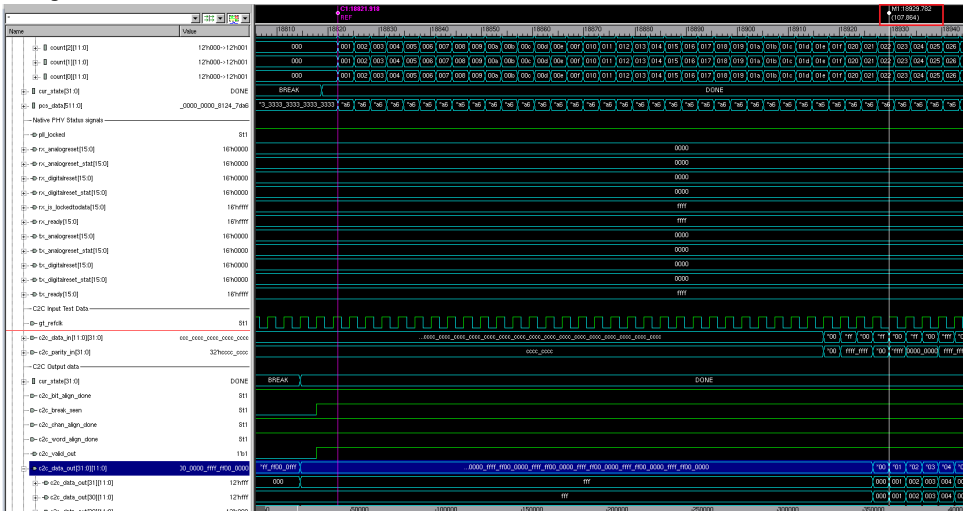


Figure 14. Latency Measurement for 10G Design—107.864 ns

This simulation waveform shows the measured latency for 10G design after adding pipieline registers to close timing.



1.8. Latency Measurement for 10G Design

Because hardware testing is not performed for 10G design, the latency measurement for 10G is derived by considering 5G design calculations and added pipelines to close timing in 10G designs. Refer to the *Appendix: Latency Measurement for 5G Design* for more information.

Related Information

Latency Measurement for 5G Design on page 26

1.8.1. Latency Calculations

Speed	FPGA Clocking (MHz)	Latency on User Interface (C2C) (ns)	Latency on Transceiver Parallel Data (Native PHY FPGA Interface) (ns)
5GT/s (5GSps)	156.25	179.2	140.8
10GT/s (10GSps) (1)	312.5	99.2	80

For designs with the lane rate of 5 Gbps:

- PHY receiver clock = (Lane rate/32 = 5000e6/32) = 156.25 MHz.
- Clock period of 156.25 MHz = 6.4 ns.
- Number of clock cycles between pulse injected at the ADC input to pulse observed at C2C output – 28 clock cycles of PHY receiver clock.
- Latency measured at the C2C output = 6.4 ns * 28 = **179.2 ns**.
- Since the C2C design in the FPGA has a constant delay of 6 clock cycles, this can be excluded from overall latency so that the latency can get up to receive transceiver.
- Latency measured at the transceiver output = 6.4 ns * (28-6) = **140.8 ns**.

Considering the above calculation, the latency for 10G lane rate is derived and shown below:

- PHY receiver clock = (lane rate/32 = 10000e6/32) = 312.5 MHz.
- Clock period of 312.5 MHz = 3.2 ns.
- Number of clock cycles between pulse injected at ADC input to pulse observed at C2C output – 28 clock cycles of PHY receiver clock.
- Latency measured at the C2C output = 3.2 ns * 28 = **89.6 ns**.
- C2C latency in clock cycles = 6 clock cycles.
- Latency measured at the transceiver output = 3.2 ns * (28-6) = **70.4 ns**.

After considering the pipeline delays in 10G designs, the latency is further increased to 3 clock cycles:

Note: Pipeline stages is added to close timing in 10G designs.

- Total latency measured = 3.2 ns * 31 = **99.2 ns**.
- C2C latency in clock cycles = 6 clock cycles.
- Latency measured at transceiver output = 3.2 ns * (31-6) = **80 ns**.

(1) The latency for 10G lane rate is derived by considering the 5G design calculations and added pipelines to close timing in 10G designs (3 clock cycles).

1.9. Register Map

Table 2. FPGA Register Map

Address	Register Name	R/W	Bit Field Name	Bits	Description
0001	C2C_SCRAMBLE_EN	R/W	c2c_scramble_en	[0]	Scrambler enable signal.
0010	C2C_OUTPUT_EN	R/W	c2c_output_en	[0]	C2C output enable signal as the C2C Valid.
0011	C2C_CHAN_ALIGN_BG_PATT	R/W	c2c_chan_align_bg_patt	[15:0]	16-bit channel alignment BG pattern register.
0100	C2C_CHAN_ALIGN_PATT	R/W	c2c_chan_align_patt	[15:0]	16-bit channel alignment pattern register.
0101	C2C_BREAK_PATT	R/W	c2c_break_patt	[15:0]	16-bit Break Pattern register.
0110	C2C_PARITY_MODE	R/W	c2c_parity_mod	[0]	Parity Mode select (0/1).
0111	DATA_PATT_SM_START	R/W	DATA_PATT_SM_START	[0]	Data capture start signal to initiate the calibration sequence.
1000	PHY_STATUS	R	Phy Status	[31:0]	Transceiver PHY status.
1001	LANE POLARITY	R/W	Lane polarity	[0]	To take the Transceiver Lane polarity in FPGA.
1010	C2C_STATUS	R	C2c status	[31:0]	C2C status.
1011	BIT REVERSE	R	Bit reversal	[0]	To reverse the PHY output.

