JEDEC STANDARD

Serial Interface for Data Converters

wenzaifadian

JESD204C.1

(Revision of JESD204C December 2017)

DECEMBER 2021

JEDEC SOLID STATE TECHNOLOGY ASSOCIATION

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or refer to www.jedec.org under Standards and Documents for alternative contact information.

weldfadran

Published by
©JEDEC Solid State Technology Association 2021
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

This document may be downloaded free of charge; however JEDEC retains the copyright on this material. By downloading this file the individual agrees not to charge for or resell the resulting material.

PRICE: Contact JEDEC

Printed in the U.S.A.
All rights reserved

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

PLEASE!

DON'T VIOLATE
THE
LAW!

This document is copyrighted by JEDEC and may not be
reproduced without permission.

For information, contact:

JEDEC Solid State Technology Association
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

or refer to www.jedec.org under Standards-Documents/Copyright Information.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

CONTENTS_Toc62286702

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

CONTENTS (cont'd)

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

CONTENTS (cont'd)

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

CONTENTS (cont'd)

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

CONTENTS (cont'd)

weldradian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

CONTENTS (cont'd)

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

FIGURES

Weisfadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

FIGURES (cont'd)

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

FIGURES (cont'd)

weidaguan

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

TABLES

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

TABLES (cont'd)

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Foreword

The JESD204 standard separates the communication mechanism between logic devices and data converters into three layers, each with a distinct function: physical (specified in clause 5), link (64B/66B and 64B/80B encoding specified clause 0, and 8B/10B encoding specified in clause 0), and transport (specified in clause 0). The JESD204 physical layer encompasses the PMA and PMD Ethernet layers; the JESD204 link layer encompasses the FEC and PCS Ethernet layers and additional CRC and alignment functionality. Finally, the JESD204 transport layer maps to the MAC Ethernet layer (Figure 1).

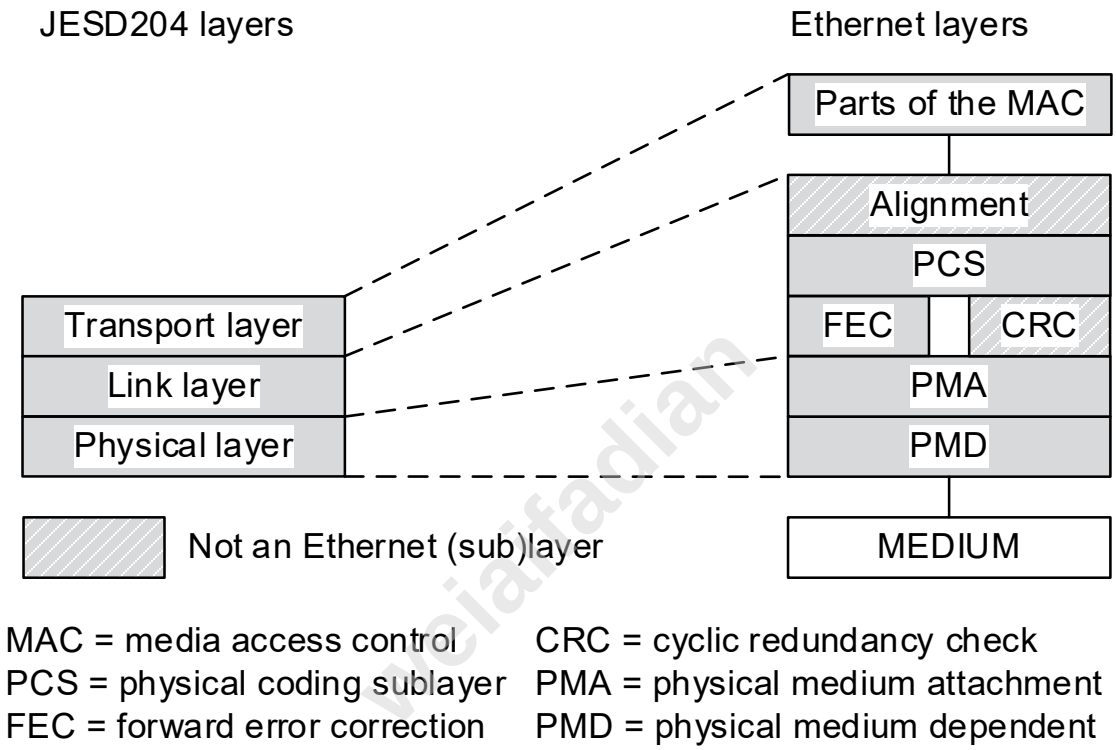


Figure 1 — JESD204 layer relationship to the IEEE Ethernet model

The JESD204C.01 standard replaces JESD204C, which replaced JESD204B standard.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

SERIAL INTERFACE FOR DATA CONVERTERS

(From JEDEC Board Ballot JCB-21-56, formulated under cognizance of the JEDEC JC-16 Committee on Interface Technology.)

1 Scope

This standard describes a serialized interface between data converters and logic devices. It contains normative information to enable designers to implement devices that communicate with other devices covered by this specification. Informative annexes are included to clarify and exemplify the document.

Due to the range of applications involved, the intention of this standard is to completely specify only the serial data interface and the link protocol. Certain signals common to both the interface and the function of the device, such as device clocks and control interfaces, have application-dependent requirements. The JESD204 standard does not require a specific implementation of the control interface, however a serial interface is the recommended implementation. Devices may also have application-dependent modes, such as a low power / shutdown mode that will affect the interface. In these instances, the specification merely constrains other device properties as they relate to the interface, and leaves the specific implementation up to the designer.

Revision A of the standard was expanded to support serial data interfaces consisting of single or multiple lanes per converter device. In addition, converter functionality (ADC or DAC) can be distributed over multiple devices:

- All parallel running devices are implemented or specified to run synchronously with each other using the same data format.
- Normally this means that they are part of the same product family.

Revision B of the standard supports the following additional functions:

- Mechanism for achieving repeatable, programmable deterministic delay across the JESD204 link.
- Support for serial data rates up to 12.5 Gbps.
- Transition from using frame clock as the main clock source to using device clock as the main clock source. Device clock frequency requirements offer much more flexibility compared to requiring a frame clock input.

Revision C of the standard now supports the following additional functions:

- Data interfaces up to 32.45 Gbps.
- Three link layers – 64B/66B, 64B/80B, and 8B/10B. The 8B/10B link layer is similar to the link layer defined in JESD204B.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

1 Scope (cont'd)

Figure 2 compares the scope of the original JESD204 specification and its revisions. Although not illustrated in the figure, it is possible to apply multiple, independent instances of the JESD204 standard to the same device. The logic device (e.g., ASIC or FPGA) is always assumed to be a single device.

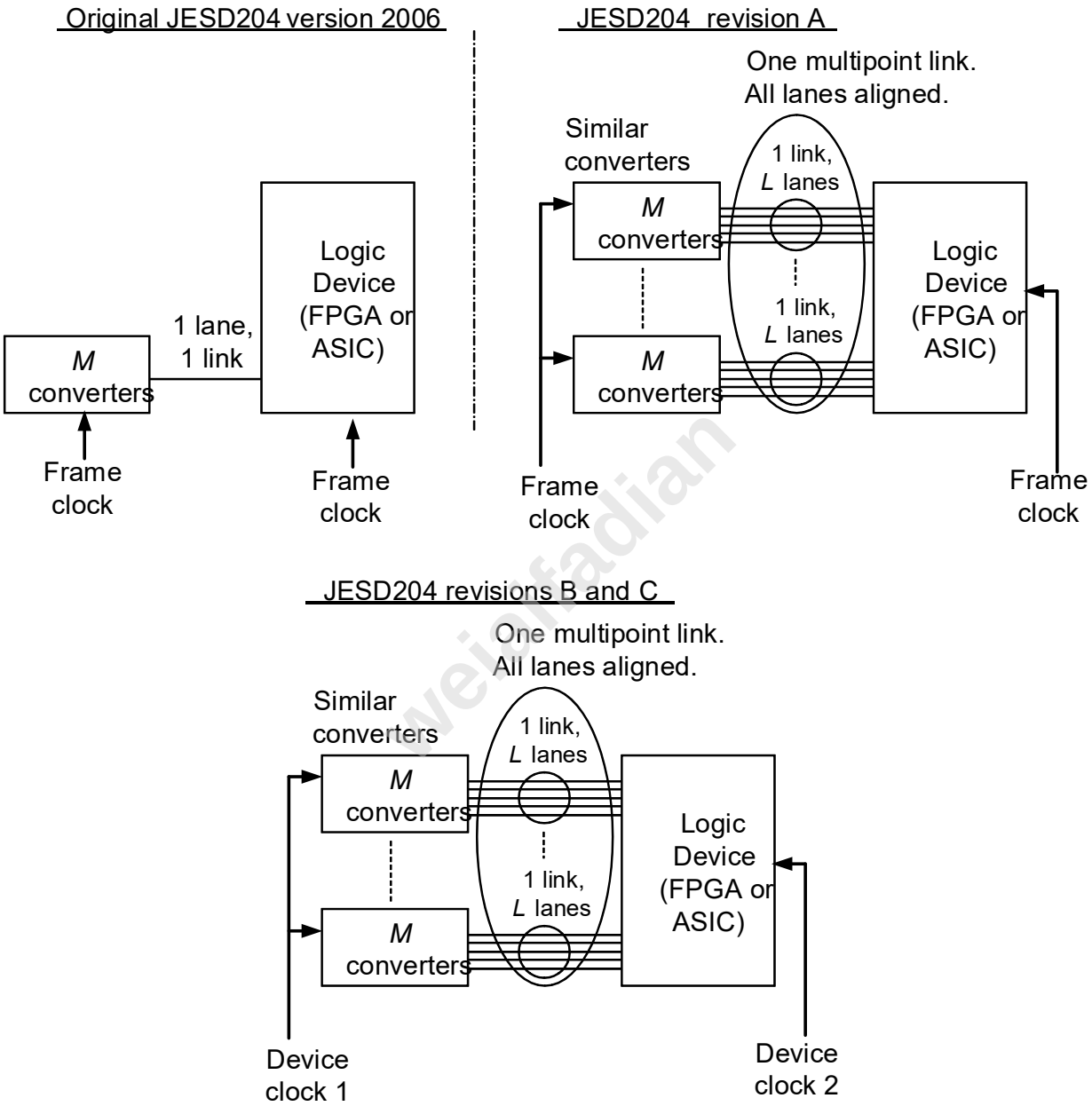


Figure 2 — Scope of original JESD204 and revisions A, B, and C

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

2 Normative references

The following normative documents contain provisions that, through reference in this text, constitute provisions of this standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies.

1. JEDEC JESD99, Terms, Definitions, and Letter Symbols for Microelectronic Devices.

2. IEEE Standard for Ethernet, IEEE Std. 802.3™-2018 section 3, clause 36, *Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer, type 1000BASE-X*. https://standards.ieee.org/standard/802_3-2018.html.

3. IEEE Standard for Ethernet, IEEE Std. 802.3™-2018 section 4, annex 48A, *Jitter test patterns.* https://standards.ieee.org/standard/802_3-2018.html.

4. OIF-SxI-5-01.0, System Interface Level 5 (SxI-5): Common Electrical Characteristics for 2.488 – 3.125 Gbps Parallel Interfaces, Optical Internetworking Forum, October 2002. http://www.oiforum.com/wp-content/uploads/OIF-SxI5-01.0.pdf.

5. OIF-CEI-04.0, Common Electrical I/O (CEI) - Electrical and Jitter Interoperability agreements for 6G+ bps and 11G+ bps and 25G+ bps I/O, Optical Internetworking Forum, February 18, 2017 04.0http://www.oiforum.com/wp-content/uploads/OIF_CEI_04.0.pdf.

6. INCITS TR-35-2004 (R2009), Fibre Channel - Methodologies for Jitter and Signal Quality Specification (FC-MJSQ), available from http://webstore.ansi.org.

7. INCITS 450-2009, *Information technology - Fibre Channel - Physical Interface - 4 (FC-PI-4)*, Annex F, *Scrambled test patterns.* Available from http://webstore.ansi.org.

8. INCITS TR-46-2011, *Information technology - Fibre Channel - Methodologies for Signal Quality Specification (FC-MSQS)*, clause 9, *Test patterns*. Available from http://webstore.ansi.org.

For readability and ease-of-use of this specification, in some cases complete clauses, figures and/or tables of reference standards have been copied (with permission) verbatim; for proper attribution and ease of (cross)referencing, the original clause, table and/or, figure is indicated directly and the phrase “Reprinted” is used (for example, Figure 22) In some other cases, the content of the original clause, table and/or figure has been used as the starting point of the material in this specification, but has been modified (some portions may have been deleted, phrasing modified, content augmented, etc.) so as to better fit the intended application. In these cases, the original material is indicated preceded by the phrases “Based on” or “Adapted and reprinted” (for example 5.2.3.1). In all cases, appropriate copyright holders are indicated.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3 Terminology

For the purposes of this standard, the terms and definitions given in JESD204, JESD99 (reference 1), and the following apply:

3.1 Terms and definitions

8B/10B: A DC-balanced octet-oriented data encoding sequence (see IEEE Std. 802.3 [5].)

64B/66B: An encoding type consisting of 64 unaltered user data bits concatenated with a 2-bit sync header (see 0).

64B/80B: An encoding type consisting of 64 unaltered user data bits concatenated with a 2-bit sync header and 14 fill bits (see 0.)

active edge of a clock: The clock edge on which the logic changes state.

NOTE 1 The active edge can be the rising edge, the falling edge, or both.

NOTE 2 The drawings in this standard show the rising edge as active edge.

ADC device: A device containing one or more analog-to-digital converters connected to the JESD204 link.

ADC link: A link from an ADC device to a logic device.

analog-to-digital converter: A functional block converting an analog signal to a digital sampled data stream. If the analog signal is converted to a complex data stream, the ADC is defined as the subblock generating either the real or the imaginary component of the data stream.

BERT: Bit Error Ratio Test or Tester (ref. OIF-CEI-04.0.)

block (64B/80B): A structure starting with a 2-bit sync header containing 80 bits total.

block (64B/66B or 64B/80B): A structure starting with a 2-bit sync header containing either 66 or 80 bits total.

CDR: Clock Data Recovery (ref. OIF-CEI-04.0.)

ceil(x): The smallest integer greater than or equal to x.

character: A symbol produced by 8B/10B encoding of an octet.

NOTE 1 While all octets can be encoded as data characters, certain octets can also be encoded as control characters.

NOTE 2 The same character may exist as two different code groups, depending on running disparity.

weirdfadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.1 Terms and definitions (cont'd)

clock generator: A circuit used to generate synchronous, phase aligned device clocks to various devices in the JESD204 system.

NOTE A clock generator circuit can include one or more clock generator devices, but they must use a common source clock.

code group: A set of ten bits that, when representing data, conveys an octet (applies to 8B/10B encoding, see IEEE Std. 802.3 [5].)

command channel: Data stream using extra bandwidth afforded from sync headers (see 7.3.7.)

control interface: An application-specific interface used to pass information (usually status and control information) between a converter device and a logic device and/or between a device and a higher layer application level.

NOTE The JESD204 specification does not require a specific implementation of the control interface, however a serial interface is the recommended implementation.

converter: An analog-to-digital converter (ADC) or digital-to-analog converter (DAC).

NOTE In this standard, a converter is assumed to interface via a single stream of digital samples.

converter device: A component package containing one or more converters.

NOTE This standard specifies the interactions between one logic device and one or more converter devices.

DAC device: A device containing one or more digital-to-analog converters connected to the JESD204 link.

DAC link: A link from a DAC device to a logic device.

data link: An assembly that is controlled by a link protocol enabling data to be transferred from a data source to a data sink and consists of parts of two devices and the interconnecting data circuit, that is ("terminal" replaced by "device" in Ref. ATIS Telecom Glossary [1].)

data rate: speed at which data is transferred measured in bits per second, denoted by the symbol f_b.

descrambler: The inverse of a scrambler. (Ref. ATIS Telecom Glossary [1].)

NOTE The descrambler output is a signal restored to the state that it had when it entered the associated scrambler, provided that no errors have occurred.

device clock: A main clock signal from which a device must generate its local clocks.

digital-to-analog converter: A circuit converting a digital sampled data stream to an analog signal. If the analog signal is converted from a complex data stream, the DAC is defined as the subblock accepting either the real or the imaginary component of the data stream.

EMB_LOCK: A state asserting that extended multiblock alignment has been achieved.

extended multiblock: A set of data containing one or more multiblocks.

fill bit: A bit used to artificially extend the block size in 64B/80B encoding mode.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.1 Terms and definitions (cont'd)

floor(x): The greatest integer less than or equal to x.

frame: A set of consecutive octets in which the position of each octet can be identified by reference to a frame alignment signal (adapted from Ref. ATIS Telecom Glossary.)

frame clock: A signal used for sequencing frames or monitoring their alignment.

NOTE A physical frame clock is not required in this version of this standard, but may be required in the implementation of previous versions.

idle mode: An operating mode used for a converter that is not currently sampling data.

invalid code group: A code group that is not found in the proper column of the 8B/10B decoding tables, according to the current running disparity (see IEEE Std. 802.3 [5].)

lane: A differential signal pair for data transmission in one direction.

link: Synonym for "data link".

local clock: A clock derived inside a device from the device clock and used in the implementation of the JESD204 link within the device.

NOTE 1 It is possible to align a local clock to an external signal, e.g., SYSREF.

NOTE 2 An internal copy of the device clock is not a local clock.

logic device: A component package containing exclusively or primarily digital logic; e.g., an ASIC or FPGA.

NOTE This standard specifies the interactions between one logic device and one or more converter devices.

max(x, y): The largest of x and y.

MB: Multiblock.

medium: The transmission path along which a signal propagates.

min(x, y): The smallest of x and y.

mod(x, y): The remainder after dividing x by y (x modulo y).

multiblock: A set of data containing 32 64B/66B or 64B/80B blocks.

weiaifaqian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.1 Terms and definitions (cont'd)

multiframe: A set of consecutive frames in which the position of each frame can be identified by reference to a multiframe alignment signal (ref. ATIS Telecom Glossary [1].) Applicable to the 8B/10B encoding option (see 0.)

NOTE 1 The multiframe alignment signal does not necessarily occur in each multiframe.

NOTE 2 In JESD204, a multiframe consists of *K* frames and is transmitted over a single lane.

multiframe clock: A signal used for sequencing multiframes or monitoring their alignment. Applicable to the 8B/10B encoding option (see 0.)

multipoint link: A data communications link that interconnects three or more devices ("terminal" replaced by "device" in Ref. ATIS Telecom Glossary [1].)

nibble: A group of four data bits (ref. IEEE 802.3 [3].)

normal operation: Operation of a link for the purpose of transmitting converter samples. That is, not a test mode or a power down mode.

octet: A group of eight adjacent binary digits (similar to a byte.)

receiver: A circuit attached to a lane for reconstructing a serial bit stream into time-aligned frames.

NOTE A receiver consists of one physical layer block and one link layer block.

receiver block: The combination of the receiver transport layer and all receiver link layer and physical layer blocks connected to a link.

receiver device: A component package containing one or more receiver blocks.

relative wander: Components of wander that are uncorrelated between any two in band signals (ref. OIF-CEI-04.0.)

rising edge of a differential signal(P,N): The simultaneous transitions that occur when signal(P) changes from the low logic level to the high logic level and signal(N) changes from the high logic level to the low logic level.

running disparity: A binary parameter having a value of + or –, representing the imbalance between the number of ones and zeros in a sequence of 8B/10B code-groups (see IEEE Std. 802.3 [5].)

sample: The instantaneous value of a signal measured or determined at a discrete time (adapted from ASIS Telecom Glossary, "sampled data" [1].)

NOTE In the context of JESD204, a sample is always the digital representation of a signal.

scrambler: A randomizing mechanism that is used to eliminate long strings of consecutive identical transmitted symbols and avoid the presence of spectral lines in the signal spectrum without changing the signaling rate (ref. IEEE 802.3 [3].)

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.1 Terms and definitions (cont'd)

SH_LOCK: A state that asserts the sync header alignment has been achieved.

skew: The magnitude of the time difference between two events that ideally would occur simultaneously (ref. JESD65B.)

steady-state voltage: The voltage at the transmitter output or receiver input resulting from the transmission of a series of consecutive logic-ones or logic-zeros, after the initial transient part of the waveform and any transmit filter compensation, have ended.

source clock: An oscillator from which the various other clock signals are derived. This oscillator is typically a VCO inside a clock generator device, or an external VCO within a clock generator circuit.

symbol: The smallest unit of coded data on the medium (simplified from IEEE 802.3 [3].)

