AFE76xx Alarm Training
Part I: JESD204B Alarms

Kang Hsia

10/10/2018

Agenda

- Brief Overview of JESD204B Protocol
- Details of JESD204B errors
  - FIFO Errors
  - Lane Errors
- TXDAC Data Path Protection Logic to Auto-Zero Output Upon Detection of Errors

Covered in Future Topics

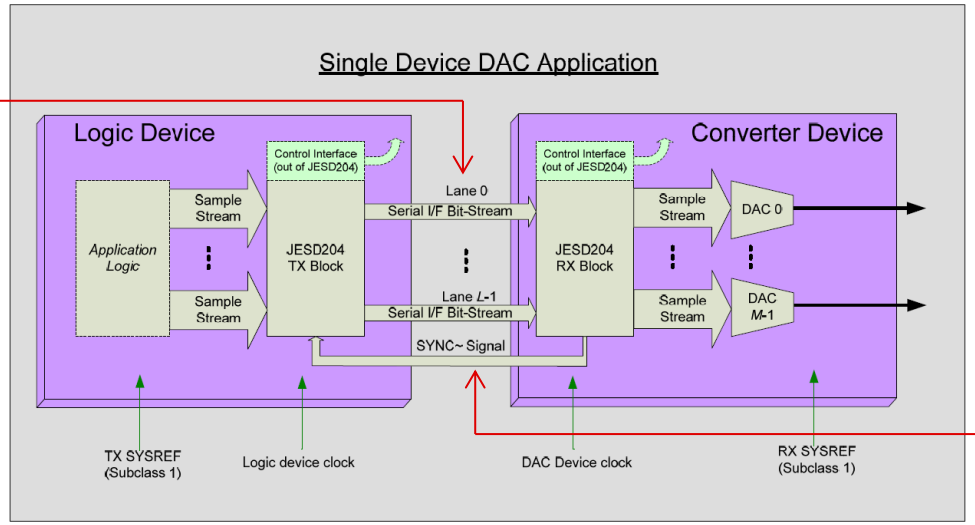
- Power Amplifier Protection Error
- SYSREF Alarms
- Clock Alarms
- On-chip PLL check
- SERDES PLL Check

JESD204B Overview for Logic Device (ASIC/FPGA) to DAC (converter device)

Crash Course of JESD204B ASIC/FPGA Logic Device to DAC Device

- Logic Device = ASIC or FPGA. This is the JESD204B Transmitter.
- DAC Device = digital-to-analog converter. This is the JESD204B Receiver.

8b/10b encoded Serial Interface Lanes



~SYNC Interface from JESD204B Receiver to JESD204B Transmitter

Figure B.2 — Single Device DAC Application

- Data are aggregated into high speed serial lanes (SERDES). Single lane or multiple lanes can form an effective data link. Each lanes are 8B/10B encoded.
- Link status are communicated via ~SYNC (SYNC_REQUEST) signal from the DAC device and also other software interfaces. The ASIC/FPGA logic devices need to handle link status accordingly.

JESD204B RX Standard Section 7

- The ~SYNC or SYNCB signal is an important signal for JESD204B link establishment. The JESD204B receiver basically “raises hand” whenever an error occurs that impacts the quality of the link.
- The JESD204B Standard Section 7 highlighted the three main functionality of SYNCB or ~SYNC signal
  - Initial request of synchronization upon initialization
    The minimum duration for a synchronization request on the SYNC~ signal is 5 frames plus nine octets. Further timing requirements are specified in 4.9.
  - Synchronization request upon errors requiring re-initialization
  - Detection of an error not requiring re-initialization

7.6.4 Error reporting via SYNC interface

On detection of an error requiring re-initialization, the receiver shall activate the SYNC~ signal for the duration of a whole number of frames. The minimum duration is five frames plus nine octets.

On detection of an error not requiring re-initialization, the required behavior is dependent on the deterministic latency subclass of the device.

- Subclass 1 or Subclass 2 receiver devices shall indicate the detection of such an error by activating the SYNC~ signal for exactly 2 frame periods. For one or more errors occurring during a particular LMFC period, the SYNC~ signal shall be activated two frame periods prior to the end of the next LMFC period and deactivated at the LMFC boundary associated with the end of this period. Other errors scheduled for signaling during this multiframe period may be ignored, unless they would require link re-initialization.
  - For backwards compatibility with JESD204A, Subclass 1 or 2 receiver devices may optionally support SYNC~ error reports of 1 frame length, and/or the ability to disable error reporting completely.

JESD204B Errors

Minimum Required

Optional Error Reporting

7.6 Error handling

7.6.1 Error kinds

Table 18 shows the minimum set of errors to be detected in each receiver:

Table 18 — Minimum set of errors to detect per receiver

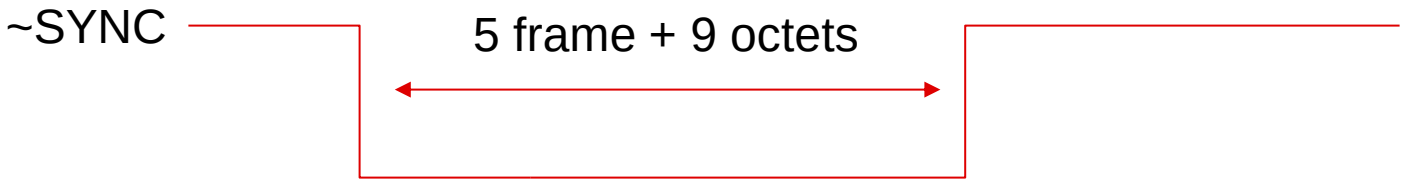
Error	Description
Disparity error	The received code group exists in the 8B/10B decoding table, but is not found in the proper column according to the current running disparity.
Not-in-table error	The received code group is not found in the 8B/10B decoding table for either disparity
Unexpected control character	A control character is received that is not expected at the given character position
Code group synchronization error	The state machine for code group synchronization has returned to the CS_INIT state

In addition, many other kinds of errors may occur, which may not always require detection or action in each application, e.g.,

- Frame realigned (previous conversion samples may be in error)
- Lane realigned (previous conversion samples may be in error)
- Uncorrectable frame alignment error
- Initial lane alignment failure
- Uncorrectable lane alignment error
- Link configuration data error (parameters in TX and RX do not match)
- Lane alignment sequence decoding error (wrong octets decoded)

SYNC REQUEST

Synchronization request upon errors requiring re-initialization



Register: TXDUC_REG139

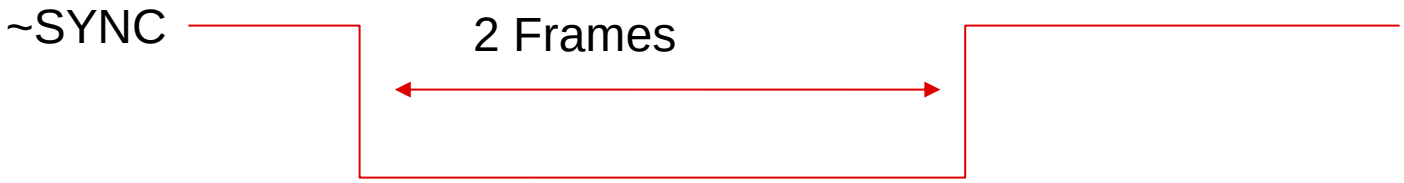
Description	:
Offset	: 0x8B
System Address	: 0x08B
Addressing Mode	: 8-bits

7	6	5	4	3	2	1	0
sync_request_ena							
RW - 0xFF							

Bit	Name	Description	SW Access HW Access Protection	Reset
7:0	sync_request_ena	These bits select which errors cause a sync request. Sync requests take priority over the error notification; so if sync request isn't desired; set these bits to a '0'. bit7 = multi-frame alignment error bit6 = frame alignment error bit5 = link configuration error bit4 = elastic buffer overflow (bad RBD value) bit3 = elastic buffer end char mismatch (match_ctrl match_data) bit2 = code synchronization error bit1 = 8b/10b not-in-table code error bit0 = 8b/10b disparity error	RW - -	0xFF

Minimum Required per JESD204B standard. Other selection depends on application requirement. **TI recommends customers to evaluate their applications needs. For maximum reliability, select all of settings.**

Error Reporting
Detection of an error not requiring re-initialization



Register: TXDUC_REG140

Description	:
Offset	: 0x8C
System Address	: 0x08C
Addressing Mode	: 8-bits

7	6	5	4	3	2	1	0
error_ena							
RW - 0xFF							

Bit	Name	Description	SW Access HW Access Protection	Reset
7:0	error_ena	These bits select the errors generated are counted in the err_c for the link. The bits also control what signals are sent out the pad_syncb pin for error notification. bit7 = multi-frame alignment error bit6 = frame alignment error bit5 = link configuration error bit4 = elastic buffer overflow (bad RBD value) bit3 = elastic buffer and char mismatch (match_ctrl match_data) bit2 = code synchronization error bit1 = 8b/10b not-in-table code error bit0 = 8b/10b disparity error	RW - -	0xFF

Selection depends on application requirement. For instance, the selection may be all settings not selected by “SYNC_REQUEST”

Error Counter

- The AFE76xx JESD204B RX IP has error counter feature. Basically, if the error reporting feature is enabled, the error counter can count the number of “selected” errors that has been programmed in the error_ena register.
- User will need to first clear the counter, and then read back the counter for the actual error that has been accumulated.
- This feature is useful for statistical gathering of error over period of time.
- The user may mask out various errors and only focus on the counting of particular errors (i.e. 8b/10b not-in-table and 8b/10b disparity error for link bit error count).

AFE76xx Commands for ~SYNC Request and Error Reporting

Process	Register Address (Hex)	Register Value (Hex)	Comment	TI Note
W	10	55	Open all TXDUCP0 page for all four DACs	
W	8B	FF	Enable ~SYNC request to go to logic LOW on either the LVDS SYNCBOUT or CMOS SYNCOUT driver for 5 frames + 9 octets during JESD204B RX error events	see associated powerpoint for SYNC request and error reporting adjustment
W	8C	FF	Enable ~SYNC request to go to logic LOW on either the LVDS SYNCBOUT or CMOS SYNCOUT driver for 2 frames during JESD204B RX error events	
W	8D	bit 7 set to 0	Make sure error reporting feature is enabled over ~SYNC output	
W	92	bit 3 set to 1	clears the JESD204B RX IP error counter	
W	92	bit 3 set to 0	starts the JESD204B RX IP error counter	
R	109	jesd_errcnt[7:0]	JESD204B RX IP error counter, lower 8 bits	
R	109	jesd_errcnt[15:8]	JESD204B RX IP error counter, upper 8 bits	
W	11	0	Close Page: Traffic Controller	

JESD204B Section 7.6.5
Other non-JESD204B interface errors

- The JESD204B standard also specifies other error reporting structure beside the ~SYNC signal
- This is implemented at the discretion of the vendor.

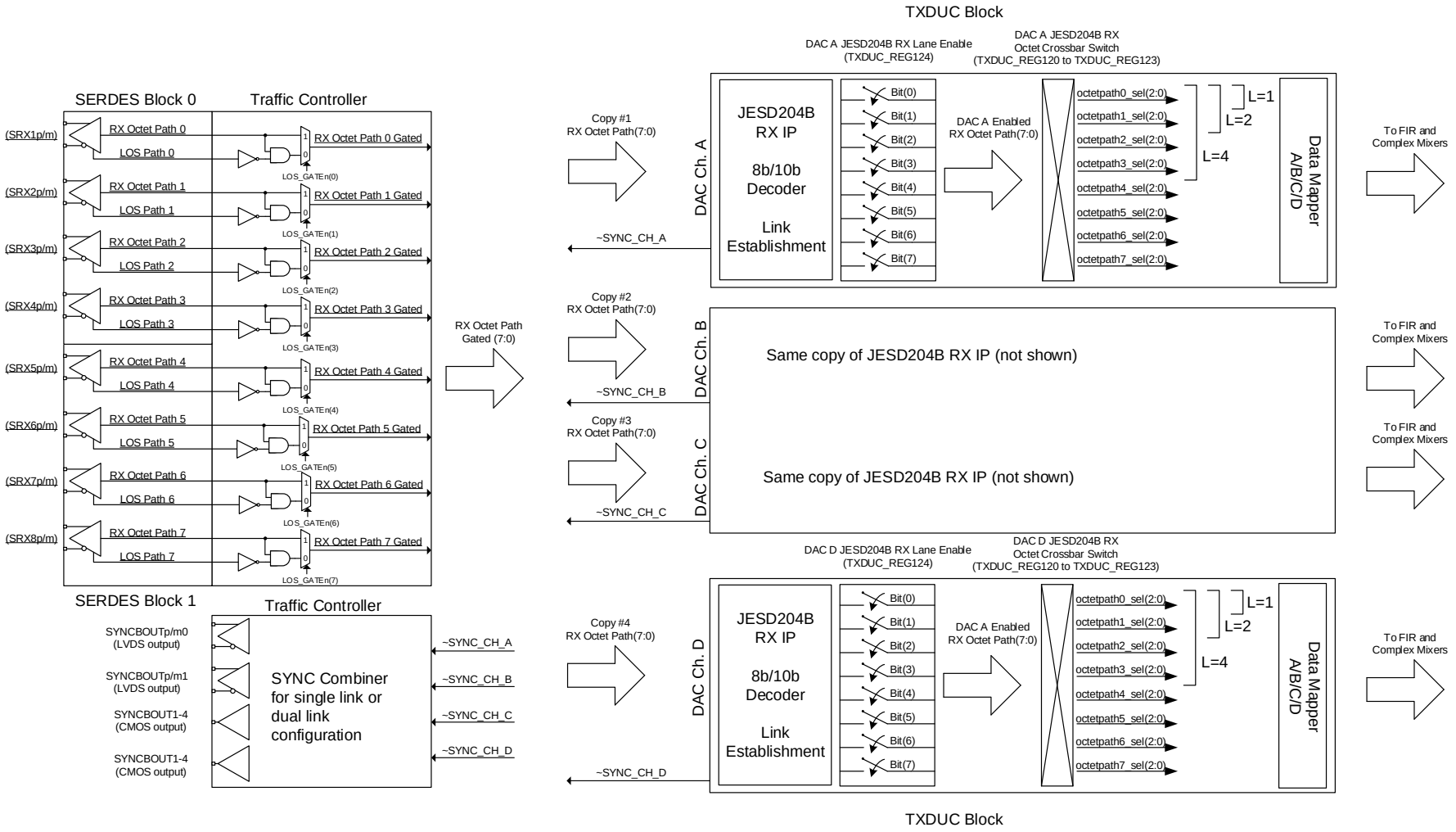
7.6.5 Error reporting via control interface

A receiver in a logic device shall have the ability to report decoding errors via a control interface to higher application layers. The details of this reporting are application- and implementation-dependent. A receiver in a DAC may report errors via a control interface to higher application layers, either directly or via the logic device.