1.10. Document Revision History for AN 882: Using ADI AD9217 with Intel Stratix 10 Devices

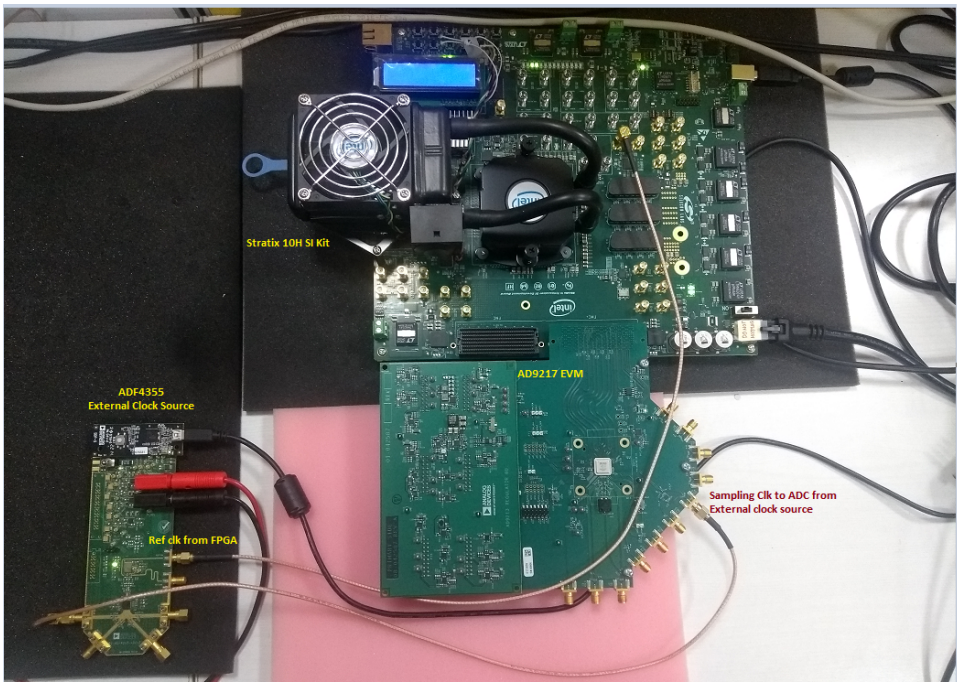
Document Version	Changes
2020.08.17	Updated Figure: *Hardware Setup—10G Design*.
2020.08.14	Initial release.

1.11. Appendix: 5G Design Example

1.11.1. Hardware Setup

An Intel Stratix 10 GX H-Tile Signal Integrity Development Kit is used with the ADI AD9217 EVM attached to the FPGA Mezzanine Card Plus (FMC+) connector of the development board.

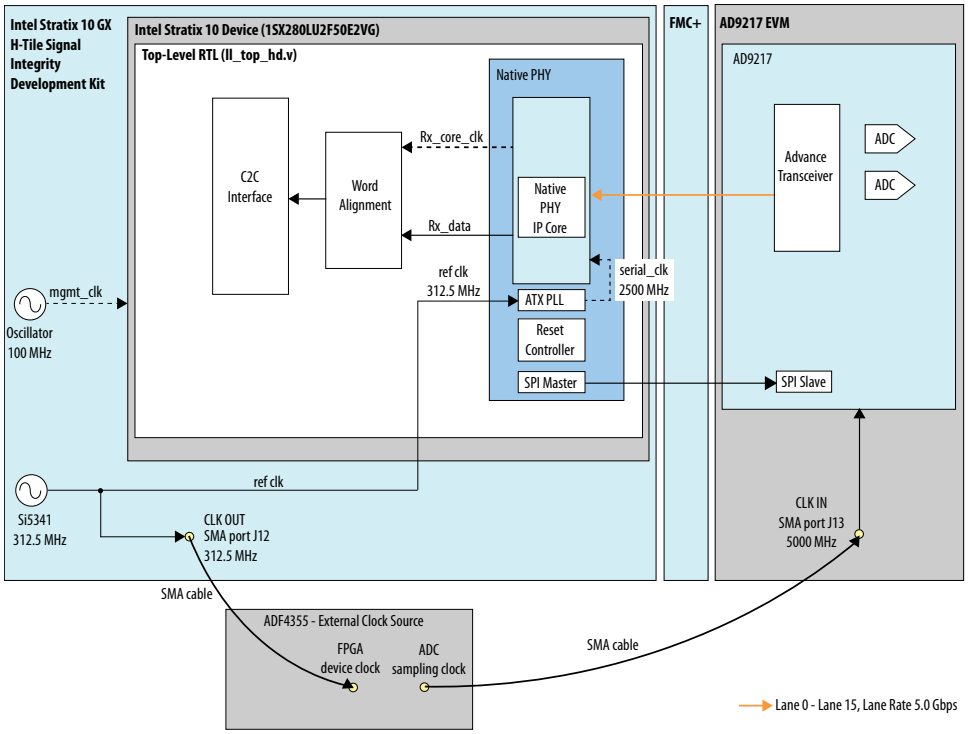
Figure 15. Hardware Setup—5G Design



- The ADC AD9217 derives power from the FMC pins.
- The field-programmable gate array (FPGA) device clock is supplied by Si5341 clock generator on the development kit.
- Si5341 clock generator provides reference clock to the ADF4355 clock source and FPGA.
- Sampling clock to the ADC AD9217 EVM is given by the external clock source ADF4355.

