XAPP876 (v1.0.1) February 22, 2010

Virtex-5 FPGA Interface to a JESD204A Compliant ADC

Author: Marc Defossez

Summary

This application note describes how to interface the Virtex®-5 LXT, SXT, TXT, and FXT devices featuring GTP/GTX transceivers to an analog-to-digital (ADC) converter compliant to JEDEC Standard No. 204A (JESD204A) *Serial Interface for Data Converters* [Ref 1]. With some restrictions that are highlighted in the text, this application note can also be used for ADC devices compliant to the older JESD204 standard.

Introduction

The JESD204A standard describes a serialized interface between data converters and logic devices. It contains normative information to enable the implementation of designs that communicate with devices covered by the JESD204A standard. This application note discusses the implementation of a two-lane dual ADC with each lane having a 14-bit resolution and running at 125 MSPS. It provides an overview of how to implement the serial data interface and the link protocol described in the JESD204A standard. Although some implementation modes are discussed in this application note, not all possible implementation modes are provided in the accompanying reference design.

The JESD204A standard describes the protocol for implementation with general high-speed SERDES devices. The Virtex-5 TXT device contains GTX transceivers. The JESD204A standard is interpreted accordingly, and a compliant interface is delivered for GTX transceivers.

Figure 1 shows a comparison between the JESD204A standard and the older JESD204 standard. The implementation described in this application note is for a single device containing two converters (M), using one link of two lanes (L) connected to the FPGA. For completeness, the FPGA is always assumed to be a single device.

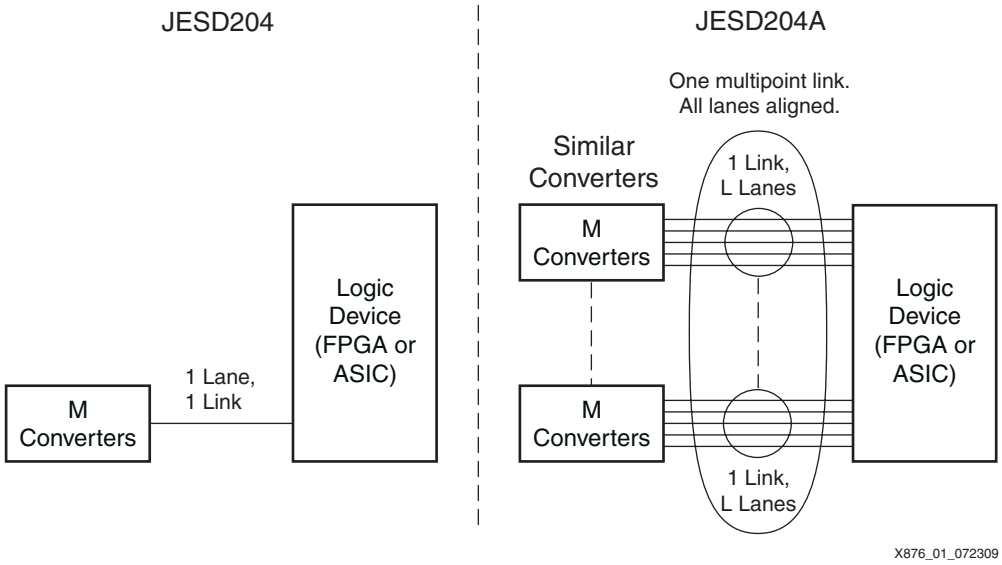

Figure 1: **Comparison of JESD204 and JESD204A Standards**

GTP Transceiver Clocking

In all current ADCs with a serial low-voltage differential signaling (LVDS) interface, the sample rate of the converter determines the communication speed between the converter and the interface FPGA. Equation 1 can be used to calculate the DDR bit clock between the converter and the interface FPGA.

$$F_{adcbitclk} = \frac{Sample\_clock \times Adc\_Resolution}{2} \quad \text{Equation 1}$$

For example, Equation 2 determines the clock speed a 125 MSPS, 12-bit ADC with a single serial LVDS interface communicates with an interface FPGA.

$$F_{adcbitclk} = \frac{125 \times 12}{2} = 750 \text{ MHz} \quad \text{Equation 2}$$

To increase the sample rate of the converters and still provide a workable LVDS interface converter, manufacturers split the communication between the converters and the FPGA into multiple lanes.

The converter needs a precision sample clock for analog signals. This sample clock is used as a reference clock for high-speed transceivers built into the converter. The JESD204 standard is written so that the sample rate of the converter is called a frame clock. At the high-speed transceiver side of the converter, the frame clock is used to generate two other clocks: a character clock and a line clock.

The frame clock is the converter's sample clock. Each frame clock cycle produces *n* data bits, where *n* is the resolution of the converter. These *n* samples are grouped into octets (bytes). The resolution of the converter can cause some octets to be used only partially. The transceiver uses the 8B/10B technique to transfer data. The byte arranged samples are converted to 10-bit values according to the 8B/10B valid character list. The line clock is the effective transmission clock and is therefore ten times the character clock.

Table 1 shows the relationship between the clocks, resolution, and channels of a converter device. This relationship is independent of multi-lane configurations and of the possible ways to pack the sampled data into octets.

Table 1: **Relationship between Converter Parameters and Clock Rates**

ADC Parameters		Data Frame (Octets)	Clocks	
Channels	Resolution (Bits)		Character Clock (MHz)	Line Clock (MHz)
1	12	2	160	1600
2	12	3	240	2400
4	12	6	480	4800
1	14	2	160	1600
2	14	4	320	3200
4	14	7	560	5600
1	16	2	160	1600
2	16	4	320	3200
4	16	8	640	6400

For example, assuming that all converters run at 80 MSPS, the frame clock is then 80 MHz. The Data Frame column of Table 1 is the product of the first two columns, Channels and Resolution (Equation 3):

$$\text{Data\_Frame} = \text{Channels} \times \text{Resolution} \quad \text{Equation 3}$$

The Character Clock column in Table 1 is the product of the data frame and the frame clock (Equation 4). The frame clock in this case is 80 MHz.

$$\text{Character\_Clock} = \text{Data\_Frame} \times \text{Frame\_Clock} \qquad \textit{Equation 4}$$

The line clock is ten times the frame clock because ten characters are transmitted for every octet over the serial connection.

The different methods of packing sampled data into octet boundaries and the possible gain or loss in transmission efficiency are discussed in Data Transport. Table 1 demonstrates that for a certain combination of channels and resolution, one GTP transceiver lane is not sufficient. Multi-lane converters—the same solution used for serial LVDS based converters—must be used. Figure 2 displays the conversion from sampled data to transmitted data.

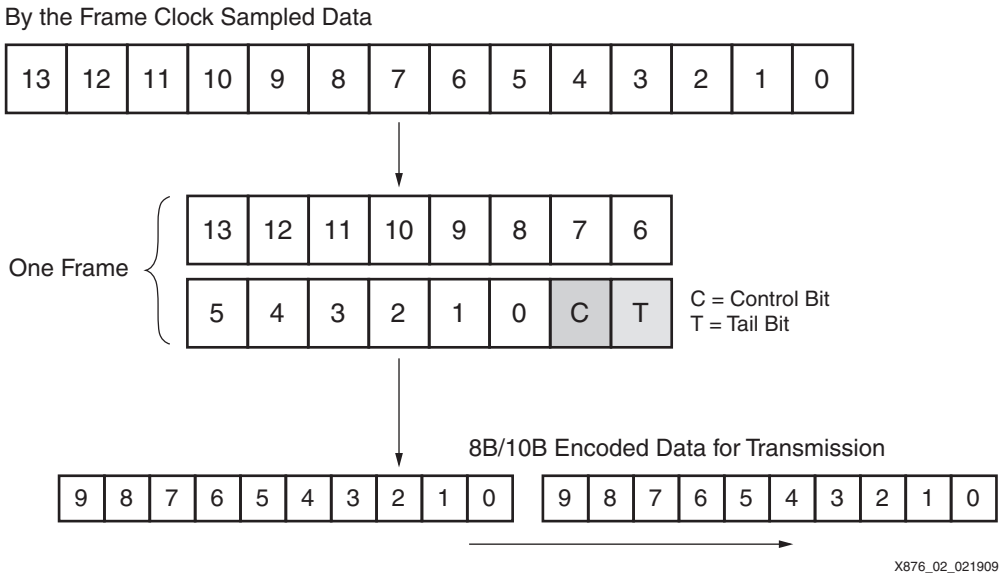

Figure 2: **Conversion of Sample Data to Transmission Data**

In the JEDEC standard, the frame clock is used as a reference clock to interface to the high-speed transceiver. The high-speed transceiver devices recover clock and data from the incoming data stream. Therefore, it is not necessary for the converter to deliver a clock and frame signal to capture the data into the interface, as it is with LVDS based solutions.

Only one pair of connections exists per channel between the converter and the GTP transceiver. Thus, traces only have to be matched per channel. This leads to several possibilities for component, PCB, and device (apparatus) solutions:

- Layout of traces between the converter and the FPGA can be simplified. Only two matched traces (one each for the P and N side) are needed per channel instead of the six traces (for the bit clock, frame clock, and data differential traces) required in a one-channel serial LVDS-based converter.
- The converter(s) and the FPGA can be placed on different PCBs.
- The converter PCB and interface FPGA PCB can be assembled in different cabinets.

Note: Refer to the skew budget number given in the JESD204A standard.

Several clock connection possibilities are shown in Figure 3, Figure 4, and Figure 5. Figure 3 shows a setup similar to the serial LVDS based converters. The converter requires a high-precision, low-jitter clock to sample the analog signals and delivers a digitized version of the clock for the FPGA interface.