NOTE In this standard used as synonym to "character".

sync header: Two bits, which guarantee a transition, preceding every block in the 64B/66B and 64/80B link layer (see 0.)

sync word: A group of 32 encoded sync transitions in the 64B/66B and 64/80B link layer (see 0.)

SYSREF: A periodic, one-shot (strobe-type), or "gapped" periodic signal used to align the boundaries of local clocks in Subclass 1 devices. SYSREF must be source synchronous with the device clock.

transition time: Collective name given to the rise and fall time of a signal.

transmitter: A circuit that serializes the input frames and transports the resulting bit stream across a lane.

NOTE A transmitter consists of one link layer block and one physical layer block.

transmitter block: The combination of the transmitter transport layer and all transmitter link layer and physical layer blocks connected to a link.

transmitter device: A component package containing one or more transmitter blocks.

valid code group: A code group that is found in the proper column of the 8B/10B decoding tables, according to the current running disparity (see IEEE Std. 802.3 [5].)

word: A character string or a binary element string that it is convenient to consider as an entity.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.2 Symbols and abbreviated terms

AC: Alternating Current.

ADC: Analog-to-Digital Converter.

$\boldsymbol{A_{DD}}$: Dual-Dirac jitter added to received signal after the decision feedback equalizer (see Table 27, Table 28 and Table 29).

$\boldsymbol{A_{fe}}$: Far-end aggressor transmitter peak differential output voltage (see Table 27, Table 28 and Table 29).

$\boldsymbol{A_{ne}}$: Near-end aggressor transmitter peak differential output voltage voltage (see Table 27, Table 28 and Table 29).

ASIC: Application-Specific Integrated Circuit.

$\boldsymbol{A_v}$: Transmitter differential output voltage voltage (see Table 27, Table 28 and Table 29).

BER: Bit Error Rate – The rate at which errors occur, in errors per number of bits transmitted.

BERT: Bit Error Rate Tester.

BGA: Ball Grid Array.

***BkW*:** Block Width – The number of bits in a 64B/66B or 64B/80B block.

B-3: A class of devices that support data rates up to 3.125 Gbps.

B-6: A class of devices that support data rates up to 6.375 Gbps.

B-12: A class of devices that support data rates up to 12.5 Gbps.

$\boldsymbol{b_{max}[j]}$: Normalized maximum decision feedback equalizer coefficient magnitude voltage (see Table 27, Table 28 and Table 29).

$\boldsymbol{b(n)}$: Normalized magnitude of n^{th} decision feedback equalizer coefficient magnitude voltage (see Table 27, Table 28 and Table 29)

C: Control bit (used in figures, applies to the transport layer specification.)

CBHPJ: Correlated Bounded High-Probability Jitter.

***CF*:** Number of control words per frame clock period per link.

CGS: Code Group Synchronization.

Cmd: Command – as related to the command channel.

COM: Channel Operating Margin (see IEEE Std. 802.3-2018.)

Cr: Converter (used in figures, applies to the transport layer specification.)

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.2 Symbols and abbreviated terms (cont'd)

CRC: Cyclic Redundancy Check.

***CS*:** Number of control bits per conversion sample.

CTLE: Continuous Time Linear Equalizer.

C-S: A class of devices that support data rates up to 32.45 Gbps, with a short-reach channel.

C-M: A class of devices that support data rates up to 32.45 Gbps, with a medium-reach channel.

C-R: A class of devices that support data rates up to 32.45 Gbps, with a reflective channel.

$\boldsymbol{c}(-1)$: Normalized value of the pre-cursor (bit before current bit) coefficient in the functional model for the feed-forward transmitter equalizer voltage (see Table 27, Table 28 and Table 29).

$\boldsymbol{c}(\mathbf{0})$: Normalized value of the cursor (current bit) coefficient in the functional model for the feed-forward transmitter voltage (see Table 27, Table 28 and Table 29).

$\boldsymbol{c}(+1)$: Normalized value of the post-cursor (bit after current bit) coefficient in the functional model for the feed-forward transmitter equalizer voltage (see Table 27, Table 28 and Table 29)

D: Dx.y, any 8B10B data symbol (used in figures, applies to the 8B/10B link layer specification.)

DAC: Digital-to-Analog Converter.

DC: Direct Current.

DCD: Duty Cycle Distortion.

$\boldsymbol{DER_0}$**:** Target detector (bit) error rate; used in COM.

DFE: Decision Feedback Equalizer.

$\mathbf{D_p}$**:** Differential steady-state transmitter output voltage linear fit pulse delay.

$\mathbf{D_w}$**:** Differential steady-state transmitter output voltage linear fit equalizer delay.

***E*:** The number of multiblocks in an extended multiblock.

EBUJ: Effective Bounded Uncorrelated Jitter.

DA: Electronic Design Automation.

EMB: Extended Multiblock.

EoMB: End-of-multiblock sequence (00001).

EoEMB: End of extended multiblock identifier bit.

ERJ: Effective Random Jitter.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.2 Symbols and abbreviated terms (cont'd)

***F* (as symbol):** Number of octets per frame on a single lane.

FEC: Forward error correction.

FEXT: Far-End cross Talk.

FFE: Feed Forward Equalizer.

FIR: Finite Impulse Response.

FOM: Figure of Merit.

f_b**:** (1) Used data rate.

(2) Data rate at which a device, channel or system is compliance tested.

f_{min}: Start frequency of COM voltage (see Table 27, Table 28 and Table 29).

f_r: Receiver 3 dB bandwidth voltage (see Table 27, Table 28 and Table 29).

GJ: Gaussian Jitter.

g_{DC}: Continuous time filter, CTLE, DC gain voltage (see Table 27, Table 28 and Table 29).

***HD*:** High Density user data format.

H(z): Transmitter feed-forward equalizer discrete-time transfer function.

IBIS-AMI: Input/output Buffer Information Specification Algorithmic Modeling Interface (see Annex P reference 10.)

IFT: Inverse Fourier transform.

***IL*:** Insertion Loss.

ILAS: Initial Lane Alignment Sequence. Applies to 8B/10B encoding.

***ILD(f)*:** Insertion Loss Deviation as a function of frequency (see Annex F.2.)

ISI: Inter-Symbol Interference.

***K* (as symbol):** (1) Number of frames per multiframe. Applies to the 8B/10B link layer, see 8.4.5.2.

(2) Number of frames per extended multiblock. Applies to the 64B/66B and 64B/80B link layers, see 7.1.4.

NOTE In the 8B/10B link layer, *K* is an independent design parameter. In the 64B/66B and 64B/80B link layers, *K* is dependent on *E* and *F*: $K = 256 \cdot E/F$.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.2 Symbols and abbreviated terms (cont'd)

***L*:** (1) Number of lanes per converter device.

(2) Number of signal levels (see Table 27, Table 28 and Table 29)**.**

LCM: Lowest Common Multiple.

LEMC: Local Extended Multiblock Clock.

LFSR: Linear Feedback Shift Register.

LMFC: Local Multi Frame Clock.

***M*:** (1) Number of converters per device (as used in the transport layer.)

(2) Number of samples per unit interval; used in COM.

mod: Modulo. x mod y is used as alternative notation for mod(x, y).

***N*:** Converter resolution (as used in the transport layer.)

N_b: Number of taps of the receiver decision feedback equalizer (see Table 27, Table 28 and Table 29).

N_{max}: Number of samples of the sampled pulse response.

NEXT: Near-End crosstalk.

NG: Nibble Group.

N_p**:** Differential steady-state transmitter output voltage linear fit pulse length.

NRZ: Non-Return-to-Zero.

N_w**:** Differential steady-state transmitter output voltage linear fit equalizer length.

N': Total number of bits per sample in the user data format.

n: (1) Coefficient index in a decision feedback equalizer (see Table 27, Table 28 and Table 29). $0 \leq n \leq N_b$.

PCB: Printer Circuit Board.

PCS: Physical Coding Sublayer.

PMA: Physical Medium Attachment Sublayer.

PRBS: Pseudo-Random Bit Sequence.

***PSFEXT(f)*:** Far-end crosstalk power sum.

***PSNEXT(f)*:** Near-end crosstalk power sum.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.2 Symbols and abbreviated terms (cont'd)

PVT: A term denoting the variations of a device across process corner, supply voltage, and temperature.

$\boldsymbol{p_{max}}$**:** Peak value of the linear fit to the differential steady-state transmitter output.

NOTE Depending on the context, $p(k)$ applies specifically to the case that the 'cursor' TX equalizer coefficient is set to its maximum value and the other equalizer coefficients are zero, or to any configuration of the transmit equalizer.

p.u.l.: per unit length

$\boldsymbol{RBD}$**:** Receive Buffer Delay.

RD: Running Disparity. Applies to 8B/10B encoding (see IEEE Std. 802.3 [5].)

RL: Return loss.

$\boldsymbol{RL_{CD}(f)}$**:** Differential-to-common-mode return loss, i.e. the ratio between differential-mode incident power and common-mode reflected power.

$\boldsymbol{RL_{CM}(f)}$**:** Common-mode return loss, i.e. the ratio between common-mode incident power and common-mode reflected power.

$\boldsymbol{RL_{DD}(f)}$**:** Differential-mode return loss, i.e. the ratio between differential-mode incident power and differential-mode reflected power.

$\boldsymbol{RL_{RXF_{ref}}(f, f_b)}$**:** Reference differential return loss of the receiver test fixture as a function of frequency for a specified data rate (see 5.2.4.2.)

$\boldsymbol{RL_{TXF_{ref}}(f, f_b)}$**:** Reference differential return loss of the transmitter test fixture as a function of frequency for a specified data rate (see 5.2.3.2.)

RMS: Root Mean Square.

$\boldsymbol{R_o}$: Reference single-ended resistance of COM (see Table 27, Table 28 and Table 29).

RX: Receiver.

$\boldsymbol{S}$**:** Number of samples transmitted per single converter per frame cycle.

SERDES: Serializer/Deserializer.

SH: Sync Header.

$\boldsymbol{SJ_{LF}}$**:** Low-frequency sinusoidal jitter to be tolerated by a compliant receiver.

$\boldsymbol{SJ_{LH}}$**:** High-frequency sinusoidal jitter to be tolerated by a compliant receiver.

***SL<p>*:** Transmitter or receiver differential pair positive terminal voltage.

***SL<n>*:** Transmitter or receiver differential pair negative terminal voltage.

weijiafadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.2 Symbols and abbreviated terms (cont'd)

***SNDR*:** Signal-to-Noise-and-Distortion Ratio.

***SNR*:** Signal-to-Noise Ratio.

$\boldsymbol{SNR_{TX}}$: Transmitter signal-to-noise ratio (see Table 27, Table 28 and Table 29).

SSPS-64: Short Stress Pattern SDH64 (see OIF-CEI-04.0.)

T: Tail bit (applies to the transport layer specification.)

TJ: Total jitter.

$\boldsymbol{T_{LLC}}$**:** The duration of one cycle of the link layer clock.

$\boldsymbol{T_{SYNCCLK}}$**:** The duration of one cycle of the SYNC~ generation or detection clock.

$\boldsymbol{T_{TLC}}$**:** The duration of one cycle of the transport layer clock.

TUJ: Effective Total Uncorrelated Jitter.

TX: Transmitter.

UBHPJ: Uncorrelated Bounded High-Probability Jitter.

UI: Unit Interval = the duration of a serial bit.

$\boldsymbol{U_k}$**:** Number of laminate package lengths of the reference model.

$\boldsymbol{U_{LLC}}$**:** The maximum expected number of unit intervals (serial bit durations) in one cycle of the link layer clock.

$\boldsymbol{U_{TLC}}$**:** The maximum expected number of unit intervals (serial bit durations) in one cycle of the transport layer clock.

$\boldsymbol{v_{cm}}$**:** Common-mode voltage, i.e. the average of the voltages at the "true" (or positive) and "complement" (or negative) input or output terminals of a circuit.

$\boldsymbol{v_d}$**:** Differential voltage, i.e. the voltage between the "true" (or positive) and "complement" (or negative) input or output terminals of a circuit.

$\boldsymbol{v_f}$**:** Differential zero-to-peak voltage at the transmitter test/measurement point resulting from the transmission of a series of consecutive logic-ones, after the initial transient part of the waveform has ended, when the transmit equalizer "cursor" (current bit) coefficient has been set to its maximum value and all other pre- and post-cursor equalizer coefficients have been nulled.

$\Delta\boldsymbol{f}$: Maximum frequency step in (see Table 27, Table 28 and Table 29).

$\boldsymbol{\eta_o}$: One-sided noise spectral density of reference receiver model (see Table 27, Table 28 and Table 29).

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

3.2 Symbols and abbreviated terms (cont'd)

σ_{RJ}: Random jitter added to received signal after the decision feedback equalizer (see Table 27, Table 28 and Table 29).

/A/: K28.3 character, lane align. Applies to 8B/10B encoding.

/Dx.y/: Data code group Dx.y encoded according to the current running disparity (see IEEE Std. 802.3 [5].)

/F/ (as abbreviation): K28.7 character, frame sync. Applies to 8B/10B encoding.

/K/ (as abbreviation): K28.5 character, code group synchronization. Applies to 8B/10B encoding.

/Kx.y/: Control code group Kx.y encoded according to the current running disparity (see IEEE Std. 802.3 [**5**].)

/R/: K28.0 character. Applies to 8B/10B encoding.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4 Introduction and common requirements

4.1 Application overview

4.1.1 Background

In many data converter applications, a side effect of advances in converter technology is that device packages, power consumption and circuit board area are now driven by interconnect and data transfer requirements, instead of the intrinsic needs of the application. High resolution data converters typically need more board space for the interconnection (board traces and vias) than for the chip package itself. Single-ended digital interfaces cause switching noise, which affects the performance of the analog part, and parallel differential-signaling interfaces (such as LVDS) exacerbate the board area cost of the interconnect.

While technological advances have led to significantly lower power consumption in the intrinsic part of converters, similar reductions in the I/O power consumption and the size of the chip package have not taken place because of the large number of I/O lines needed. These problems become even more evident when multiple converters are integrated in the same package, e.g., for direct conversion radio transceivers. This I/O pin count limit also affects multi-channel data acquisition applications, such as beam forming, in which signal processors synthesize the output of an array of A/D converters into a single data stream.

Various approaches, such as integrating converters with signal processing logic or multi-chip modules, have addressed this problem but have drawbacks where converter performance is critical. The optimal processes for the converter function and digital VLSI are different. Integrating the two can impose performance limits on mixed-signal design, increases the risk of coupling digital switching noise into sensitive analog circuitry and can add cost to the digital design. Over multiple generations of a system design, requirements for the two areas evolve independently. From a practical standpoint, the two functions are typically supplied by different manufacturers. For many applications, then, the preferred configuration remains separate ADCs, DACs, and logic devices for driving the DAC devices and receiving data from the ADC devices.

Serialized differential links offer a solution in this case. Although such links have been widely adopted in data transfer applications, existing industry standards such as PCI Express provide services better suited to processors or dedicated interface devices. Such PC-centric interfaces impose unacceptable additional complexity and overhead on the simple, continuous, constant-bandwidth data stream considered here. What is needed is a definition of signals and data formats that enables front-end data acquisition systems to take advantage of high-speed serial links with minimal overhead.

The purpose of this specification is to allow manufacturers of converter devices (ADC devices and DAC devices) and logic devices (ASICs and FPGAs) to guarantee interoperability between devices, which will promote reuse of components across applications and enable system integrators to independently select the best device for each function. Implementation of this standard only requires logic blocks that are already commonly available to both FPGA and ASIC designers. Standard logic libraries, PLLs and SERDES macros are sufficient to implement this standard.

The intended application is one with short, point-to-point links between devices, either on the same printed circuit board, across one or more impedance-controlled connectors, or across short-reach cables. The electrical layer signal generally represents what is commercially referred to as Current Mode Logic or "CML", terminated at 1.2 V.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.1.1 Background (cont'd)

The sampled converter data is sent over a serial data interface as a continuous, encoded data stream. The data interface consists of one or more lanes, i.e., pairs of differential signals. The collection of lanes and circuitry connecting two devices to each other is called a link. Several links connecting to the same logic device can be combined into a multipoint link on which all lanes are time-aligned.

4.1.2 Physical layer overview

JESD204C supports data rates for a single lane ranging from 0.3125 to 32.45 Gbps (raw bit rate), which supports a range of applications with varying requirements of converter resolution, sample rate, and number of channels. In multi-channel applications, several lower bandwidth channels are aggregated on a single link. In addition, the specification optionally supports links consisting of multiple lanes. Multiple lanes may be utilized when the converter data rate is higher than any single lane can accommodate or when lower rates are desired for reasons of simplicity or economy. Further, the specification optionally supports multiple data converters connected to the same logic device. Embedded alignment data enable the receiver(s) to realign associated information across multiple lanes and multiple links.

JESD204C defines two categories of classes, category B and category C. Three classes are defined within category B (B-3, B-6, and B-12), each with a different maximum data rate. Category C also contains three classes, however all of the category C classes have the same maximum data rate of 32.45 Gbps. Table 2 specifies the minimum and maximum data rate for each class.

Table 1 — Data rate range for each data interface

Data interface class	Minimum data rate (Gbps)	Maximum data rate (Gbps)
B-3	0.3125	3.125
B-6	0.3125	6.375
B-12	6.375	12.5
Category C	6.375	32.45

Three category C classes are defined to minimize link power dissipation for a variety of channel types: C-S(hort), C-M(edium) and C-R(eflective), each class being a superset of the previous one. Table 3 describes the most important architecture differences between the classes. A device may implement one or more classes or operating modes and still be considered JESD204C compliant.

Table 2 — JESD204C 32.45 Gbps interface device class features

Class	Relative power	Transmit filter boost	Receive filter boost	Receiver DFE taps
C-S	Low	9.5dB	6dB	0
C-M	Medium	9.5dB	9dB	3
C-R	High	9.5dB	12dB	14

JESD204C category C class interface compliance is assessed by calculating a figure of merit called channel operating margin or COM. A link's COM may be calculated using the JESD204C channel operating margin (JCOM) reference implementation tool, which is derived from the IEEE802.3 channel operating margin tool. A JESD204 compliant link is one that has a COM higher than 2dB. Subsuming channel compliance to an overall link compliance using JCOM affords greater flexibility in managing the trade-offs of channel impairments and SERDES architectural complexity.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.1.3 Transport and link layer overview

4.1.3.1 Data encoding and organization

JESD204C provides three encoding options, which are listed in Table 4. For the four ranges of data rates shown in the table, each link layer is either required, recommended, optional, or not recommended. For example, 64B/66B encoding is recommended (but not required) between 6.375 and 12.5 Gbps, and required above 12.5 Gbps. 8B/10B encoding may not be optimal when paired with decision feedback equalizers (DFEs) as it may not provide adequate spectral richness required for successful DFE adaptation. For this reason, as well as the relative power efficiency of 64B/66B encoding, 8B/10B encoding is not recommended above 16 Gbps.

Table 3 — Supported link layers versus data rate

Data rate (DR) (Gbps)	8B/10B	64B/66B	64B/80B
DR <= 6.375	Required	Not Recommended	Not Recommended
6.375 < DR <= 12.5	Required	Recommended	Optional
12.5 < DR <= 16	Optional	Required	Optional
16 < DR <= 32	Not Recommended	Required	Optional

The 8B/10B link layer organizes the data into multiframes, which contain $K \cdot F$ octets. The 64B/66B and 64B/80B link layer organizes the data into multiblocks, which contain 32 blocks, with each block containing eight octets.

Data scrambling is optional when using 8B/10B encoding, however it is mandatory when using 64B/66B or 64B/80B encoding.

4.1.3.2 Clocking

The JESD204 standard specifies the use of a device clock. The device clock addresses a key problem of the system designer – the problem of having to transmit clocks of different frequencies to different devices in the system. Such transmission causes not only complications in the clock generation scheme but is also prone to spurious effects from sub-harmonic energies existing on the board. The ideal scenario for a system designer therefore would be to transmit clocks of identical frequencies to all the devices in the system. If the different devices in turn have provisions to generate their desired internal clocks from this main clock, it would vastly simplify the problem of system clocking. The device clocks transmitted to all the converter devices are expected to be identical, and may or may not be identical to the device clock transmitted to the logic device.