- TI implements the alarms on both the register read back and also GPIO for quick handling of the interrupt service routine.

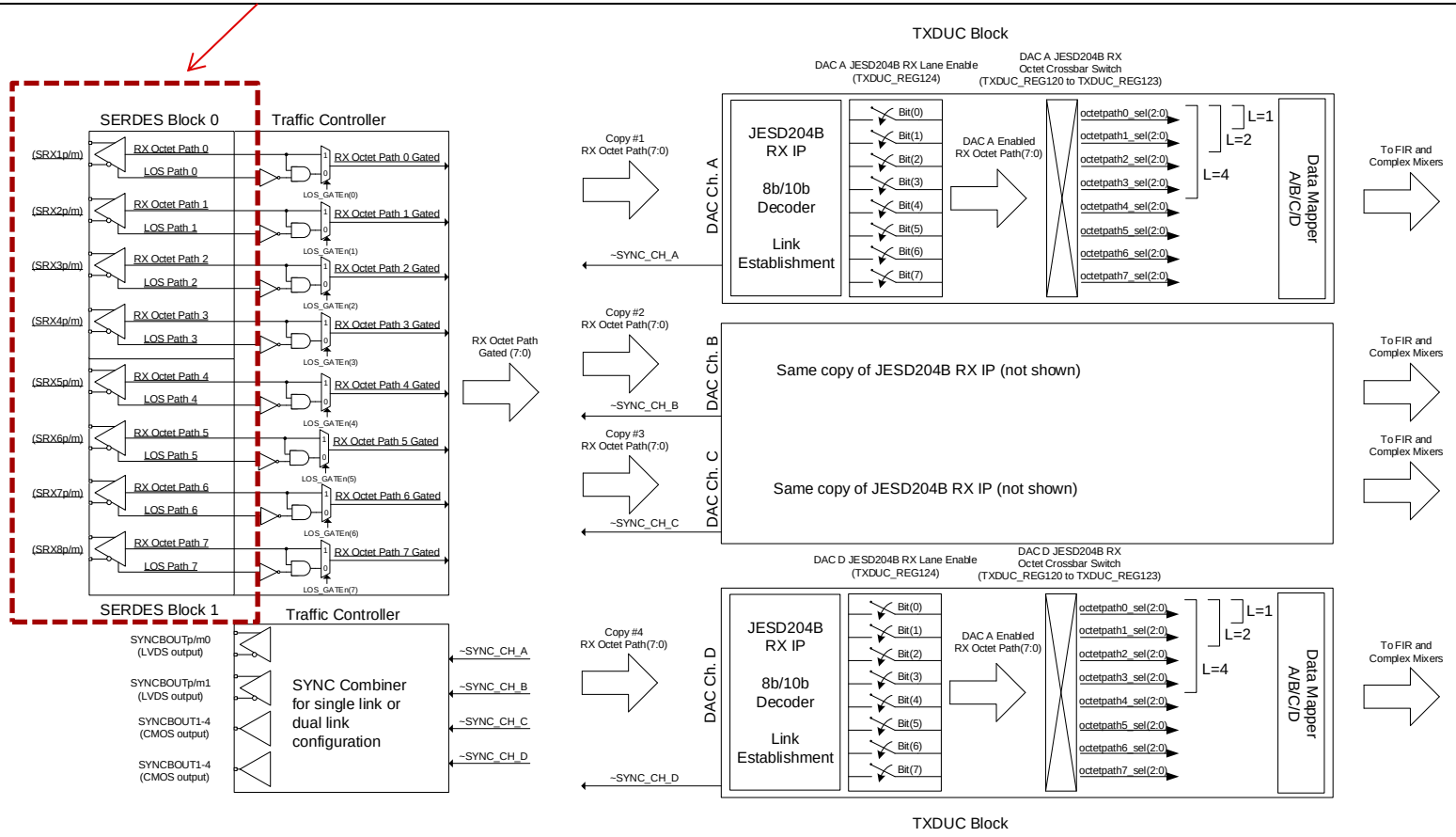
AFE76xx JESD204B RX IP Logic in the DAC

TI AFE76xx DAC JESD204B Implementation



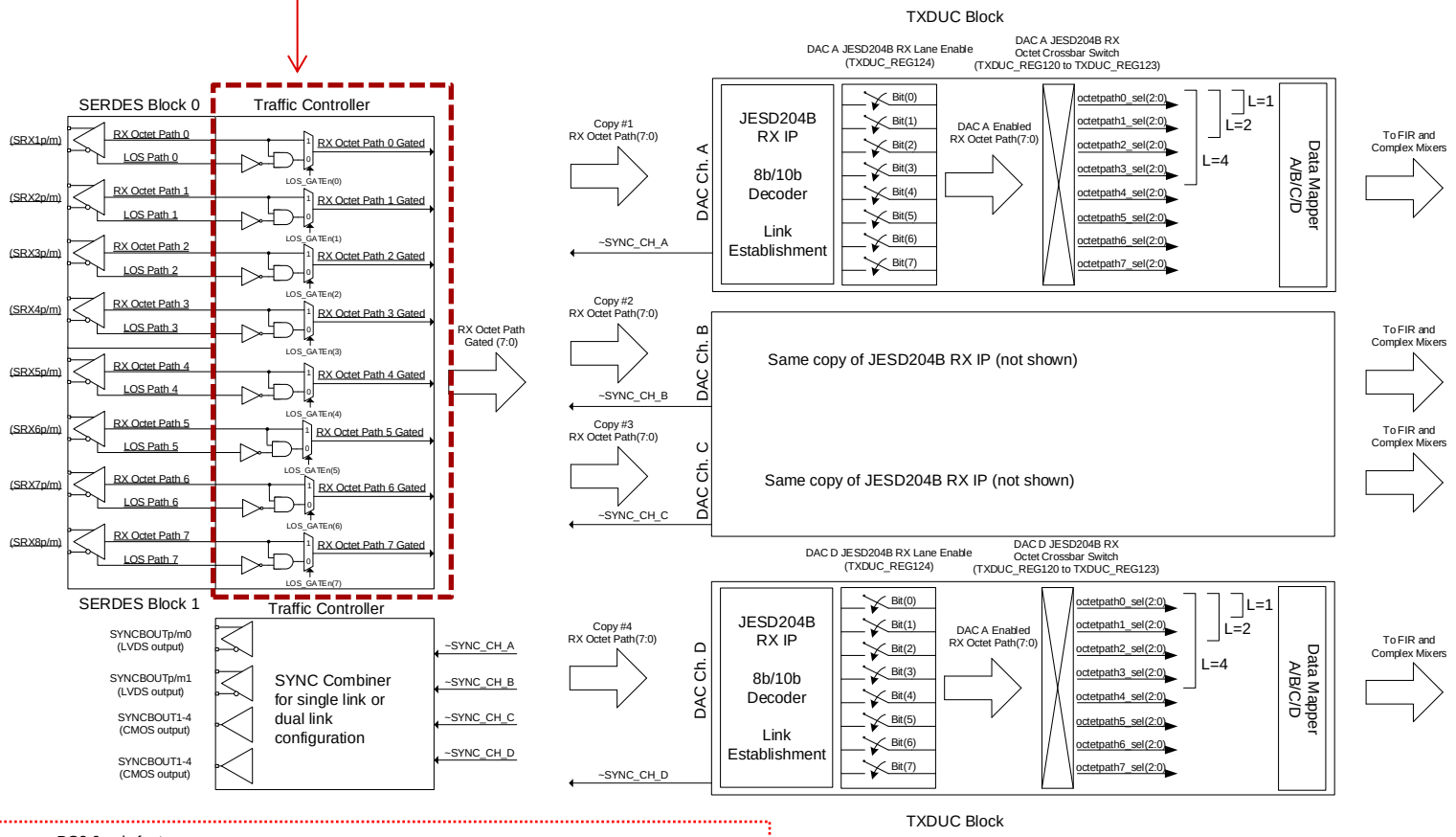
TI AFE76xx DAC JESD204B Implementation

Serial interface receivers (SRX1 to SRX8) physical ports: This is also known as SERDES (serializer/deserializer). The receivers receive 8B/10B encoded serialized data and deserialize to the rest of the data path.



TI AFE76xx DAC JESD204B Implementation

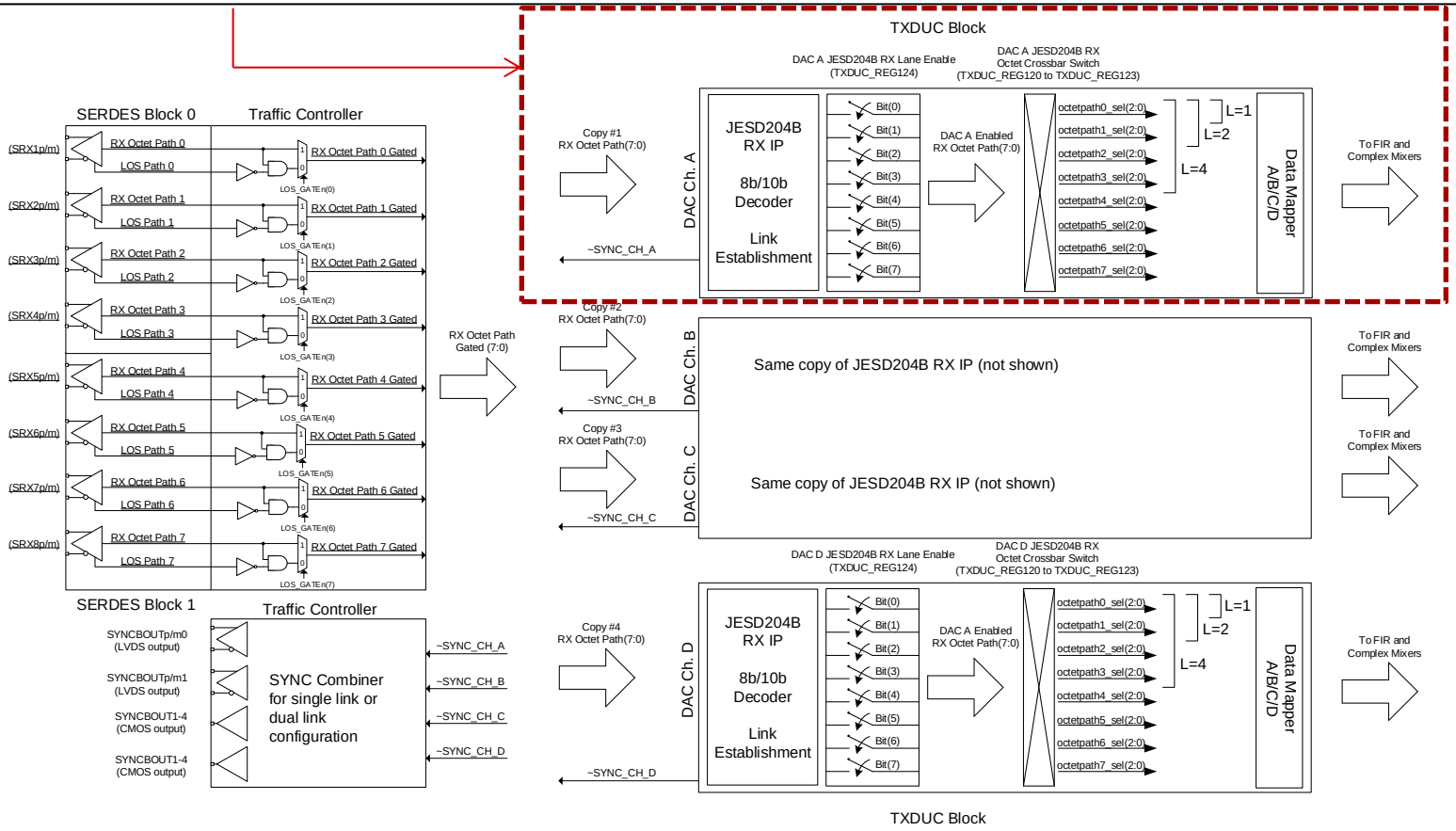
Loss of Signal (LOS) Gating: whenever electrical idle on the SRX1 to SRX8 lines are detected, the LOS signal will trigger to zero out the 10B encoded data path. This will trigger JESD204B alarm to trigger signal path protection.



1. PG3.0 only feature.
2. Without LOS gating, the default LOS output may be a proper 10B code, which translates to a large DC signal and may damage further downlink components.
3. Signal path protection includes power amplifier protection logic and other native zeroing logic in the downlink TXDAC datapath.