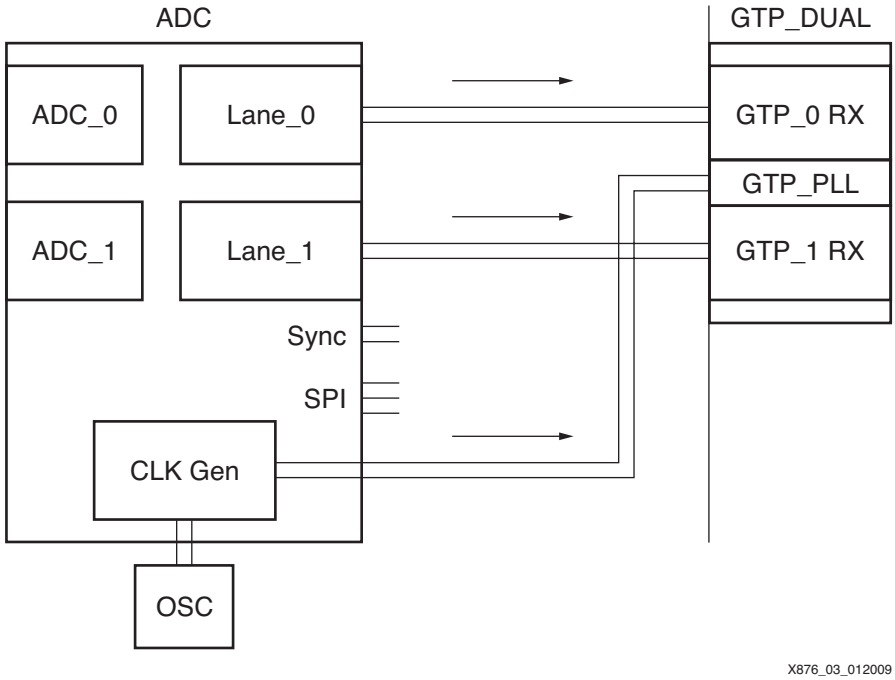

Figure 3: **Converter Delivers the Reference Clock for the Interface**

The setup in Figure 3 is useful when the converter and FPGA are placed on the same PCB. Only one clock connection to the FPGA is needed regardless of the number of converter channels. Inside the FPGA, the reference clock is spread via dedicated routing amongst the different GTP_DUAL tiles (Figure 4).

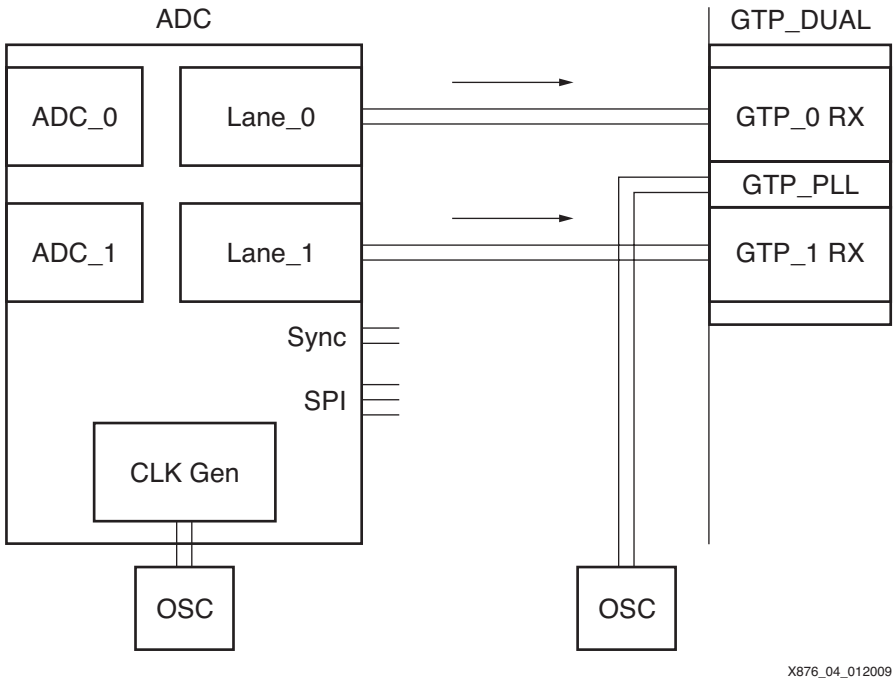

Figure 4: **Reference Clock Usage of Converter and Interface**

Both oscillators (OSC) in Figure 4 must have the same frequency. When the ADC has a sample clock rate below the minimum required reference clock rate for the GTP transceiver, the setup can be modified to use an oscillator running at the character clock frequency. This dual-oscillator setup is a good solution for these cases:

- The converter and interface FPGA are placed on different PCBs.
- The ADC and FPGA are placed so far apart on the same PCB that it is not possible to match the length, impedance, and other parameters.
- Different converters with different but related sample frequencies are connected to one interface FPGA. The phase-locked loop (PLL) in each of the GTP_DUAL tiles of the FPGA is capable of generating the correct high-speed clocks for capturing the data.

The setup shown in Figure 5 is useful when only one clock oscillator is used with multiple converters and one FPGA.

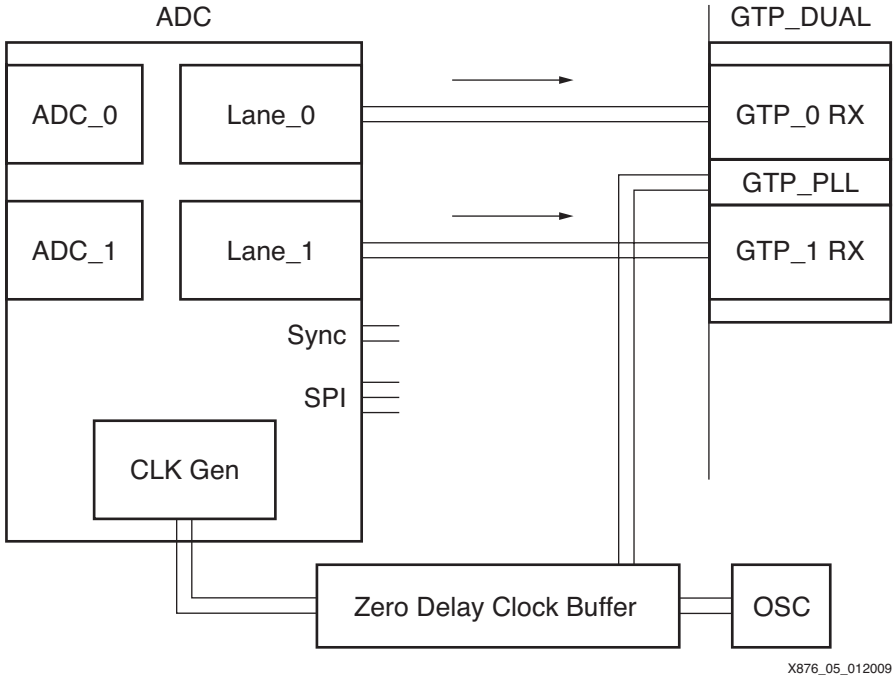

Figure 5: **Clocking Solution with Zero Clock Delay Buffer**

Reference Design

The reference design uses a dual-converter device with 14-bit resolution and a sample rate of 125 MSPS. The additional clock specifications are:

- The sample and frame clocks are both 125 MHz.
- Two lanes are used. Table 1 shows that, for the required resolution and speed, a single-lane solution does not work.
- The two converters each use one lane. This is equivalent to two single-lane devices operating according to the JESD204A standard.
- A 14-bit resolution ADC requires 2 octets.
- With the ADC sampling at 125 MHz, the frame rate must be equal to 2 octets × 125 MHz = 250 MHz. Because each octet is transmitted as a 10-bit word, the line rate must be 10 × 250 MHz = 2.5 GHz.

Equation 5 gives the settings for the PLL inside the GTP_DUAL tile.

$$PLL\_Clock = CLKIN \times \frac{PLL\_CLKDIV\_FB \times DIV}{PLL\_CLKDIV\_REF} \qquad \text{Equation 5}$$

Substituting the reference design default values for PLL_Clock, CLKIN, and DIV gives Equation 6.

$$2500 = 125 \times \frac{PLL\_CLKDIV\_FB \times 5}{PLL\_CLKDIV\_REF} \qquad \textit{Equation 6}$$

The parameters to solve for in Equation 6 are PLL_CLKDIV_FB and PLL_CLKDIV_REF. PLL_CLKDIV_FB can be 1, 2, 3, 4, or 5, and PLL_CLKDIV_REF can be 1 or 2. The ratio between the required frequency and the input frequency is 20. The numerator of Equation 5 must already be multiplied by the fixed value of 5. Because 20 divided by 5 is equal to 4, the only good selection for the parameters is:

PLLCLKDIV_FB = 4

PLL_CLKDIV_REF = 1

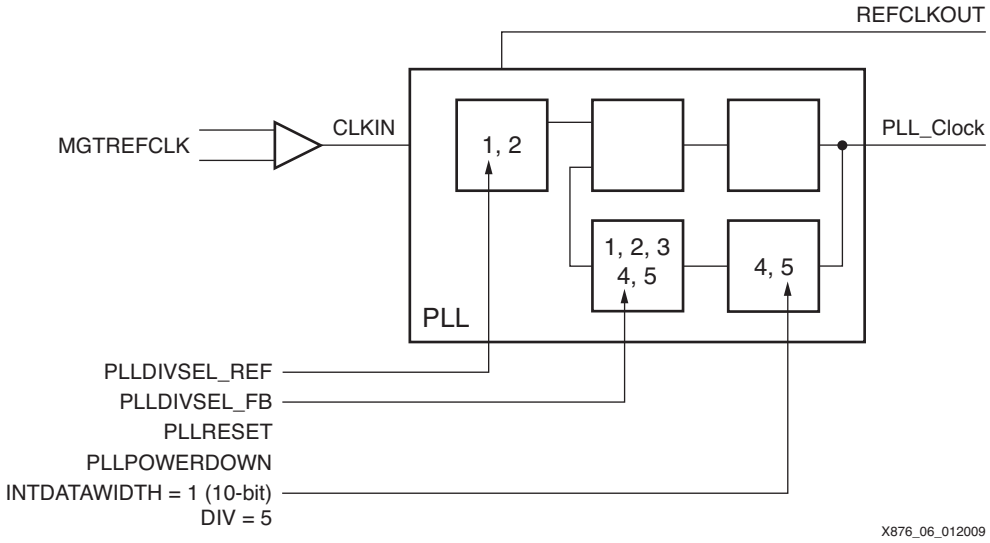

Figure 6: **GTP_DUAL Tile Shared PLL Settings**

The RX PMA operates on both edges of the high-speed clock generated by the PLL in the GTP_DUAL tile. Therefore, the generated clock must be divided by two, as shown in Figure 7.