The sample clock may be internally generated from the device clock, of which a copy is provided to each device. Since most converter applications demand a low jitter sample clock to minimize noise, the device clock provided to the converter should have this desired low jitter. Also in low jitter applications, it would be desirable to generate the sample clock from the device clock through a simple process of clock division.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.1.3.2 Clocking (cont'd)

The process of generating the internal clocks from the device clock may involve operations like clock division, which require a mechanism for phase synchronization across chips. As an example, multiple ADC devices interfacing to the same logic device might generate their internal sampling clocks based on a process of division – if the phase synchronization between these divided clocks is not established, it would result in a relative skew between the sampling instants of the different ADCs. In applications like beamforming for instance, such a relative skew would not be tolerable. The 8B/10B link layer offers a solution to this problem of phase synchronization by exploiting the existence of a Local Multiframe clock (LMFC). Similarly, the 64B/66B and 64B/80B link layer instead uses the LEMC (Local Extended Multiblock Clock). While the LMFC marks multiframe boundaries in the 8B/10B link layer, the LEMC is used to mark extended multiblock boundaries in the 64B/66B and 64B/80B link layer. The LEMC period spans multiple multiblocks, and should be chosen to be long enough to fit a whole number of frames and to allow for a certain amount of delay uncertainty on the lanes, the physical layers, and the link layers.

The 8B/10B link layer attempts to phase align the LMFCs between all the devices through two different means in two different subclasses, subclass 1 and subclass 2. However, the 64B/66B and 64B/80B link layer only supports subclass 1. In subclass 1, the LEMC/LMFCs are synchronized by a source synchronous signal called SYSREF, which is transmitted to all the devices with controlled timing. In subclass 2 (again, this is only supported when using the 8B/10B link layer), the logic device generates its LMFC through a process of division of its device clock. The logic device then communicates the phase of its LMFC to the converter devices, each of which adjust their LMFC phases with respect to the common logic main device so that clocks are in phase.

It is recommended that the pin configuration of a subclass 1 device would have the SYSREF pin in close proximity to the device clock pins. It is also recommended that the electrical characteristics of the SYSREF signal be identical to that of the device clock. While SYSREF is expected to be a much slower clock than the device clock, its rise/fall times are expected to be similar to that of the device clock. As a result, the signal integrity of the SYSREF signal might be as much of a concern as that of the device clock.

A related concern to be kept in mind while controlling the timing of the SYSREF signal relative to the device clock is the fact that SYSREF, by virtue of being a slow signal would have logic high and low voltage levels that would be close to the saturation levels. In comparison, a high speed device clock is more likely to have logic high and low voltage levels that are further away from the saturation levels. This can lead to a mismatch in the rise/ fall times of the SYSREF signal relative to the device clock, and therefore needs to be considered while analyzing the setup and hold times available for the device clock edge to latch the SYSREF. In a case where SYSREF is provided as a low frequency periodic signal, the data converter vendor as well as the system implementer would do well to analyze the spurious effects from having such a sub-harmonic clock, and minimize its effects on the converter and system performance.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.1.3.3 Sync header stream

The 64B/66B and 64B/80B link layer define a sync header stream, which allows the transmission of information parallel to the user data. This information is encoded using the sync header portion of the 66B or 80B word block. Specifically, one sync header per block is decoded to a single bit, and 32 of these bits across a multiblock make up the 32-bit sync word. Four formats of the sync word are supported – one that supplies Forward Error Correction (FEC) parity information, two that supply Cyclic Redundancy Check (CRC) parity information along with a command channel, and one that only supplies the command channel. The command channel provides a mechanism for the transmitter to send various commands and data to the receiver.

4.1.3.4 Deterministic latency

Yet another key system issue that is addressed in the JESD204 standard is the problem of deterministic latency. In the context of multiple ADC devices interfacing to a common logic device, deterministic latency is simply the following – at a particular instant, the frames appearing at the final output of the receiver must be relatable to a specific sampling instant of each ADC. These sampling instants for each ADC must, within margins of aperture error, correspond to the same absolute time instant. This relation between the ADC sampling instants and the frame-based output at the receiver output must be maintained the same over temperature and supply voltage, across devices, and must be repeatable over power-up cycles. For the case of a logic device transmitter interfacing to multiple DAC devices, deterministic latency implies a known deterministic relation between the frames going into the logic device transmitter and the analog output transitions at the output of the multiple DAC devices. While the 8B/10B link layer utilizes the LMFC to achieve deterministic latency, the 64B/66B and 64B/80B link layer uses the LEMC to achieve deterministic latency. Please refer to clause 4.2 for further details on the implementation of deterministic latency.

4.1.4 Data link properties

With the 8B/10B link layer, JESD204 uses the SYNC interface for synchronization and error reporting. When using the 64B/66B or 64B/80B encoding options, sync headers within the encoded data are used for the synchronization process and the reporting of errors is left to the application-layer software.

With the 8B10B link layer, the alignment between multiple converter devices happens through the alignment of their LMFCs as determined by SYSREF in subclass 1, and SYNC~ in subclass 2. With the 64B66B and 64B80B link layer, the alignment between multiple converter devices happens through the alignment of their LEMC as determined by SYSREF in subclass 1. Each converter device can then adjust its LEMC phase to match with the common LEMC of the logic device. Subclass 2 may still be used with the 8B/10B link layer.

When one or more converters inside a device are in power-down mode, the link can still be kept active and synchronized. This state is called idle mode. In idle mode, samples of inactive converters are replaced by dummy samples. On the link level, no difference can be seen between an active and an inactive converter. The presence of dummy samples is indicated to the receiver via the control interface or via dedicated control bits in the sample stream.

When all converters in a device are inactive for a long period of time, it may be advantageous for the link to enter a shut-down mode. Operation in shut-down mode is similar to the continuous mode, but with power-down and power-up procedures applied between the periods of activity. These power-down and power-up procedures are not part of the current specification and their implementation is left to the application.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.1.4.1 Variants and modes

The interface supports a wide range of data converter devices over single and multiple serial links. In addition, each link may consist of a single lane or multiple lanes as needed to support the required data throughput. Each lane contains synchronization and alignment data that allows the data to be reassembled in the receiver, effectively rebuilding the structure of the original converter data. Although one or more converter devices are supported by this standard, only one logic device is supported.

The configuration variants can be classified according to the following scheme:

1. Single vs. multiple converters per converter device.
2. Single lane vs. multiple lanes per converter device.
3. Single converter device vs. multiple converter devices connected to one logic device.

The interface can work in one of the following modes:

4. Data mode vs. test mode.
5. Active mode vs. idle mode.

Test modes can be implemented on the physical layer level, e.g., by transmitting a PRBS pattern, on the link layer level, e.g., by transmitting a stream of synchronization characters, or on the transport layer level, by transmitting a stream of test samples. In the idle mode one or more converters in a device may be shut down and their samples are replaced by dummy samples.

4.1.5 Configuration examples

4.1.5.1 General

The following examples illustrate configurations for a single link, connecting a logic device to a single converter device. Configurations with multiple converter devices consist of multiple single links in parallel, which share the same logic device. Multiple links can be independent of each other, or they can use certain functionality to align their data streams, as described in clauses 4.2 and 4.3. While specified for data converters, the standard is also suitable for all systems that transmit real-time streaming data between devices. Therefore, the configurations outlined below provide a few examples of the many possibilities.

4.1.5.2 Single-device ADC application

Figure 3 illustrates a typical implementation of the interface as it applies to a single ADC device communicating with a single logic device. With this configuration, a single package contains one or more ADCs. As described earlier, the converters' output data is aggregated by the JESD204 TX block and is sent across one or more lanes to the receiving logic device.

The logic device de-serializes the data in the JESD204 RX block. If multiple lanes are involved, the logic device RX block is also responsible for ensuring alignment of the data streams being sent to the application logic.

In subclass 1, the zero-to-one transitions on the signals marked as TX SYSREF and RX SYSREF are sampled by the ADC Device clock and the Logic Device clock respectively, thereby synchronizing the LMFC/LEMC phase inside each of the devices.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.1.5.2 Single-device ADC application (cont'd)

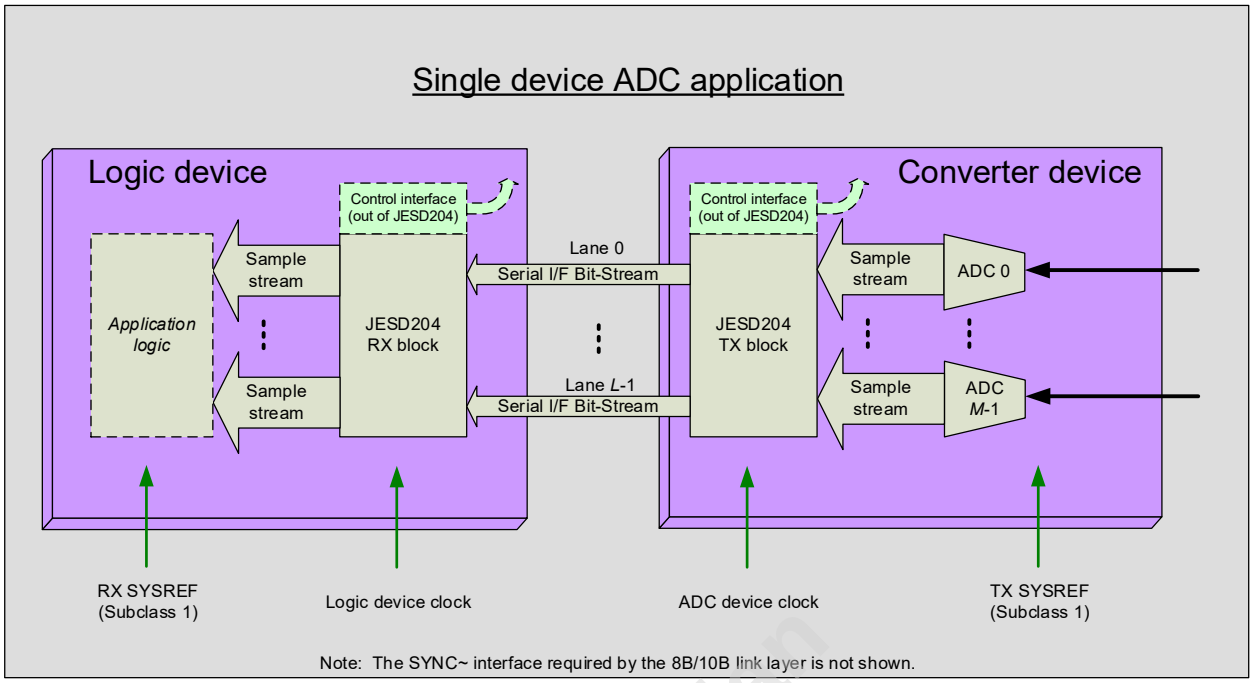


Figure 3 — Single device ADC application

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.1.5.3 Single-device DAC application

Figure 4 shows an application using multiple DACs in a single package. In this application, the logic device provides DAC samples using one or more lanes. In the DAC device's JESD204 RX block, the data is de-serialized and provided to the DACs.

In subclass 1, the zero-to-one transitions on the signals marked as TX SYSREF and RX SYSREF are sampled by the logic device clock and the DAC device clock respectively, thereby synchronizing the LMFC phase inside each of the devices.

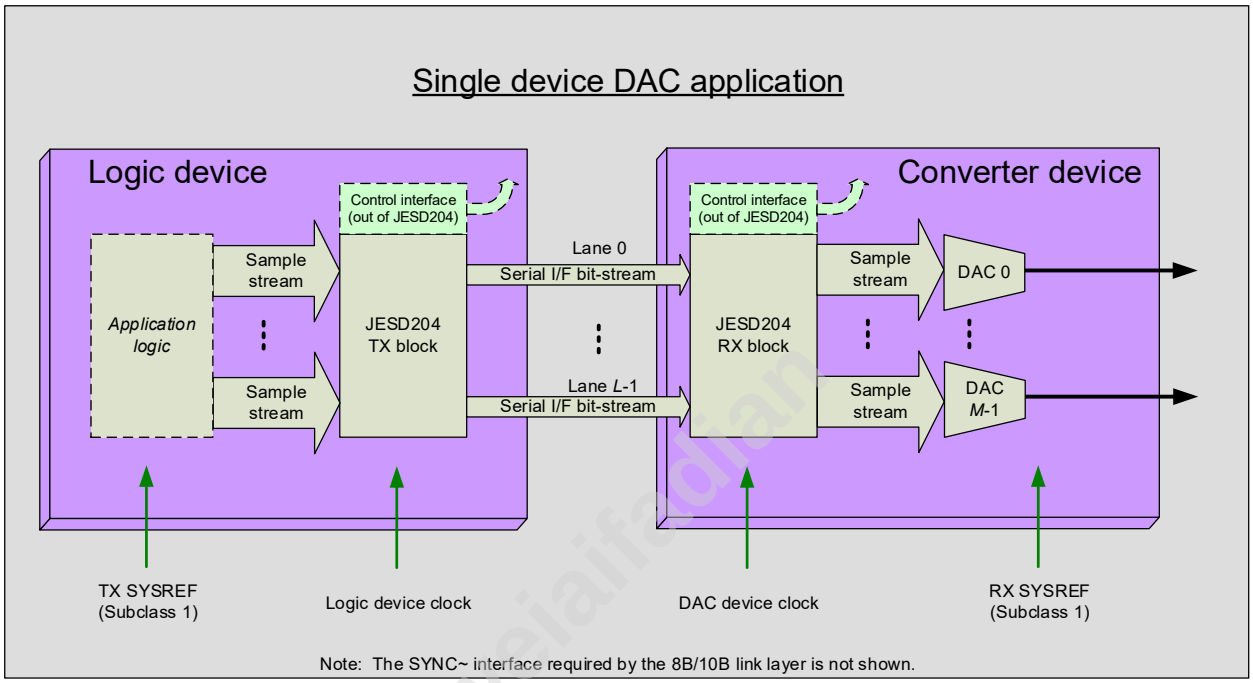


Figure 4 — Single device DAC application

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.2 Deterministic latency

4.2.1 Introduction and general requirements

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. The JESD204 standard provides the option for what's called "deterministic latency" to avoid such ambiguity.

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. This latency across the link 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.

NOTE The frame-based input and output points for defining latency are at the interface between the transport layer and the application layer. Therefore, the implementation of the deterministic latency may partly or wholly take place in the transport layer or application layer.

The use of a local multiframe clock (LMFC) or local extended multiblock clock (LEMC) is required to achieve deterministic latency. The data in the TX and RX devices shall be aligned to the LMFC/LEMC in the following way:

In the TX device, the start of any multiframe or extended multiblock shall be initiated simultaneously across all lanes at a well-defined moment in time.

- The 'well-defined moment in time' for TX transmission is a deterministic period of time from all active LMFC/LEMC edges.

In the RX device, a buffer shall exist to hold all lane data for release (i.e., data allowed to propagate) simultaneously at a well-defined moment in time.

- The 'well-defined moment in time' for RX buffer release is a programmable number of steps after an active LMFC/LEMC edge. This programmable number of steps is referred to as the RX Buffer Delay (*RBD*).
- The full adjustment range of *RBD* shall cover the full size of the RX elastic buffer or at least one LMFC/LEMC cycle, whichever is smaller.
- The minimum size required for the RX elastic buffer corresponds to the maximum possible skew at skew point SP5 for the relevant subclass (see 4.3.8).
- The oldest data in the buffer available for release shall correspond to the start of a multiframe or extended multiblock.
- The *RBD* adjustment resolution shall not be greater than 255 steps.
- The *RBD* adjustment resolution shall be at least 16 steps. If more than one multiframe or multiblock per lane fits in the buffer, the *RBD* adjustment resolution shall be at least 16 steps per multiframe or multiblock.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.2.1 Introduction and general requirements (cont'd)

The *RBD* adjustment is required to move the release of the elastic buffer away from any point in time that may lie within the range of lane data arrival skew (see example below). It must be possible, through the use of *RBD*, to select a release opportunity that does not lie in this range. Smaller and more numerous *RBD* adjustment steps are preferred to meet this requirement and to assist in selecting a release opportunity just after the latest possible lane arrival, which is useful to minimize total latency through the link. In order that the system implementer can select the optimum *RBD* value, the RX device shall be able to report over the control interface the arrival time of the multiframe or extended multiblock boundary relative to the active LMFC/LEMC edge for at least the slowest received lane.

The moments of generation of the multiframe or extended multiblock boundaries and release of the RX buffer mentioned above are related to the LMFC/LEMCs in the TX and RX devices. Thus, the achievement of deterministic latency with minimum uncertainty relies on achievement of a fixed offset between the LMFC/LEMCs within TX and RX devices. To achieve proper performance of the deterministic latency protocol while utilizing the 8B/10B link layer, the period of the LMFC must be larger than the maximum possible delay variation across any link or multipoint link. To achieve proper performance of the deterministic latency protocol while utilizing either the 64B/66B link layer or the 64B/80B link layer, the period of the LEMC must be larger than double the maximum possible delay variation across any link or multipoint link.

To achieve deterministic latency for JESD204 links, the RX device must be able to buffer the incoming data of all lanes until the RX elastic buffer can be released. To release the elastic buffer, data must be present that corresponds to the start of a multiframe or extended multiblock for all active lanes. The buffer must be released after an active LMFC/LEMC edge according to the *RBD* value.

The selection of the moment to release the buffers in a JESD204C system is illustrated in the timing diagram of Figure 5. As part of the initial lane alignment, the TX device transmits a stream of /K/ characters, followed by an initial lane alignment sequence (ILAS). In this example, ILAS starts exactly on an active LMFC edge. The RX device will then detect the start of ILAS on each lane, and will feed this data into per-lane elastic buffers. The 'release opportunity' of these buffers is set to *RBD* character durations after each active LMFC edge. On the first release opportunity after the RX device receives valid ILAS data on all lanes, it will release all the elastic store buffers. The resulting data output from the RX device is aligned with a fixed latency across the link.

Uncontrolled Manufacturer

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.2.1 Introduction and general requirements (cont'd)

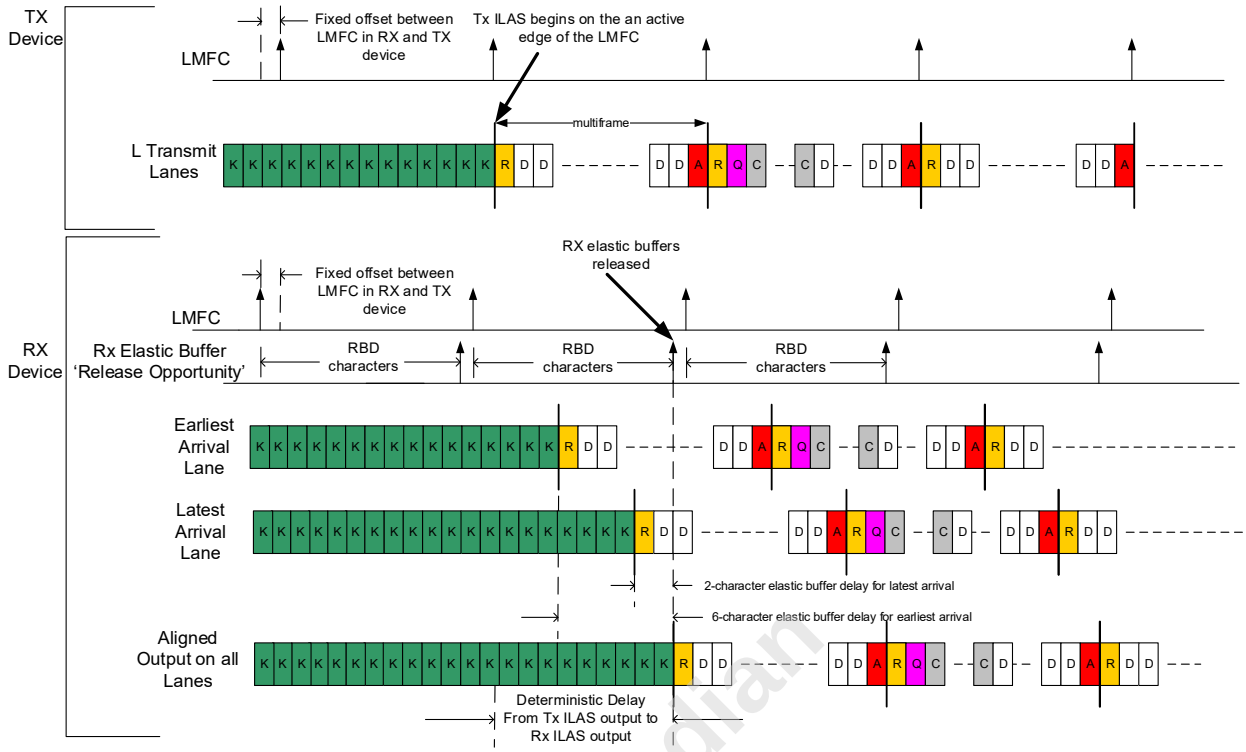


Figure 5 — Timing diagram illustrating the RX elastic buffer release opportunity in a JESD204C 8B/10B system

The selection of the *RBD* value should take into account margin for latency variations due to e.g., power reset or resynchronization of the link. The following might occur between two subsequent power-ups in the same system if this were not the case:

1) The start of a multiframe or extended multiblock on the slowest lane arrives at the Rx just soon enough for a particular release opportunity. In this case the data is released from the buffer with little delay.