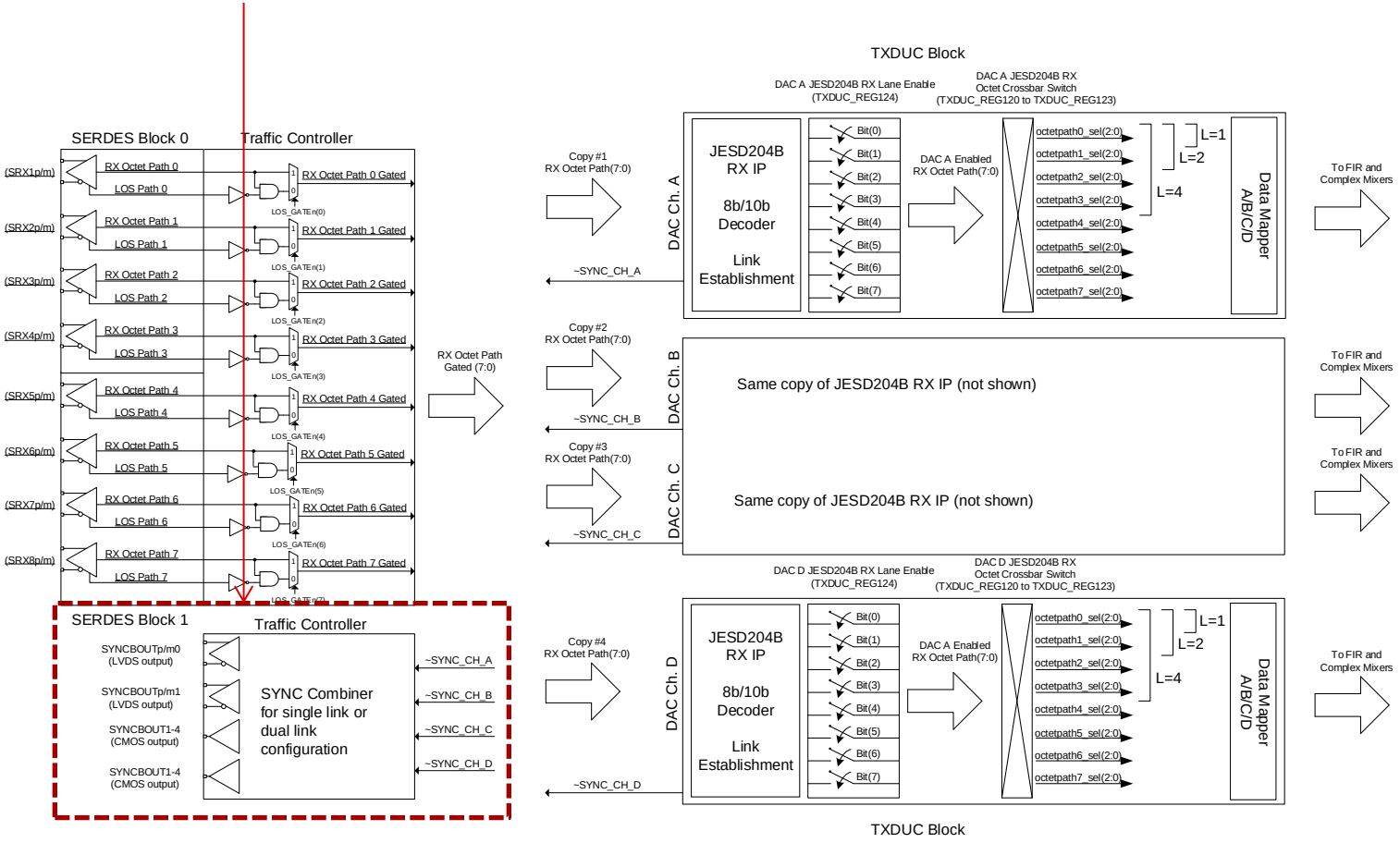
TI AFE76xx DAC JESD204B Implementation

TXDUC JESD204B Logic: four copies of identical logic path that are routed to individual channels. It is responsible for decoding the serialized 10B code into proper 8B code (octets). The logic also map and route the data lanes to user configurable data paths, and then route to the digital up-conversion logics (DUCs)

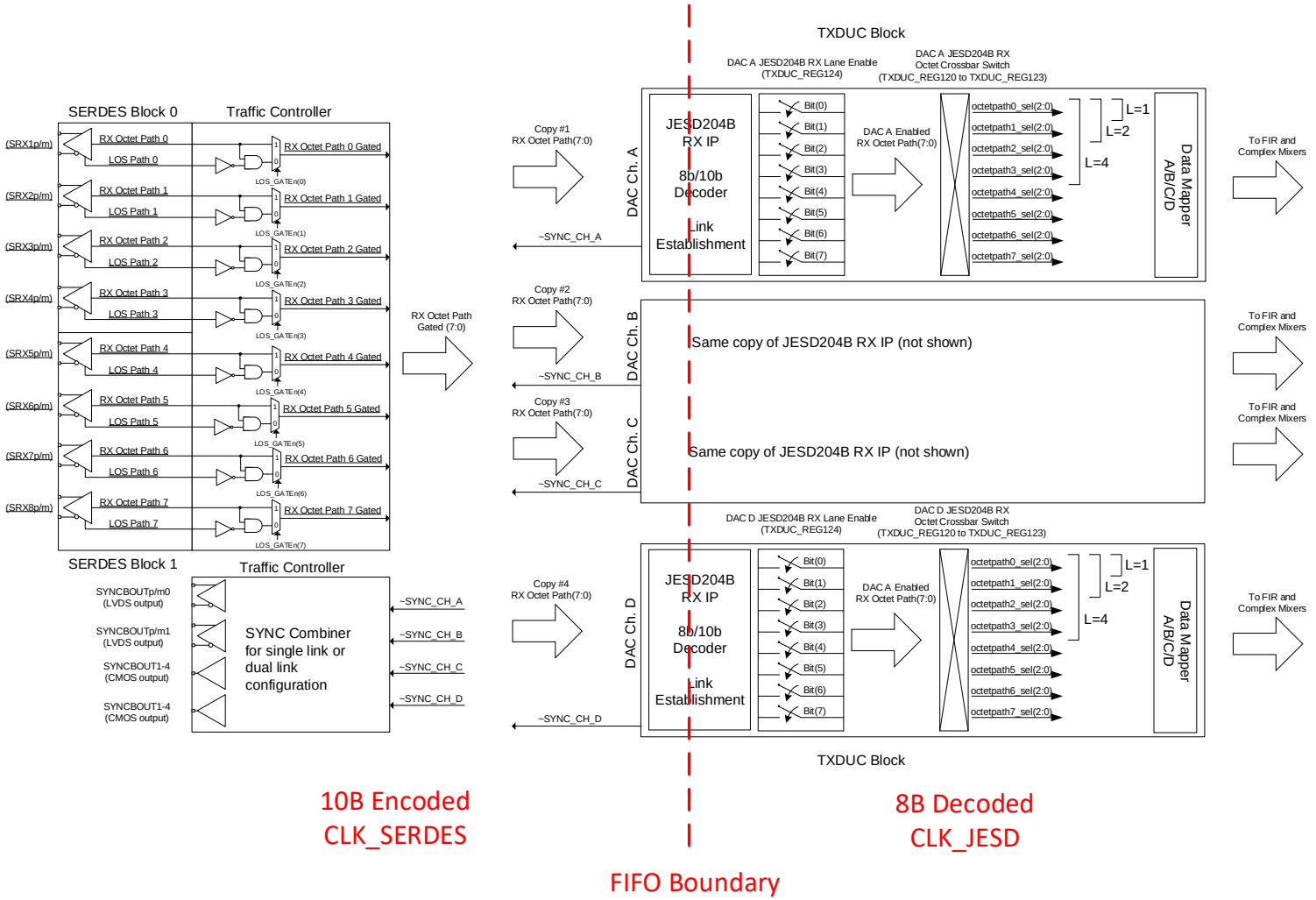


TI AFE76xx DAC JESD204B Implementation

~SYNC over LVDS or CMOS: aggregate single or multiple TXDUC JESD204B RX logics into single link or multiple links. Multiple links are intended for redundancy reasons to avoid single lane error causing the entire link to drop. This will maintain throughput of the JESD204B link.



FIFO Boundary Between 10B Encoded and 8B Decoded Data



AFE76xx Alarm Structure

Alarm Behaviors

- Alarms: error reporting structure in the AFE76xx TX DAC section to report any warnings or potential problems in the AFE76xx chip.
- Errors includes JESD204B RX IP error, PLL lock/unlock error, lane errors.
- Sticky: alarms in the AFE76xx are generally “sticky”. This means the alarms will *not* self-clear upon circuit/logic recovery. This allows the user to poll the non-critical errors regularly.
- Sticky alarms will need to be cleared before updating the latest result.
- Sticky nature of the alarm design is to allow the user to be able to read any errors occurred in the past, even if the errors have been self-fixed in the circuit. The user simply read the alarms first without clearing the alarms.

TXDUC Alarms

Under TXDUCP1 pages for all four DACs:

Alarm Status:

- Alarm Status Registers are located from 0x82 to 0x91. These reflect the actual alarm status. The alarms are sticky, which means the error will be reflected as bit logic 1b'1 and will not self clear. The errors will be recorded until manually being cleared by user.

Alarm Clear:

- Alarm Clear Registers are located from 0xAA to 0xB7. These bits will clear the Alarms Status Registers corresponding from ox82 to 0x91. A bit logic level from 1b'1 to 1b'0 clears the sticky alarm in order to reflect on the latest status. The Alarm Clear Registers must be set to 1b'0 in order for the Alarm Status Registers to reflect properly.

Alarm Master Clear

- A Master Alarm Clear bit will clear all the sticky alarms, and is located in 0x25 register, bit 6. A bit logic level from 1b'1 to 1b'0 (or 0xFF to 0x00 for entire register operation) clears the sticky alarm in order to reflect on the latest status. The Master Alarm Clear bit must be set to 1b'0 in order for the Alarm Status Registers to reflect properly.

Alarm Masking:

- Alarm Mask Registers are located from 0x96 to 0xA5. These register bits will mask the alarm bits to prevent the error from presenting on the TXALARM pins. The TXALARM pins are mainly used for interrupt service routines. If the critical alarms need to be present to the TXALARM pin for interrupt service routines, then the associated alarm bits must be unmasked by writing 1b'0 to the bits.

Alarm Master Clear

- A Master Mask Register will mask all the alarm bits, and is located in 0x25 register, bit 7. Setting this bit to 1b'1 will mask all alarms.

Summary:

- Alarm Status Registers: 0x82 to 0x91
- Alarm Clear Registers: 0xAA to 0xB7 (0xFF to 0x00 transition)
- Alarm Mask Registers: 0x96 to 0xA5.

JESD204B RX Lane Alarms

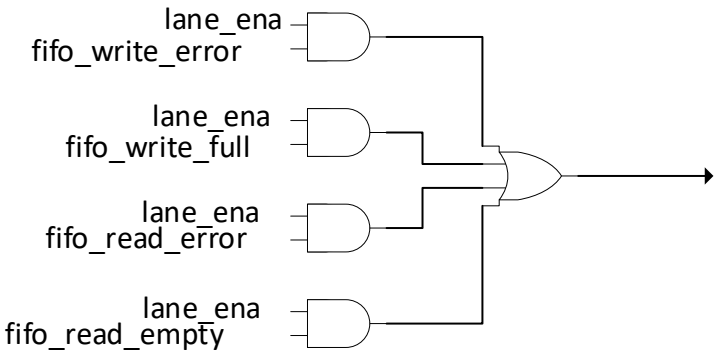
JESD204B DUC Registers

0x84, 0x85, 0x86, 0x87

- Upper 4-bits indicates SRX2, SRX4, SRX6, and SRX8 lane
- [7:4] Lane 1_3_5_7 FIFO errors
- Lower 4-bits indicates SRX1, SRX3, SRX5, and SRX7 lane
- [3:0] Lane 0_2_4_6 FIFO errors

- bit3 = write_error : High if write request and FIFO is full (NOTE: only released when JESD block is initialized with mem_init_state)
- bit2 = write_full : FIFO is FULL. Ignore in actual application. Error used for TI internal design purpose
- bit1 = read_error : High if read request with empty FIFO (NOTE: only released when JESD block is initialized with mem_init_state)
- bit0 = read_empty : FIFO is empty. Ignore in actual application. Error used for TI internal design purpose

JESD204B RX Lane Alarms
Alarm Data Path to Register Only



- Lane X FIFO Alarms
- X ranges from 0 to 7
- JESD204B Alarms located in the TXDUCP1 page, 0x84 to 0x87 registers. One register for each two lanes.
- fifo_write_full and fifo_read_empty alarms may be ignored.

Note: the associated lane must be enabled in each JESD204B path (one JESD204B path per TXDUC)

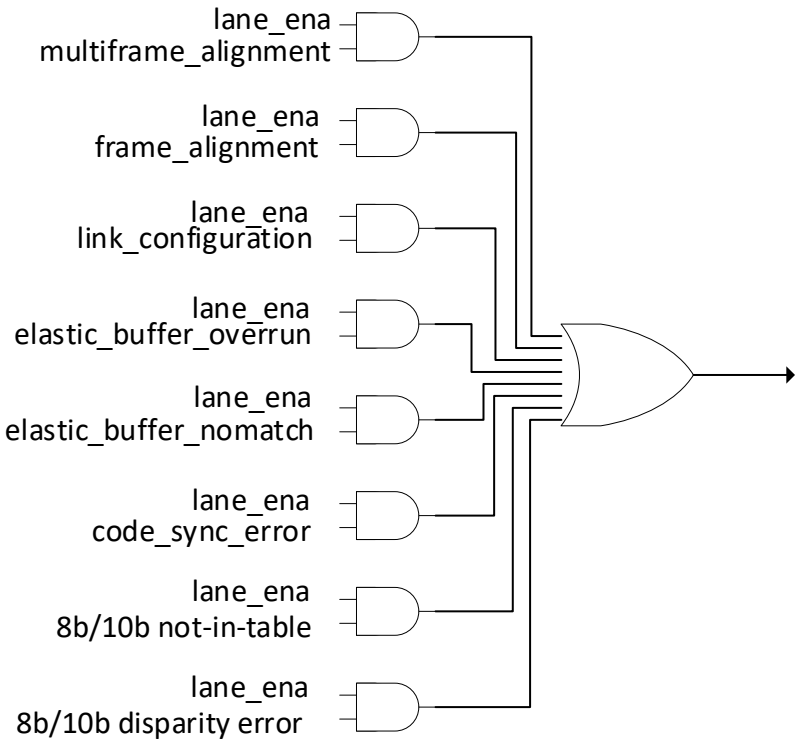
JESD204B RX IP Status Alarms

0x88, 0x89, 0x8A, 0x8B, 0x8C, 0x8D, 0x8E, 0x8F Registers

JESD204B Lane errors. Please read-back only the associated JESD204B lane for the specific DAC.