The following system-level diagram shows how the different modules connect in this design.

Figure 16. System Diagram—5G Design



In this setup, the data rate of the native PHY transceiver lanes is 5 Gbps. An external clock source card ADF4355 provides sampling clock of 5000 MHz to the ADC (AD9217) through the SMA cables. The Si5341 oscillator on the development kit board provides reference clock to the FPGA and the external clock source board ADF4355. The reference clock to the FPGA is provided to the ATX PLL, which generates the serial clock to transceiver. SPI is used to access the registers in ADC (AD9217).

1.11.2. Parameterization

1.11.2.1. Design Parameters

- NO OF TRANSCEIVERS = 16
- NUM_PHY = 1
- NUM_LINK = 1
- C2C_NUM_DATA_LANE = 12
- C2C_ALIGN_CNT_WIDTH = 2

1.11.2.2. Derived Parameters

- Lane rate = 5 Gbps
- Number of lanes = 16
- ADC sampling clock = 5 GHz
- ADC samples = 5 GSps

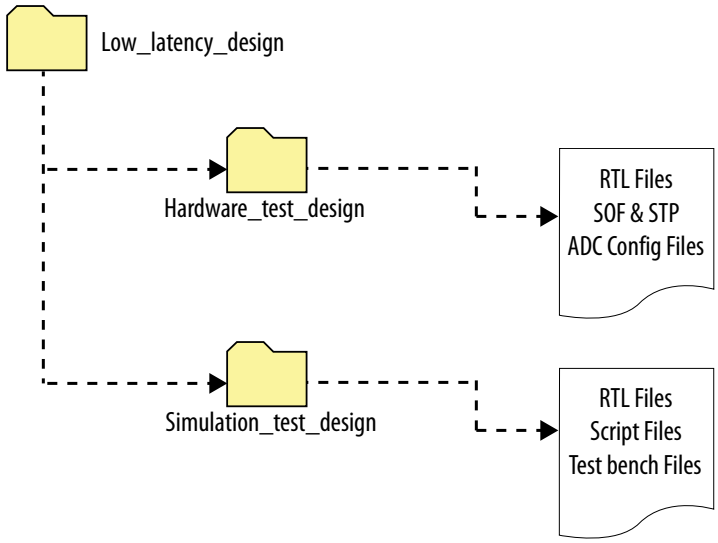
- Sample bits = 12
- Bit rate = 5 Gsps * 12 = 60 Gbps
- *Note:* There are 12 C2C data lanes. Hence, lane rate = 60 Gbps/12 = 5 Gbps.

1.11.2.3. Clocks

- Master_clk = 100 MHz
- Mgt_Refclk = 312.5 MHz (Actual reference clock to generate serial clock)
- Rx_Phy Clock = (Lane rate / 32) = 5 Gbps/32 = 156.25 MHz
- Serial Clock = (Lane rate / 2) = 5 Gbps/2 = 2500 MHz

1.11.3. Directory Structure

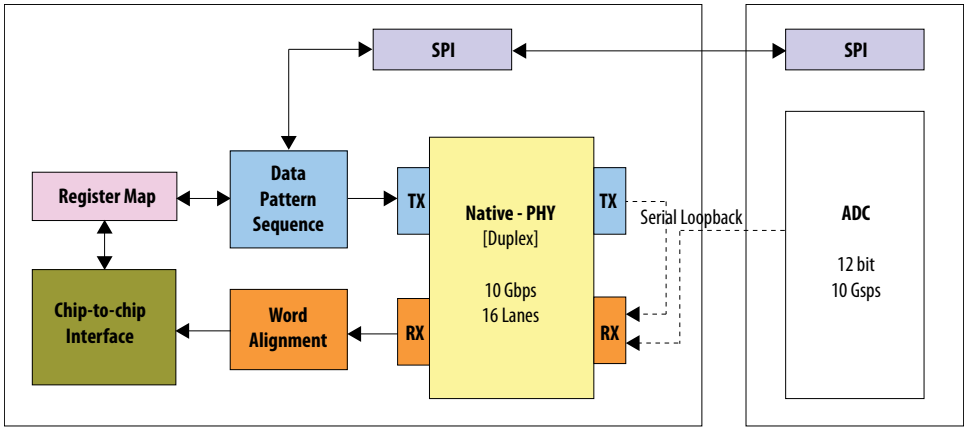
Figure 17. Directory Structure for the AD9217 Low Latency 5G Design



- Hardware test design includes RTL files, SOF file, Signal Tap file, and script files to perform hardware validation and latency measurement.
- Simulation test design includes RTL files, testbench files, and script files to perform functional simulation and latency measurement.

1.11.4. Simulation

Figure 18. Simulation Environment



The data pattern sequence block generates the calibration sequence data, which is serial loopbacked and is provided as an input to the PHY receiver (RX).

1.11.4.1. Procedure

To run the functional simulation using VCS simulation tool, follow these steps:

1. In a Terminal window, open the `simulation_test_design` file, which is present in the design folder.
2. Change the working directory to `simulation_test_design/ll_nphy_tb`.
3. In the command line, type `sh run_vcs.sh` to invoke the VCS Discovery Visual Environment (DVE).
4. In the VCS DVE, select **File ➤ Load Session** from the main menu and select `simulation_test_design/load_wave.tcl`.
5. Click **Load** to load the session.
6. Run the simulation for 0.5 ms.
7. When the simulation is completed, observe the output displayed. The following diagrams show the simulation waveforms of the native PHY status output, data pattern generation and C2C states, and latency measurements for 5G design.