2) The start of a multiframe or extended multiblock on the slowest lane arrives at the Rx just too late for a particular release opportunity. In this case the data is held for the next release opportunity and the buffer inserts close to a whole LMFC/LEMC period worth of delay.

Deterministic latency would not be preserved in the above example because the second instance would add exactly an LMFC/LEMC period worth of total link delay compared with the first instance.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.2.2 No support for deterministic latency (device subclass 0 and device subclass 1 using MULTIREF)

In subclass 0 operation, the RX buffers are released at the earliest opportunity after the detection of a multiframe or extended multiblock boundary in each lane. Subclass 0 operation requires no alignment of LMFC/LEMCs across transmitter and receiver devices. Subclass 0 is intended for interfacing to JESD204A devices and for applications in which minimum link delay is desired, without need for deterministic latency.

In subclass 1 operation, the MULTIREF signal may be used to align the LMFC/LEMCs across converter devices (see 4.3.6). In this use case, the MULTIREF signal does not align the LMFC/LEMCs between the logic device and the converter devices. Therefore, this use case of the MULTIREF signal will not create deterministic latency. The MULTIREF signal replaces the lane-to-lane inter-device synchronization interface for subclass 0 DAC devices of the JESD204B standard as well as lane alignment across subclass 0 ADC devices using sync signal combination.

4.2.3 Deterministic latency using SYSREF (device subclass 1)

For subclass-1 devices, the creation of properly aligned LMFC/LEMC signals within TX and RX devices is achieved using a signal designated as ‘SYSREF’, which must be distributed to all converter and logic devices.

Delay uncertainty in the system is minimized by using highly accurate SYSREF / device clock signal pairs. While it is not mandatory, it is recommended that SYSREF be generated from the same device producing the TX and RX device clocks.

Due to the variety of allowable SYSREF signal types (periodic, one-shot (strobe-type), or “gapped” periodic), the variety of clock generator devices that may not support all the options for SYSREF generation, and to accommodate systems where SYSREF will be ‘turned off’ during normal operation, the following requirements for subclass-1 devices apply:

- RX and TX logic Devices shall have the ability to issue a ‘generate SYSREF’ request, which enables the clock generator (or other SYSREF generating device) to generate one or more SYSREF pulses to all devices in the system or at least those connected to the link.
- All TX and RX devices shall have the ability to determine whether or not to adjust the phase alignment of their LMFC/LEMCs based on the next detected SYSREF pulse. Implementation details for this function are left to the device implementer but three possible options are listed below:
    - Every SYSREF pulse may be examined by the device to determine if the existing phase alignment of the LMFC/LEMC requires adjustment.
    - A device may be instructed via device input pin or control interface command to use the next detected SYSREF pulse to force a phase alignment of the LMFC/LEMC.
    - A device may be instructed via device input pin or control interface command to ignore all future SYSREF pulses.

NOTE For subclass 1 devices, LMFC/LEMC phase realignment based on SYSREF will only be necessary in cases where a device is being initialized or where a link has exhibited a failure and wants to rule out LMFC/LEMC misalignment as a cause.

4.2.4 Deterministic latency using SYNC~ detection (device subclass 2)

Please refer to 8.5.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3 Physical timing

4.3.1 Device clock

The device clock is the external timing reference of each transmitter and receiver converter device in the JESD204 system. It is not mandatory that all devices share a common device clock. However, all device clocks in the system shall be phase locked to a common source, which is known as the source clock. The devices shall be able to generate and receive data at a constant rate derived from their device clock. Inside the devices, all internal clocks that influence the delay on the link shall be derived from the device clock.

The permitted relationships between the periods of the TX and RX device clock and the durations of data structures on the link depend on the JESD204 subclass as follows:

- Subclass 0: The permitted relationships between the device clock period and the duration of the frame, 8B/10B character, 64B/66B block or 64B/80B block shall be specified by the device implementer. The device implementer shall specify possible extra restrictions on the device clock period for backwards compatibility with JESD204A or JESD204B, if applicable.
- Subclass 1: The duration of the multiframe or extended multiblock shall be a whole number of device clock periods.
- Subclass 2 (8B/10B link layer only): The multiframe duration shall be a whole number of device clock periods. Additionally, the TX device clock period shall be a whole number of RX device clock periods, or the RX device clock period shall be a whole number of TX device clocks periods.

NOTE The above requirements for subclasses 1 and 2 are necessary, in order that the devices can align the LMFC or LEMC edges (see 4.3.4) to the device clock edges.

The electrical characteristics of the device clock signal are not specified in this document, because other considerations may determine these characteristics. It is expected that Low Voltage CMOS type electrical characteristics, as specified by such JEDEC standards as JESD8-5 (2.5 V) or JESD8-7 (1.8 V) would be used for this signal. Higher clock rates might demand the use of a differential signal format, such as LVPECL, LVDS or CML.

The device clock input's frequency and phase stability (jitter), however, have a direct bearing on the operation of this interface. For this reason, manufacturers shall specify the required jitter for the device clock.

Other circuitry in the transmitter or receiver device may require more restrictive jitter characteristics and may also have additional requirements such as duty cycle.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.2 Link layer clock

In this standard, the link layer clock is defined as the clock that sequences the signal processing in the link layer and the parallel interface between the link layer and the physical layer. The skew budget is based on the following requirement for the duration T_{LLC} of one link layer clock cycle:

$$T_{LLC} \leq \begin{cases} 20\ \text{UI}, & f_b \leq 4\ \text{Gbps} \\ 40\ \text{UI}, & 4\ \text{GHz} < f_b \leq 8\ \text{Gbps} \\ 80\ \text{UI}, & f_b > 8\ \text{Gbps} \end{cases}$$

where f_b is the serial bit rate and UI is the duration of a serial bit.

JESD204 does not specify the device implementation. The device implementer may deviate from the above assumptions, but such a deviation shall not lead to a violation of the skew budget (see 4.3.8).

4.3.3 Transport layer clock

In this standard, the transport layer clock is defined as the clock that sequences the signal processing in the transport layer and the parallel interface between the transport layer and the link layer. The skew budget is based on the following requirement for the duration T_{TLC} of one transport layer clock cycle:

For 8B/10B and 64B/80B encoding: $T_{TLC} \leq \begin{cases} 20\ \text{UI}, & f_b \leq 4\ \text{Gbps} \\ 40\ \text{UI}, & 4\ \text{GHz} < f_b \leq 8\ \text{Gbps} \\ 80\ \text{UI}, & f_b > 8\ \text{Gbps} \end{cases}$

For 64B/66B encoding: $T_{TLC} \leq 66\ \text{UI}$

where f_b is the serial bit rate and UI is the duration of a serial bit.

JESD204 does not specify the device implementation. The device implementer may deviate from the above assumptions, but such a deviation may not lead to a violation of the skew budget (see 4.3.8).

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.4 Local multiframe and extended multiblock clocks (LMFC and LEMC)

The local multiframe clock (LMFC, used with 8B/10B encoded data) and the local extended multiblock clock (LEMC, used with 64B/66B and 64B/80B encoded data) are internal clocks used to align the boundaries of the multiframes or extended multiblocks between lanes and – in deterministic latency devices – to an external reference (SYSREF or MULTIREF in subclass 1 and SYNC~ in subclass 2). The use of an LMFC / LEMC is mandatory in subclasses 1 and 2 and optional in subclass 0.

NOTE Subclass 2 is only applicable to 8B/10B encoding.

The local multiframe / extended multiblock clocks in device subclasses 1 and 2 shall comply with the following requirements:

- The LMFC / LEMC period in all transmitter and receiver devices shall be identical.
- The LMFC / LEMC shall be derived from the device clock.
- At each side of the link, the LMFC / LEMC shall be common to all lanes of the link.
- In subclass 1 devices, there shall be a fixed or programmable offset between the rising or falling edge of the LMFC / LEMC and the device clock edge upon which the SYSREF / MULTIREF signal is detected as being active.
- Subclass 1 devices shall be able to measure the amount of device clock cycles by which the detected active edge of the SYSREF / MULTIREF signal deviates from its expected position and not to realign the LMFC / LEMC if the deviation from the expected position is less than a programmable number of device clock cycles.

If a device supports multiple JESD204 links, the above requirements are applied separately to each link.

4.3.5 SYSREF signal (device Subclass 1)

In Subclass 1 deterministic latency systems, a signal named SYSREF is distributed to all transmitter and receiver devices in the system. The purpose of SYSREF is to identify the device clock edge that shall be used to align the phase of the LMFC or LEMC.

SYSREF can be either a periodic, one-shot (strobe-type), or “gapped” periodic signal. It is an active high signal, which is sampled by the device clock. Devices shall be able to support sampling SYSREF on the device clock rising edge, but may optionally allow SYSREF to be sampled on the device clock falling edge.

For periodic or “gapped” periodic SYSREF signals, the period shall be an integer multiple of the LMFC or LEMC period. The phase of the LMFC or LEMC within a device shall be determined by the device clock sampling edge upon which the sampled SYSREF value has transitioned from “0” to “1”.

It is not mandatory that a common SYSREF signal is generated to all devices in the system. However, it is required that SYSREF signals be generated to all devices in a manner that ensures a deterministic relationship between the moments when SYSREF is sampled active by all devices in the system. A timing diagram illustrating this concept (using device clock rising edge to sample SYSREF) and showing simultaneous sampling of the active SYSREF signal at both devices is shown in Figure 6.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.5 SYSREF signal (device Subclass 1) (cont'd)

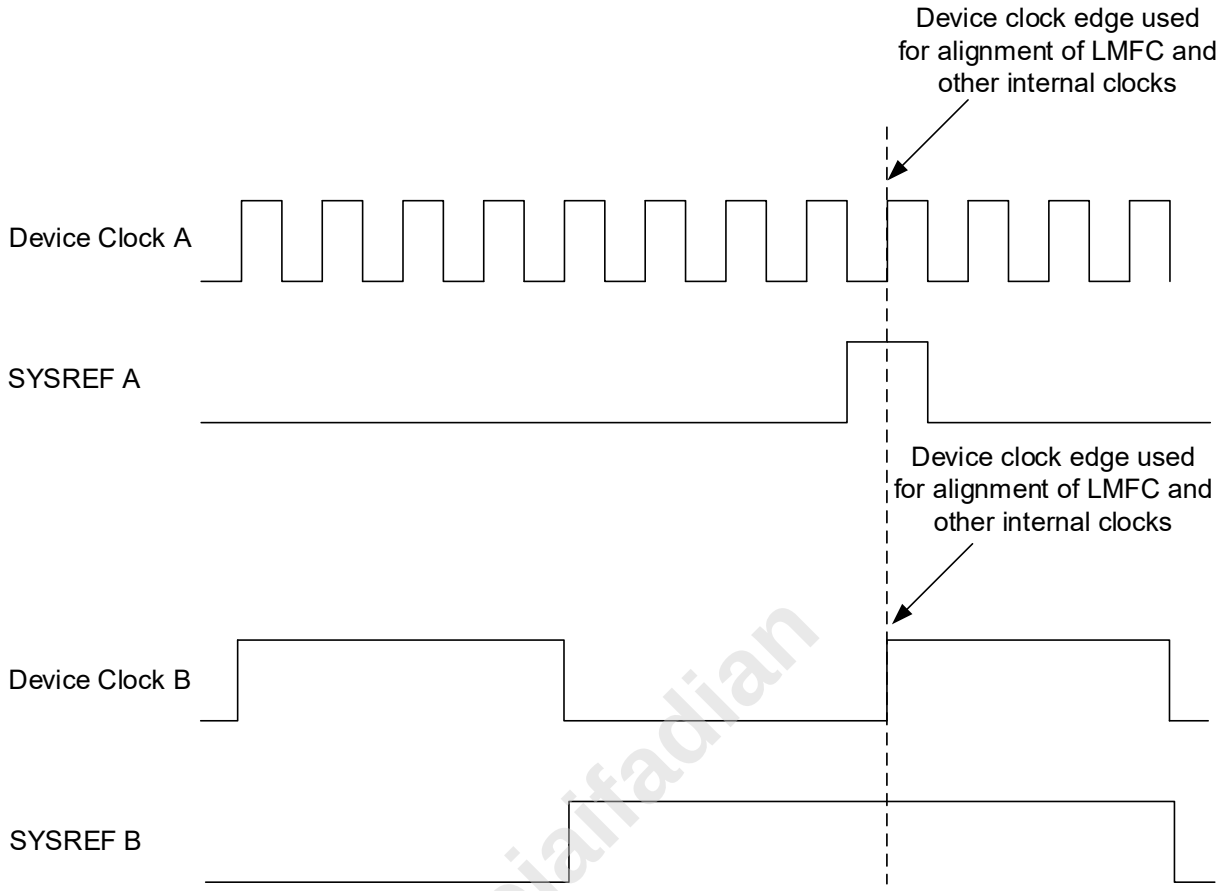


Figure 6 — SYSREF sampling illustration

weiaifadian

Delay uncertainty across the link is caused by the uncertainty in phase alignment between the LMFCs / LEMCs in the TX and RX devices within the system. Minimizing delay uncertainty relies on the generation of the LMFC / LEMC in both the TX and RX to be governed by events that happen at the same time instant. For Subclass-1 devices, this time instant corresponds to the device clock cycle upon which a "0" to "1" transition on SYSREF is detected. Therefore, it is recommended that the system implementer follows these guidelines for minimizing phase alignment offset between TX and RX LMFCs / LEMCs.

1. The device clock and SYSREF signal to each device should be routed close to each other.
2. SYSREF (source synchronous to device clock) should be distributed within the system for maximum setup/hold compliance at the device receiver.
3. SYSREF should ideally have the timing relationship relative to the device clock as shown in Figure 7. However, with very high speed device clocks, it may not be possible to meet SYSREF setup/hold time requirements. The resulting amount of delay uncertainty across the link can be minimized by controlling the phase of the SYSREF signal relative to the setup/hold time requirements of the device that is sampling it.

It is strongly recommended to use the same type of signal type for SYSREF and the device clock, to maintain an accurate timing relationship.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.5 SYSREF signal (device Subclass 1) (cont'd)

Figure 7 shows the critical timing specifications relating to the SYSREF signal. The values for these parameters are not specified here but the transmitter and receiver device specifications shall specify these values.

$\mathbf{t}_{SU\_S}$ (min) and $\mathbf{t}_{H\_S}$ (min): Setup and hold times of SYSREF with respect to device clock at the device SYSREF pin. Subclass-1 transmitter and receiver devices shall specify these parameters.

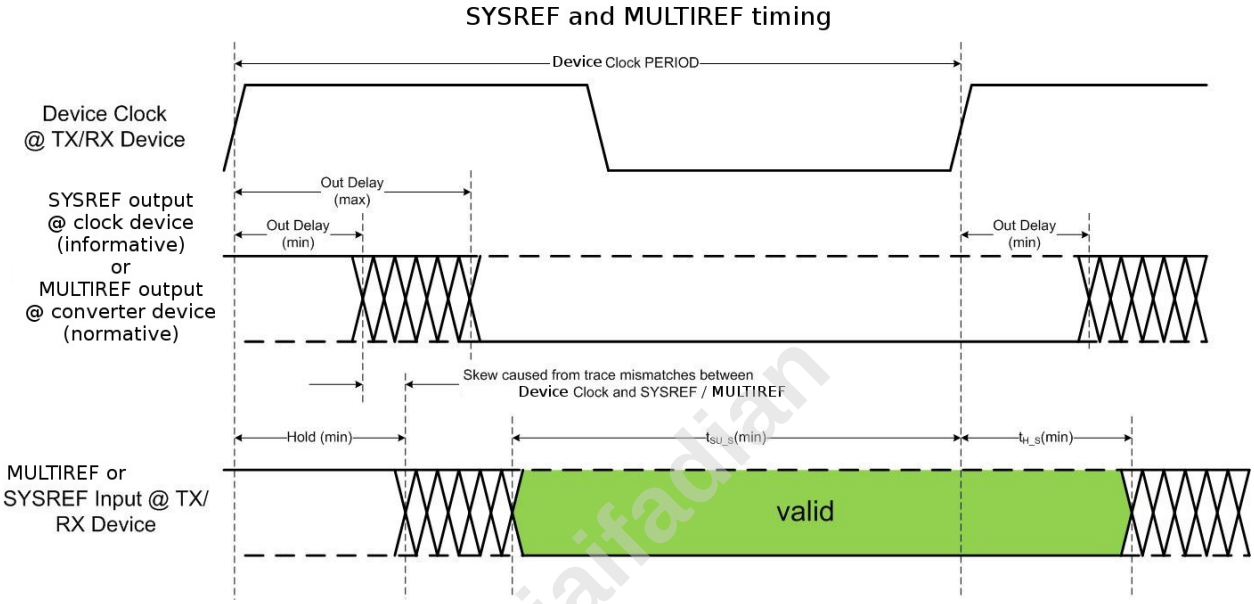


Figure 7 — SYSREF and MULTIREF signal timing

4.3.6 MULTIREF signal (device subclass 1)

In applications where lane alignment between multiple converter devices is required, but no deterministic latency is needed, the SYSREF signal may be replaced by the optional MULTIREF signal. The MULTIREF signal is fed to the SYSREF input of the converter devices. The converter devices shall use the MULTIREF signal to align their LMFCs or LEMCs, exactly in the same way as SYSREF. However, unlike SYSREF, the MULTIREF signal is generated in a converter device, not in the clock generator. For this reason, it may be more difficult to meet the setup and hold requirements relative to the device clock at the device inputs. Hence, the alignment obtained by means of the MULTIREF signal is possibly less accurate than what can be obtained by the SYSREF signal. Despite its disadvantages of no deterministic latency and possibly reduced alignment accuracy, the MULTIREF signal may be a useful option, because it simplifies the clocking architecture.

Figure 7 shows the critical timing specifications relating to the MULTIREF signal. The specification of the setup and hold times is common with the SYSREF signal. In addition, the specification of the devices that are capable of transmitting the MULTIREF signal shall specify the delay of the MULTIREF output relative to the active device clock edge. It is recommended that this delay is programmable, to facilitate compliance with the setup and hold times of the MULTIREF receiving device(s).

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.6 MULTIREF signal (device Subclass 1) (cont'd)

The timing of the MULTIREF signal is controlled by a main device. A main ADC device shall align the rising edge of the transmitted MULTIREF signal to the active edge of its LMFC or LEMC. A main DAC device shall align its LMFC or LEMC and the rising edge of the transmitted MULTIREF signal to the start of the received multiframe or extended multiblock on the latest lane. Optionally, a main device may advance the transmitted MULTIREF signal by a programmable number of device clock cycles relative to the LMFC or LEMC, to compensate the transmission delay to the secondary device(s). After alignment of its LMFC to the latest lane signal, a main DAC device shall delay the received lane signals relative to the LMFC or LEMC as specified by the parameter *RBD* (see 4.2.1).

Secondary devices accept the MULTIREF signal at their SYSREF input. A secondary device shall align its LMFC or LEMC to the rising edge of the received MULTIREF signal. After alignment of its LMFC or LEMC to the received MULTIREF signal, a secondary DAC device shall delay the start of the multiframe or extended multiblock of the received lane signals relative to its LMFC or LEMC as specified by the parameter RBD, see 4.2.1.

A logic device receiver connected to a multipoint ADC link that is aligned using MULTIREF, shall behave as if it were a subclass 0 receiver: the RX buffers are released at the earliest opportunity after the detection of a multiframe or extended multiblock boundary in each lane.

There are two kinds of MULTIREF devices possible: one-directional and repeater type. One-directional devices can either receive the MULTIREF signal or transmit it, but not simultaneously. Repeater devices receive the MULTREF signal at their SYSREF input and forward it to a MULTIREF output. One-directional devices can be used in a multidrop topology, and repeater devices in a daisy chain topology, as illustrated in Figure 8. When using a daisy chain topology, it is recommended to use a periodic or "gapped" periodic MULTIREF signal, in order that the repeater device can regenerate the MULTIREF signal and align it to the received MULTIREF signal with essentially zero delay. A secondary repeater device may optionally advance the transmitted MULTIREF signal by a programmable number of device clock cycles relative to its LMFC or LEMC, to compensate the transmission delay to the next secondary device.