- bit7 = multiframe alignment error
- bit6 = frame alignment error
- bit5 = link configuration error
- bit4 = elastic buffer overflow (bad RBD value)
- bit3 = elastic buffer match error. The first non-/K/ doesn’t match 'match_ctrl' and 'match_data' programmed values
- bit2 = code synchronization error
- bit1 = 8b/10b not-in-table code error
- bit0 = 8b/10b disparity error

JESD204B RX IP Status Alarms
Alarm Data Path to Register Only



- lane_ena[7:0] located in 0x7C in TXDUCP0 page

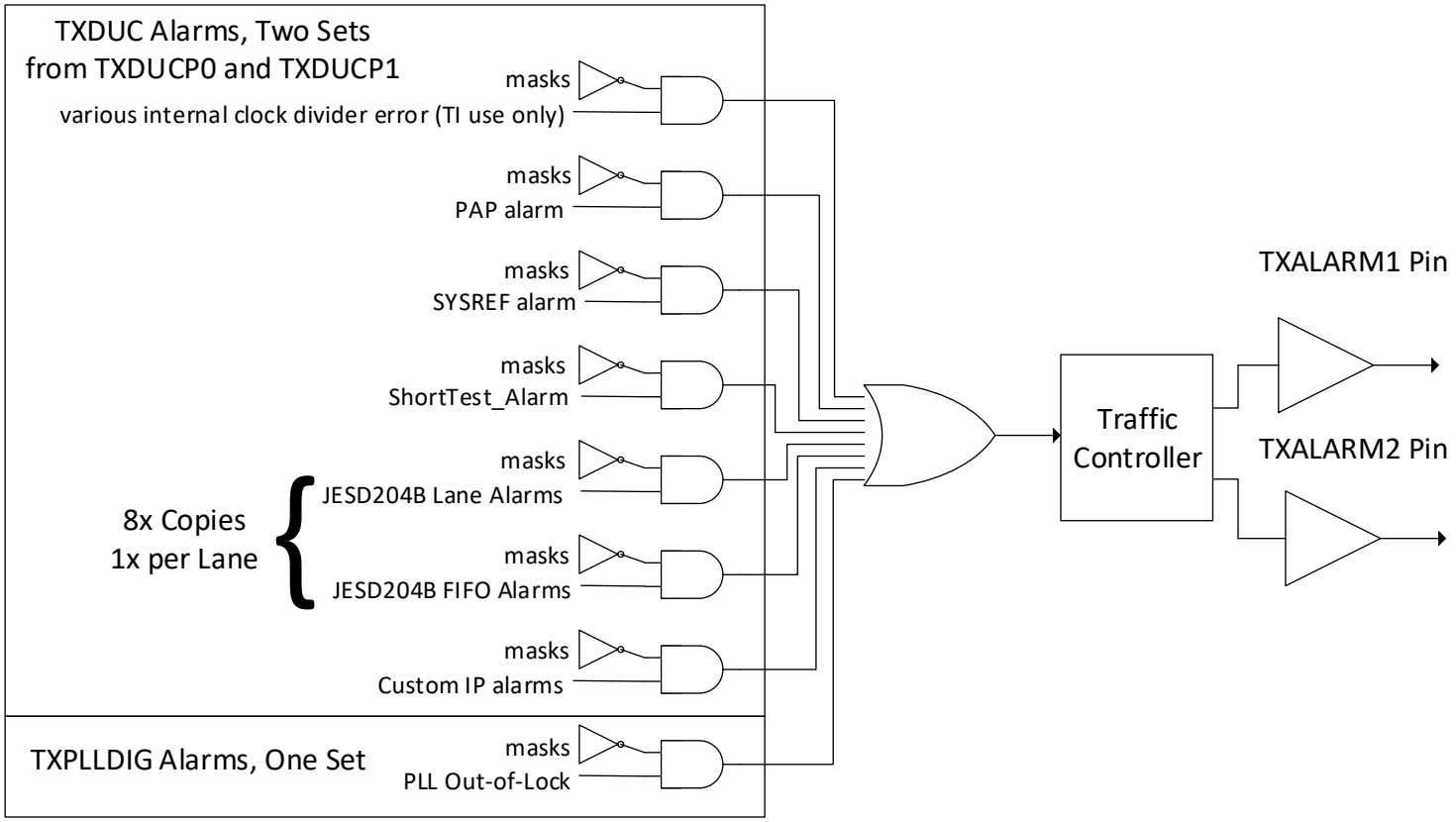
- Lane X JESD204B Alarms
- X ranges from 0 to 7
- JESD204B Alarms located in the TXDUCP1 page, 0x88 to 0x8F registers. One register for each lane

Note: the associated lane must be enabled in each JESD204B path (one JESD204B path per TXDUC)

Interrupts through TXALARM1 and/or TXALARM2 CMOS GPIO

- Interrupt: GPO output from the AFE76xx device that route to the master controller such as FPGA/ASIC.
- Upon any critical alarms (interruptible alarms), the AFE76xx will send interrupt through the TXALARM1 and TXALARM2 GPO to the master controller such that the master controller will start the interrupt service routine (i.e. ISR) to fix the error immediately.
- Interruptible alarms: alarms in the AFE76xx TX DAC section that are *not* masked. Non-masked alarms will be routed to the interrupt output on the TXALARM1 and/or TXALARM2 output.
- The end user will need to determine the alarms that are sufficiently critical for the interrupt, and also design the appropriate interrupt routine to recover the error or minimize the potential harm.

TXALARM1 and TXALARM2 Interrupt Alarm Data Path to GPIO Routing

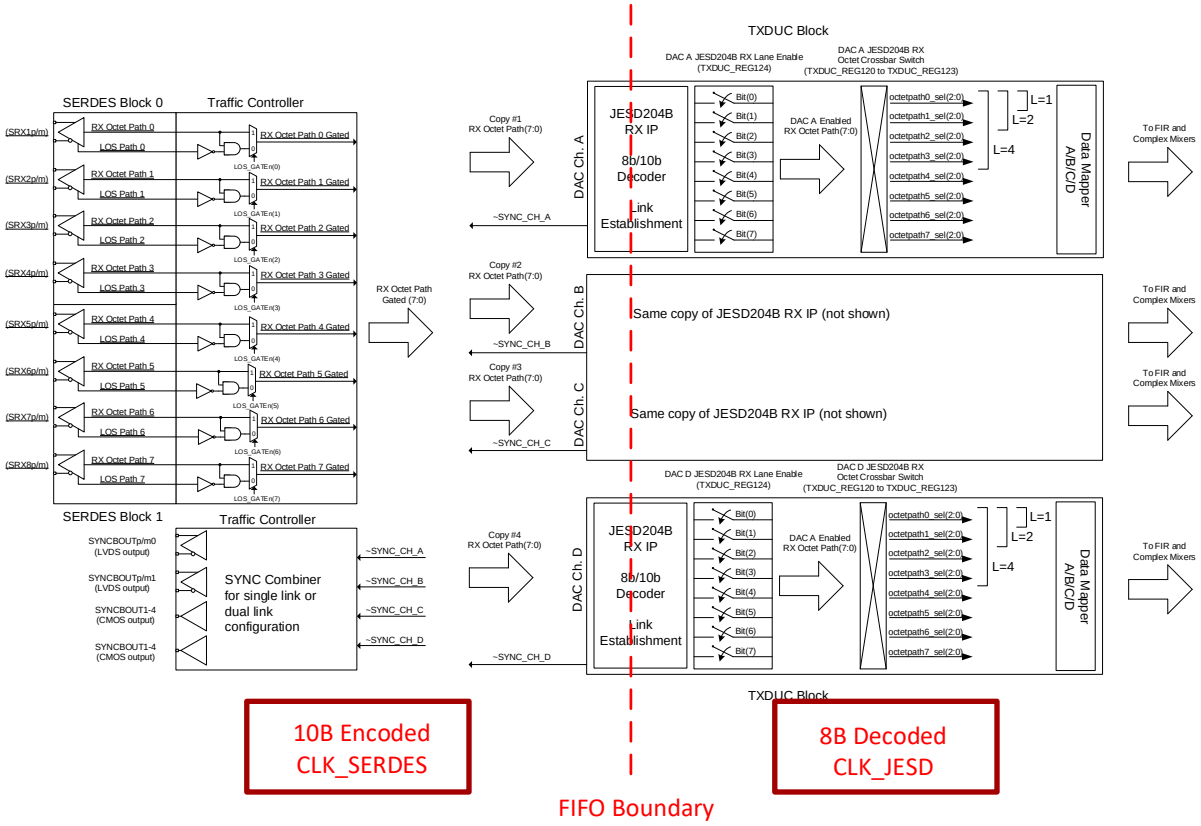


Details of Each Alarms

FIFO Errors

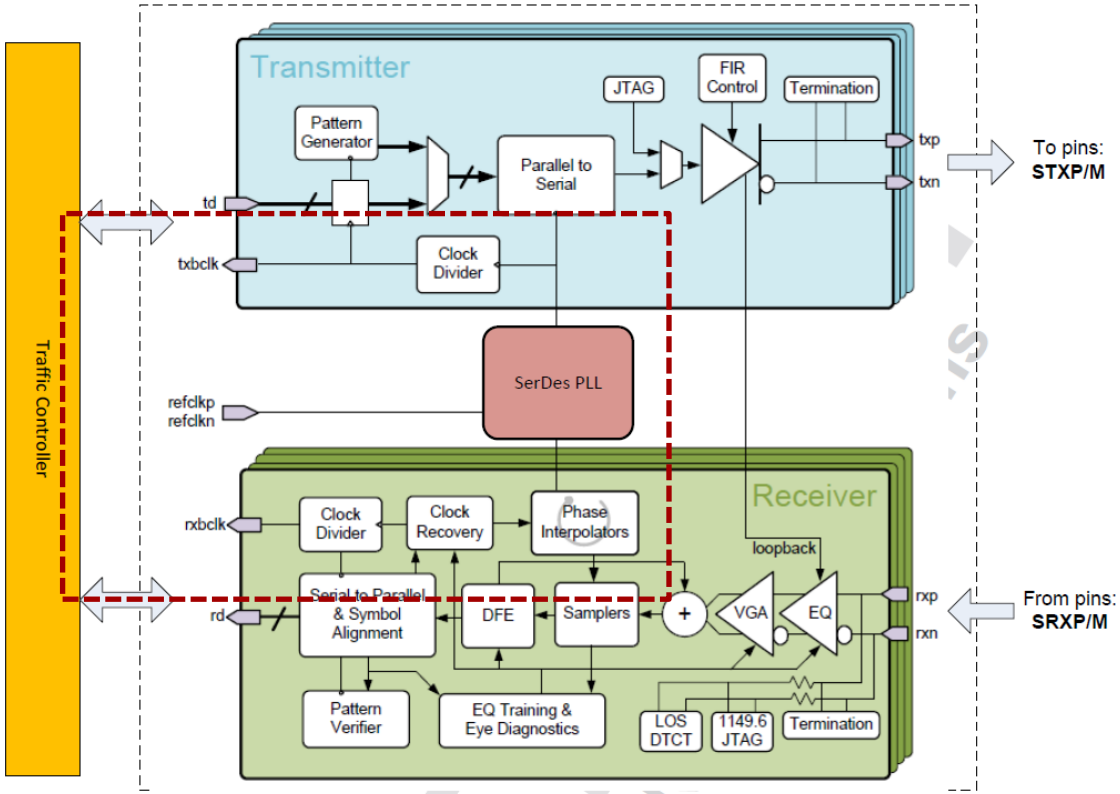
FIFO Errors

- FIFO error occurs whenever the CLK_SERDES and CLK_JESD are either missing or have mismatch in clock rate.
- FIFO error can be cleared through the forced JESD204B RX IP reset, assuming both CLK_SERDES and CLK_JESD are stable



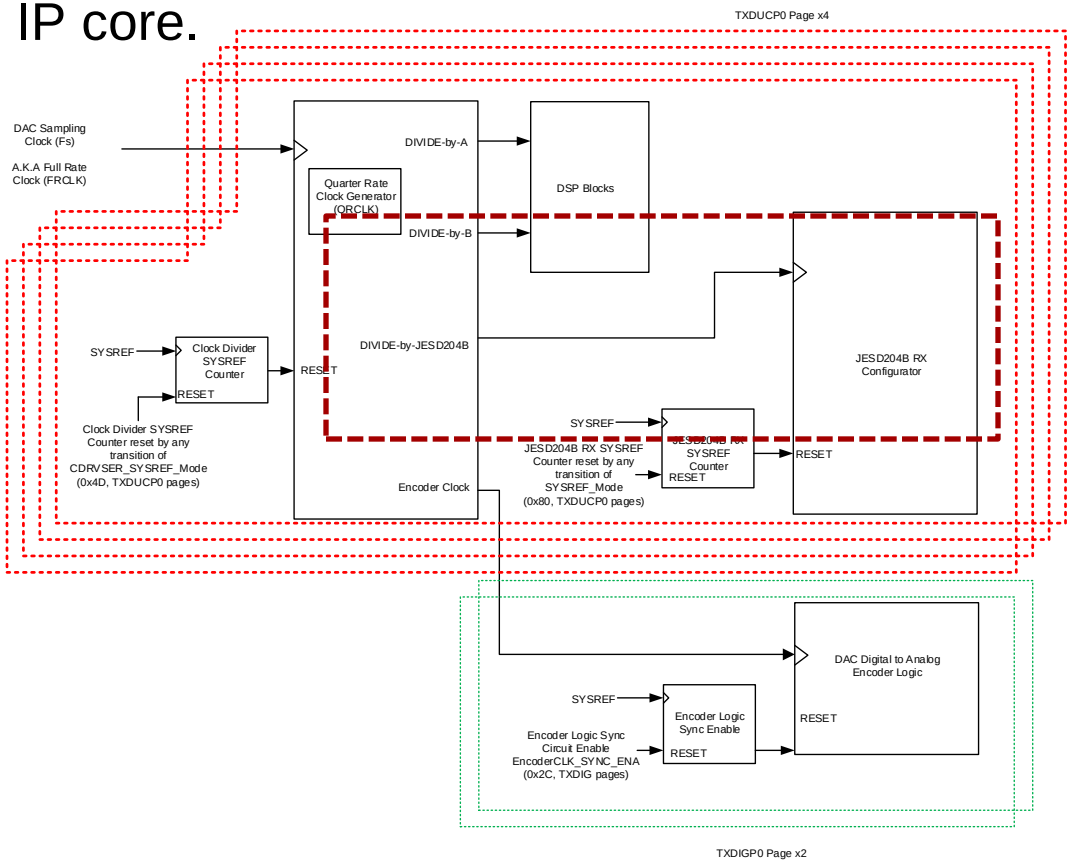
CLK_SERDES Generation

- CLK_SERDES (a.k.a. RXBCLK) is generated from the SERDES PLL with some clock information from the clock recovery circuit
- Reference clock to the SERDES PLL is generated from the divided down DACCLK (either from the on-chip PLL or externally provided).



CLK_JESD Generation

- CLK_JESD is sourced from the main DACCLK (either through the on-chip PLL or externally supplied clock source)
- An internal clock divider divides down the necessary CLK_JESD for the JESD204B IP core.