Figure 19. Native PHY Status Output

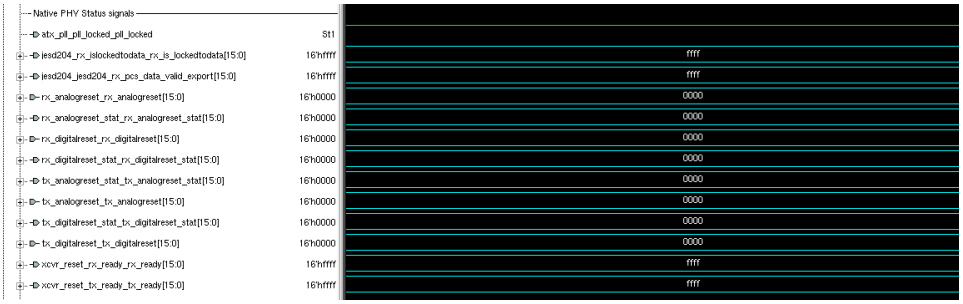


Figure 20. Data Pattern Generation and C2C States

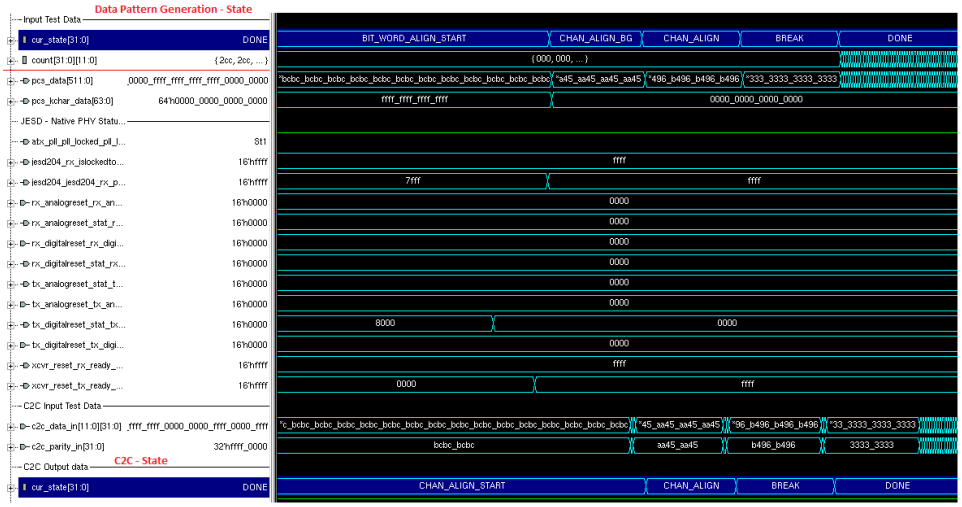


Figure 21. Data at C2C Output

This simulation waveform has the data at C2C output. Incremental data is provided in each lane and it is observed at C2C output.

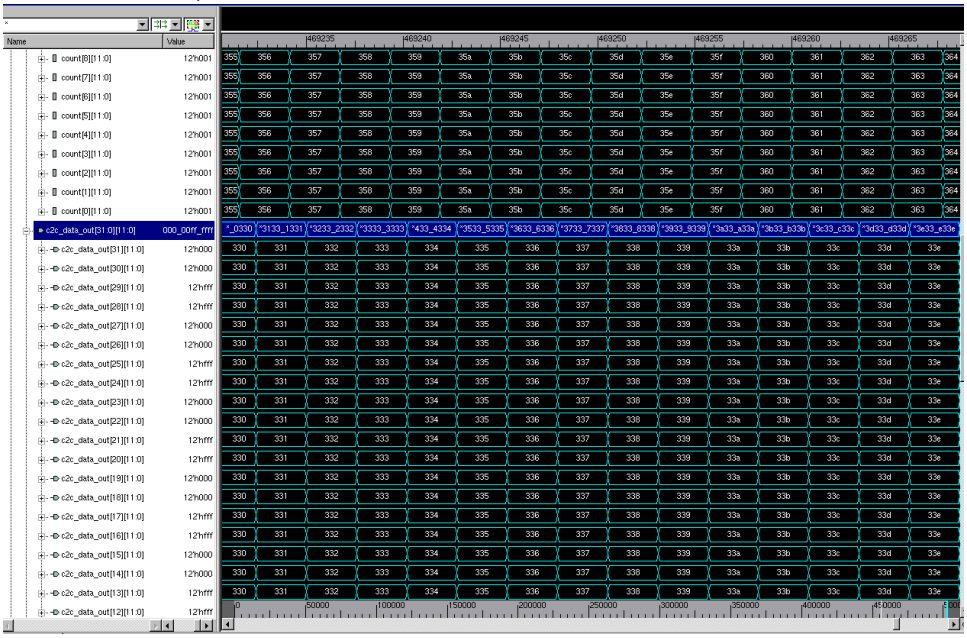


Figure 22. Latency Measurement from PHY Transmitter to C2C Output

This simulation waveform shows the latency measured from the data injected at the PHY transmitter to C2C output. The data 0x001 is a reference for latency measurement, which is highlighted below.