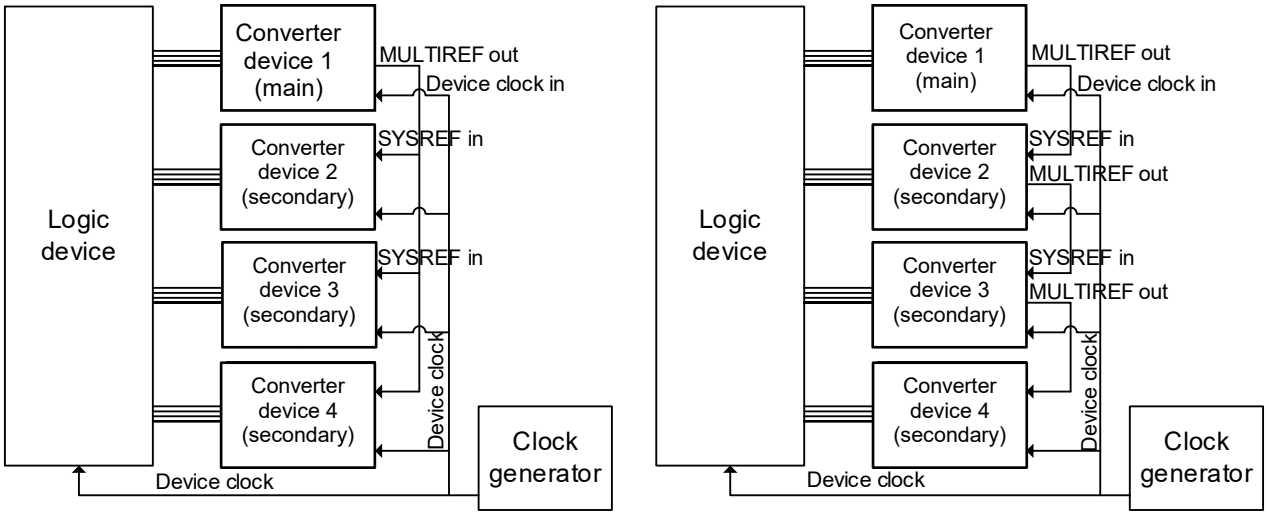


Figure 8 — Examples of MULTIREF distribution. Left: multidrop, and right: daisy chain topology

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.7 SYNC~ generation and detection clocks (8B/10B link layer)

The SYNC~ generation clock is the clock used in the RX device for generating the SYNC~ signal to the hard-wired SYNC interface (see 8.8.2). The SYNC~ detection clock is the clock used in the TX device for sampling the SYNC~ signal from the hard-wired SYNC interface (see 8.8.2).

In most of the expected applications, it will be possible to use the link layer clock as SYNC~ generation or detection clock. However, it may be necessary to use another clock in the following cases:

- for backward compatibility with a previous version of this standard (see Annex M);
- if the multiframe duration is not a whole number of link layer clock periods;
- to provide a sufficiently fine detection resolution in a subclass-2 TX (see 8.5.3.4).

In device subclasses 1 and 2, the SYNC~ generation and detection clocks shall operate at a harmonic of the LMFC and be phase aligned to the LMFC.

The duration $T_{SYNCCLK}$ of one cycle of the SYNC~ generation or detection clock shall comply to the following requirements:

For subclass 1 devices: $T_{SYNCCLK} \leq 80 \text{ UI}$

For subclass 2 devices: $T_{SYNCCLK} \leq \begin{cases} 20 \text{ UI}, & f_b \leq 4 \text{ Gbps} \\ 40 \text{ UI}, & 4 \text{ GHz} < f_b \leq 8 \text{ Gbps} \\ 80 \text{ UI}, & f_b > 8 \text{ Gbps} \end{cases}$

where

f_b is the serial bit rate and UI is the duration of a serial bit.

NOTE The above requirements ensure that the TX device can distinguish a synchronization request from an error report (see 8.8.2). The requirement for subclass 2 devices ensures in addition that the SYNC~ generation and detection clocks will not be slower than the slowest expected link layer clock, so that the skew budget can be based on the assumptions for the link layer clock.

weiaradian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget

Skew is introduced at various places in the link(s) and within the clock distribution circuit. The implementation of the system shall be such that certain amounts of skew can be tolerated without affecting system performance. In JESD204 this means that all data generated simultaneously at the TX side will be aligned to the same clock edge at the RX side. Deterministic latency requires in addition alignment to a predictable device clock edge.

NOTE Skew compares signals at either the receiving or the transmitting side of the link, not signals at opposite sides of the link.

The total skew budget can be broken down into the following parts:

a) **Transmission skew:** the skew contribution due to the different propagation delays in the transmission medium for different lanes.

b) **Processing skew:** the skew contribution due to the different signal processing delays inside the devices for different lanes. Processing skew includes the effects of both physical delays and clock domain crossings.

c) **Timing reference skew:** the skew contribution due to the different arrival times of timing reference signals at the converter devices connected to a multipoint link. In JESD204, these timing reference signals are the device clock and, depending on the type of the link layer and the device subclass, SYSREF (optionally MULTIREF) and SYNC~.

NOTE Wander is not counted in the skew value, see OIF-CEI-04.0 subclause 1.6 [12].

Transmission skew and processing skew are generally larger for multipoint links than for single links. Inside a link, the lane lengths in the transmission medium are usually matched. Between links, there may be a significant difference in lane lengths. Inside the devices, the input or output pins belonging to a single link are usually situated next to each other. The pins of different links of a multipoint link may be situated at different edges of the device package, which could lead to a significant difference in propagation delays inside large (logic) devices. Although this difference in propagation delay can be compensated by extra buffering in the shorter paths, the physical distance between the links will cause increased sensitivity to PVT and clock skew in the device. Between converter devices, the delays caused by e.g., clock boundary transitions can be different. The skew budgets have therefore an allowance for extra skew on multipoint links. This allowance is modelled by the items inter-link transmission skew and inter-link processing skew.

All deterministic latency devices are required to support multipoint links. The system implementer will generally adjust the same deterministic latency on all links of a multipoint link. Therefore, the receive buffer on the fastest lane in the system shall be able to compensate for the extra delay on the slowest lane. Hence, the receive buffer of a deterministic latency device shall be dimensioned for the maximum skew on the multipoint link. Also, the system implementer will generally adjust the same deterministic latency on each board. Even if there is only a single link, this link could be “fast” on one board and “slow” on another board. Hence, also in case of a single link, the receive buffer shall be dimensioned for the maximum skew on the multipoint link.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

Multipoint links can use one of the alignment reference signals: SYNC~, SYSREF or MULTIREF, to align lanes across devices, depending on the type of link layer and the device subclass. In general, the lanes are aligned to the active device clock edge on which a low-to-high transition of the alignment reference signal is detected. If the alignment reference signal would have a fixed timing relationship to the device clock at the device pins and setup and hold times were always met, the total effect of the timing reference skew on the timing misalignment between the lanes would thus be equal to the skew of the alignment reference signals. However, because the above conditions are not always met, the total effect may be larger by one RX or TX device clock cycle, or the sum of one RX and one TX device clock cycle, depending on which side(s) of the link the timing reference signals are received.

The skew budgets assume pessimistically that all items add up with the same sign. However, by prudent design of the layout on the board and in the devices, the system or device designer may be able to ensure that certain skew items add with opposite sign and the total skew is less than assumed in the below budgets.

The items in the skew budget are illustrated in Figure 9 and Figure 10.

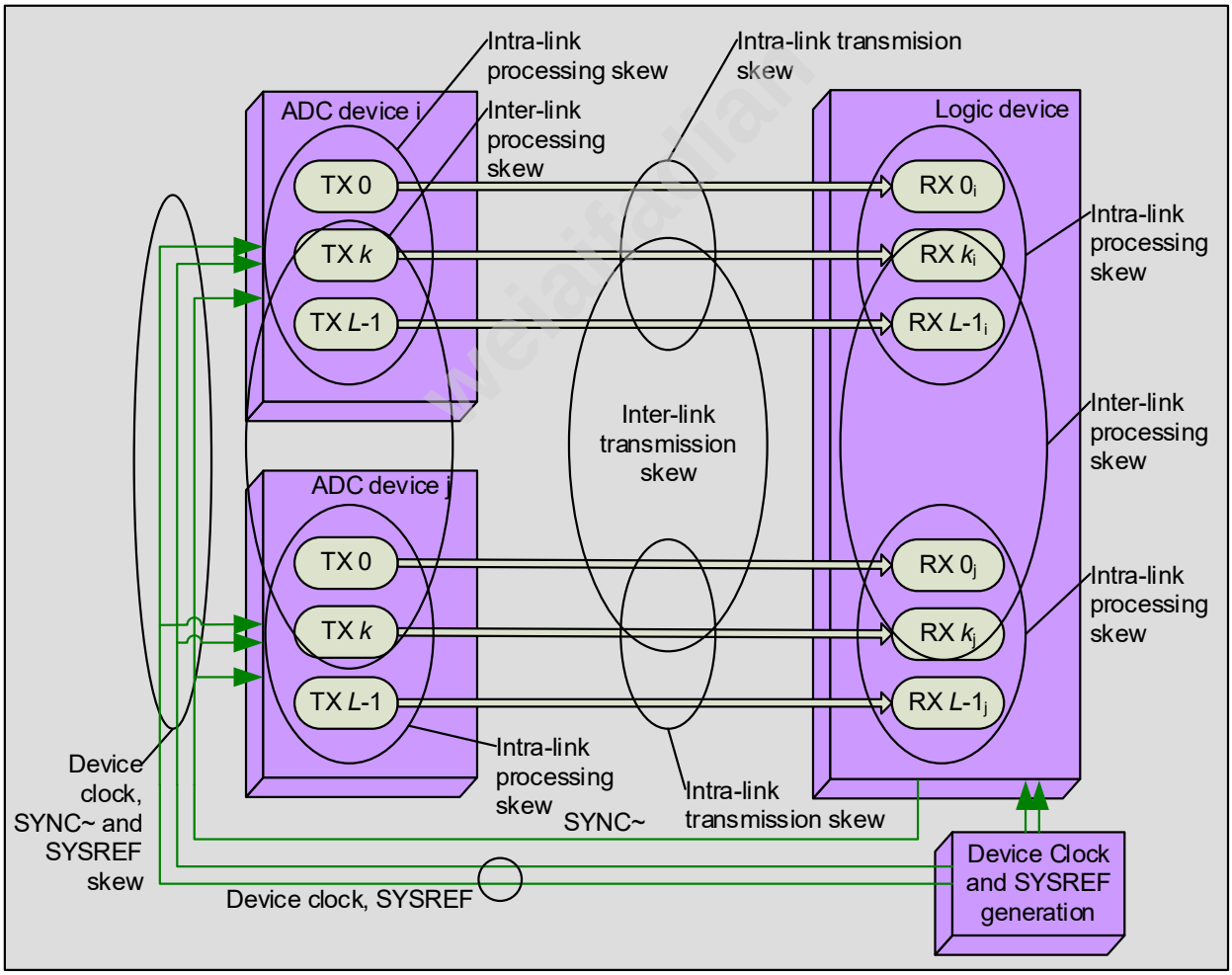


Figure 9 — Illustration of skew items in configuration with multiple ADC devices

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

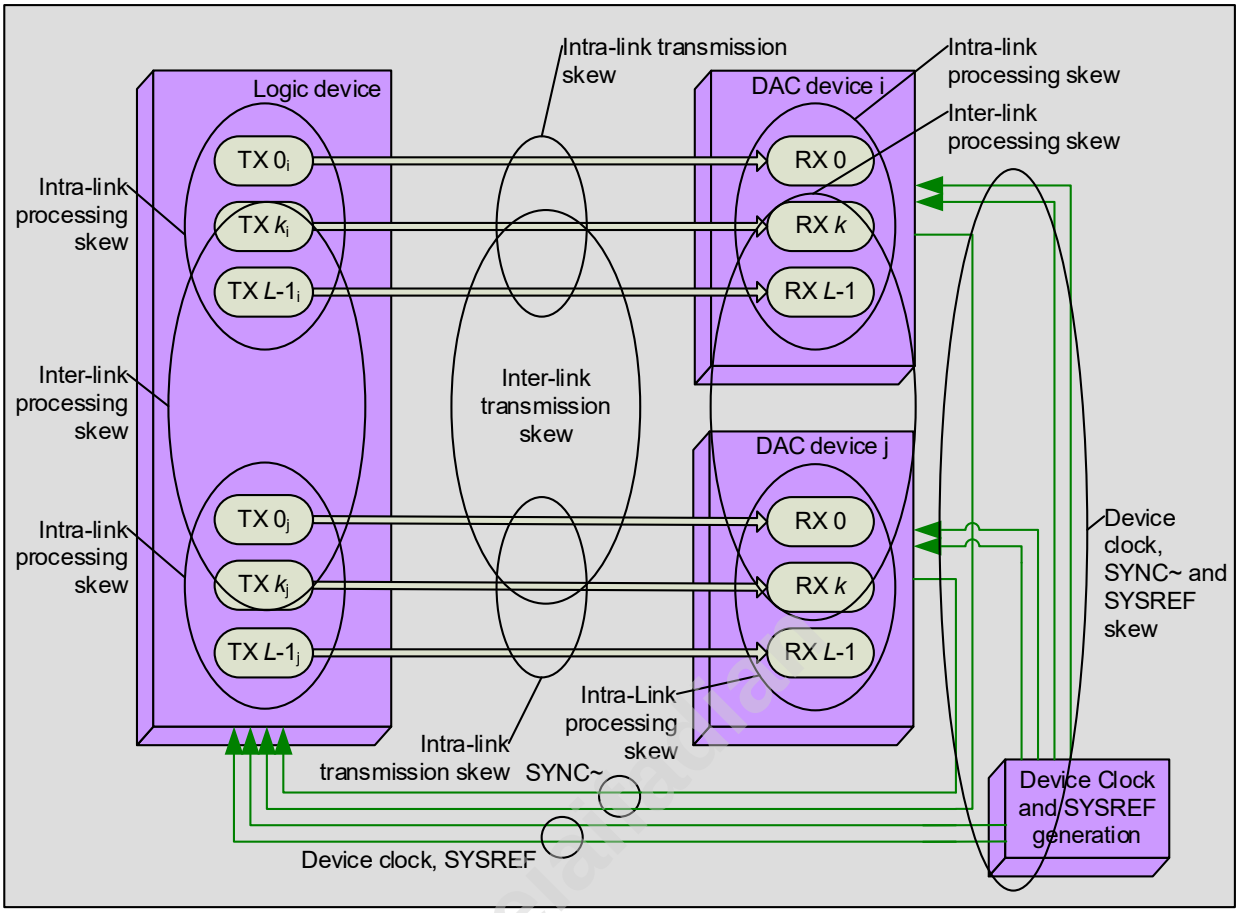


Figure 10 — Illustration of skew items in configuration with multiple DAC devices

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

The skew is specified at the following skew points, as shown in Figure 11 and Figure 12:

- SP0 = the input of the TX transport layer = reference point for deterministic latency.
- SP1 = the input of the TX link layer.
- SP2 = the output of the TX physical layer / input of the transmission medium.
- SP3 = the output of the transmission medium / input to the RX physical layer.
- SP4 = the output of the RX link layer.
- SP5 = the output of the RX transport layer = reference point for deterministic latency.

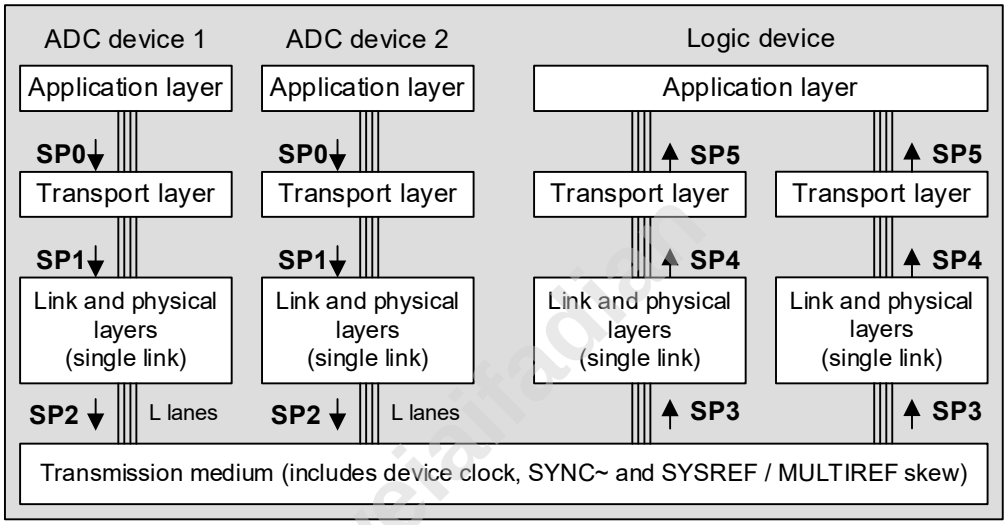


Figure 11 — Definition of the skew points on a multipoint ADC link

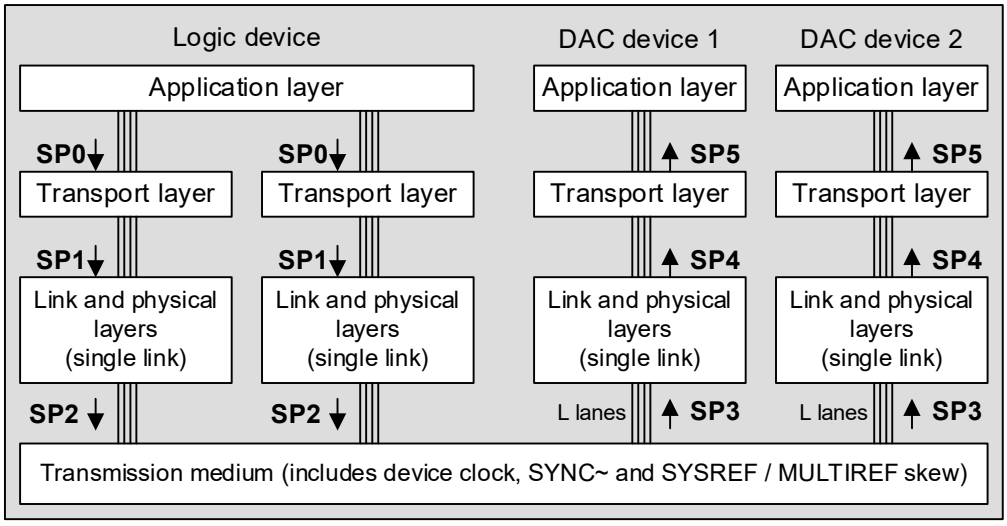


Figure 12 — Definition of the skew points on a multipoint DAC link

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

The signal path contains several clock domain crossings. Annex C provides an analysis of their effects on the skew. Each clock domain crossing gives generally a delay uncertainty of one clock cycle in the receiving clock domain. In some of the clock domain crossings it may be possible to avoid this uncertainty by careful design, but in general one cannot rely on this.

In logic devices, there may be a significant distance between the transport layer – in the user logic – and the link and physical layer, which are IP blocks by the logic device vendor situated at the periphery of the device. Therefore, in logic devices the inter-link processing skew is mainly caused by the delay differences in the connection between the transport layer and the link layers.

In converter devices, assuming a single link per device, each link is in principle identical and each device operates essentially at the same temperature and supply voltage. Therefore, if the skew budget for a single converter device allows already the maximum possible skew of one receiving clock cycle per clock domain crossing, no extra margin for inter-link processing skew is necessary, other than for tolerances in the physical delays in the devices.

The distribution of the device clock, SYNC~, SYSREF and MULTIREF, as well as the generation of the SYSREF, are under control of the system implementer. Therefore, the skew budget allocates these skew items to the transmission medium, which is also under control of the system implementer.

Table 5 through Table 9 specify the maximum skew between lanes at the various skew points. On 64B/66B and 64B/80B links, the skew at these points is defined as the difference between the times of the earliest lane and latest lane for the bit transition in the sync header of the first block in a multiblock. On 8B/10B links, the skew is defined as the difference between the times of the earliest lane and latest lane for the bit transition between the last /K/ character to the first data bit or the start of the initial lane alignment sequence. In the transmit direction, the specification is for generated skew. In the receive direction, the specification is for the skew tolerance, such that the lanes can be aligned to the desired clock edge at the output of the JESD204 link or multipoint link. The background of the numbers in the tables is provided in Annex C.