Error Handling and Things to Check

- When error occurs, check the SERDES STX lanes and JESD204B transmitter IP on the FPGA/ASIC logic device are phase locked to the SERDES SRX lanes and JESD204B receiver IP.
- The CLK_SERDES are partially contributed from the recovered SERDES clock from the SERDES SRX IP. If there are any shift in the recovered clock to the CLK_JESD, then FIFO error will occur
- Check the line rate configured on both the logic device and also the AFE76xx are within expectation.
- Check the SERDES PLL lock status of the AFE76xx.
- Typically, the FIFO error indicates the FIFO input clock (CLK_SERDES) and FIFO output clock (CLK_JESD) are disturbed. Other clocking path in the system may need to be thoroughly checked.
- JESD204B RX IP on the DAC side need to have a re-sync (JESD re-init) procedure performed after all the clocks are recovered since the JESD re-init procedure also re-init/reset the FIFO.

SERDES FIFO Clear Feature (PG3.0 Only)

- Key Register: 0x2D SERDES_FIFO_Err_Clear in TXDUCP0 page
- A 1b’0 to 1b’1 transition to clear the FIFO write error and FIFO read error.
- In order for this bit to function, both the CLK_SERDES and the CLK_JESD must be stable.
- Without the clocks, even when SERDES_FIFO_Err_Clear is set to 1b’1, it will not be reset the internal logic that generates the FIFO error bits.
- This is a good, alternative way to check for stability for the FIFO clocks without going through JESD204B IP reset (which resets the FIFO also).

Frame and Multi-Frame Error

Frame and Multi-Frame Setup

- Frame (F): a set of consecutive octets in which the position of each octet can be identified by reference to a frame alignment signal.
- Example below show the frame format for TXDAC of 42111 mode and RXADC of 42220 mode.

Table 6-58. JESD204B Frame Format for DACs: L-M-F-S-Hd = 42111 (1TX)

OCTET	1
Lane SRX0	TXA_i0[15:8]
Lane SRX1	TXA_i0[7:0]
Lane SRX2	TXA_q0[15:8]
Lane SRX3	TXA_q0[7:0]

Table 6-90. JESD204B Frame Format for ADCs: L-M-F-S-Hd = 42220 (1RX, complex mode)

OCTET	1	2
Lane STX0	RXA_i0_s0[15:0]	
Lane STX1	RXA_i0_s1[15:0]	
Lane STX2	RXA_q0_s0[15:0]	
Lane STX3	RXA_q0_s1[15:0]	

- Multi-frame (K) : a set of consecutive frames in which the position of each frame can be identified by reference to a multiframe alignment signal
- JESD204B standard specify maximum of K = 32.

Illustration of Frame and Multi-frame in JESD204B Link

- Frame is highlighted in blue

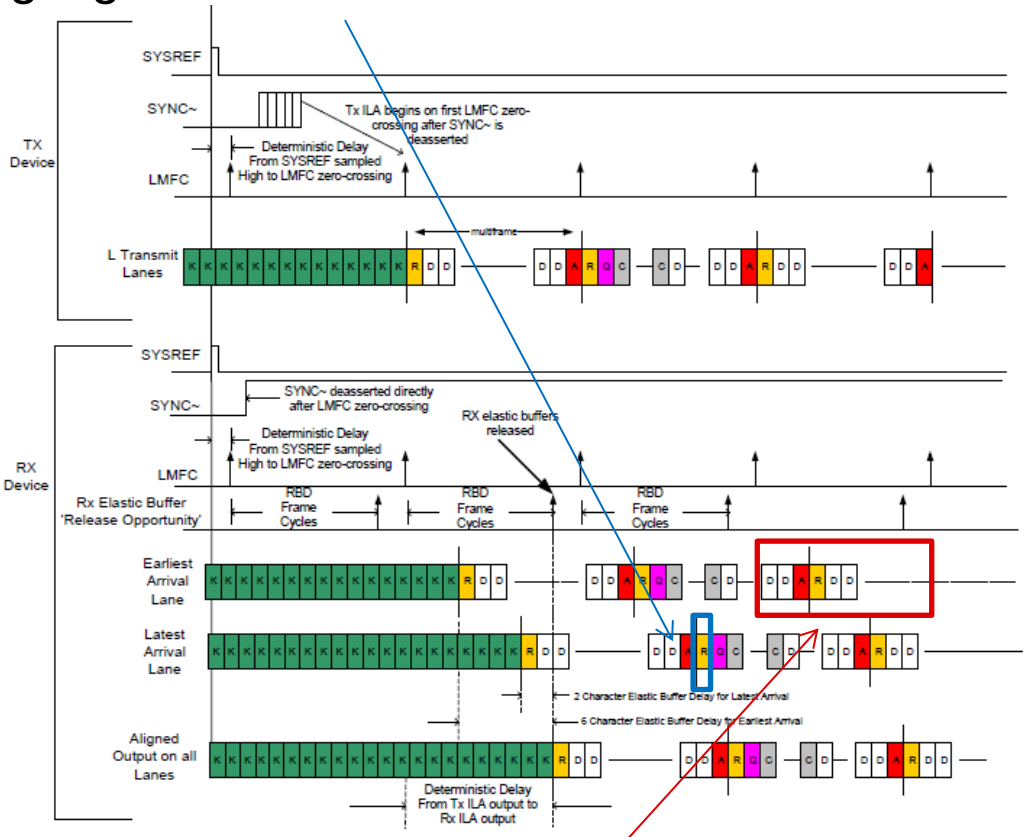


Figure 37 — Timing Diagram Illustration for achieving minimum deterministic latency

- Multi-frame is highlighted in red

Reason for Frame and Multi-frame Boundary

- Frame boundary is needed for both the JESD204B transmitter and JESD204B receiver to have pre-determined way to “process” the bit packing and octet packing of the data octets. This is performed in the transport layer of the JESD204B specification.
- Multi-frame boundary is needed such that the JESD204B transmitter and JESD204B receiver can perform logics in the data path using their own perspective “local clock”. This is the local multi-frame clock (or LMFC), which defines the logical processing clock to process the data octets.
- The ideas is that the LMFC of each perspective JESD204B block can have known, well-defined, and deterministic delay between the data source and data receiver. The clocks do not have to have absolute delay defined in order to simplify clocking solution.
- The deterministic delays can be absorbed by the release buffer in the JESD204B receiver.

Frame Alignment Alarm

- Frame alignment monitoring is constantly performed by check that replacement characters /F/ and /A/ arrive at expected positions in the frame and multiframe.
- When these characters are received, they are replaced with actual data.
- If they are consistently received at unexpected positions, then frame alignment error or the multiframe alignment error would be triggered to indicate the alignment has been lost.
- Frame alignment correction is *not* supported. Therefore, the host need to perform re-alignment and resynchronization of the JESD204B link based on alarm feedback
- The following sections of the JESD204B Standard highlights the Frame and Multi-frame Alignment:
  - 7.2 Initial Frame Synchronization
  - Figure 45 – State Machine for Frame Synchronization in Receivers Supporting Re-initialization
  - 7.3 Frame Alignment Monitoring and Correction
  - 7.4 Initial Lane Synchronization
  - 7.5 Lane Alignment Monitoring and Correction
  - 8.2 Initial Lane Alignment Sequence
  - 8.3 Link Configuration Data and Encoding
  - 5.3.3.4 Frame Alignment Monitoring and Correction.

Possible Source of Frame Alignment and Multi-Frame Alignment Alarms

- "Drift" of SYSREF reset of LMFC over time.
  - SYSREF reset of LMFC change in the "periodic" time instance over time.
  - The overall LMFC boundary changes and potentially cause alignment errors.
  - The SYSREF can be programmed to be ignored in the JESD204B logics of the AFE76xx, both TXDAC and RXADC
- "Drift" of the clock source to various logics.
- "Drift" of incoming data on the JESD204B transmitter over time.
- The incoming data are sufficiently periodic and not sufficiently random. The character replacement rules for the /A/ and /F/ characters are not exercised sufficiently, and hence the periodic frame and multi-frame are not checked frequently.
- Incorrect initialization of the logical block in the JESD204B transmitter. The JESD204B IP was not reset properly upon start-up. Glitch occurs after various temperature cycle.

Link Configuration Error

Overview of Link Configuration Data

- The frame alignment block of the AFE76xx JESD204B RX IP performs the initial frame synchronization, verification of the link configuration parameters, and frame alignment monitoring.
- When the initial lane alignment sequence arrives in this block, the first non-/K/ character is marked as the first octet of the first frame. The link configuration data is expected to arrive in the second multiframe. This block verified that the JESD204B TX lanes are programmed with the same configuration as the JESD204B RX lane.
- Any mismatches in the configuration parameter would trigger the link configuration error.
- See next slide for better pictorial representation.

Initial Lane Alignment JESD204B Configuration Check Location

Link configuration data is expected to arrive on the second multi-frame of the initial lane alignment sequence. See blue area for detail. Basically the second /R/ indicates the ILAS

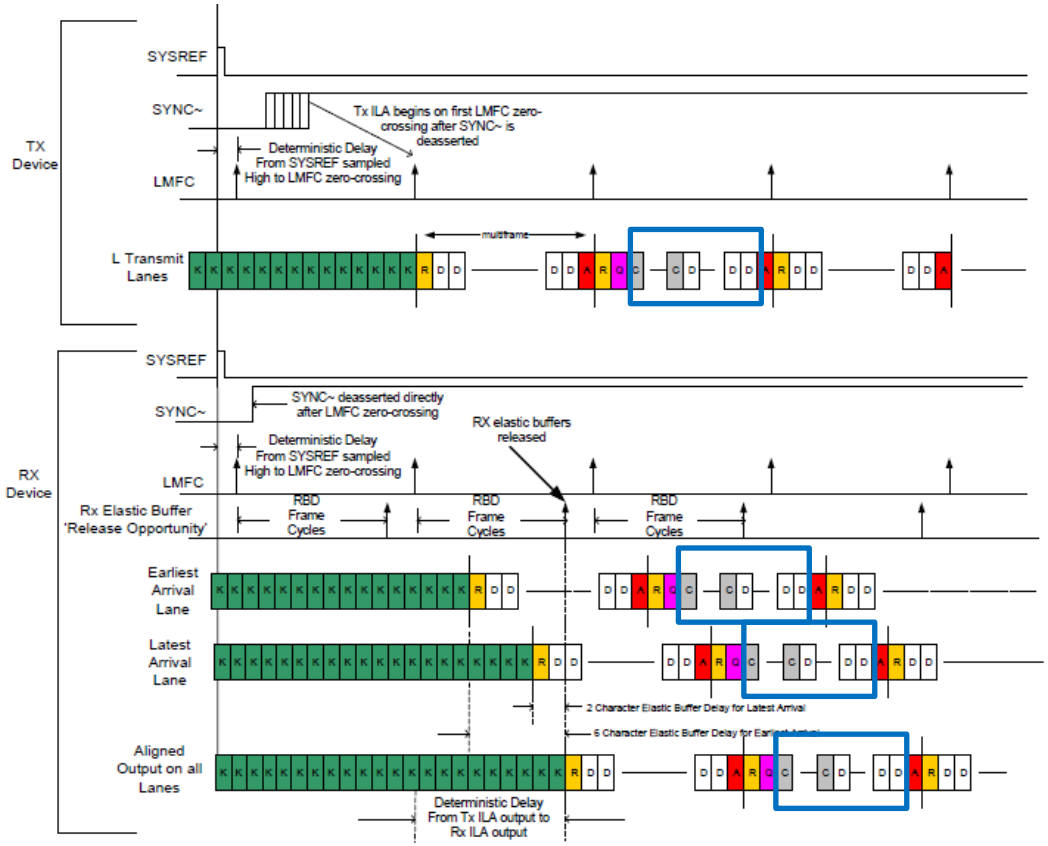


Figure 37 — Timing Diagram Illustration for achieving minimum deterministic latency

Initial Lane Alignment JESD204B Configuration Check

- Each lane should have unique lane ID that is tied to the physical lane.
- Correct programming of the ILAS configuration are needed for both side of the JESD204B link (i.e. ASIC/FPGA and AFE76xx DAC logic) in order for the ILAS check to be successful.

Overview of the JESD204B Link Configuration

Link Configuration Data Requirement

8.3 Link configuration data and encoding

Table 20 — Link configuration parameters

Parameter	Description	Parameter Range	Field	Encoding
ADJCNT	Number of adjustment resolution steps to adjust DAC LMFC. Applies to Subclass 2 operation only.	0 … 15	ADJCNT<3:0>	Binary value
ADJDIR	Direction to adjust DAC LMFC 0 – Advance 1 – Delay Applies to Subclass 2 operation only	0 … 1	ADJDIR<0>	Binary value
BID	Bank ID – Extension to DID	0 … 15	BID<3:0>	Binary value
CF	No. of control words per frame clock period per link	0 … 32	CF<4:0>	Binary value*
CS	No. of control bits per sample	0 … 3	CS<1:0>	Binary value
DID	Device (= link) identification no.	0 … 255	DID<7:0>	Binary value
F	No. of octets per frame	1 … 256	F<7:0>	Binary value minus 1
HD	High Density format	0 … 1	HD<0>	Binary value
JESDV	JESD204 version 000 – JESD204A 001 – JESD204B	0 … 7	JESDV<2:0>	Binary Value
K	No. of frames per multiframe	1 … 32	K<4:0>	Binary value minus 1
L	No. of lanes per converter device (link)	1 … 32	L<4:0>	Binary value minus 1
LID	Lane identification no. (within link)	0 … 31	LID<4:0>	Binary value
M	No. of converters per device	1 … 256	M<7:0>	Binary value minus 1
N	Converter resolution	1 … 32	N<4:0>	Binary value minus 1
N'	Total no. of bits per sample	1 … 32	N'<4:0>	Binary value minus 1
PHADJ	Phase adjustment request to DAC Subclass 2 only.	0 … 1	PHADJ<0>	Binary value
S	No. of samples per converter per frame cycle	1 … 32	S<4:0>	Binary value minus 1
SCR	Scrambling enabled	0 … 1	SCR<0>	Binary value
SUBCLASSV	Device Subclass Version 000 – Subclass 0 001 – Subclass 1 010 – Subclass 2	0 … 7	SUBCLASSV <2:0>	Binary Value
RES1	Reserved field 1	0 … 255	RES1<7:0>	Binary value
RES2	Reserved field 2	0 … 255	RES2<7:0>	Binary value
CHKSUM	Checksum Σ(all above fields)mod 256	0 … 255	FCHK<7:0>	Binary value

* *CF*==*L* shall always be encoded as 31: control words on all lanes. *CF*==31 can only occur when *L*==31, see 5.1.3.