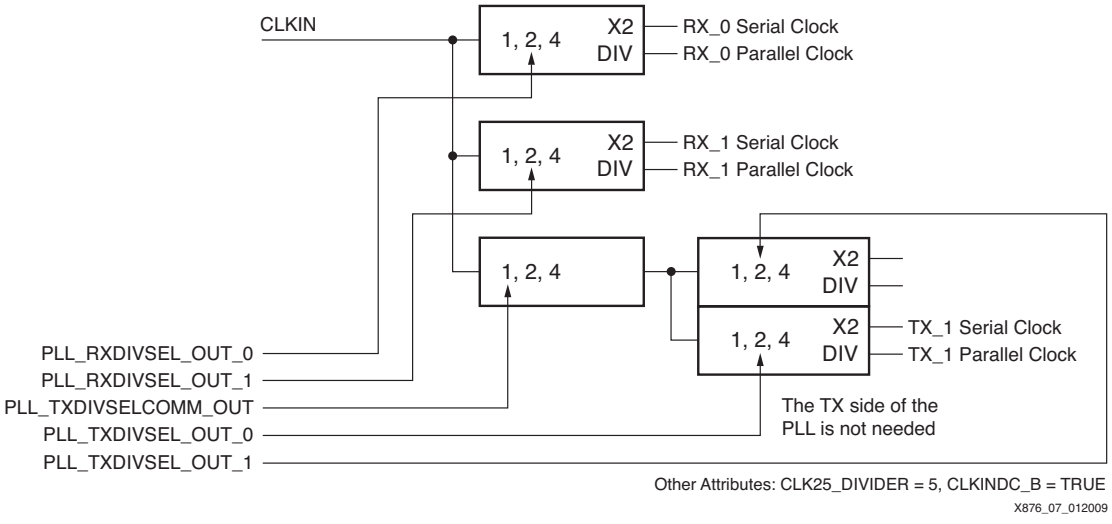

Figure 7: **PLL Output Clock Dividers**

Interface Clocking

The PLL in the GTP_DUAL tile ensures that data can be received in the correct order. From the frame clock, the PLL generates the high-speed bit clock used at data transmission as well as the clocks necessary to pass the serially received bits into the FPGA logic.

The GTP transceiver cannot perform all the functions described in the JESD204A standard. Therefore, some logic build is needed inside the FPGA behind the GTP_DUAL tile. This logic build is also needed for the application running behind the whole interface to be able to retrieve data from storage such as block RAM or distributed RAM.

Thus, the reference clock (frame clock) supplied to the GTP transceiver must be passed to a PLL or digital clock manager (DCM) inside the FPGA so that clocks can be generated for the necessary logic. The PLL implementation in the GTP transceiver has a direct clock output, REFCLKOUT, reflecting the input reference clock. This direct clock output of the GTP transceiver can be routed to a clock management tile (CMT) and used by the PLL or one of the DCMs of the CMT (Figure 8).

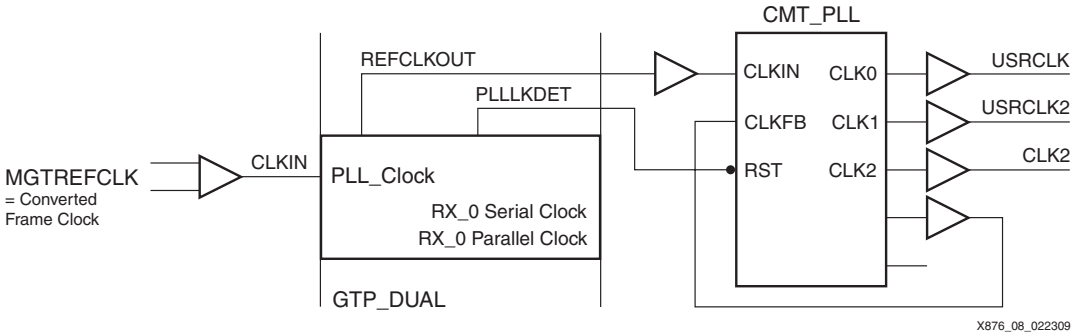

Figure 8: **GTP Transceiver and Logic Clocking**

GTP_DUAL Tile Parameters

The parameters of the GTP_DUAL tile are:

- The GTP transceiver reference clock is 125 MHz or the sample clock of the ADC.
- The PLL_Clock generated in the GTP_DUAL tile is 2.5 GHz.
- The RX serial clock is half the PLL clock rate because the deserializer samples on both edges of the clock.
- The parallel clock, equal to the character clock of the JESD204A standard, is 250 MHz. The clock is 250 MHz because two bytes or octets make one data frame.

The PLL in the Clock Management Tile (CMT) must be set to match the character clock because the output data of the GTP transceiver RX interface is set to the byte-wise configuration mode. The input clock of the CMT_PLL is a copy of the sample clock or 125 MHz. The calculations of the CMT_PLL output clocks for a -3 speed grade Virtex-5 device are:

F_{VCOMax} = 1,440 MHz

$F_{VCO} = F_{CLKIN} \times (M/D) = 125 \times (10/1) = 1,250$ MHz

$F_{OUT0} = F_{VCO}/O = 1,250/5 = 250$ MHz

$F_{OUT1} = F_{VCO}/O = 1,250/5 = 250$ MHz

$F_{OUT2} = F_{VCO}/O = 1,250/10 = 125$ MHz

The calculations of the CMT_PLL output clocks for a -2 speed grade Virtex-5 device are:

F_{VCOMax} = 1,200 MHz

$F_{VCO} = F_{CLKIN} \times (M/D) = 125 \times (8/1) = 1,000$ MHz

$F_{OUT0} = F_{VCO}/O = 1,000/4 = 250$ MHz

$F_{OUT1} = F_{VCO}/O = 1,000/4 = 250$ MHz

$F_{OUT2} = F_{VCO}/O = 1,000/8 = 125$ MHz

The CMT_PLL output_0 (USRCLK) and output_1 (USRCLK2) are the same because they are chosen for an 8-bit data output. The CMT_PLL also generates the sample clock (CLK2) for the application that uses the reassembled frame data.

Data Transport

The JESD204A standard describes the mapping of data for different converter setups and interface devices. These setups are possible:

- A single converter to a single-lane link
- A single converter to a multi-lane link
- Multiple converters in the same device to a single-lane link
- Multiple converters in the same device to a multi-lane link (used in this application note)

A method of grouping sampled data into octets has been developed to provide a solution to these converter-to-interface setups. This method is referred to as F for the remainder of this application note.

In many applications, the frame clock has the same frequency as the sample clock. A data sample and/or a partial sample is grouped into a frame of F octets. However, JESD204A allows more than one sample per converter to be transmitted in one frame cycle. This is represented by the number S as samples per converter per frame cycle and must always be an integer.

Each sample, converted to octets, is transmitted as a group of N' bits consisting of N data bits together with optional control and tail bits. Additionally, tail bits at the end of the frame might be necessary to fill a whole number of octets per lane per frame cycle. The converter parameters defined up to this point are given in Table 2.

Table 2: **First Set of Converter Parameters**

Parameter	Description	Range
F	Octets per frame	1–256
S	Samples per converter per frame cycle	1–32
N	Converter resolution	1–32
N'	Total number of bits per sample	1–32

Single-Lane Format

One device can contain multiple converters. The number of converters is referred to as M. The M converters all produce data samples with a length of N bits. These samples are then converted into octets and transmitted using 8B/10B encoding. Figure 9 shows the mapping of data samples to octets and lane data.

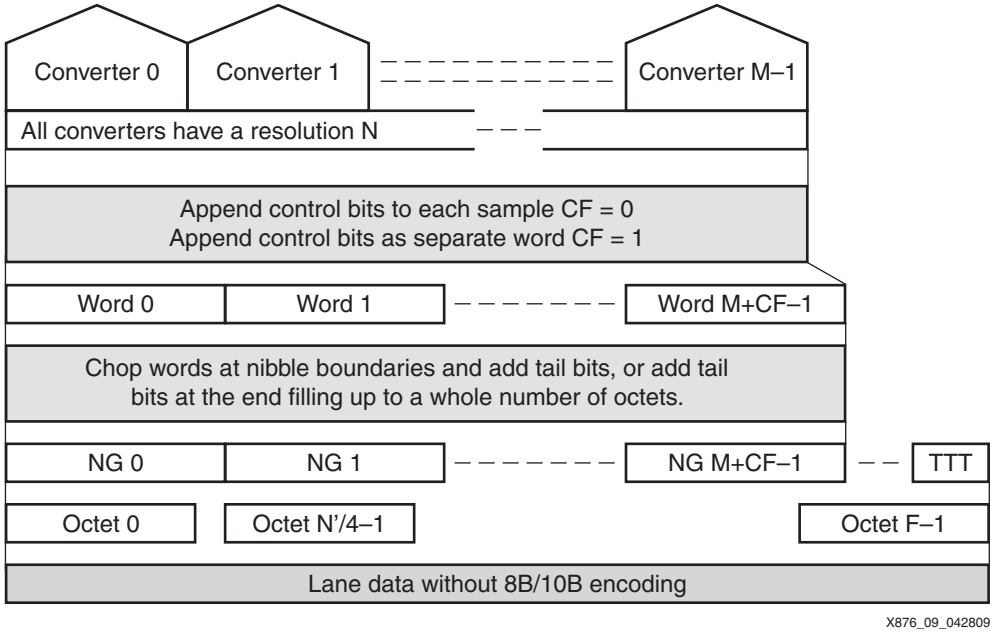

Figure 9: **Data Format for a Single Lane**

The mapping process occurs as follows:

1. Starting with converter 0, the N-bit samples of all M converters are mapped to a linear axis until all samples have been mapped.
2. The samples are mapped to words. When the samples contain no control bits (out-of-range or other indication), the words are identical to the samples. When sample-specific control bits are available, either of these two options can be done, as specified in the JESD204 standard:
   a. A relevant control bit is appended after the LSB of each conversion sample (CS), as shown in Figure 10. This figure shows a single-lane, four-converter, 12-bit resolution device with its required parameters.