In the tables below, the numbers U_{LLC} and U_{TLC} denote the maximum expected number of unit intervals (serial bit durations) in the link layer clock and the transport layer clock, respectively. U_{LLC} and U_{TLC} depend on the serial bit rate f_b as follows:

$$U_{LLC} = \begin{cases} 20, & f_b \leq 4 \text{ Gbps} \\ 40, & 4 \text{ GHz} < f_b \leq 8 \text{ Gbps} \\ 80, & f_b > 8 \text{ Gbps} \end{cases}$$

$$U_{TLC} = \begin{cases} U_{LLC}, & \text{for 8B/10B and 64B/80B encoding} \\ 66, & \text{for 64B/66B encoding} \end{cases}$$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

Table 4 — Skew budget for subclass 0 ADC and DAC links

Skew point	Maximum skew*	Maximum added skew†	Added by
SP0	0	0	Interface application-to transport layer
SP1	1.5 ns	1.5 ns	Interface transport-to-link layer
SP2	(U_{LLC}+1) UI + 1.5 ns	(U_{LLC}+1) UI	TX link and physical layers
SP3	(U_{LLC}+1) UI + 2.5 ns	1 ns	Intra-link transmission skew
SP4	(2·U_{LLC}+ U_{TLC} +1) UI + 2.5 ns	(U_{LLC}+U_{TLC}) UI	RX link and physical layers
SP5	(2·U_{LLC}+ U_{TLC} +1) UI + 4 ns	1.5 ns	Interface link-to-transport layer

* Maximum skew is normative at skew points that are accessible by the device user, otherwise informative.

† Maximum added skew is normative between skew points that are accessible by the device user, otherwise informative.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

Table 5 — Skew budget for subclass 1 ADC links

Skew point	Maximum skew*, with SYSREF | MULTIREF	Maximum added skew†	Added by
SP0	0	0	Interface application-to transport layer
SP1	4.8 ns	1.5 ns	Interface transport-to-link layer
		3.3 ns	Delay tolerance between ADC devices
SP2	(U_{LLC}+1) UI + 4.8 ns	(U_{LLC}+1) UI	TX link and physical layers
SP3	(U_{LLC}+1) UI + (14.8 | 24.8) ns	1 ns‡	Intra-link transmission skew
		3 ns‡, §	Inter-link transmission skew
		6 ns‡, §	Device clock and SYSREF
		16 ns‡, §	Device clock and MULTIREF
SP4	(2·U_{LLC}+ U_{TLC} +1) UI + (14.8 | 24.8) ns	(U_{LLC}+U_{TLC}) UI	RX link and physical layers
SP5	(2·U_{LLC}+ U_{TLC} +1) UI + (19.6 | 29.6) ns	1.5 ns	Interface link-to-transport layer
		3.3 ns	Delay tolerance between links in logic device

* Maximum skew is normative at skew points that are accessible by the device user, otherwise informative.

† Maximum added skew is normative between skew points that are accessible by the device user, otherwise informative.

‡ The system implementer shall ensure that the total effect of the transmission skews, device clock and SYSREF | MULTIREF skew allocated for SP3 is no more than specified.

§ Either the SYSREF or the MULTIREF skew is added, depending on which signal is in use.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

Table 6 — Skew budget for subclass 1 DAC links

Skew point	Maximum skew*, with SYSREF | MULTIREF	Maximum added skew†	Added by
SP0	0	0	Interface application-to transport layer
SP1	4.8 ns	1.5 ns	Interface transport-to-link layer
		3.3 ns	Delay tolerance between links in logic device
SP2	(U_{LLC}+1) UI + 4.8 ns	(U_{LLC}+1) UI	TX link and physical layers
SP3	(U_{LLC}+1) UI + (14.8 | 24.8) ns	1 ns‡	Intra-link transmission skew
		3 ns‡	Inter-link transmission skew
		6 ns‡, §	Device clock and SYSREF
		16 ns‡, §	Device clock and MULTIREF
SP4	(2·U_{LLC}+ U_{TLC} +1) UI + (14.8 | 24.8) ns	(U_{LLC}+U_{TLC}) UI	RX link and physical layers
SP5	(2·U_{LLC}+ U_{TLC} +1) UI + (19.6 | 29.6) ns	1.5 ns	Interface link-to-transport layer
		3.3 ns	Delay tolerance between DAC devices

* Maximum skew is normative at skew points that are accessible by the device user, otherwise informative.

† Maximum added skew is normative between skew points that are accessible by the device user, otherwise informative.

‡ The system implementer shall ensure that the total effect of the transmission skews, device clock and SYSREF | MULTIREF skew allocated for SP3 is no more than specified.

§ Either the SYSREF or the MULTIREF skew is added, depending on which signal is in use.

Weitaidian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

Table 7 — Skew budget for subclass 2 ADC links

Skew point	Maximum skew	Maximum added skew[†]	Added by
SP0	0	0	Interface application-to transport layer
SP1	4.8 ns	1.5 ns	Interface transport-to-link layer
		3.3 ns	Delay tolerance between ADC devices
SP2	(U_{LLC}+1) UI + 4.8 ns	(U_{LLC}+1) UI	TX link and physical layers
SP3	(2·U_{LLC}+1) UI + 14.8 ns	1 ns[‡]	Intra-link transmission skew
		3 ns[‡]	Inter-link transmission skew
		U_{LLC} UI + 6 ns[‡]	Device clock and SYNC~
SP4	(3·U_{LLC}+ U_{TLC} +1) UI + 14.8 ns	(U_{LLC}+U_{TLC}) UI	RX link and physical layers
SP5	(3·U_{LLC}+ U_{TLC} +1) UI + 19.6 ns	1.5 ns	Interface link-to-transport layer
		3.3 ns	Delay tolerance between links in logic device

[*] Maximum skew is normative at skew points that are accessible by the device user, otherwise informative.

[†] Maximum added skew is normative between skew points that are accessible by the device user, otherwise informative.

[‡] The system implementer shall ensure that the total effect of the transmission skews, device clock and SYNC~ skew allocated for SP3 is no more than specified.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

Table 8 — Skew budget for subclass 2 DAC links

Skew point	Maximum skew	Maximum added skew[†]	Added by
SP0	0	0	Interface application-to transport layer
SP1	4.8 ns	1.5 ns	Interface transport-to-link layer
		3.3 ns	Delay tolerance between links in logic device
SP2	(U_{LLC}+1) UI + 4.8 ns	(U_{LLC}+1) UI	TX link and physical layers
SP3	(2·U_{LLC}+1) UI + 14.8 ns	1[‡]	Intra-link transmission skew
		3[‡]	Inter-link transmission skew
		U_{LLC} UI + 6 ns[‡]	Device clock and SYNC~
SP4	(3·U_{LLC}+ U_{TLC} +1) UI + 14.8 ns	(U_{LLC}+U_{TLC}) UI	RX link and physical layers
SP5	(3·U_{LLC}+ U_{TLC} +1) UI + 19.6 ns	1.5 ns	Interface link-to-transport layer
		3.3 ns	Delay tolerance between DAC devices

[*] Maximum skew is normative at skew points that are accessible by the device user, otherwise informative.

[†] Maximum added skew is normative between skew points that are accessible by the device user, otherwise informative.

[‡] The system implementer shall ensure that the total effect of the transmission skews, device clock and SYNC~ skew allocated for SP3 is no more than specified.

weiafadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

Table 9 and Table 10 specify the maximum variation between lanes. Skew variation is defined as the change in skew between any two lanes over the entire time that the link is in operation, excluding the effects of reinitialization. Each layer shall produce no more skew variation than specified and shall maintain alignment of the data to the chosen clock edge when the skew variation at its input is within the specification. The background of the numbers in the tables is provided in Annex C.

Table 9 — Maximum skew variation for subclass 0 ADC and DAC links

Skew point	Maximum skew variation* (ns)	Maximum added skew variation† (ns)	Added by
SP0	0	0	Interface application-to transport layer
SP1	0.2	0.2	Interface transport-to-link layer
SP2	0.6	0.4	TX link and physical layers
SP3	0.8	0.2	Transmission skew
SP4	1.2	0.4	RX link and physical layers
SP5	1.4	0.2	Interface link-to-transport layer

* Maximum skew variation is normative at skew points that are accessible by the device user, otherwise informative.

† Maximum added skew variation is normative between skew points that are accessible by the device user, otherwise informative.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

4.3.8 Skew and latency variation budget (cont'd)

Table 10 — Maximum skew variation for subclass 1 and 2 links

Skew point	Maximum skew variation* (ns)	Maximum added skew variation† (ns)	Added by
SP0	0	0	Interface application-to transport layer
SP1	0.2	0.2	Interface transport-to-link layer
SP2	0.6	0.4	TX link and physical layers
SP3	3.2	0.2	Transmission skew
		2.4‡	Device clock and SYSREF / MULTIREF / SYNC~
SP4	3.6	0.4	RX link and physical layers
SP5	3.8	0.2	Interface link-to-transport layer

* Maximum skew variation is normative at skew points that are accessible by the device user, otherwise informative.

† Maximum added skew variation is normative between skew points that are accessible by the device user, otherwise informative.

‡ Tolerance of the maximum specified skew variation of device clock and SYSREF/ MULTIREF / SYNC~ may require the capability of the RX and TX devices not to realign the LEMC / LMFC if the detected SYSREF / MULTIREF / SYNC~ edge deviates no more than a specified amount from its expected position.

4.4 Control interfaces

The application may require one or more control interfaces to pass information (status, control, etc.) between the devices mutually and/or between the devices and a higher layer application level. The JESD204C specification does not require a specific implementation of the control interface, however a serial interface is the recommended implementation. The control interfaces will generally be sequenced by dedicated clocks, which are generally not the same as the device clock.

4.5 Device classification

As indicated in the Scope, there is a wide application range for this standard. Converter manufacturers may decide not to support all configuration options in a certain device. However, manufacturers shall declare which classes, subclasses, and features are supported from the lists in Table 5, Table 6, and Table 7. In addition to the classes and features listed in these tables, the maximum supported line rate and the supported link configuration parameter values must also be declared.

4.5.1 Classes

Table 11 — Device classification

Device class category	Device class	Supported data interface
B	B-3	The 3.125 Gbps (max) data interface
	B-6	The 6.375 Gbps (max) data interface
	B-12	The 12.5 Gbps (max) data interface
C	C-S	The C-S (short) class 32.45 Gbps (max) data interface
	C-M	The C-M (medium) class 32.45 Gbps (max) data interface
	C-R	The C-R (reflective) class 32.45 Gbps (max) data interface

4.5.2 Device subclassification

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

In addition to the device classes defined in the previous section, a device subclass is defined in order to reference the device's ability to support deterministic latency across the JESD204 link. The converter and logic device manufacturer shall classify each device for a JESD204 link according to the scheme of Table 12.

Table 12 — Device subclassification

Device subclass	Supported data interface
Subclass 0	No Support for Deterministic Latency – Backwards compatibility with JESD204A, but no longer support for lane alignment across converter devices.
Subclass 1	Deterministic Latency supported using SYSREF signaling. When used with the MULTIREF signal, there is no deterministic latency, but still lane alignment across converter devices.
Subclass 2	Deterministic Latency supported using SYNC~ sampling. This subclass is only applicable to the 8B/10B link layer.

4.5.3 Features to be declared

The manufacturer shall declare which of the following features are supported in the device:

Table 13 — Supported features to be declared by manufacturers

Letter identifier	Reference clause	Supported feature
a	clause 0	8B/10B link layer
b	clause 0	64B/66B link layer
c	clause 0	64B/80B link layer
d	clause 0	The command channel when using the 64B/66B or 64B/80B link layer
e	clause 0	Forward error correction (FEC) when using the 64B/66B or 64B/80B link layer
f	clause 0	CRC3 when using the 64B/66B or 64B/80B link layer
g	clause 0	A physical SYNC~ pin when using the 8B/10B link layer
h	clause 0, clause 0	Subclass 0
i	clause 0, clause 0	Subclass 1
j	clause 0	Subclass 2
k	clause 0, clause 0	Lane alignment within a single link
l	clause 0, clause 0	Subclass 1 with support for lane alignment on a multipoint link by means of the MULTIREF signal
m	clause 0	SYNC~ interface timing compatibility with JESD204A
n	clause 0	SYNC~ interface timing compatibility with JESD204B

The letter identifier is provided as an optional short-hand that can be used to reference a feature set. For example, a manufacturer might declare that they support the JESD204 features a, b, d, j, l, l, and n.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5 Physical layer specification

5.1 Category B physical layer specification

5.1.1 Electrical specification overview

The electrical layer of this specification supports three physical layer classes for unidirectional, point-to-point, serial coded data rates:

- Class B-3 (LV-OIF-SxI5-based operation): from 312.5 Mbps to 3.125 Gbps (adapted from OIF SxI-5, [2]).
- Class B-6 (LV-OIF-6G-SR-based operation): from 312.5 Mbps to 6.375 Gbps (adapted from CEI-6G-SR, [5]).
- Class B-12 (LV-OIF-11G-SR-based operation): from 312.5 Mbps to 12.5 Gbps (adapted from CEI-11G-SR, [5]).

The converter devices are connected to a logic device in one of three ways:

- All devices are on the same printed circuit board and communicate at up to 3.125, 6.375, or 12.5 Gbps (depending on the physical layer class in use) using copper signal traces.
- The devices are on separate cards or boards that are connected by a backplane using one or more controlled-impedance connectors.
- The devices are on separate cards or boards that are connected by one or more cables.

When the converter devices and the logic device communicate across a controlled-impedance connector or a cable, the implementer must ensure the reliability of the link at the implementer-determined maximum signaling rate (up to 3.125, 6.375, or 12.5 Gbps depending on the physical layer class in use).

The transmission medium is point-to-point and unidirectional. The signaling defined in this specification is low-swing, differential, suitable for low-voltage IC technologies, and generally represents what is commercially referred to as “CML.”

Compliant transmitters and receivers are expected to achieve a BER less than 10^{-12} for class B-3, and less than 10^{-15} for classes B-6 and B-12.

Compliance of classes B-6 and B-12 may be verified at a BER of 10^{-12} by applying adjustments to the Gaussian Jitter (GJ) portion of the transmit and receive masks, and to the GJ portion of the reference transmitter test signal to compensate for the lower test populations (consult OIF-CEI-04.0 Clause 2 [5]).

Voltage specifications in this document follow the convention shown in Figure 13.

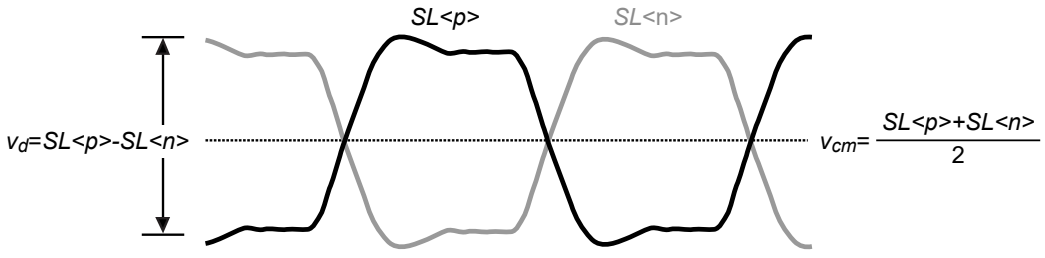


Figure 13 — Transmitter output voltage definitions. v_d is the differential voltage, v_{cm} is the common-mode voltage.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.2 Compliance types

Compliance is measured at the system-level integration of transmitter, receiver, and transmission medium and is a characteristic of the transmitter and receiver devices' external behavior.

For transmitter and receiver devices, there are two *types* of compliance – one of which may be claimed by device manufacturers:

- DC-compliance: DC parameters of the device conform to the specifications in 5.1.4, 5.1.5, or 5.1.6 (depending on the class in use).
- AC-compliance only: DC parameters of the device do not fully conform to the specifications in 5.1.4, 5.1.5, or 5.1.6 (depending on the class in use).

If either the transmitter or receiver device or both are not DC-compliant, the devices must be coupled via external capacitors. External coupling capacitors may also be used with DC-compliant devices unless the device specification specifically prohibits it. Note that a device may internally use AC-coupling in order to meet the DC electrical specifications and thus claim DC-compliance at its pins.

The system integrator must ensure that the transmitter and receiver compliance types match and that the transmission medium is appropriate for the compliance type chosen.

For the transmitter and receiver circuits, and for the system, there is only one *level* of compliance recognized for each of the three defined physical layer variants – i.e., full compliance with performance specifications detailed in this clause for the particular class being used. There is only one exception to this: a device may specify a range of data transfer rates that is a subset of the full range of data rates supported by one of the classes. A compliant device does not have to support the full data rate range.

weiaifaqen

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.3 Transmission medium

The data interface supports unidirectional, point-to-point, serial coded data rates from 312.5 Mbps to 3.125, 6.375, or 12.5 Gbps (depending on the class in use) between converter devices and a logic device using controlled impedance traces on printed circuit boards (PCBs). The data interface may also be implemented across a backplane (using controlled-impedance connectors) or across a cable. Exactly one transmitter and one receiver are allowed to be present at the ends of the transmission medium (point-to-point unidirectional connection).

The primary intended application is as a point-to-point interface of up to approximately 200 mm (approximately 8 in) between integrated circuits. Up to one connector is expected for operation above 3.125 Gbps and up to two connectors for operation below 3.125 Gbps. The performance of an actual transceiver transmission medium is highly dependent on the implementation. The link performance depends on effective channel characteristics like attenuation rather than on physical length. For operation below 6.375 Gbps a 200 mm reach is expected to be possible with a low-cost material, but above 6.375 Gbps a premium material will likely be necessary to achieve the same reach (see [2] and [5]).

The transmission medium requirements are as follows:

- The nominal characteristic impedance of the signal traces shall be 100 Ω differential.
- Depending upon the transmitter's and receiver's compliance capabilities, AC-coupling capacitors may be used in the transmission medium. The value of the capacitor depends on the minimum frequency present in the code used. Therefore the value of the AC-coupling capacitors is not specified in this document and is determined by the application.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.3.1 Transmission medium insertion loss

The transmission medium insertion loss is a measure of the signal quality of the transmission medium itself. Insertion loss will be affected by channel length, dielectric material, number of connectors, and board topology.

- The fitted transmission medium insertion loss (when calculated using the linear method) shall meet the mask requirements of the transfer model shown in Figure 14 from 50 MHz to 0.75 times the data rate. For details on calculating the linear-fitted insertion loss of the transmission medium, please refer to Annex E.
- The insertion loss deviation (ILD, calculated as the absolute difference between the fitted insertion loss and the measured insertion loss) shall not exceed 1.5 dB from 50 MHz to 0.75 times the data rate.

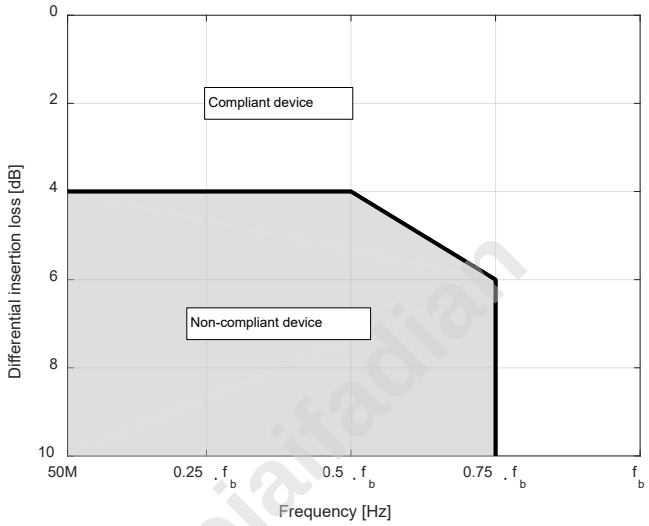


Figure 14 — Transmission medium differential insertion loss mask. Data rate f_b in Hertz.

weiaifadian

5.1.4 Class B-3 (LV-OIF-SxI5)

5.1.4.1 Compliance

The device implementer shall consult OIF SxI-5 [4] regarding the exact interpretation of the items specified in 5.1.4.2 and 5.1.4.3 and OIF-CEI-04.0 [5] regarding their measurement techniques.

A transmitter using the 64B/66B or 64B/80B link layers (see 0) shall have the ability to output PRBS31, and either PRBS9 or a repeating (clock) test pattern at the maximum data rate supported, regardless of the converter data it receives. The clock pattern consists of 8 logic ones and 8 logic zeros. A receiver using the 64B/66B or 64B/80B link layers shall have the ability to receive a PRBS31 pattern (see Annex H) and report bit errors, if any, over a given observation period.