Programmer Tool

Microsoft Excel Worksheet

AFE76xx Configuration for Link Configuration Check

Process	Register Address (Hex)	Register Value (Hex)	Comment	TI Note
W	10	55	Open page for all four TX DUC path	Please note each DUC path may be independently programmed for different ILAS sequence check
W	85	TBD	ADJDIR and ADJCNT	
W	86	TBD	BID	
W	87	TBD	PHADJ, CS, and CF	
W	88	TBD	DID	
W	89	TBD	RES1	
W	8A	TBD	RES2	
W	8D	TBD	HD and L	
W	8E	TBD	M	may be different than actual JESD204B Mode
W	8F	TBD	F	may be different than actual JESD204B Mode
W	90	TBD	K	may be different than actual JESD204B Mode
W	91	TBD	S	may be different than actual JESD204B Mode
W	95	TBD	LID0	all DUCs check the same JESD204B lanes (8 lanes to the AFE76xx SRX)
W	96	TBD	LID1	all DUCs check the same JESD204B lanes (8 lanes to the AFE76xx SRX)
W	97	TBD	LID2	all DUCs check the same JESD204B lanes (8 lanes to the AFE76xx SRX)
W	98	TBD	LID3	all DUCs check the same JESD204B lanes (8 lanes to the AFE76xx SRX)
W	99	TBD	LID4	all DUCs check the same JESD204B lanes (8 lanes to the AFE76xx SRX)
W	9A	TBD	LID5	all DUCs check the same JESD204B lanes (8 lanes to the AFE76xx SRX)
W	9B	TBD	LID6	all DUCs check the same JESD204B lanes (8 lanes to the AFE76xx SRX)
W	9C	TBD	LID7	all DUCs check the same JESD204B lanes (8 lanes to the AFE76xx SRX)
W	10	0	Close page	

Error Handling and Things to Check

- If link cannot be established, try to ignore ILAS check in SYNC_REQUEST to check if the JESD204B link can be established properly.
  - Note the keyword is “ignore”, not skip. Skipping the ILAS will default to subclass 0 mode, which is not supported by AFE76xx.
- Check the ASIC/FPGA ILAS setting and sequence.
- Probe the SERDES line with high speed scope to trigger on first non-K28.5 character to decode the ILAS sequence
- Go through the sequence to see if the octets are properly replaced for ILAS
- Run link layer test to test for ILAS sequence.

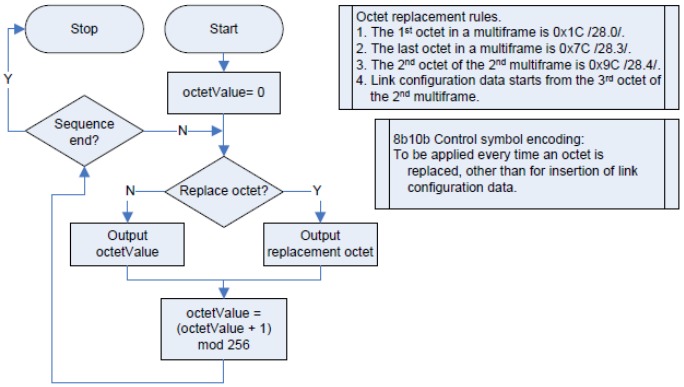


Figure 50 — Character specification for initial lane alignment sequence

Elastic Buffer Overrun

DAC JESD204B K value Setting

- For the AFE76xx DAC JESD204B logics, the number of octets in RBD buffer is 64 octets.
  - More octets requires more logics for the RBD buffer, which consumes silicon area and power consumption.
  - 64 octets of RBD buffer is used in most of the TI DAC JESD204B logics (and most of the FPGA HDL implementation)
- F = number of octets per frame
- K = number of frames in multi-frame
- 64/F = optimal number of values within the RBD buffer
  - For instance, in TX = 44210 mode, F = 2. The optimal K set to allow full-usage of RBD buffer is 64/2 = 32, or K = 32.

DAC JESD204B K value Setting

- K greater than 64/F can be allowed. However, not all spaces in the RBD will be valid. A sweep of the optimal RBD value is required.
  - For instance, for TX setting of 24410 mode, F = 4. The optimal range of RBDs are 64/4 = 16.
  - However, K = 32 can still be set.
  - The optimal range of 16 is within the K = 32 if K = 32 is set.
- The valid range of RBD depends on system delay such as SERDES traces, SYSREF delay, SYNC delay, etc
- The valid RBD range may be from 0 to 15, 1 to 16, 2 to 17, or even 17 to 32, etc. The length of range would be the same, however, the starting or ending point may be different depending system parameters.

DAC JESD204B K Value Setting

- Setting K value to be the minimum:
  - Pros:
    - Easy to find the optimal RBD range. Most of the time, RBD range is greater than or equal to K.
    - Lowest latency across JESD204B. The LMF clock is set by the K value. The lower the K, the faster than LMF clock
  - Cons:
    - The FPGA or ASIC may not support lowest value of Ks. The system clock is the LMF clock, and may need to run faster with lower K values

RBD Adjustment Location

Process	Register Address (Hex)	Register Value (Hex)	Comment	TI Note
Comment			TX DAC JESD204B and DUC configuration	
Comment			TXDIG Section	
W	11	3	Open TXDIGP0 for DACA/B and DACC/D	
W	21	7	Invert the SERDES RX clock capture edge in the TXDIGP0 block	
W	2C	3	turn on the DAC encoder clk_sync_ena	
W	11	0	Open TXDIGP0 for DACA/B and DACC/D	
W	11	3	Open TXDIGP0 for DACA/B and DACC/D	
W	20	3	set JESD204B RX INIT_STATE = 1b1' for all DACs	
W	11	0	Close TXDIGP0 for DACA/B and DACC/D	
Comment			TXDUC Section	
W	10	55	Open TXDUCP0 page for DACA/B and DACC/D	
W	25	0B	22210_12x JESD mode (jesd_mode = 5b'01011. jesd_mode registers are bits4:0)	Application Dependent
W	26	11	Select 18x interpolation	Application Dependent
W	7D	12	Select phasemode(jesd_phase_mode = 2b'00. Jesd_phase_mode = bits6:5), subclass=1,204B=1	Application Dependent
W	28	0	turn off mc_sum_ena and set dual_iq=0	Application Dependent
W	4E	45	set clk_jesd_div and clk_jesd_out_div	Application Dependent
W	7E	application dependent	RBD value for JESD204B RX for DAC	Application Dependent. Please note the RBD range may need to be adjusted based on lane delays
W	7F	application dependent	K value for JESD204B RX for DAC	Application Dependent
W	84	3	turn on match control and match specific character for RBD release	
W	8B	1F	multi-frame, frame, and link config do not cause SYNC request	Application Dependent. Refer to DAC JESD204B RX IP sync request documentation
W	8C	0	no error reporting over SYNC	Application Dependent. Refer to DAC JESD204B RX IP sync request documentation
W	53	0	turn off gating of SYSREF to DSP blocks in TXDUC	
Comment			TX DAC encoder section	
W	11	3	Open TXDIGP0 for DACA/B and DACC/D	
W	2B	33	enable QRCLK and FRCLK for all four DAC cores	
W	29	3	enable performance mode for all four DAC cores	
W	11	0	Close TXDIGP0 for DACA/B and DACC/D	
Comment			SYSREF usage	
W	10	55	Open TXUDCP0 page for DACA/B and DACC/D	
W	4D	0	CDRV_SER init uses no SYSREF	
W	4D	2	CDRV_SER init uses only the next SYSREF	
W	80	0	JESD204B RX IP init uses no SYSREF	
W	80	A	JESD204B RX IP init uses skip two and then next	
W	10	0	Close TXDIGP0 for DACA/B and DACC/D	

JESD204B Specification

6 Deterministic Latency

6.1 Introduction

Many JESD204 systems contain various data processing elements that are distributed across different clock domains and lead to ambiguous delays through the interface. These ambiguities lead to non-repeatable latencies across the link from power-up to power-up or over link-reestablishments. JESD204A did not provide for a mechanism to make the interface latency deterministic. JESD204B provides two possible mechanisms for this purpose, defined as Subclass 1 and Subclass 2 operation.

The deterministic latency across the link is defined from the parallel frame-based data input on the TX device to the parallel frame-based data output on the RX device, all measured within the frame clock domain (Refer to Figure C.4 and Figure C.7 for reference). This latency across the link shall be programmable in units at least as small as the frame clock period, and shall be repeatable from power-up cycle to power-up cycle and across link re-synchronization events, provided that the auxiliary timing signals meet the required specifications at the device inputs.

The achievement of deterministic delay across a link involves two requirements:

1. In the TX device, ILA generation must be initiated simultaneously across all lanes at a well-defined moment in time. (This also ensures that user data following the ILA is initiated simultaneously across all lanes at a well-defined moment in time).
   a. The 'well-defined moment in time' for ILA generation (and hence user data generation) in the TX device is the first LMFC boundary after the detection of the SYNC~ rising edge. While TX devices must be able to generate the ILA on the first LMFC boundary after the detection of the SYNC~ rising edge, devices may also support a programmable number of additional LMFC boundaries to wait before starting the ILA sequence.
2. In the RX device, the incoming data on each lane must be buffered to account for skews across TX SERDES lanes, physical channels, and RX SERDES lanes. The RX buffers must be released (i.e., data allowed to propagate) simultaneously across all lanes at a well-defined moment in time.
   a. The 'well-defined moment in time' for RX buffer release is a programmable number of frame cycles after an LMFC boundary. This programmable number of frame cycles is referred to as the Rx Buffer Delay (RBD). For details on which LMFC boundary to use, see the Example given in this clause.

The ILA generation and Rx Buffer Release alignments mentioned above are related to the LMFCs in the TX and RX devices. Thus, the achievement of deterministic latency with minimum uncertainty relies on aligning the LMFCs within TX and RX devices as closely as possible.

RBD used to be released on Multi-frame (K)

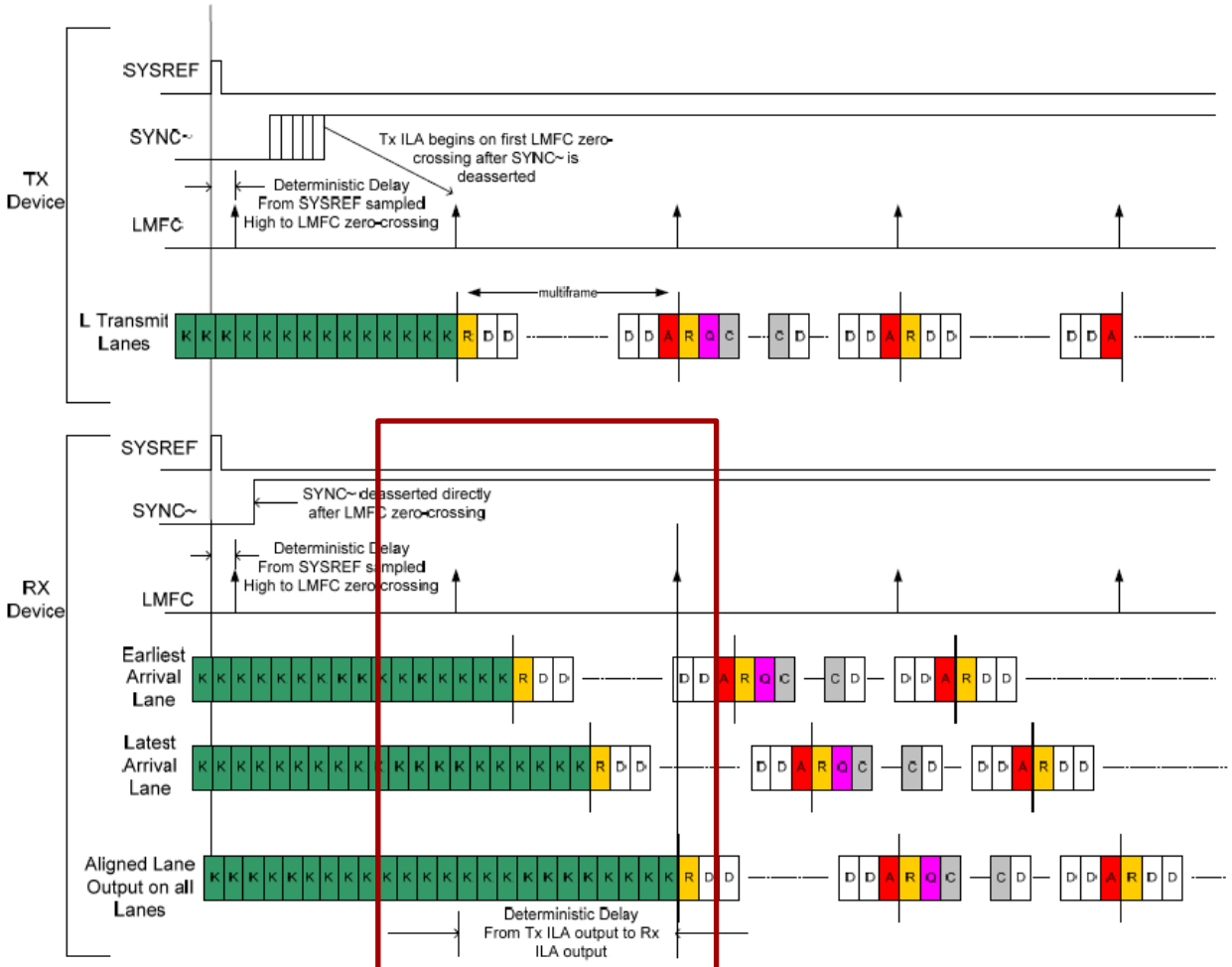


Figure 36 — Timing Diagram Illustration for deterministic latency equal to multiple of multiframe period