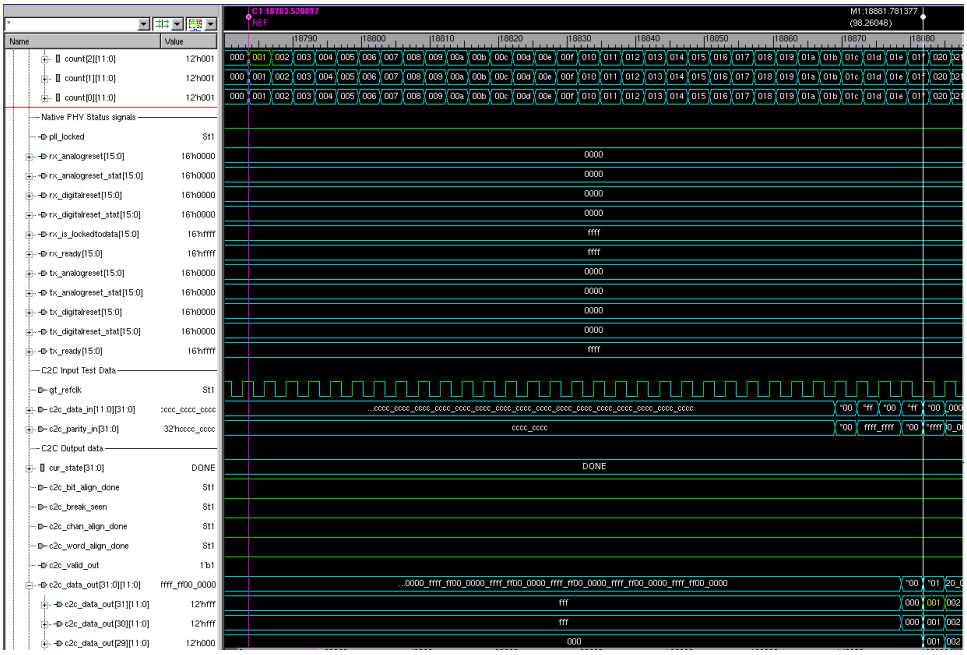
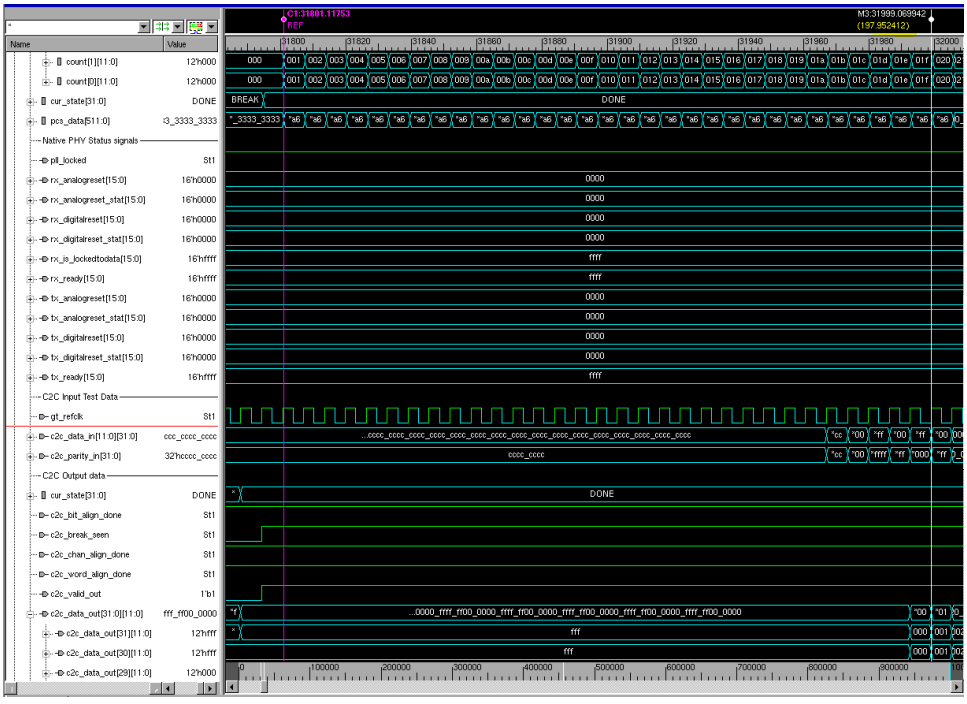