Devices only supporting the 8B/10B link layer (see 0) may alternatively be compliance-tested with the "modified RPAT" or the JSPAT pattern (see 8.4.8) for transmitters or the JTSPAT pattern for receivers. INCITS TR-35-2004 [6] is a useful resource for signal quality and jitter tolerance measurement methodologies.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.4.2 Transmitter

Class B-3-compliant transmitters shall meet all the requirements specified in Table 14.

Table 14 — General differential output DC and AC characteristics for class B-3

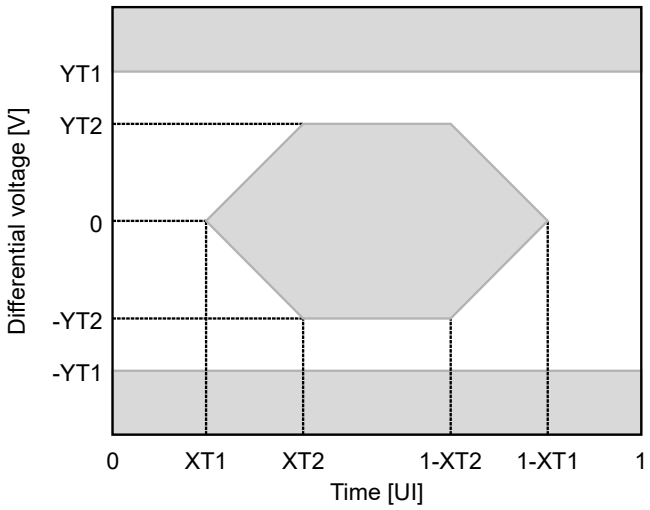
Symbol	Parameter	Conditions	Min.	Max.	Units
f_b	Data rate	NOTE 1	0.3125	3.125	Gbps
UI	Unit interval	NOTE 1	320	3,200	ps
T_{DRF}	Rise and fall times	20%–80% into 100 Ω load	50	NOTE 2	ps
V_{cm}	Transmitter common-mode voltage	Required only if DC-compliance claimed. Termination circuit of Figure 16 applied at the transmitter terminals and transmitter ground with 1.05 V $< V_{tt} <$ 1.35 V, 75 Ω $< z_{rterm} <$ 125 Ω, 0 Ω $< z_{vtt} <$ 30 Ω	0.72	1.23	V
I_{DSHORT}	Transmitter short circuit current	Transmitter terminal(s) shorted to any voltage between −0.25 V and 1.45 V, power on or off	−50	+50	mA
R_{SE}	Single-ended output impedance	At DC	35	65	Ω
R_0	Differential impedance	At DC	75	125	Ω
R_{HS}	Single-ended return loss	From 0.004 · f_b to 0.75 · f_b relative to 50 Ω	7.5	–	dB
RL_{DIFF}	Differential return loss	From 0.004 · f_b to 0.75 · f_b relative to 100 Ω	7.5	–	dB
NOTE 1	$f_b = 1/\text{UI}$. A subset of this range may be supported.				
NOTE 2	Maximum rise time is specified by the eye mask.				

The transmit eye mask specifies the signal amplitude and jitter. The eye mask is measured into a differential 100 Ω load with more than 20 dB return loss between DC and 1.6 times the data rate.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.4.2 Transmitter (cont'd)



XT1 [UI]	XT2 [UI]	YT1 [V]	YT2 [V]	DJ [p-p UI]	TJ [p-p UI]
0.175	0.45	0.50	0.25	0.17	0.35

NOTE 1 Total jitter (TJ) = deterministic jitter (DJ) + random jitter (RJ).
NOTE 2 $XT1 = TJ/2$.
NOTE 3 Unit interval (UI) is specified in Table 14.
NOTE 4 Random jitter (RJ) is defined with respect to a BER of 10^{-12} (Q = 7.04).

Figure 15 — Transmit eye mask for class B-3-based operation

5.1.4.3 Receiver

Compliant receivers shall meet all the requirements specified in this Table 15. The measurement points are at the receive device pins. The line termination shown in Figure 16 is considered part of the receiver.

Table 15 — General differential input DC and AC characteristics for class B-3

Symbol	Parameter	Conditions	Min.	Max.	Units
f_b	Data rate		0.3125	3.125	Gbps
V_{RCM}	Input common-mode voltage	Required only if DC-compliance claimed	0.70	V_{tt}	V
V_{tt}	Termination voltage	Required only if DC-compliance claimed	1.10	1.30	V
z_{vtt}	V_{tt} source impedance	At DC	–	30	Ω
z_{INDIFF}	Receiver differential impedance	At DC	75	125	Ω
L_{DR}	Differential return loss	From $0.004 \cdot f_b$ to $0.75 \cdot f_b$ relative to 100 Ω	10	–	dB

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.4.3 Receiver (cont'd)

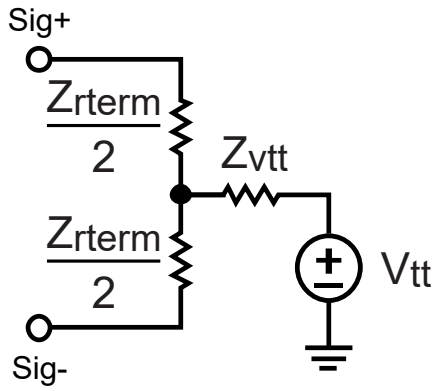
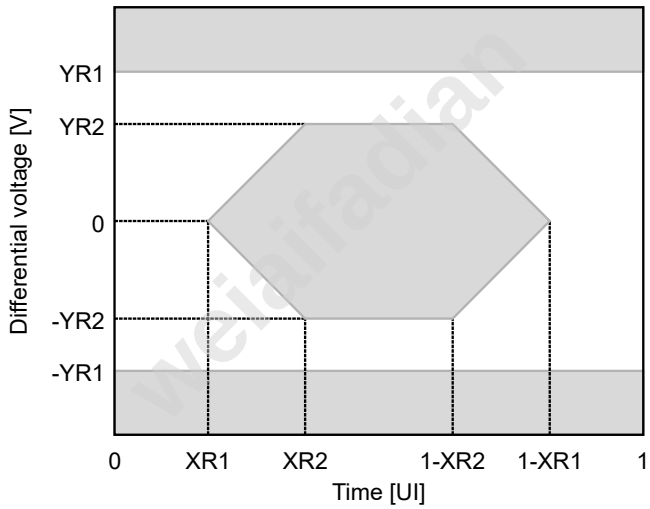


Figure 16 — Line termination at receiver

The receive eye mask specifies the signal amplitude and jitter tolerance requirements for the receiver. The eye mask is measured into a differential 100 Ω load with more than 20 dB return loss between DC and 1.6 times the data rate.



XR1 [UI]	XR2 [UI]	YR1 [V]	YR2 [V]	DJ [p-p UI]	TJ [p-p UI]
0.28	0.39	0.50	0.0875	0.32	0.56

NOTE 1 Total jitter (TJ) = deterministic jitter (DJ) + random jitter (RJ).
NOTE 2 XR1 = TJ/2.
NOTE 3 Unit interval (UI) is specified in Table 14.
NOTE 4 Random jitter (RJ) is defined with respect to a BER of 10^{-12} (Q = 7.04).

Figure 17 — Receive eye mask for class B-3-based operation

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.5 Class B-6 (LV-OIF-6G-SR)

5.1.5.1 Compliance

The device implementer shall consult OIF-CEI 3.1 [5] regarding the exact interpretation of the items specified in 5.1.5.2 and 5.1.5.3 and their measurement techniques.

A transmitter using the 64B/66B or 64B/80B link layers (see 0) shall have the ability to output PRBS31, and either PRBS9 or a repeating (clock) test pattern at the maximum data rate supported, regardless of the converter data it receives. The clock pattern consists of 8 logic ones and 8 logic zeros. A receiver using the 64B/66B or 64B/80B link layers shall have the ability to receive a PRBS31 pattern (see Annex H) and report bit errors, if any, over a given observation period.

Devices only supporting the 8B/10B link layer (see 0) may alternatively be compliance-tested with the "modified RPAT" or the JSPAT pattern (see 8.4.8) for transmitters or the JTSPAT pattern for receivers. INCITS TR-35-2004 [6] is a useful resource for signal quality and jitter tolerance measurement methodologies.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.5.2 Transmitter

Compliant transmitters shall meet all the requirements specified in Table 16.

Table 16 — General differential output DC and AC characteristics for class B-6

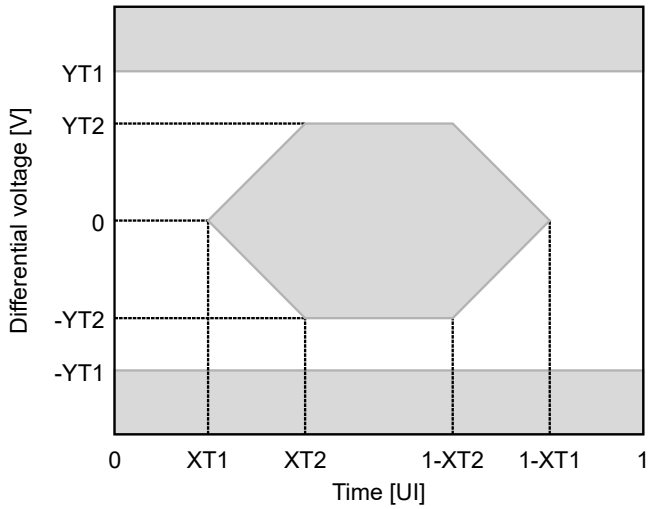
Symbol	Parameter	Conditions	Min.	Max.	Units
f_b	Data rate		0.3125	6.375	Gbps
UI	Unit interval	NOTE 1	156.9	3,200	ps
t_r, t_f	Rise and fall times	20%–80% into 100 Ω load	30	NOTE 2	ps
V_{cm}	Output common-mode voltage	Applies only to AC coupling. Termination circuit of Figure 16 applied at the transmitter terminals and transmitter ground with parameters of NOTE 3	0	1.8	V
		Required only if DC-compliance claimed. Termination circuit of Figure 16 applied at the transmitter terminals and transmitter ground with parameters of NOTE 4	735	1135	mV
V_{cm}	Output common-mode voltage	Required only if DC-compliance claimed. Termination circuit of Figure 16 applied at the transmitter terminals and transmitter ground with parameters of NOTE 5	550	1060	mV
V_{cm}	Output common-mode voltage	Required only if DC-compliance claimed. Termination circuit of Figure 16 applied at the transmitter terminals and transmitter ground with parameters of NOTE 6	490	850	mV
V_{diff}	Transmitter differential voltage	Into floating 100 Ω load	400	750	mVpp
I_{DSHORT}	Transmitter short circuit current	Transmitter terminal(s) shorted to each other or ground, power on	−100	+100	mA
R_d	Differential impedance	At DC	80	120	Ω
S_{DD22}	Differential output return loss	From 100 MHz to 0.75 · f_b	8	–	dB
S_{CC22}	Common-mode return loss	From 100 MHz to 0.75 · f_b	6	–	dB

NOTE 1 $f_b = 1/\text{UI}$. A subset of this range may be supported.
NOTE 2 Maximum rise and fall times are specified by the eye mask.
NOTE 3 $z_{tt} \geq 1$ kΩ.
NOTE 4 $z_{tt} \leq 30\ \Omega$, $80\ \Omega < z_{rterm} < 120\ \Omega, V_{tt} = 1.2$ V nominal.
NOTE 5 $z_{tt} \leq 30\ \Omega$, $80\ \Omega < z_{rterm} < 120\ \Omega, V_{tt} = 1.0$ V nominal.
NOTE 6 $z_{tt} \leq 30\ \Omega$, $80\ \Omega < z_{rterm} < 120\ \Omega, V_{tt} = 0.8$ V nominal.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.5.2 Transmitter (cont'd)

The transmit eye mask specifies the signal amplitude and jitter. The eye mask is measured into a differential 100 Ω load with more than 20 dB return loss between DC and 1.6 times the data rate.



XT1 [UI]	XT2 [UI]	YT1 [V]	YT2 [V]	T_UBHPJ [p-p UI]	T_DCD [p-p UI]	TJ [p-p UI]
0.15	0.4	0.375	0.2	0.15	0.05	0.3

NOTE 1 T_UBHPJ = Transmit uncorrelated bounded high probability jitter.
NOTE 2 T_DCD = Transmit duty cycle distortion.
NOTE 3 XT1 = TJ/2.
NOTE 4 Unit interval (UI) is specified in Table 16.
NOTE 5 The Gaussian jitter (GJ) portion of the total jitter (TJ) is defined with respect to a BER of 10^{-15} (Q = 7.94).

Figure 18 — Transmit eye mask for class B-6-based operation

weiafadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.5.3 Receiver

Compliant receivers shall meet all the requirements specified in Table 17. The measurement points are at the receive device pins. The line termination shown in Figure 16 is considered part of the receiver.

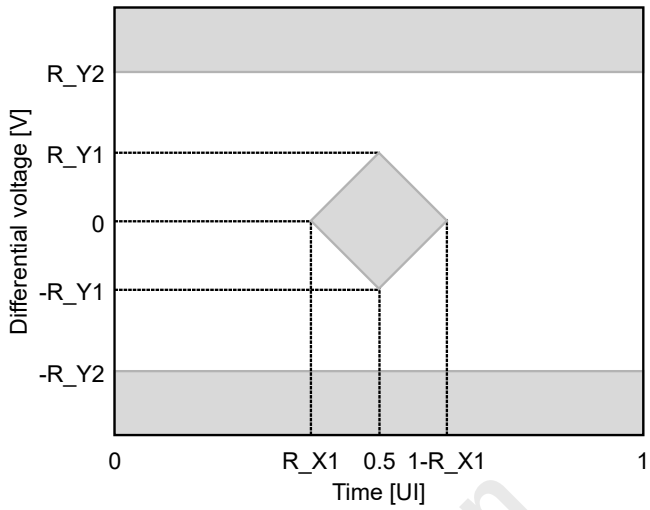
Table 17 — General differential input DC and AC characteristics for class B-6

Symbol	Parameter	Conditions	Min.	Max.	Units
f_b	Data rate		0.3125	6.375	Gbps
V_{tt}	Termination voltage	Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 1	$1.2 - 8\%$	$1.2 + 5\%$	V
		Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 2	$1.0 - 8\%$	$1.0 + 5\%$	V
V_{tt}	Termination voltage	Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 3	$0.8 - 8\%$	$0.8 + 5\%$	V
V_{cm}	Input common-mode voltage	Applies only to AC coupling	–0.05	1.85	V
		Required only if DC-compliance claimed. Line termination circuit of Figure 16 with parameters of NOTE 1	720	$V_{tt} - 10$	mV
		Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 2	535	$V_{tt} + 125$	mV
		Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 3	475	$V_{tt} + 105$	mV
V_{diff}	Input differential voltage		125	750	mVpp
z_{vtt}	V_{tt} source impedance	At DC	–	30	Ω
R_{din}	Receiver differential impedance	At DC	80	120	Ω
S_{DD11}	Differential input return loss	From 100 MHz to $0.75 \cdot f_b$ relative to 100 Ω.	8	–	dB
S_{CC11}	Common-mode input return loss	From 100 MHz to $0.75 \cdot f_b$ relative to 100 Ω	6	–	dB
NOTE 1	$80\ \Omega < z_{rterm} < 120\ \Omega, V_{tt} = 1.2$ V nominal.				
NOTE 2	$80\ \Omega < z_{rterm} < 120\ \Omega, V_{tt} = 1.0$ V nominal.				
NOTE 3	$80\ \Omega < z_{rterm} < 120\ \Omega, V_{tt} = 0.8$ V nominal.				

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.5.3 Receiver (cont'd)

The receive eye mask specifies the signal amplitude and jitter tolerance requirements for the receiver. The eye mask is measured into a differential 100 Ω load with more than 20 dB return loss between DC and 1.6 times the data rate.



R_X1 [UI]	1-R_X1 [UI]	R_Y1 [V]	R_Y2 [V]	R-SJ-hf [p-p UI]	R-SJ-max [p-p UI]	R_BHPJ [p-p UI]	TJ [p-p UI]
0.3	0.7	0.0625	0.375	0.05	5	0.45	0.6

NOTE 1 R-SJ-hf = Receive sinusoidal jitter, high frequency.
NOTE 2 R_BHPJ = Receive bounded high probability jitter. Breakdown is 0.15 p-p UI uncorrelated, 0.30 p-p UI correlated.
NOTE 3 R_X1 = TJ/2.
NOTE 4 Unit interval (UI) is specified in Table 16
NOTE 5 Total jitter (TJ) includes high-frequency sinusoidal jitter (R-SJ-hf).
NOTE 6 The Gaussian jitter (GJ) portion of the total jitter (TJ) is defined with respect to a BER of 10^{-15} (Q = 7.94).

Figure 19 — Receive eye mask for class B-6–based operation

5.1.6 Class B-12 (LV-OIF-11G-SR)

5.1.6.1 Applicability above 11.1 Gbps

Whilst LV-OIF-11G-SR defines data rates between 9.95 Gbps to 11.1 Gbps, this data interface defines the maximum data rate as 12.5 Gbps. The device implementer shall consult OIF CEI 3.1 [5], but increase the T_Baud and R_Baud maximum value to 12.5 Gbps in Table 8-1 and Table 8-4 of that document.

For data rates of up-to 11.1 Gbps, the normalized bit time (UI) used for transmitter and receiver eye-masks shall be 1/data rate. However, for data rates greater than 11.1 Gbps, the normalized bit time (UI) used for transmit and receive eye-masks shall be 1/11.1 Gbps = 90.09 ps.

weiaifabian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.6.2 Compliance

The device implementer shall consult OIF-CEI 3.1 [5] regarding the exact interpretation of the items specified in 5.1.6.3 and 5.1.6.4 and their measurement techniques.

A transmitter using the 64B/66B or 64B/80B link layers (see 0) shall have the ability to output PRBS31, and either PRBS9 or a repeating (clock) test pattern at the maximum data rate supported, regardless of the converter data it receives. The clock pattern consists of 8 logic ones and 8 logic zeros. A receiver using the 64B/66B or 64B/80B link layers shall have the ability to receive a PRBS31 pattern (see Annex H) and report bit errors, if any, over a given observation period.

Devices only supporting the 8B/10B link layer (see 0) may alternatively be compliance-tested with the "modified RPAT" or the JSPAT pattern (see 8.4.8) for transmitters or the JTSPAT pattern for receivers. INCITS TR-35-2004 [6] is a useful resource for signal quality and jitter tolerance measurement methodologies.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.6.3 Transmitter

Compliant transmitters shall meet all the requirements specified in Table 18.

Table 18 — General differential output DC and AC characteristics for class B-12

Symbol	Parameter	Conditions	Min.	Max.	Units
f_b	Data rate		6.375	12.5	Gbps
UI	Unit interval	NOTE 1	80	156.9	ps
t_r, t_f	Rise and fall times	20%–80% into 100 Ω load	24	NOTE 2	ps
V_{cm}	Output common-mode voltage	Applies only to AC coupling. Termination circuit of Figure 16 applied at the transmitter terminals and transmitter ground with parameters as per NOTE 3	0	1.8	V
		Required only if DC-compliance is claimed. Termination circuit of Figure 16 applied at the transmitter terminals and transmitter ground with parameters as per NOTE 4	735	1135	mV
V_{cm}	Output common-mode voltage	Required only if DC-compliance is claimed. Termination circuit of Figure 16 applied at the transmitter terminals and transmitter ground with parameters as per NOTE 5	550	1060	mV
		Required only if DC-compliance is claimed. Termination circuit of Figure 16 applied at the transmitter terminals and transmitter ground with parameters as per NOTE 6	490	850	mV
V_{diff}	Transmitter differential voltage	Into floating 100 Ω load	360	770	mVpp
I_{DSHORT}	Transmitter short circuit current	Transmitter terminal(s) shorted to each other or ground, power on	−100	+100	mA
R_d	Differential impedance	At DC	80	120	Ω
S_{DD22}	Differential output return loss	From 100 MHz to $0.75 \cdot f_b$	8	–	dB
S_{CC22}	Common-mode return loss	From 100 MHz to $0.75 \cdot f_b$	6	–	dB

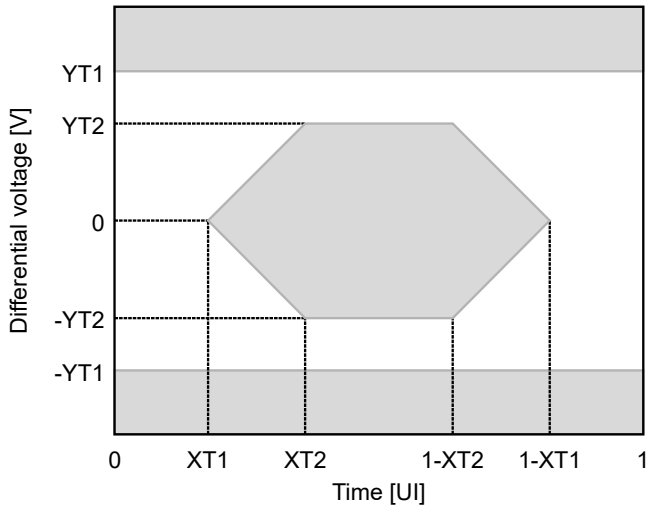
NOTE 1 $f_b = 1/\text{UI}$. A subset of this range may be supported.
NOTE 2 Maximum rise and fall times are specified by the eye mask.
NOTE 3 $z_{tt} \geq 1$ kΩ.
NOTE 4 $z_{tt} \leq 30\ \Omega$, $80\ \Omega < z_{rterm} < 120\ \Omega, V_{tt} = 1.2$ V nominal.
NOTE 5 $z_{tt} \leq 30\ \Omega$, $80\ \Omega < z_{rterm} < 120\ \Omega, V_{tt} = 1.0$ V nominal.
NOTE 6 $z_{tt} \leq 30\ \Omega$, $80\ \Omega < z_{rterm} < 120\ \Omega, V_{tt} = 0.8$ V nominal.