RBD used for minimum deterministic latency

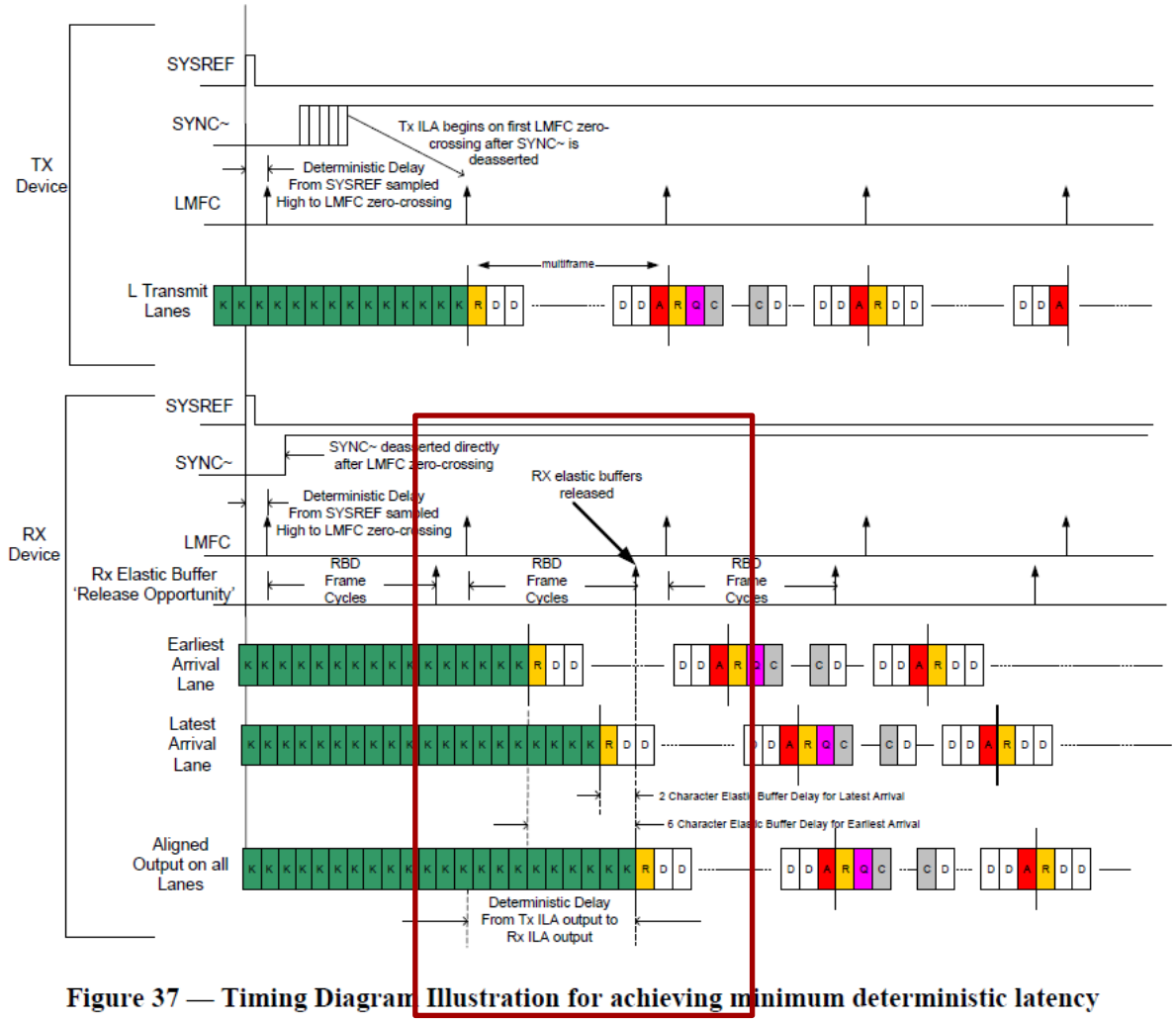
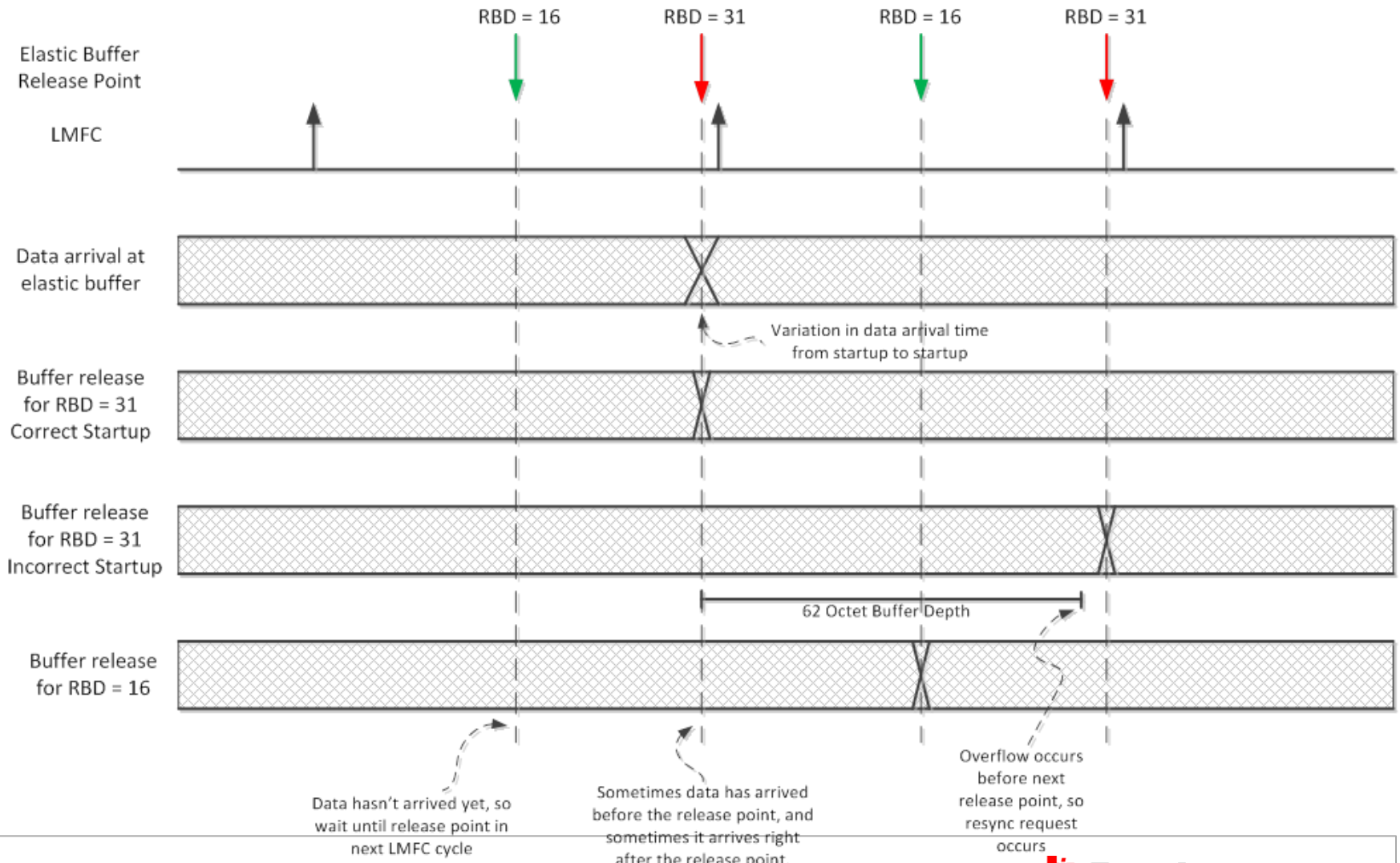


Figure 37 — Timing Diagram Illustration for achieving minimum deterministic latency

RBD release


RBD = 16
RBD = 31
RBD = 16
RBD = 31
Elastic Buffer Release Point
LMFC
Data arrival at elastic buffer
Variation in data arrival time from startup to startup
Buffer release for RBD = 31 Correct Startup
Buffer release for RBD = 31 Incorrect Startup
62 Octet Buffer Depth
Buffer release for RBD = 16
Data hasn't arrived yet, so wait until release point in next LMFC cycle
Sometimes data has arrived before the release point, and sometimes it arrives right after the release point.
Overflow occurs before next release point, so resync request occurs

Additional Technical Information

- Regarding the RBD parameter, setting it to 16 is technically earlier than 31. However, the release points are periodic. There are two conditions for releasing the buffer. First, all lanes must have arrived (meaning the ILA sequence) and raised their “ready” flag. Second, a buffer release point must occur. The release points are periodic and occur RBD frame cycles after every LMFC cycle. If we miss the release point within one LMFC cycle, then the buffer will continue buffering and simply release at the release point in the next LMFC cycle. Since we have observed that data is arriving near the end of an LMFC cycle (because RBD of 31 is showing signs of an incorrect release point), we can simply set RBD to a lower value (e.g. 16) such that the release point will occur 16 frame cycles after the LMFC edge. In this case, changing from RBD of 31 to RBD of 16 causes the buffer to release 17 frame cycles later.
- The elastic buffer is 64 octets deep. In 442 mode with K of 32, our multiframe is 64 octets long. We’re seeing the buffer overflow error because data for the earliest lane arrives before the release point and data for the latest lane arrives after the release point causing the elastic buffer release to wait until the next release point in the next multi-frame. Since the release points are periodic at the multiframe rate this means the buffer will need to buffer an additional 64 octets of data. Since the buffer is 62 octets deep we can’t store the additional 64 octets due to missing the release point and therefore the buffer overflows.
- http://www.ti.com/lit/ml/slap159/slap159.pdf

Example RBD Sweeps

- RBD is swept from 1 to K over 10 times for each RBD setting.
- For successful JESD204B bring-up, an “O’ is marked
- For unsuccessful JESD204B bring-up with errors and alarms, an “X” is marked.
- Valid RBD window can be determined from the table below, and an optimal RBD setting can be determined.

Band X	RBD Value																														
#	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	O	O	O	O	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
2	O	O	O	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
3	O	O	X	X	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
4	O	O	O	X	O	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
5	O	O	O	X	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
6	O	O	X	X	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
7	O	O	O	X	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
8	O	O	O	O	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
9	O	O	O	X	X	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
10	O	O	O	X	X	X	X	X	X	X	X	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O

Skew Indication of Various Lanes

- Key Register: 0x3C in TXDUCP0 page, lane_skew
- Measure the lane skew among the lanes within each TXDUC logic (note: this does not go beyond the boundary of the TXDUC and cannot cross the TXDUCs. i.e. cannot measure between skew of DAC-A to DAC-C).
- This measurement automatically updates and holds the output value whenever the elastic buffers are released. This could happen repeatedly if the skew causes buffer overflow, which subsequently causes repeated synchronization requests. Only a reset or init-state of the JESD204B RX IP will clear the output.

Elastic Buffer Match Error

Description of Elastic Buffer Match Error

- Key Register: 0x84 of TXDUCP0 page Match_Specific
- If set to 1b’0, the JESD204B RX buffer will start buffering with the first non-/K/ value and ignore match control character. Programming match_specif to 1b’0 should not be used unless for debugging purpose
- If set to 1b’1, the JESD204B RX buffer will start buffering only with /R/ character, immediately followed by /K/ after the completion of CGS
- If the first character after the /K/ in the CGS, then the elastic buffer match error will be flagged.

Error Handling and Things to Check

- If error occurs, please check the JESD204B TX IP to see if the IP is set to subclass 0 mode as oppose to subclass 1 mode.
- Check with high speed scope to see if the actual /R/ is detected after the /K/ characters in the CGS.
- Set match_specific to 1b’0 to see if the link establishment can proceed.
- Check if the SERDES receivers are detecting any bit errors, especially with the possibility of the AFE76xx SERDES RX being potentially saturated after long period of repeating /K/ patterns.
- Minimize the length of /K/ pattern by reducing the length variation and delay variation of the JESD204B lanes and also data logic paths.

Code Group Synchronization

Code Group Synchronization Check

- The lane alignment block performs code group synchronization and implements the elastic buffer.
- Code group synchronization is achieved when the lane has received four consecutive /K/ characters successfully after asserting a synchronization request (~SYNC) and then another four valid characters after de-asserting the synchronization request.
- If three invalid characters are received within a certain amount of time, code group synchronization would be lost.
- This would trigger the code group synchronization error.
- See the following JESD204B standard:
  - 7.1 Code Group Synchronization
  - Figure 44 – Receiver State Machine for Code Group Synchronization

Code Group Synchronization State Diagram

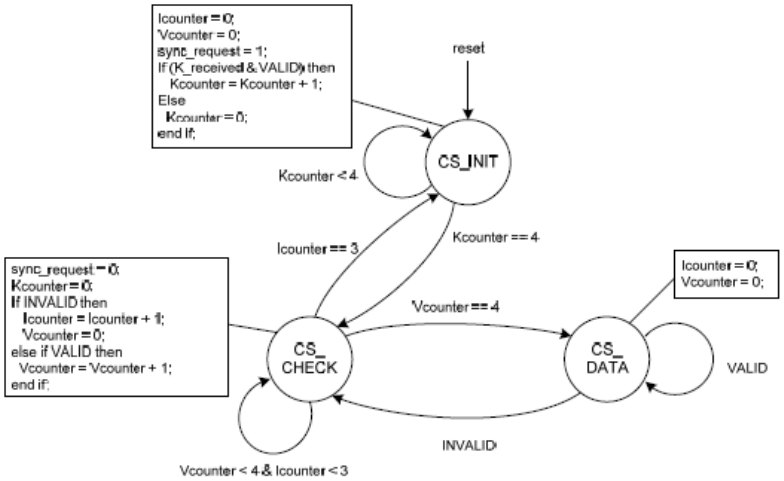


Figure 44 — Receiver state machine for code group synchronization

Table 14 — Variables used in receiver state machine for code group synchronization

Variable	Meaning
Icounter	Counter used in the CS_CHECK phase to count the number of invalid symbols
INVALID	Asserted by receiver to indicate that the current symbol is an invalid symbol given the current running disparity.
K_received	Asserted when the current symbol corresponds to control character K28.5
Kcounter	Counter used in the CS_INIT phase to count the number of valid K28.5 symbols
sync_request	Asserted by receiver when loss of code group synchronization has been detected. Note that sync_request does not drive SYNC~ directly, as SYNC~ assertion/de-assertion is based on more than just the sync_request signal described here.
VALID	Asserted by receiver to indicate that the current symbol is a valid symbol given the current running disparity.
Vcounter	Counter used in the CS_CHECK phase to count the number of successive valid symbols

The receiver is allowed and required to align the code group boundary to received comma characters only during an active synchronization request. During data transmission, commas can be detected across the border of two code groups as the result of bit errors. Spurious commas can also be generated across the border of a frame alignment symbol /K28.7/ and certain data symbols.