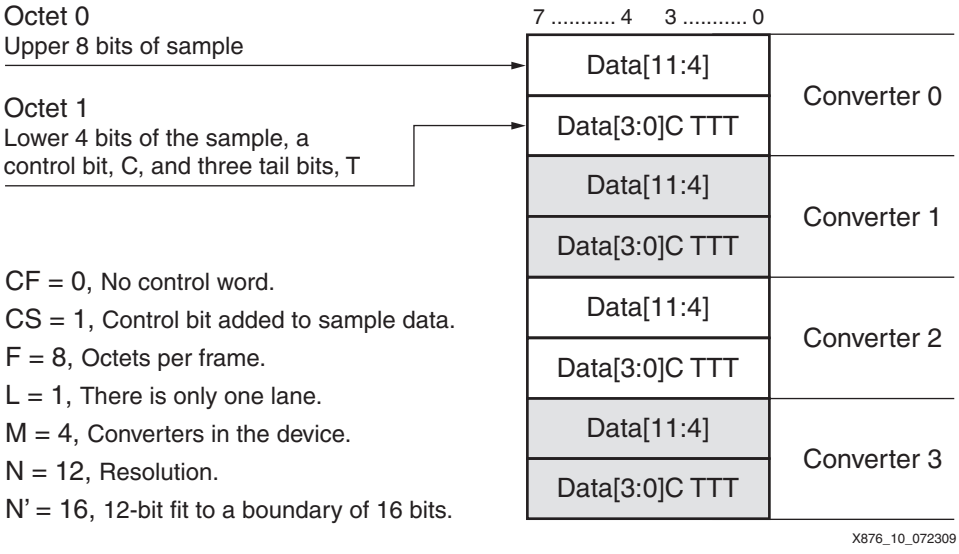

Figure 10: **Single-Lane Converter with Control Bit without Control Word**

   b. The control bits are grouped into a separate control word that is appended after the data samples. The first bit(s) of the control word correspond(s) to the control bit(s) of

converter 0, the next bit(s) in the control word correspond(s) to the control bit(s) of converter 1, and so on. Figure 11 shows a single-lane, four-converter, 14-bit resolution device with its required parameters.

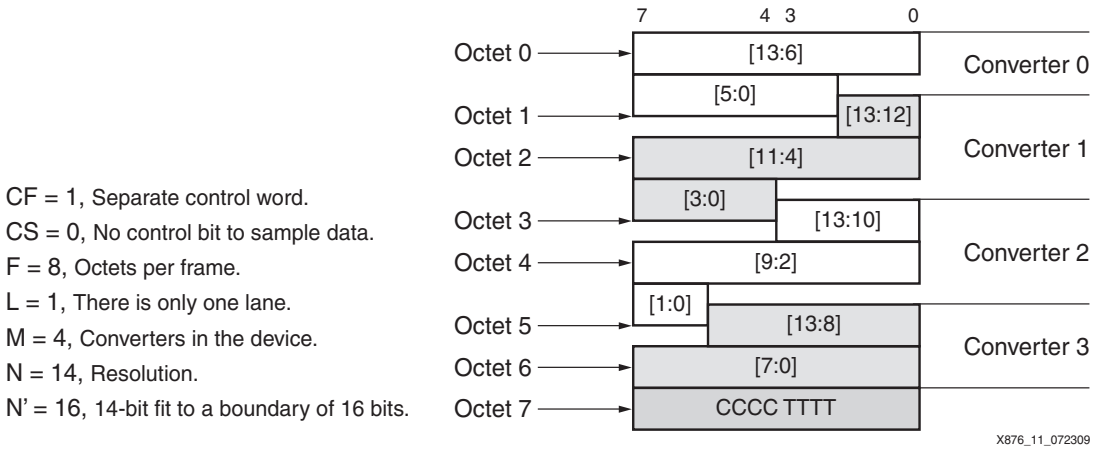

Figure 11: **Single-Lane Converter for Increased Line Efficiency with Control Word**

3. The control words in a frame are indicated by a parameter named CF. CF = 0 if there is no control bit to sample the data. CF = 1 if control bits are in a separate control word. The total number of words transmitted per frame cycle is M + CF.
4. Words not containing a whole multiple of 4 bits are extended to the smallest possible nibble group (a group of half octets) using tail bits. The extended words are indicated by NG in Figure 9 (an example of this is shown in Figure 10).
5. (Optional) This step increases line efficiency by prioritizing it against easier data sample mapping (Figure 11).
6. A conversion word can be extended by control bits, tail bits, or both to a length of N' ≥ N bits, where N' is a whole multiple of 4.
   a. Tail bits can be appended to make the total number of bits after the last step an integer multiple of 8.
   b. The sequence obtained is divided into octet boundaries. The result is F octets.

The converter parameters defined in this section are given in Table 3.

Table 3: **Additional Converter Parameters**

Parameter	Description	Range
M	Number of converters in the package	1–256
CF	Control word per frame clock per cycle	1–32
CS	Control bits per sample	1–3
T	Tail bit	1

Multi-Lane Format

For a link consisting of L lanes, the mapping method described in Single-Lane Format is used for a single lane. Instead of putting all the data into one SERDES lane, step 6 above spreads data over L × F octets. The first F octets are transmitted over lane 0. The last F octets are transmitted over lane L – 1.

To make it possible to have a high lane efficiency, a new parameter named High Density (HD) is introduced. When HD = 0, low density mode is used. This means that partial conversion words at the end of a group of F octets are avoided by adding more tail bits after the last full

nibble group in the group. In the high density mode (HD = 1), the conversion words might break at the frame boundary on the mapping axis. A multi-lane converter setup is shown in Figure 12.

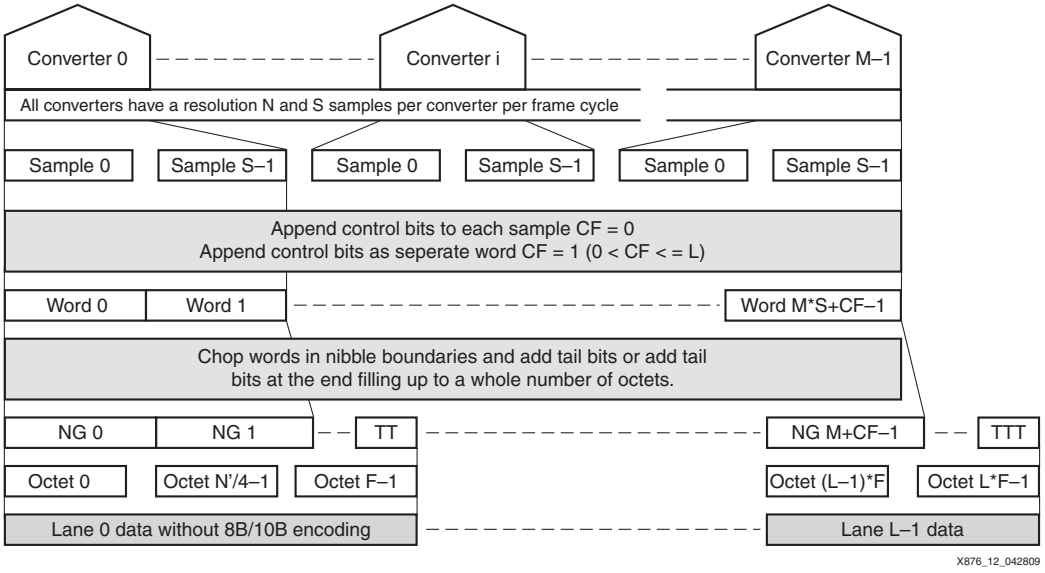

Figure 12: **Multi-Lane Converter Setup**

Figure 13 shows a practical example of data-mapping in a multi-lane setup with four 14-bit converters. In this example, the device is first mapped over two lanes using the low-density method (HD = 0), and then using high-density mapping (HD = 1).

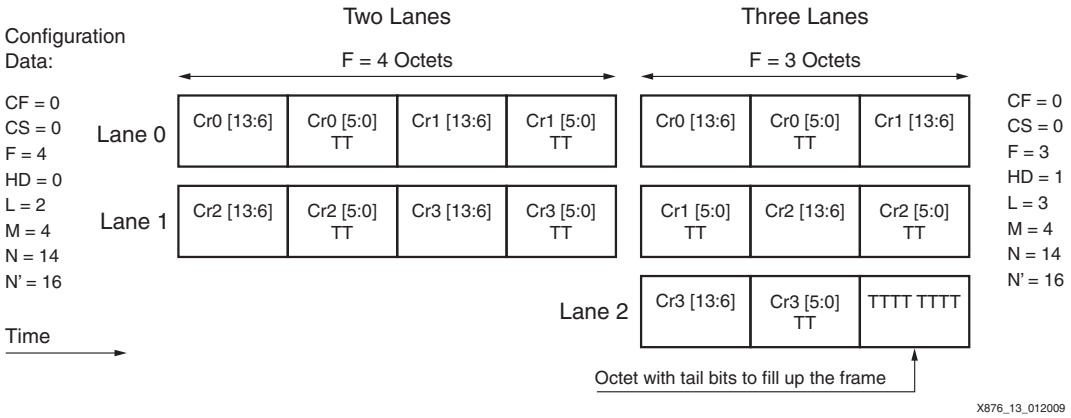

Figure 13: **Data Mapping in a Multi-Lane Setup**

Table 4 defines the parameters introduced in this section.

Table 4: **Additional Parameters Not Defined in Table 2 and Table 3**

Parameter	Description	Range
L	Number of high-speed lanes per package	1–32
HD	Data packaging format, high density or low density	0, 1

Table 5 summarizes the parameters defined in Table 2, Table 3, and Table 4, together with other parameters defined by the JESD204A standard.

Table 5: **All Link Configuration Parameters**

Parameter	Description	Range	Field
BID	Bank ID	0–15	[3:0]
CF	Control words per frame clock cycle per link	0–32	[4:0]
CS	Control bits per sample	0–3	[1:0]
DID	Device ID	0–255	[7:0]
F	Octets per frame	0–256	[7:0]
HD	High-density format	0–1	[0]
K	Frames per multi-frame	1–32	[4:0]
L	Lanes per converter device	1–32	[4:0]
LID	Lane identification	0–31	[4:0]
M	Converters per device	1–256	[7:0]
N	Converter resolution	1–32	[4:0]
N	Bits per sample	1–32	[4:0]
S	Samples per converter per frame cycle	1–32	[4:0]
SCR	Scrambling enabled	0–1	[0]
RES1	Reserved	0–255	[7:0]
RES2	Reserved	0–255	[7:0]
FCHK	Checksum of all fields (mod256)	0–255	[7:0]

Lane Format Conclusion

Several options are available to code the sampled data of the converter into bytes for transmission by high-speed transceivers. The higher the sample speeds of the converters, the less likely that a single high-speed transceiver lane is the solution.