Figure 23. Latency Measurement for 5G Design—197.95 ns



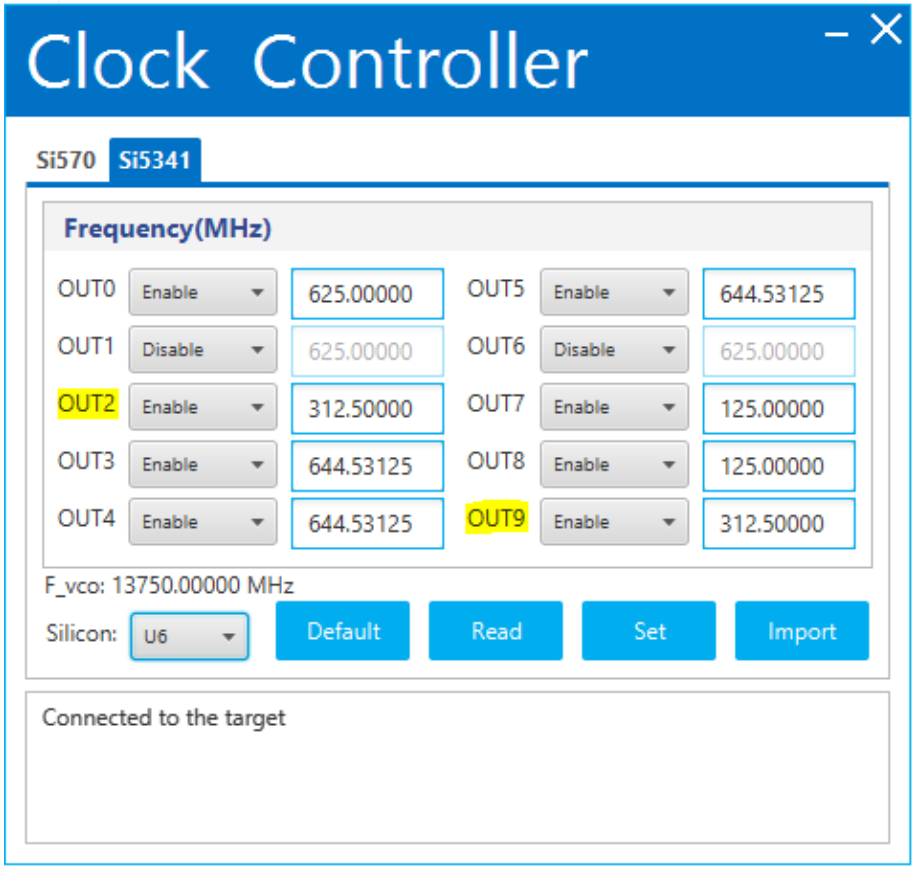
1.11.5. Hardware Testing

1.11.5.1. Procedure

The scripts to run the commands are available in the Intel Quartus® Prime Pro Edition software archive. The Intel Quartus Prime Pro Edition software archive should be extracted into the same folder.

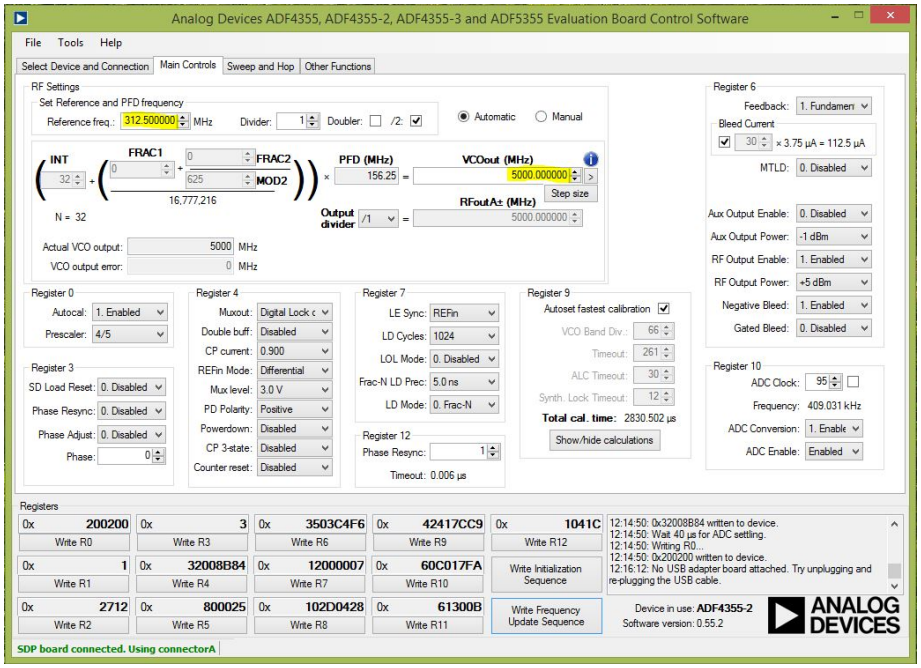
1. Ensure the device clock is available to the FPGA before proceeding to program the FPGA using SOF file.
2. The Intel Stratix 10 H-Tile Signal Integrity Development Kit has a Si5341 clock controller. Program 312.5 MHz in the Out_2 and Out_9 of Si5341 U6 silicon. Out_2 acts as device clock to the FPGA and Out_9 acts as the reference clock to the external clock source board.

Figure 24. Si5341 Clock Controller

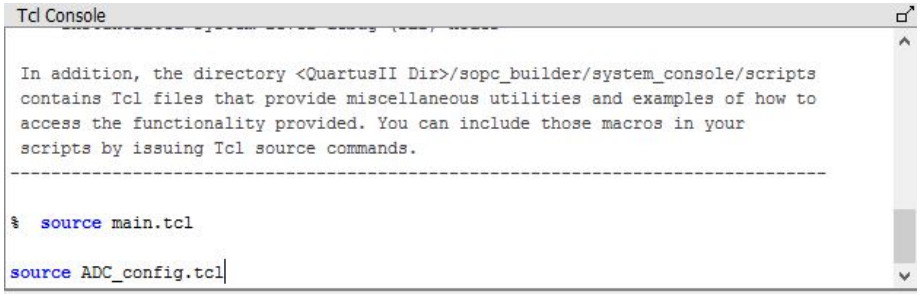


3. After providing the device clock to the FPGA, program the FPGA using the SOF file. Then, provide 5000 MHz of sampling clock to the ADC through the external clock source board. In this case, ADF4355 is used as the external clock source board to generate sampling clock to the ADC.