Error Handling and Things to Check

- If error occurs, please check the JESD204B TX IP to see if the IP initialized correctly with proper CGS state machine running
- Check with high speed scope to see if the actual /K/ are detected
- Run link layer testing to check for validity of the /K/ CGS patterns.
- Check if the SERDES receivers are detecting any bit errors, especially with the possibility of the AFE76xx SERDES RX being potentially saturated after long period of repeating /K/ patterns.
- Minimize the length of /K/ pattern by reducing the length variation and delay variation of the JESD204B lanes and also data logic paths.

8B/10B Errors

8B/10B Encoding and Decoding Errors

- Disparity error: the received code group exists in the 8b/10b decoding table, but is not found in the proper column to the current + or – running disparity.
- Not-in-table error: the received code group is not found in the 8b/10b decoding table for either disparity. Usually a true bit error

Table 18 — Minimum set of errors to detect per receiver

Error	Description
Disparity error	The received code group exists in the 8B/10B decoding table, but is not found in the proper column according to the current running disparity.
Not-in-table error	The received code group is not found in the 8B/10B decoding table for either disparity
Unexpected control character	A control character is received that is not expected at the given character position
Code group synchronization error	The state machine for code group synchronization has returned to the CS_INIT state

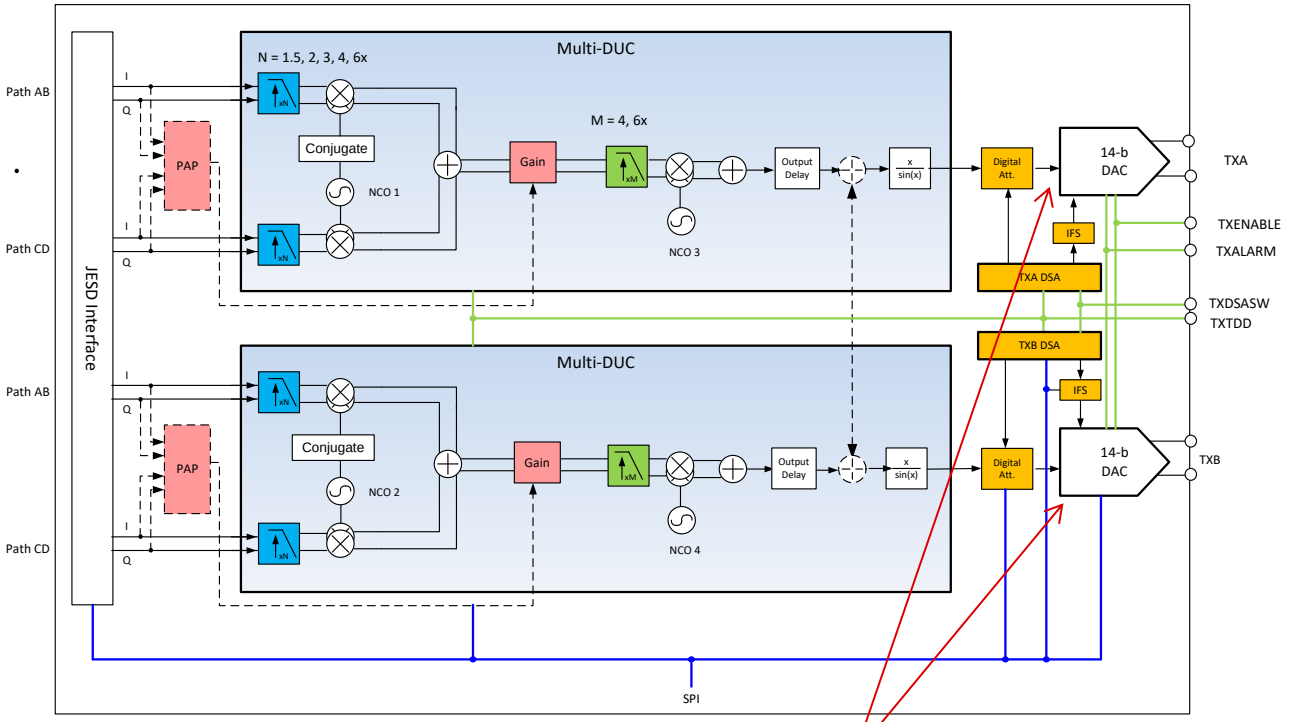
Error Handling and Things to Check

- If error occurs, please check the JESD204B TX IP to see if the IP initialized correctly with proper 8b/10b coding logics running.
- Typically, both 8b/10b not-in-table and 8b/10b disparity occur simultaneously, and indicate general bit error due to the SERDES signal conditioning.
- If only 8b/10b disparity occurs, the error may lead to the 8b/10b logic in the JESD204B not being reset properly.
- Check with high speed scope to check the eye diagram for SERDES signal quality.
- For PG3.0, PRBS pattern such as PRBS7 may be run over period of time for bit error check.
- Check if the SERDES receivers are detecting any bit errors, especially with the possibility of the AFE76xx SERDES RX being potentially saturated after long period of repeating pattern.

TX Output Zeroing Logic

Alarm_Zeros_TXENABLE_ENA TXDUCP0 Page, 0x29 Register, bit 7

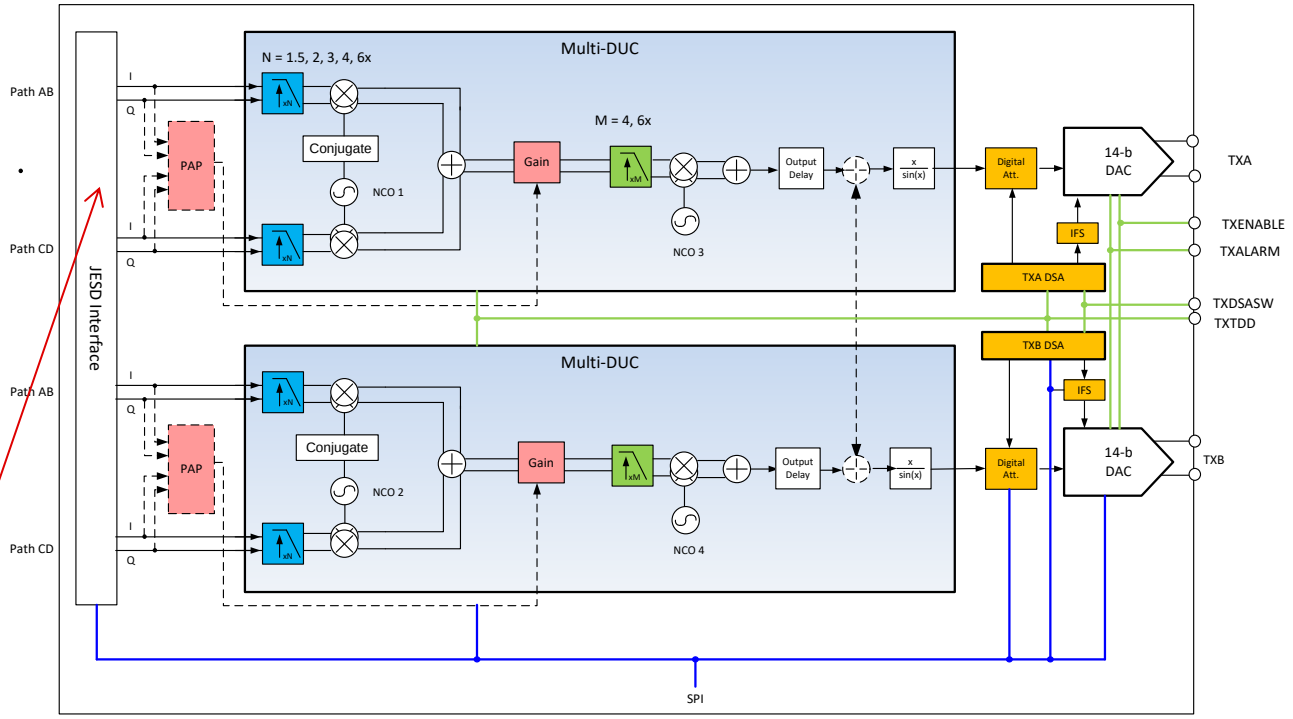
- This logics allows the DAC output (DAC output core) to be mid-scale (zeroed) upon any alarms that are not masked in the alarm register (interruptible alarms)



Zeroing in the DAC core by setting the DAC core to mid-scale upon any interrupt

Alarm_Zeros_JESD_DATA_ENA TXDUCP0 Page, 0x29 Register, bit 5

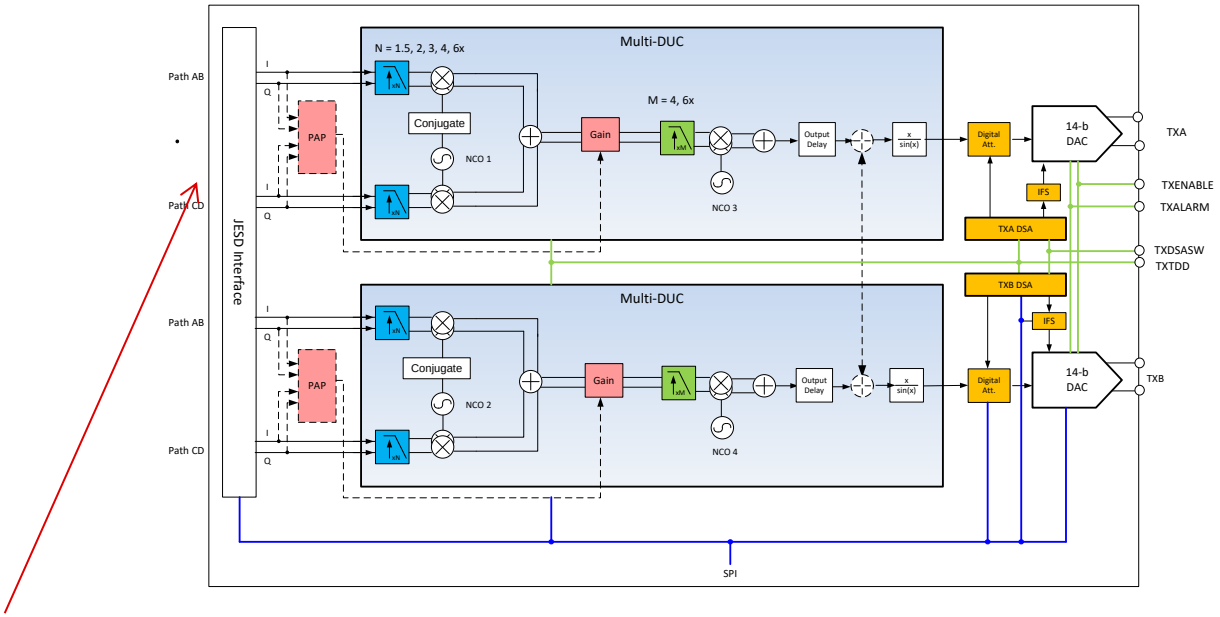
- This logics allows the JESD204B RX IP block to start sending zeros upon any interruptible alarms.



Zeroing in JESD204B RX IP upon any interruptible alarms.

FIFO_ERROR_ZEROS_DATA_ENA TXDUCP0 Page, 0x2D Register, bit 5

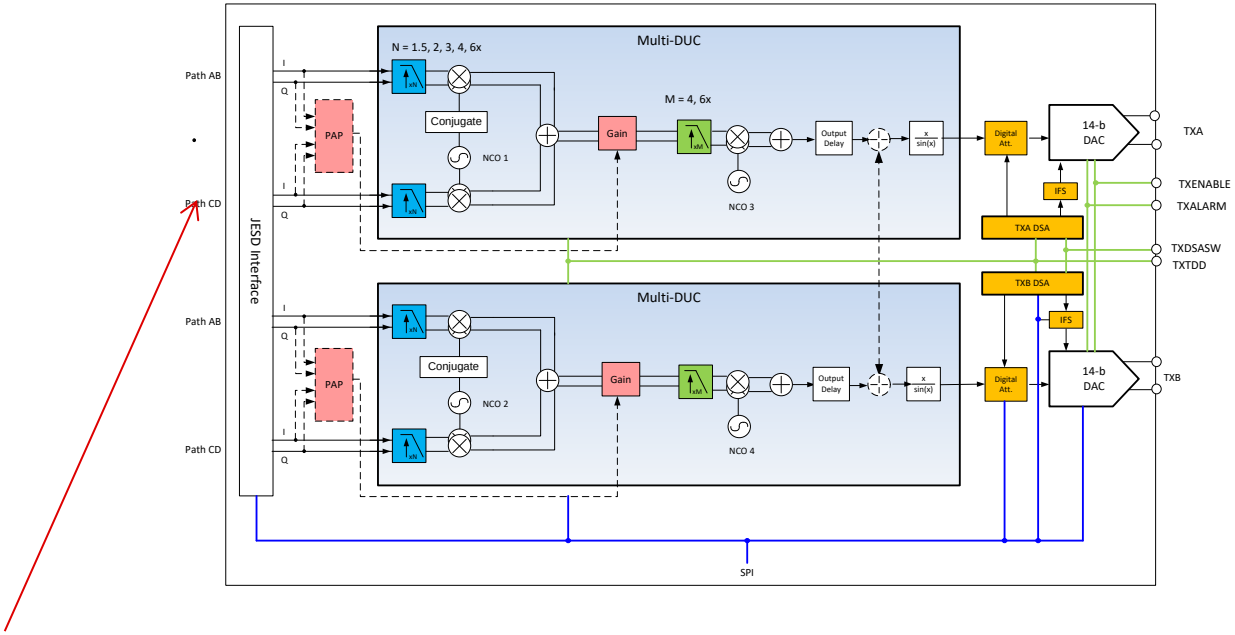
- This logics allows the JESD204B RX IP block to start sending zeros upon any errors from the SERDES RX IP -> FIFO -> JESD204B RX IP hand-off. (i.e. FIFO boundary hand-off errors)
- FIFO error is mainly contributed due to clock configuration error or PLL lock error.



Zeroing in JESD204B RX IP upon FIFO hand-off error

ZERO_INVALID_DATA
TXDUCP0 Page, 0x2A Register, bit 3

- This logics allows the JESD204B RX IP block to start sending zeros any invalid data (even if 8b/10b disparity or not-in-table error report or re-sync requests are turned off).
- Invalid data is any code group that is not found in the proper column of the 8B/10B decoding tables, according to the current running disparity (IEEE 802.3)



Zeroing in JESD204B RX IP upon invalid data