The converter manufacturer decides the number of transceiver lanes in the converter and how the data is mapped in these lanes. It is very common for a converter device to support only one data assembly implementation format (i.e., a subset of the JEDEC specification).

The receiving interface must therefore adapt to the given device specifications. This information is available in the converter data sheet or user guide and is also transmitted by the converter at operation using the parameters shown in Table 5. If some of these parameters can be changed, the converter device will likely have an SPI or I2C interface to do so.

In conclusion, no single FPGA interface solution applies to all ADC devices.

Reference Design

The reference design uses a dual-converter device with 14-bit resolution and a sample rate of 125 MSPS. It has these specifications:

- Sample clock = frame clock = 125 MHz.
- Row 5 of Table 1 shows the ADC parameters, data frame, and clocks for a single-lane solution. This shows that multiple lanes are needed because the line rate is too high.
- Two lanes are used.
- Each of the two converters uses one lane. This gives the appearance of two single-lane devices.
- A resolution of 14 bits fits in a 16-bit boundary, or 2 octets.

- A control bit is used by the converter to control overflows. Therefore, the frame setup appears as shown in Figure 14.
- The converter uses the multi-lane setup to synchronize the data of the two ADC channels.

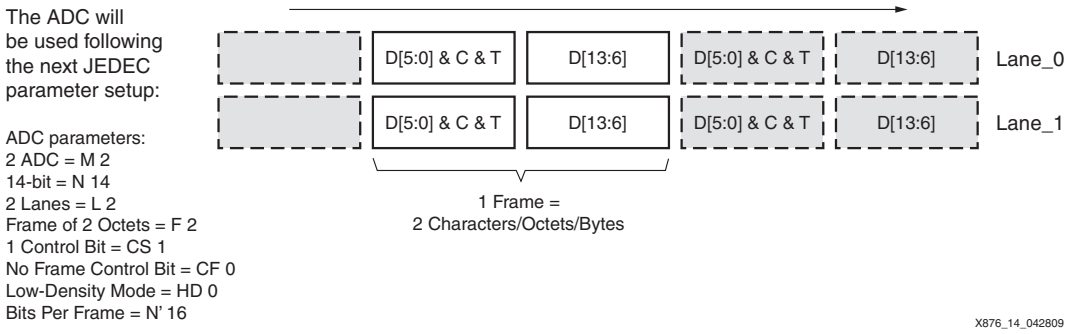

Figure 14: **Lane Organization for the Converter Used in the Reference Design**

Initial Link Synchronization

The GTP transceiver settings are configured as follows:

- The PLL of the GTP_DUAL tile is initialized.
- The datapaths in the GTP transceiver use these settings:
  - One GTP_DUAL tile is used.
  - The data arrives in 8B/10B format.
  - The data is presented at the FPGA logic in 8-bit format.
  - Comma detection is needed for link alignment.
- Channel bonding is needed to align both links.
- Oversample mode is not used.

Lane synchronization is important to allow correct functioning. Each lane must be fully synchronized and receiving valid data before the different lanes can be aligned through channel bonding, or using inter-lane alignment, as indicated by the JESD204A standard.

To synchronize data between the converters and interface (FPGA), the JESD204A standard requires a control line between the ADC and the interface device. This control line is called SYNC, as shown in Figure 15.

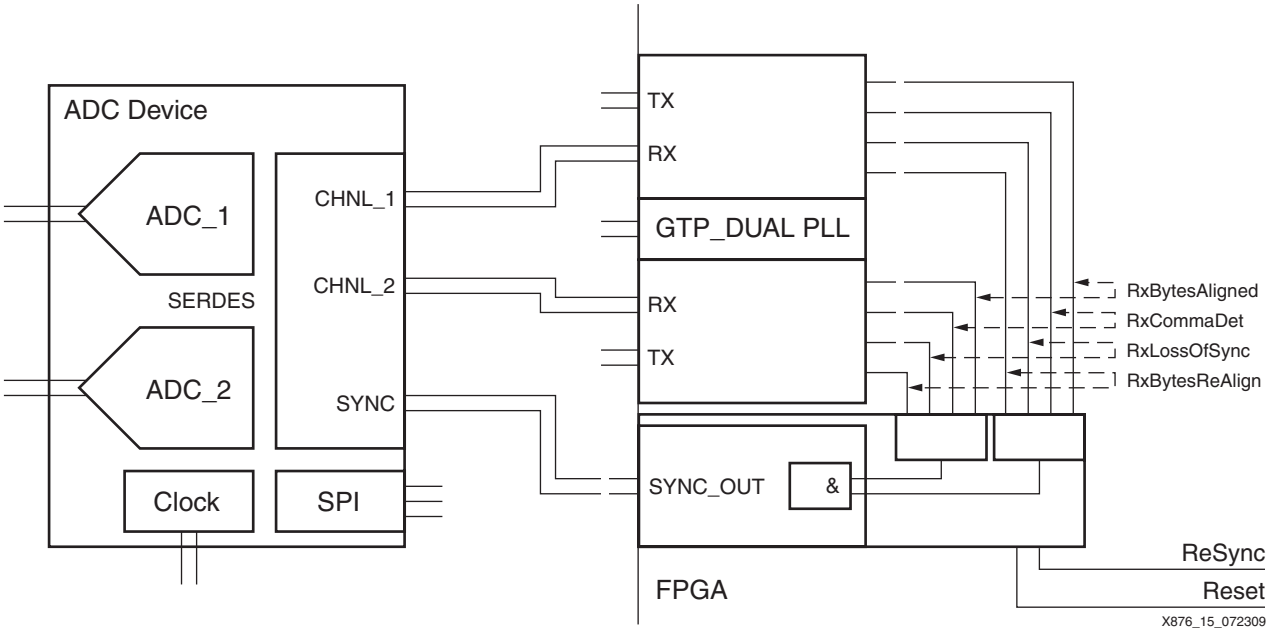

Figure 15: **The SYNC Signaling Interface**

The SYNC signal has these characteristics:

- It is an active-Low input to the ADC device.
- Only one SYNC input is used when one device contains multiple ADCs. When multiple devices are used, it is possible to have multiple SYNC signals.
- It must be synchronous with the frame clock.
- It can be a single-ended or differential signal. When the frame clock is a differential signal, SYNC is best chosen to be differential as well.

For an ADC interface, the FPGA controls the SYNC signal. When SYNC is pulled Low, the ADC device starts transmitting K28.5 characters on all lanes. The interface then synchronizes each lane so that valid K28.5 characters are detected and received. When each lane of the interface receives a minimum of four consecutive valid K28.5 characters, SYNC can be pulled High and the ADC device stops transmitting.

The JESD204A standard provides a state diagram that shows how lane synchronization must be performed. This state diagram is equivalent to the Loss-of-Sync (LOS) state machine of the GTP receiver. The interface uses the status outputs of this LOS state machine implemented in a GTP transceiver together with the RxByteIsAligned and RxCommaDet status signals.

The comma alignment and detection function of the GTP transceiver is set up to look for a K28.5 comma plus character in the incoming data stream. After the interface reset signal is released, the GTP transceiver, per RxEnPcommaAlign, starts looking for comma characters.

When the GTP transceiver hardware detects the given comma character, it pulls RxCommaDet High. When the incoming data stream is also properly aligned following byte boundaries, the RxByteIsAligned status is pulled High. At this point, at least four consecutive valid K28.5 characters should be received and detected by the LOS state machine.

When all three status signals (RxByteIsAligned, RxCommaDet, and RxByteReAlign) are satisfied for one lane, the SYNC signal of that lane is pulled High. In a multi-lane system, all separate lanes must first be properly aligned before the global SYNC signal is pulled High and the interface takes the next step.

The application running in the FPGA fabric can issue a resynchronization at any time. This forces the SYNC signal Low, and the GTP_DUAL tile waits for K28.5 character mode to be activated.

Table 6 shows the attribute settings for both channels in the GTP_DUAL tile used in this interface.

Table 6: **Channel Attribute Settings**

Attribute	Value
ALIGN_COMMA_WORD_x	1
COMMA_10B_ENABLE_x	1111111111
COMMA_DOUBLE_x	False
MCOMMA_10B_VALUE_x	1010000011 (K28.5)
MCOMMA_DETECT_x	False
PCOMMA_10B_VALUE_x	0101111100 (K28.5)
PCOMMA_DETECT_x	True (Detects only the plus comma)
RX_LOS_INVALID_INCR_x	4
RX_LOS_THRESHOLD_x	16
RX_LOSS_OF_SYNC_FSM_x	True

Notes:

1. In the above attributes, x is the GTP lane in a GTP_DUAL tile and can be 0 or 1.

These signals from the GTP_DUAL tile are monitored for both channels:

- RXBYTEISALIGNEDx: This signal is asserted when the parallel data stream is properly aligned.
- RXCOMMDETx: This signal is asserted when the comma character is detected.
- RXLOSSOFSYNCx(1): This signal is asserted when synchronization is lost.

These signals go to the GTP_DUAL tile for both channels and are controlled by logic:

- RXENPCOMMAALIGNx: Assertion of this signal turns on the alignment procedure.

Inter-Lane Alignment (Channel Bonding)

After the initial link synchronization, each lane is synchronized and receiving K28.5 control characters. Next, inter-lane alignment must be performed. Figure 16 shows the standard inter-lane alignment sequence for the JESD204A standard. At this point, the interface is in the state just before point A in Figure 16 for these reasons:

- The monitored signals for lane synchronization are asserted several clock cycles before aligned data (K28.5 in this case) is available at the GTP RX data outputs.
- After the SYNC signal is made inactive by the FPGA, several clock cycles elapse before the converted device stops transmitting K28.5 characters and new data is available at the RX data outputs.