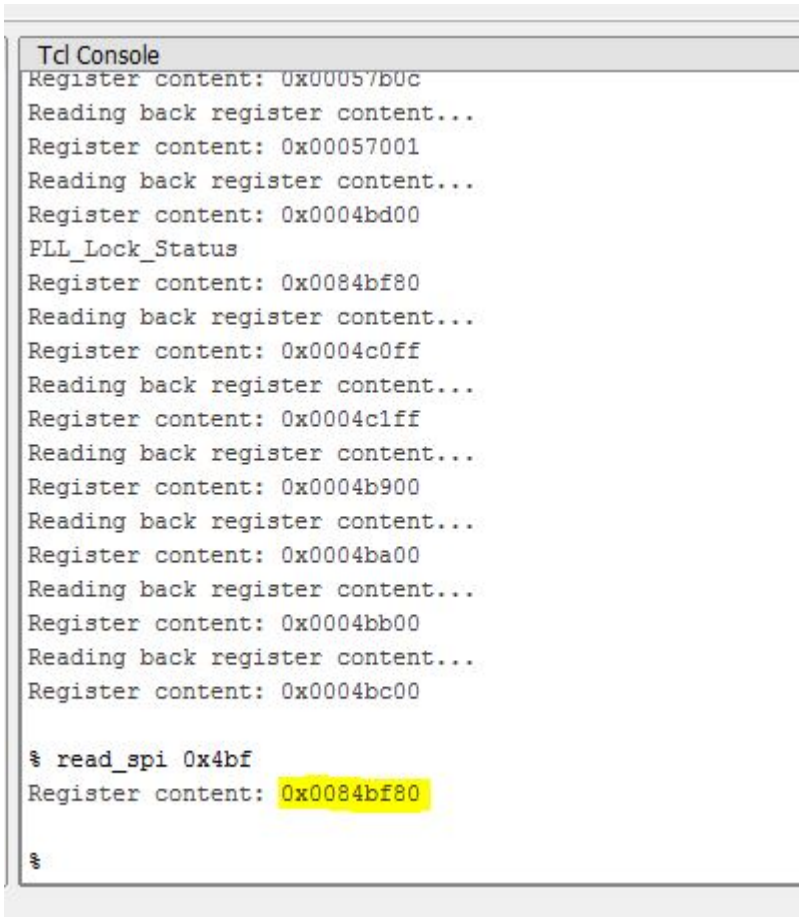
Figure 25. ADF4355 External Clock Source



4. After programming the clock source and SOF, in the Intel Quartus Prime Pro Edition software, select **Tools ➤ System Debugging Tools ➤ System Console** to launch the system console.
5. In the Tcl console pane, type `cd <full_path>` to change directory to your project folder. If your project is already opened in the Intel Quartus Prime Pro Edition software, the default path of the system console is your project folder.
6. All the examples shown are mainly for the mode where L is 16, lane rate is 5 Gbps, and ADC Sampling clock is 5000 MHz.
7. Type the following commands in sequence and check the link up in the Signal Tap Logic Analyzer.
   - `source main.tcl`
   - `source ADC_config.tcl`



8. After the successful link up, check the ADC PLL lock status by reading the register `0x04BF`. It should be read as `0x80` if the link up is successful.



9. Monitor the parity error output signal (`c2c_parity_err_out`) in the Signal Tap c2c instance for `0x000000h`, if there is no parity error.

Figure 26. C2C Data Capture

10. Calculate the latency measurement in the design by injecting a pulse to the ADC from the FPGA. The difference in clock counts between the pulse injected and the pulse observed at the C2C output gives the exact latency in the design. Refer Latency Measurement for 5G Design on page 26 for the detailed description.

Figure 27. C2C Hardware Design Latency

1.11.6. Latency Measurement for 5G Design

The latency measurement in the low latency chip-to-chip (C2C) interface is performed based on the following specified sequences for 5G design:

- Latency pulse is generated in the FPGA (SMA J64) and it is fed as an input to the J2 VIN of the ADC AD9217 EVM board through the SMA cable.