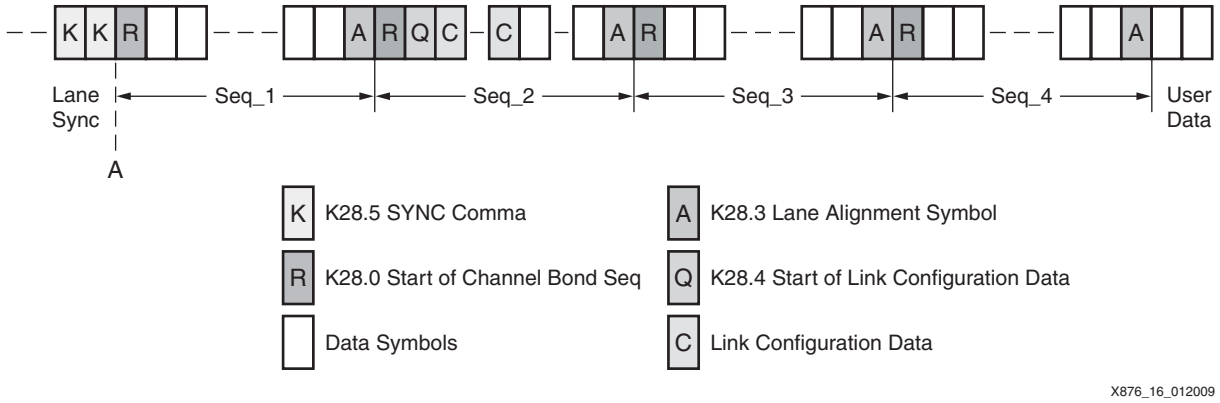

Figure 16: **JESD204A Standard Inter-Lane Alignment Sequence**

When lane alignment is done and SYNC is made inactive, the converter starts transmitting the inter-lane alignment sequence on all lanes, as shown in Figure 16.

The inter-lane alignment sequence described in the JEDEC specification allows any high-speed SERDES to synchronize different lanes. For the GTP transceiver, the processing of the sequence must be adapted so that the built-in channel bonding logic of the GTP transceiver can be used. This logic operates as follows:

1. The different GTP transceivers where channel bonding is used must be chained. One GTP transceiver functions as a master while all other transceivers function as slaves. The master GTP transceiver is the reference to which the other channels are aligned.
2. Channel bonding is performed on the data stored in the elastic data buffer of the GTP transceiver. This buffer is a FIFO written by the received data stream after it has been parallelized and read by the GTP RX interface logic. Adjusting the read pointer of all GTP transceiver components involved makes it possible to align received data.
3. Channel bonding is performed on the contents of the elastic data buffer, i.e., on 8B/10B decoded data.
4. The channels are aligned as follows:
   a. The master receives a channel bonding sequence (1 to 4 bytes long) and waits for a number of bytes (skew) before engaging channel bonding on the slaves.
   b. On the slaves, the position of the received channel bonding sequence is determined and the read pointer is modified.

   The skew is the maximum allowed delay between different channels and is typically part of the communication standard used. Skew is needed to allow the slaves to receive the channel bonding sequence. This enables the slaves to determine the distance between the master (reference channel) and the other channels to adjust the elastic buffer read pointers.

Before applying the inter-lane procedure to the channel bonding feature of the GTP transceiver, the skew budget must be determined. The RX elastic buffer is 64 bytes deep. The intra-device skew given in the JESD204A standard is 23 UI (2.3 8B/10B characters) for a single device and 68 UI (6.8 8B/10B characters) for a multi-device design. Therefore, setting the maximum skew budget for the channel bonding at eight sequences allows coverage for single- and multi-device designs.

Although not really needed for operation of the inter-lane procedure applied to the channel bonding, it can be useful to calculate the length, K, of the multi-frame. The inter-lane alignment procedure is four multi-frames long, with each multi-frame being K frames long. K is a number

between 1 and 32 such that the number of bytes per multi-frame is between 17 and 1,024. In bytes, this is expressed as Equation 7:

$$17 / F \leq K \leq 1024 / F \qquad \textit{Equation 7}$$

Where F is the number of bytes (octets) in a frame. In this reference design, F = 2 and K must thus be between 8.5 and 512. The minimum size of K is 9 frames, and this is the size used for the reference design, as shown in Figure 17.

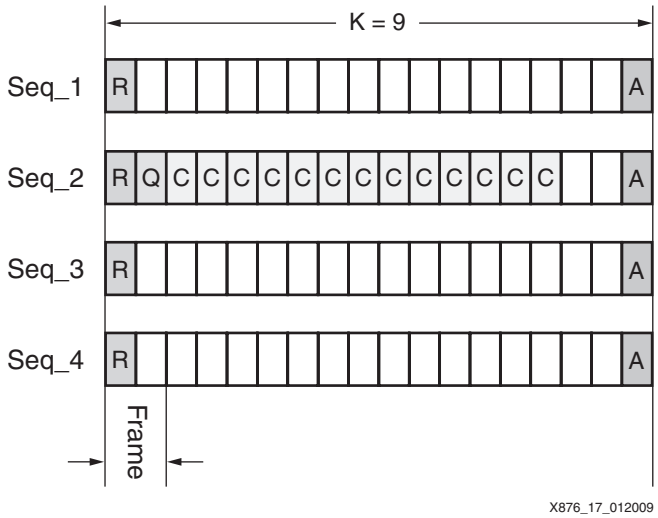

Figure 17: **Multi-Frame Alignment Procedure as Applied to the Reference Design (Two Lanes, Dual 14-Bit 125 MHz ADC)**

Table 7 contains the mapping of the link configuration data to bytes. As shown, the configuration data is 13 bytes long. An additional three control bytes (R, Q, and A) must be added to the 13 configuration bytes, for a total of 16 bytes, or 8 frames.

Table 7: **Mapping of the Link Configuration Data to Bytes**

Byte	Bits							
	7	6	5	4	3	2	1	0
0	DID[7:0]							
1					BID[3:0]			
2				LID[4:0]				
3	SCR[0]			L[4:0]				
4	F[7:0]							
5				K[4:0]				
6	M[7:0]							
7	CS[1:0]			N[4:0]				
8				N'[4:0]				
9				S[4:0]				
10	HD[0]			CF[4:0]				
11	RES1[7:0]							
12	RES2[7:0]							
13	FCHK[7:0]							

Inter-lane Alignment Procedure (JESD204A)

The inter-lane alignment procedure to perform channel bonding described in the JESD204A standard is done as follows:

- Four multi-frames are transmitted by the converter. The number of R and A control characters must be counted to determine the start of real converter data.
- After link synchronization, all lanes contain the same inter-lane alignment procedure data and each of the multi-frames looks identical, as shown in Figure 17. It is thus absolutely necessary that all lanes are aligned to the same multi-frame sequence. The two ways to accomplish this are:
  - Option 1:
    - After Sync is disabled, start looking for the first three R characters and the first two A characters on each lane.
    - When the third R character is detected on all involved lanes, enable the channel bonding for the master and use a channel bonding character, i.e., the R control character.
    - Channel bonding occurs on the last multi-frame (Seq_4 of Figure 16 or Figure 17).
  - Option 2:
    - Program a channel bonding sequence of R and Q in the GTP transceiver.
    - As soon as an R control character is detected, after SYNC is disabled, start a channel bonding operation.
    - Channel bonding is performed on the second multi-frame (Seq_2 of Figure 16 or Figure 17).

  Option 2 is easier because it requires the least logic to be developed and implemented.
- At the same time that channel bonding occurs, the master GTP RX interface looks for a Q character to register the link configuration data in distributed RAM (LUT RAM).
- The number of A characters must be counted after all channels indicate channel alignment, as shown by the status output RxChanIsAligned. Three A characters indicate the start of real converter data.

In the GTP_DUAL tile used in the reference design, GTP_0 is the master and GTP_1 is the slave. Table 8 shows the attribute settings for channel bonding.

Table 8: **Attribute Settings for Channel Bonding**

GTP Transceiver	Attribute	Setting	Description
Master	CHAN_BOND_1_MAX_SKEW_0	8	As defined in the skew budget calculation.
	CHAN_BOND_LEVEL_0	1	
	CHAN_BOND_MODE_0	MASTER	
	CHAN_BOND_SEQ_1_1_0	0 1 00011100	Regular disparity, K character, R (K28.0)
	CHAN_BOND_SEQ_1_2_0	0 1 10011100	Regular disparity, K character, Q (K28.4)
	CHAN_BOND_SEQ_1_3_0	0 0 00000000	
	CHAN_BOND_SEQ_1_4_0	0 0 00000000	
	CHAN_BOND_SEQ_1_ENABLE_0	0011	
	CHAN_BOND_SEQ_2_USE	FALSE	
	CHAN_BOND_SEQ_LEN_0	2	

Table 8: **Attribute Settings for Channel Bonding *(Cont'd)***

GTP Transceiver	Attribute	Setting	Description
Slave	CHAN_BOND_1_MAX_SKEW_0	8	As defined in the skew budget calculation.
	CHAN_BOND_LEVEL_0	0	
	CHAN_BOND_MODE_0	SLAVE	
	CHAN_BOND_SEQ_1_1_0	0 1 00011100	Regular disparity, K character, R (K28.0)
	CHAN_BOND_SEQ_1_2_0	0 1 10011100	Regular disparity, K character, Q (K28.4)
	CHAN_BOND_SEQ_1_3_0	0 0 00000000	
	CHAN_BOND_SEQ_1_4_0	0 0 00000000	
	CHAN_BOND_SEQ_1_ENABLE_0	0011	
	CHAN_BOND_SEQ_2_USE	FALSE	
	CHAN_BOND_SEQ_LEN_0	2	

Two signals from the GTP_DUAL tile are monitored for both channels:

- RXCHANBONDSEQx: This signal is asserted when RXDATA contains the start of a channel bond sequence.
- RXCHANISALIGNEDx: This signal is asserted when the channel is properly aligned.

These signals go to the GTP_DUAL tile for both channels and are controlled by logic:

- RXENCHANSYNCx: This signal enables channel bonding only on the master GTP transceiver. This signal should be tied High for slaves.
- RXCHBONDO0[2:0]: This master output should be connected to RXCHBONDI1[2:0].
- RXCHBONDI1[2:0]: This is a slave input.

Note: The x in the signal names indicates the GTP lane in a GTP_DUAL tile and can be 0 or 1.

Figure 18 shows how the basic synchronization of the channels, the channel bonding, and the extraction of device data is performed. The reference design hierarchy is constructed using the same setup.

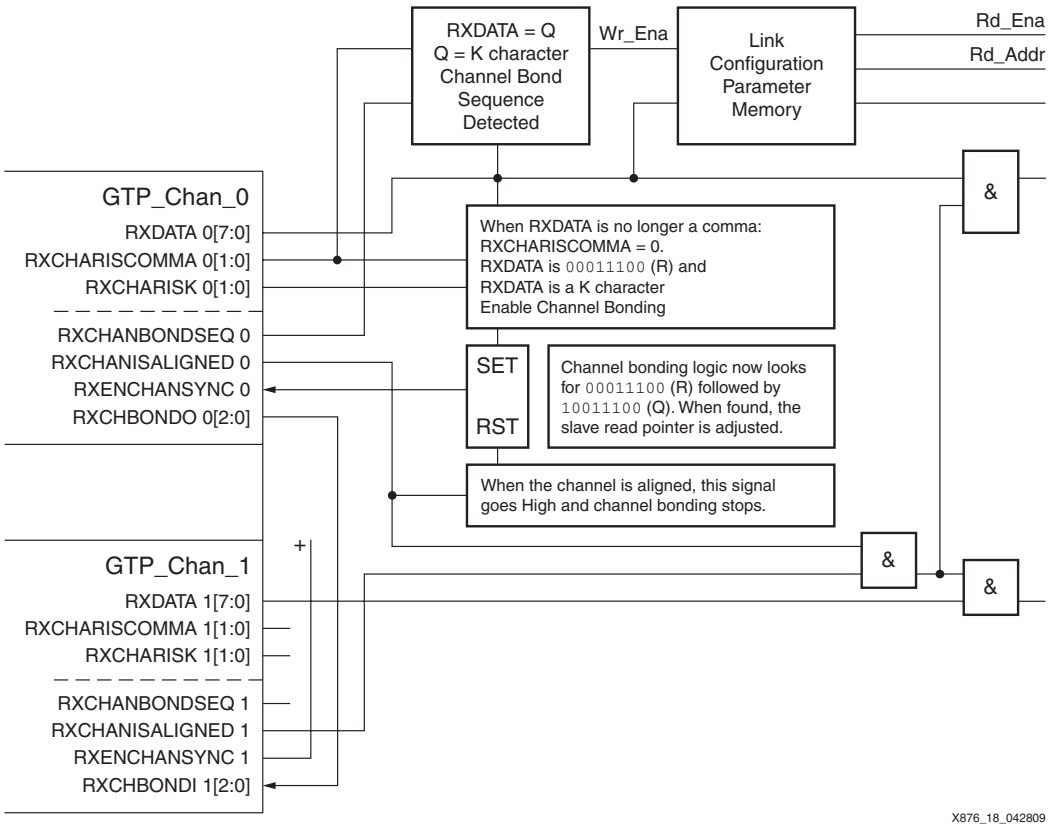

Figure 18: **Channel Bonding and Link Configuration Storage**

Note: Interfaces for devices compliant to the JESD204 standard do not need to implement this inter-lane alignment logic because only one lane is supported. The manufacturer of the converter device might be compliant to the JESD204A standard for the single-lane device. If so, at a minimum, detection and registering of the link configuration data must be performed. The moment when the real user data starts in the transmission must also must be accounted for.

Data Descrambling

After inter-lane alignment is done, real data is output on the GTP RX data ports. The SCR parameter received in the link configuration data indicates whether or not the converter is using scrambled data. To turn descrambling on or off, the link configuration data must be read and interpreted by the application.

If an SPI or I2C interface is present, the application running in the FPGA can instruct the converter to turn scrambling on or off. It is not necessary to check the configuration parameters because these reflect whether or not scrambling has been turned on.

Scrambling provides noise immunity by avoiding spectral peaks that can be produced when the same data octet repeats from frame to frame. These spectral peaks can cause electromagnetic compatibility or interference problems in sensitive applications. Spectral peaks also cause code-dependent DC offsets in the data converters via aliasing. Scrambling also makes the spectrum data-independent. This ensures that possible frequency-selective effects on the electrical interface do not cause data-dependent errors.

Equation 8 shows the scrambling polynomial set by the JESD204A standard.

$$1 + x^{14} + x^{15} \qquad \textit{Equation 8}$$

As described in the JESD204A standard, the parallel version of the descrambler is used. The descrambler is enabled when the link configuration parameter SCR is set to 1 (Figure 19). The SCR parameter is the most significant bit (MSB) at address 3. When the SCR bit is set to zero,

the descrambler is bypassed. Figure 19 shows one element of the descrambler. This element is repeated eight times for the JESD204A Virtex-5 FPGA implementation.

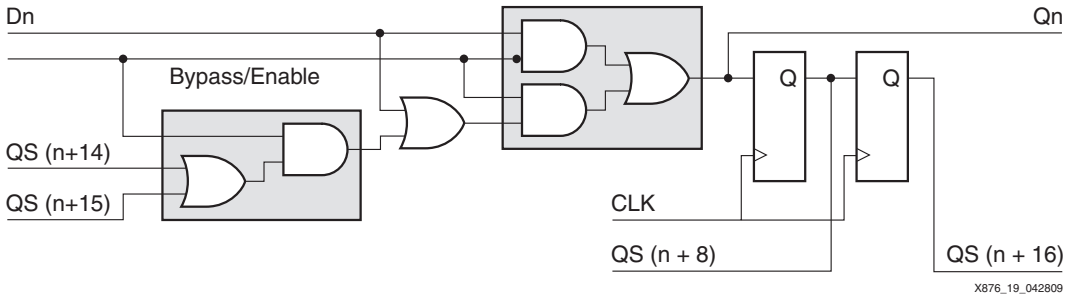

Figure 19: **Implementation of a Parallel Descrambler with Enable/Bypass**

Reference Design

The reference design files can be download at: http://www.xilinx.com/member/jedec_jesd204a_ref_des/index.htm.

The reference design checklist is shown in Table 9.

Table 9: **Reference Design Checklist**

Parameter	Description
General	
Developer Name	Marc Defossez
Target Device	Virtex-5 LXT, SXT, TXT, or FXT
Source Code Provided	Yes
Source Code Format	VHDL
Design Uses Code/IP from Existing Application Note, Reference Designs, Third Party, or CORE Generator™ Software	No
Simulation	
Functional Simulation Performed	Yes (per hierarchical block)
Timing Simulation Performed	No
Testbench Format	VDHL
Simulator Software/Version	ModelSim SE 6.4
SPICE/IBIS Simulations	No
Implementation	
Synthesis Tool/Version	XST, version 10.1.03
Implementation Software Tools/Versions Used	ISE® software, version 10.1.03
Static Timing Analysis Performed	Yes
Hardware verification	
Hardware Verified	The hardware was verified using a second FPGA functioning as an ADC. Actual ADC hardware was not available when the reference design was created.

Table 9: **Reference Design Checklist *(Cont'd)***

Parameter	Description
Hardware Platform Used for Verification	GTP transceiver inputs on the ML505 Evaluation Platform [Ref 2] were used via SATA connectors. A conversion board was used to convert from SATA to SMA.

Reference Design Utilization Summary

Table 10 shows the component resources for a dual-lane JESD204A ADC interface design implemented in a Virtex-5 FPGA.

Table 10: **Full Interface Device Utilization Summary**

Component	Percentage (%)	Number Utilized
BUFDS	16	1
BUFGs	15	5
GTP_DUAL	16	1
PLL_ADV	16	1
RAMB36s	3	2+1
Flip-Flops	1	201
LUTs	1	186

Figure 20 shows the directory setup of the reference design.

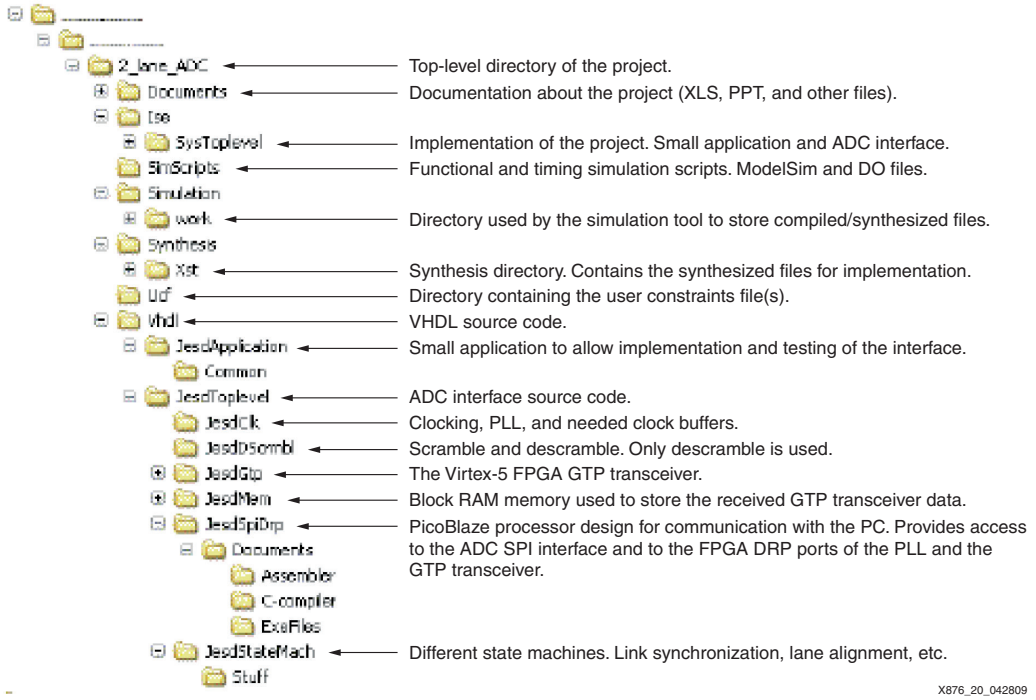

Figure 20: **Reference Design Directory Setup**

PDF documents in the directory of the VHDL source code provide detailed schematics, help files, and simulation results of the implemented source code. These documents contain block

diagrams that explain the details of the interface. The documents are provided to aid adapting the reference design to specific design needs and are supplementary to this application note.