Figure 28. Test Setup—Latency Measurement for 5G Design

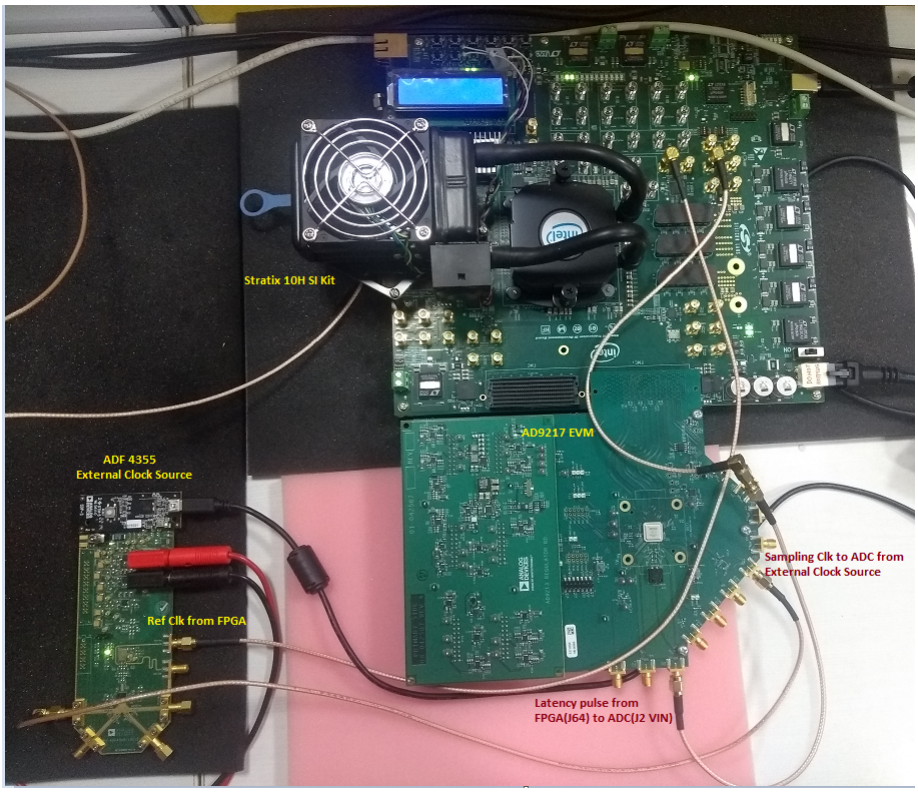
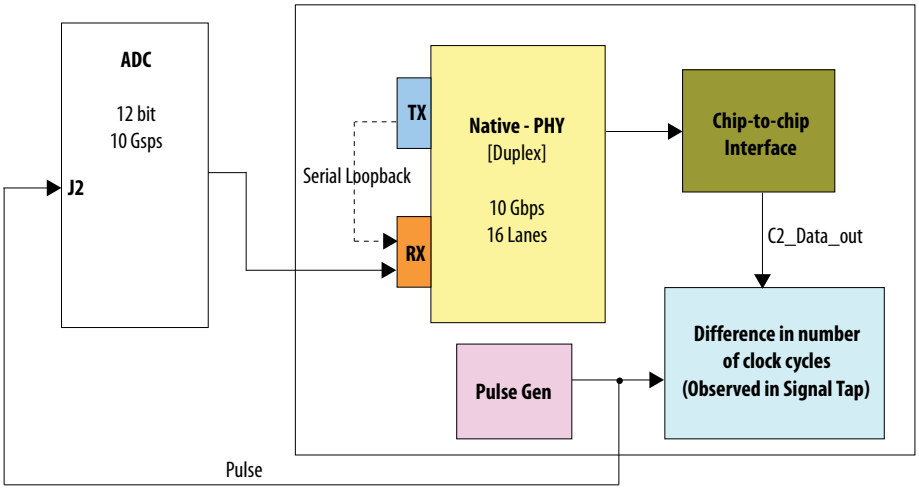
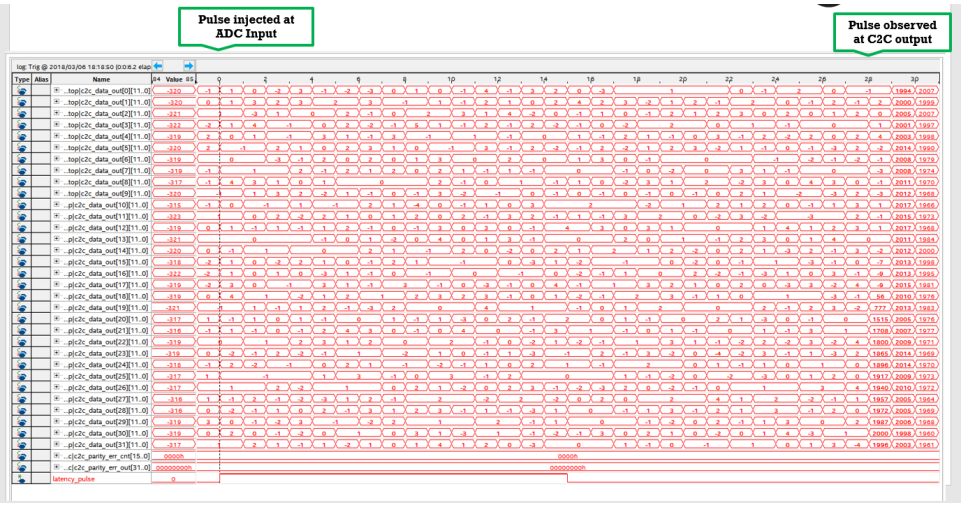


Figure 29. Block Diagram—Latency Measurement



- The pulse is injected at the input of ADC(AD9217) and is observed at the C2C output in the FPGA.
- The number of clock cycles between the pulse injected is observed at the C2C output, which is referred to as Chip-to-Chip latency.
- In Figure 30 on page 28, the latency pulse is injected at the 0th clock cycle and the latency pulse in C2C data output is observed at the 28th clock cycle.

Figure 30. Latency Measurement for 5G Design



1.11.6.1. Latency Calculations

Speed	FPGA Clocking (MHz)	Latency on User Interface (C2C) (ns)	Latency on Transceiver Parallel Data (Native PHY FPGA Interface) (ns)
5GT/s (5GSps)	156.25	179.2	140.8

For designs with the lane rate of 5 Gbps:

- PHY receiver clock = (Lane rate/32 = 5000e6/32) = 156.25 MHz.
- Clock period of 156.25 MHz = 6.4 ns.
- Number of clock cycles between pulse injected at the ADC input to pulse observed at C2C output – 28 clock cycles of PHY receiver clock.
- Latency measured at the C2C output = 6.4 ns * 28 = **179.2 ns**.
- Since the C2C design in the FPGA has a constant delay of 6 clock cycles, this can be excluded from overall latency so that the latency can get up to receive transceiver.
- Latency measured at the transceiver output = 6.4 ns * (28-6) = **140.8 ns**.