Reference Design "SysToplevel"

"SysToplevel" consists of the JESD204A interface and a small application that enables the design to run on a Xilinx® test platform such as the ML505 board. Table 11 outlines the "SysToplevel" parameters.

Table 11: **Reference Design "SysToplevel"**

Parameter	Description
Attribute/Generic	
C_NmbrOfButtons	This is the number of buttons and defines the ButtonDebounce unit.
C_DualGtpTileLoc	This is the GTP_DUAL tile location. The generic is used at the JesdToplevel hierarchical level.
C_PllAdvLoc	This is the CMT_PLL location. The generic is used at the JesdToplevel hierarchical level.
C_Ramb36Loc_0	This is the block RAM location for channel 0. The generic is used at the JesdToplevel hierarchical level.
C_Ramb36Loc_1	This is the block RAM location for channel 1. The generic is used at the JesdToplevel hierarchical level.
C_RamB36Loc_Pb	This is the block RAM location for the PicoBlaze™ processor code storage. The generic is used at the JesdToplevel hierarchical level.
C_BusOneWidth	This is the bus width if the multiplexer is the application.
C_BusTwoWidth	This is the bus width if the multiplexer is the application.
Port/Pin	
SysTop_Clk_n/p	This is the application clock.
SysTop_GtpClk_n/p	This is the reference clock of the GTP transceiver.
SysTop_Gtp_Rx0_n/p	This is the channel 0 data input.
SysTop_Gtp_Rx1_n/p	This is the channel 1 data input.
SysTop_GtpPrbsRst	This is the reset for the PRBS engine of the GTP transceiver.
SysTop_SyncIn	This is the SYNC request from the application to the interface.
SysTop_OnOff	This is the input from a pushbutton that turns the block RAM on or off.
SysTop_Reset	This is the input from a pushbutton that resets the reference design.
SysTop_GtpPrbsTst0/1	This DIP switch input is the PRBS engine start of the GTP transceiver.
SysTop_DatOutSel	This DIP switch selects the channel on the outputs.
SysTop_Alive	This LED output indicates that the CMT_PLL is working.
SysTop_PrbsErr0/1	These LED outputs indicate PRBS errors.
SysTop_Flag	This is the status of the block RAM data buffer.
SysTop_Data	This is the received data output. It is selected by "SysTop_DatOutSel."
SysTop_SyncOut	This is the SYNC output to the ADC device.
SysTop_Pb_Uart_Tx	This is the interface controller UART output.
SysTop_Pb_Uart_Rx	This is the interface controller UART input.

Table 11: **Reference Design "SysToplevel" *(Cont'd)***

Parameter	Description
SysTop_Pb_Spi_Clk	This is the interface control SPI bus for control of the JESD204A compliant ADC.
SysTop_Pb_Spi_Cs	
SysTop_Pb_Spi_Di	
SysTop_Pb_Spi_Do	

Reference Design "JesdToplevel"

"JesdToplevel" is used as a component in the "SysToplevel" source code. This part of the reference design is the implementation of the JESD204A interface (Table 12).

Table 12: **Reference Design "JesdToplevel"**

Parameter	Description
Attribute/Generic	
C_DualGtpTileLoc	This is the GTP_DUAL tile location. The generic is used at the JesdToplevel hierarchical level.
C_PllAdvLoc	This is the CMT_PLL location. The generic is used at the JesdToplevel hierarchical level.
C_Ramb36Loc_0	This is the block RAM location for channel 0. The generic is used at the JesdToplevel hierarchical level.
C_Ramb36Loc_1	This is the block RAM location for channel 1. The generic is used at the JesdToplevel hierarchical level.
C_RamB36Loc_Pb	This is the block RAM location for the PicoBlaze processor code storage. The generic is used at the JesdToplevel hierarchical level.
Port/Pin	
Jesd_Gtp_ClkIn	This is the reference clock of the GTP transceiver. The differential clock buffer is placed on a higher hierarchical level of the design.
Jesd_Gtp_Rx0_n/p	This is the channel 0 data input.
Jesd_Gtp_Rx1_n/p	This is the channel 1 data input.
Jesd_Gtp_Reset	This is the reset of the interface design, including the GTP transceiver module.
Jesd_GtpPrbsRst	This is the reset for the PRBS engine of the GTP transceiver.
Jesd_Mem_RdEna	This is the read enable for the block RAM data buffer (Port_B).
Jesd_Mem_RdRst	This is the reset for the block RAM data buffer (Port_B).
Jesd_Mem_RdClk	This is the clock for the block RAM data buffer (Port_B). This clock is normally an application clock. It is similar to the JESD204A application clock but with a different phase.
Jesd_Mem_MustRead0/1	This is the status output of the self-addressing block RAM data buffer. It indicates that the buffer is nearly full and a read must happen to prevent data loss.
Jesd_Mem_Flags0/1	These are eight status flags from the block RAM data buffer that indicate where in the data buffer data is stored or retrieved.

Table 12: **Reference Design "JesdToplevel" *(Cont'd)***

Parameter	Description
Jesd_Mem_dataOut0/1	This is the data output from the block RAM buffer. In the reference design, the data output is 24 bits wide and consists of 15 GTP transceiver status bits and 8 data bits. For a real application, the GTP transceiver status bits can be omitted by modifying the VHDL code. This allows the buffer to become deeper.
Jesd_Gtp_IntrfcEna	This is an enable for the application running behind the JESD204A interface. This enable goes High when all clocks in the interface are stable and the GTP transceiver and all logic is out of reset.
Jesd_Gto_IntrfcRst	This is a reset signal for the application running behind the JESD204A interface. This signal is released (goes Low) after all elements in the interface are out of reset.
Jesd_PllGtp_UsrClk	This is a clock from the CMT_PLL in the JESD204A interface. This is the RXUSRCLK clock used by the GTP RX interface.
Jesd_PllGtp_UsrClk2	This is a clock from the CMT_PLL in the JESD204A interface. This is the RXUSRCLK2 clock used by the GTP RX interface. When the GTP transceiver is used in 8-bit data mode, this clock is equal to the RXUSRCLK. When the GTP transceiver is used in 16-bit data mode, this clock is half of the RXUSRCLK.
Jesd_PllGtp_Clk2	This is an extra clock output of the CMT_PLL.
Jesd_Pll_AliveOut	This signal indicates that the CMT_PLL is functioning. This is a slow pulsing (heartbeat) signal that can be used to connect to an LED on the PCB.
Jesd_Dscrmbl_Bypass	This input can be used to bypass the descramble module in the interface. This signal is normally controlled by the application.
Jesd_Gtp_LnkCnfgRdEna	This enables the read port of the link configuration memory.
Jesd_Gtp_LnkCnfgRdClk	This is the clock for the link configuration memory.
Jesd_Gtp_LnkCnfgRdAddr	This is the address for the link configuration memory.
Jesd_Gtp_LnkCnfgStat	This is the status bit of the link configuration memory.
Jesd_Gtp_LnkCnfgDatOut	This is an 8-bit link configuration data output.
Jesd_Gtp_LaneAlignProcDone	This is the status bit of the interface. It indicates that all lanes of the interface are aligned and that from this point onwards, normal data flows out of the interface.
Jesd_Gtp_ChanBondStrted	This is a status bit indicating that channel bonding of the lanes has started.
Jesd_gtp_ChanBondDone	This is a status bit indicating that channel bonding is done.
Jesd_SyncIn	This is a SYNC request from the application to the interface.
Jesd_OnOff	This is an input from a pushbutton that turns the block RAM on or off.
Jesd_Pb_Uart_Tx	This is the interface controller UART output.
Jesd_Pb_Uart_Rx	This is the interface controller UART input.
Jesd_Pb_Spi_Clk	This is an interface control SPI bus for control of the JESD204A compliant ADC.
Jesd_Pb_Spi_Cs	
Jesd_Pb_Spi_Di	
Jesd_Pb_Spi_Do	

Table 12: **Reference Design "JesdToplevel" *(Cont'd)***

Parameter	Description
Jesd_GtpPrbsCntRst0/1	This resets the PRBS module in the GTP transceiver.
Jesd_GtpPrbsTst0/1	This DIP switch input is the PRBS engine start of the GTP transceiver.
Jesd_PrbsErr0/1	These LED outputs indicate PRBS errors.

The interface is built as a hierarchical structure of separate modules. Each module can be used as a stand-alone module performing a specific task of the JESD204A standard.

Conclusion

The GTP transceivers in the Virtex-5 FPGA are perfectly suited for ADC devices using the JESD204A standard. This standard makes it possible to connect the FPGA to high-speed ADC devices with a low pin count. Alignment of ADC data is also made easier using the JESD204A standard. Another advantage of using the Virtex-5 FPGA GTP transceivers is that one GTP_DUAL tile can connect a two-lane ADC and DAC device at the same time.

A single interface solution cannot work for all possible ADC setups and allow for all possibilities of the JESD204A standard. The supplied reference design needs to be modified or an interface built to follow the specifications of the ADC device used.

References

This document uses the following references:

1. JEDEC Standard No. 204A (JESD204A) *Serial Interface for Data Converters* http://www.jedec.org/download/search/JESD204A.pdf
2. UG347, *ML505/ML506/ML507 Evaluation Platform User Guide*.
3. DS202, *Virtex-5 FPGA Data Sheet: DC and Switching Characteristics*.
4. UG190, *Virtex-5 FPGA User Guide*.
5. UG196, *Virtex-5 FPGA RocketIO GTP transceiver User Guide.*
6. UG195, *Virtex-5 FPGA Packaging and Pinout Specification*.
7. UG203, *Virtex-5 FPGA PCB Designer's Guide*.

Revision History

The following table shows the revision history for this document.

Date	Version	Description of Revisions
09/18/09	1.0	Initial Xilinx release.
02/22/10	1.0.1	Updated http://www.xilinx.com/member/jedec_jesd204a_ref_des/index.htm link.

Notice of Disclaimer