Verification

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.6.3 Transmitter (cont'd)

The transmit eye mask specifies the signal amplitude and jitter. The eye mask is measured into a differential 100 Ω load with more than 20 dB return loss between DC and 1.6 times the data rate.



XT1 [UI]	XT2 [UI]	YT1 [V]	YT2 [V]	T_UBHPJ [p-p UI]	T_DCD [p-p UI]	TJ [p-p UI]
0.15	0.4	0.385	0.18	0.15	0.05	0.3

NOTE 1 T_UBHPJ = Transmit uncorrelated bounded high probability jitter.
NOTE 2 T_DCD = Transmit duty cycle distortion.
NOTE 3 XT1 = TJ/2.
NOTE 4 The Gaussian jitter (GJ) portion of the total jitter (TJ) is defined with respect to a BER of 10^{-15} (Q = 7.94).

Figure 20 — Transmit eye mask for class B-12-based operation

weiafadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.6.4 Receiver

Compliant receivers shall meet all the requirements specified in Table 19. The measurement points are at the receive device pins. The line termination shown in Figure 16 is considered part of the receiver.

Table 19 — General differential input DC and AC characteristics for class B-12

Symbol	Parameter	Conditions	Min.	Max.	Units
f_b	Data rate		6.375	12.5	Gbps
V_{tt}	Termination voltage	Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 1	$1.2 - 8\%$	$1.2 + 5\%$	V
V_{tt}	Termination voltage	Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 2	$1.0 - 8\%$	$1.0 + 5\%$	V
		Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 3	$0.8 - 8\%$	$0.8 + 5\%$	V
V_{cm}	Input common-mode voltage	Applies only to AC coupling	–0.05	1.85	V
		Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 1	720	$V_{tt} - 10$	mV
		Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 2	535	$V_{tt} + 125$	mV
		Required only if DC-compliance claimed. Termination circuit of Figure 16 with parameters of NOTE 3	475	$V_{tt} + 105$	mV
V_{diff}	Input differential voltage		110	1050	mVpp
z_{vtt}	V_{tt} source impedance	At DC	–	30	Ω
R_d	Receiver differential impedance	At DC	80	120	Ω
S_{DD11}	Differential input return loss	From 100 MHz to $0.75 \cdot f_b$ relative to 100 Ω	8	–	dB
S_{CC11}	Common-mode input return loss	From 100 MHz to $0.75 \cdot f_b$ relative to 100 Ω	6	–	dB

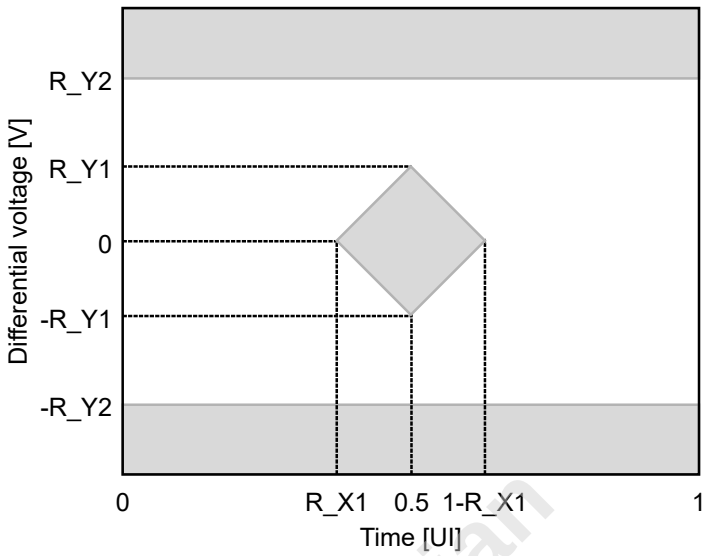
NOTE 1 $z_{tt} \leq 30\ \Omega$, $80\ \Omega < z_{rterm} < 120\ \Omega$, $V_{tt} = 1.2$ V nominal.
NOTE 2 $z_{tt} \leq 30\ \Omega$, $80\ \Omega < z_{rterm} < 120\ \Omega$, $V_{tt} = 1.0$ V nominal.
NOTE 3 $z_{tt} \leq 30\ \Omega$, $80\ \Omega < z_{rterm} < 120\ \Omega$, $V_{tt} = 0.8$ V nominal.

Preliminary

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.1.6.4 Receiver (cont'd)

The receive eye mask specifies the signal amplitude and jitter tolerance requirements for the receiver. The eye mask is measured into a differential 100 Ω load with more than 20 dB return loss between DC and 1.6 times the data rate.



R_X1 [UI]	1-R_X1 [UI]	R_Y1 [V]	R_Y2 [V]	R-SJ-hf [p-p UI]	R-SJ-max [p-p UI]	R_BHPJ [p-p UI]	TJ [p-p UI]
0.35	0.65	0.055	0.525	0.05	5	0.45	0.7

NOTE 1 R-SJ-hf = Receive sinusoidal jitter, high frequency.
NOTE 2 R_SJ-max = Receive sinusoidal jitter, maximum.
NOTE 3 R_BHPJ = Receive bounded high probability jitter. Breakdown is 0.25 p-p UI uncorrelated, 0.20 p-p UI correlated.
NOTE 4 R_X1 = TJ/2.
NOTE 5 Total jitter (TJ) includes high-frequency sinusoidal jitter (R_SJ-hf).
NOTE 6 The Gaussian jitter (GJ) portion of the total jitter (TJ) is defined with respect to a BER of 10^{-15} (Q = 7.94).

Figure 21 — Receive eye mask for class B-12-based operation

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2 Category C physical layer specification

5.2.1 Overview

Category C devices shall operate with data rates from 6.375 Gbps up to 32.45 Gbps with raw[1] bit error rate at or below 10^{-15} using line coding[2], or a subset of this range. A category C transmitter has a channel equalizer providing at least 9 dB of boost. Three receiver classes are defined to minimize link power dissipation for a variety of channel types: C-S(hort), C-M(edium) and C-R(eflective), each class being a superset of the previous one. Table 20 describes, as information, the most important receiver architectural differences between the classes and recommended maximum insertion loss, IL, in the channel at half data rate, $f_b/2$. A receiver may implement one or more classes and still be considered JESD204C-compliant.

Table 20— Category C receiver features

Class	Relative power	Receiver CTLE	Receiver DFE taps	Recommended maximum channel IL @ $f_b/2$
C-S	Low	6 dB	0	12.5
C-M	Medium	9 dB	3	22.5
C-R	High	12 dB	14	27

A category C link compliance is assessed by calculating the Channel Operating Margin (COM). The Channel Operating Margin (COM) of the channel is computed using the procedure in Annex 93A of IEEE Std 802.3. A category C-compliant link is one that has a COM margin equal or higher than 2 dB.

5.2.2 Compliance

5.2.2.1 Compliance types

Compliance is measured at the system-level integration of transmitter, receiver, and transmission medium, and is a characteristic of the transmitter and receiver devices' external behavior.

For transmitter and receiver devices, there are two types of compliance – one of which may be claimed by device manufacturers:

- DC-compliance: DC parameters of the device conform to the specifications in 5.2.3 and 5.2.4.
- AC-compliance only: DC parameters of the device do not fully conform to the specifications in 5.2.3 and 5.2.4.

If either the transmitter or receiver device or both are not DC-compliant, the devices must be coupled via external capacitors. External coupling capacitors may also be used with DC-compliant devices unless the device specification specifically prohibits it. Note that a device may internally use AC-coupling to meet the DC electrical specifications and thus claim DC-compliance at its pins.

The system integrator must ensure that the transmitter and receiver compliance types match and that the transmission medium is appropriate for the compliance type chosen. If AC-coupling is included the low-frequency 3 dB cutoff of the AC-coupling shall be less than 50 kHz.

[1] Absent error correcting codes.
[2] Options include 8B/10B, 64B/66B, and 64B/80B.

weijifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.2.1 **Compliance types (cont'd)**

For the transmitter and receiver circuits, and for the system, there is only one level of compliance recognized for each of the three defined physical layer variants – i.e., full compliance with performance specifications detailed in this clause for the category (transmitter) or class (receiver) being used. There is only one exception to this: a device may specify a range of data transfer rates that is a subset of the full range of data rates supported by the category (transmitter) or one of the classes (receiver). A compliant device does not have to support the full data rate range.

5.2.2.2 System compliance

A category C system consists of a transmitter and a receiver that are connected by a transmission medium. The system can also contain interfering transmitters. The transmitter and the receiver shall be with parameters as given in Table 27, Table 28 and Table 29.

for each class respectively. The system shall have a COM margin of at least 2 dB at a bit error rate (BER) of 10^{-15}, a DER_0 value of 10^{-15}. COM shall be computed using the procedure in Annex 93A of IEEE Std 802.3. The minimum COM margin of 2 dB allocates margin for practical limitations on the receiver implementation as well as the allowed transmitter equalization coefficients.

5.2.3 Transmitter characteristics

5.2.3.1 Transmitter electrical specifications

IEEE Std. 802.3-2018 93.8.1. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The category C shall have a transmitter with a feed-forward equalizer (FFE) of at least 3 taps. The transmitter electrical characteristics shall be measured at test point TP0a (Figure 22) and shall meet the specifications of Table 21.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.1 **Transmitter electrical specifications (cont'd)**

Table 21 — Category C transmitter electrical specification at TP0a

Characteristic	Symbol	Ref.	Min.	Max.	Unit
Data rate	f_b	5.2.3.3	6.375	32.45	Gbps
Output voltage[a,b]		5.2.3.4			
Transmitter disabled				30	mV
Transmitter enabled, minimum	$v_{TX_{min}}(fs_{TX})$			400	mV
Transmitter enabled, maximum	$v_{TX_{max}}(fs_{TX})$			1200	mV
Output common-mode voltage		5.2.3.4			
DC	v_{cm}		0	1.9	V
AC				12	mV RMS
Rise/fall time (20%-80%[c])	t_{tt}	5.2.3.5	0.25		UI
Differential output return loss	RL_{DD}	5.2.3.6	See eq. (3)		dB
Common-mode output return loss	RL_{CM}	5.2.3.6	See eq. (4)		dB
Output waveform		5.2.3.7			
Steady-state voltage	v_f		200	600	mV
Linear fit pulse peak	p_{max}		$0.71 \cdot v_f$		V
Root mean square error	σ_e			$0.027 \cdot v_f$	V
Signal-to-noise-and-distortion ratio	SNR_{TX}	5.2.3.8	27		dB
Output jitter		5.2.3.13			
Even-odd jitter	DCD			0.035	UI
Effective bounded uncorrelated jitter	$EBUJ$			0.10	p-p UI
Effective total uncorrelated jitter	TUJ			0.26	p-p UI
Transmit equalizer minimum boost (NOTE 2)		5.2.3.7	9		dB

NOTE 1 IEEE Std. 802.3-2018 Table 93-4. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

NOTE 2 If the transmit filter is implemented with a finite impulse response (FIR) filter, the boost may be achieved with pre-cursor taps only, with post-cursor taps only, or a combination of pre- and post-cursor taps.

[a] Regardless of transmit equalizer setting.
[b] A compliant transmitter may support a smaller voltage range when enabled.
[c] Unit interval corresponds to maximum transmitter data rate.

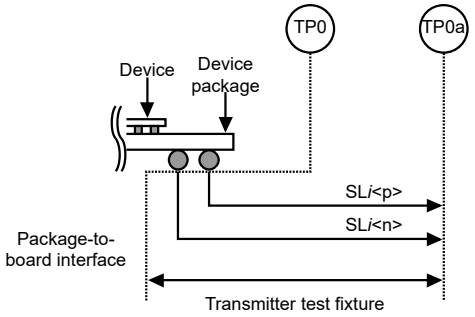
A category C transmitter shall have the ability to output PRBS9, PRBS15, and a repeating (clock) test pattern at the maximum data rate supported, regardless of the converter data it receives. The test pattern consists of 8 logic ones and 8 logic zeros. It is recommended that the transmitter should also have the ability to output other test patterns: PRBS7, SSPS-64 (advantageous to correctly model the inter-symbol interference found in 64B/66B-encoded data, see OIF-CEI-04.0 2.D.4 [OIF-CEI-04.0, Common Electrical I/O (CEI) - Electrical and Jitter Interoperability agreements for 6G+ bps and 11G+ bps and 25G+ bps I/O, Optical Internetworking Forum, February 18,]), programmable, etc. The control interface (see subclause 4.4) shall be used to make the transmitter output a test pattern, select which pattern to output, etc. The PRBS generation is specified in Annex H.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.2 Transmitter test fixture

IEEE Std. 802.3-2018 93.8.1.1. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Unless otherwise noted, measurements of the transmitter are made at the output of a test fixture (TP0a) as shown in Figure 22.



NOTE IEEE Std. 802.3-2018 Figure 93-5. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Figure 22 — Transmitter test fixture and test points

The differential insertion loss of the test fixture (in dB) at half of the transmitter maximum data rate, $f_b/2$, shall not deviate by more than 15% from the reference insertion loss of the test fixture (in dB, see (1)). Additionally, the magnitude of the differential insertion loss deviation of the test fixture shall be less than or equal to 0.1 dB from 0.05 GHz to half of the transmitter maximum data rate (see Annex F.2).

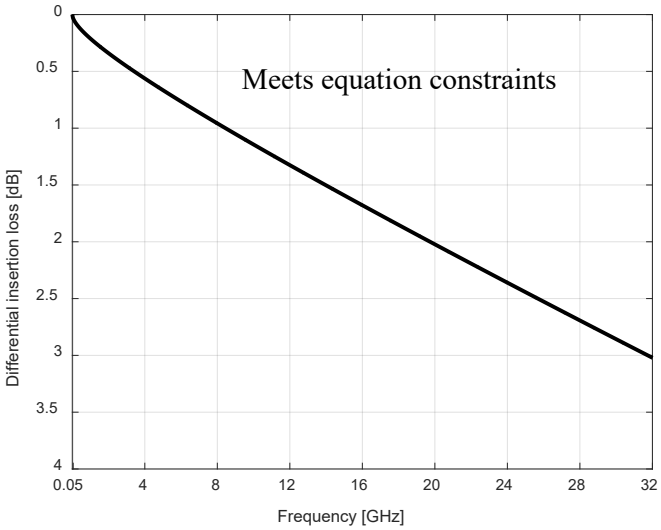
The reference insertion loss of the test fixture is defined by (1).

$$IL_{TXF_{ref}}(f, f_b) = -0.0015 + 0.144 \cdot \sqrt{f} + 0.069 \cdot f \quad 0.05 \text{ GHz} \le f \le f_b \quad \text{[dB]} \tag{1}$$

where f_b denotes the maximum transmitter data rate in GHz, and f is the frequency in GHz. The effects of differences between the insertion loss of an actual test fixture and the reference insertion loss are to be accounted for in the measurements. The reference insertion loss is illustrated in Figure 23.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.2 **Transmitter test fixture (cont'd)**



NOTE IEEE Std. 802.3-2018 Figure 93-3. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Figure 23 — Transmitter test fixture differential reference insertion loss with f_b = 32 Gbps

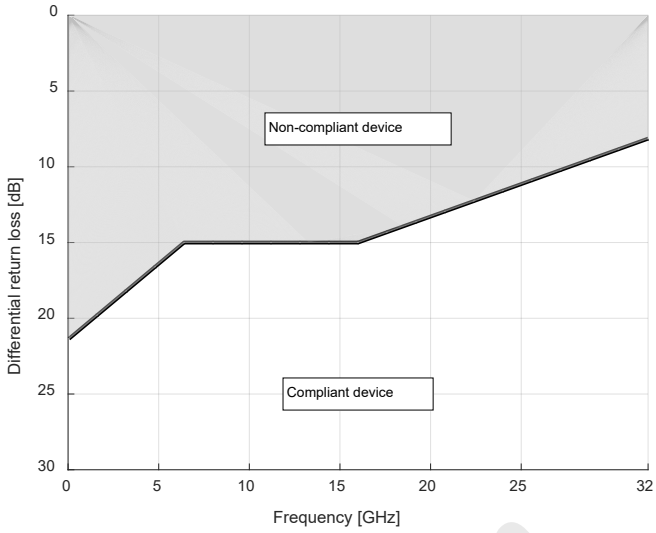
The differential return loss of the test fixture, in decibels, shall meet (2) where f_b denotes the maximum transmitter data rate in GHz. The differential return loss limit is shown in Figure 24.

$$RL_{TXF_{ref}}(f, f_b) \geq \begin{cases} 20 - 25 \cdot \frac{f}{f_b} & 0.05 \leq f \leq 0.2 \cdot f_b \\ 15 & 0.2 \cdot f_b < f \leq 0.52 \cdot f_b \\ 20.57 - 10.715 \cdot \frac{f}{f_b} & 0.52 \cdot f_b < f \leq f_b \end{cases} \text{ [dB]} \quad (2)$$

weiaffadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.2 **Transmitter test fixture (cont'd)**



NOTE IEEE Std. 802.3-2018 Figure 93-4. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Figure 24 — Transmitter test fixture differential return loss limit with f_b = 32 Gbps

The common-mode return loss of the test fixture shall be greater than or equal to 10 dB from 0.05 to half of the maximum data rate, $\frac{f_b}{2}$.

A test system with a fourth-order Bessel-Thomson low-pass response with a 3 dB bandwidth of $1.32 \cdot f_b$, where f_b represents the maximum data rate of the device, is to be used for all transmitter signal measurements, unless otherwise specified.

5.2.3.3 Signaling rate and range

IEEE Std. 802.3-2018 93.8.1.2. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The category C transmitter signaling range shall be between 6.375 Gbps±100 ppm and 32.45 Gbps±100 ppm; devices are allowed to operate in a subset of this range.

5.2.3.4 Signaling levels

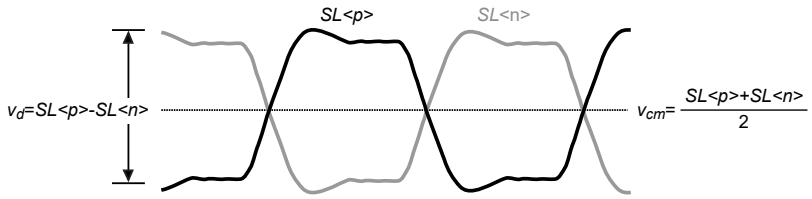
IEEE Std. 802.3-2018 93.8.1.3. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The differential output voltage v_d is defined to be *SL*<*p*> minus *SL*<*n*>. The common-mode output voltage v_{cm} is defined to be one half of the sum of *SL*<*p*> and *SL*<*n*>. These definitions are illustrated by Figure 25.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.4 **Signaling levels (cont'd)**



NOTE IEEE Std. 802.3-2018 Figure 93-6. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Figure 25 — Transmitter output voltage definitions. v_d is the differential voltage, v_{cm} is the common-mode voltage

The differential output shall be a non-return-to-zero (NRZ) signal, which shall have positive voltage when a logic 1 is to be transmitted and a negative voltage when a logic 0 is to be transmitted. The voltage resulting from the transmission of a series of consecutive logic-zeros shall be essentially the negative of the voltage resulting from the transmission of a series of consecutive logic-zeros. The peak-to-peak value of the differential output voltage shall not go above 1200 mV or below 400 mV regardless of the transmit equalizer setting.

The DC common-mode output voltage shall be between 0 V and 1.9 V with respect to signal ground. The AC common-mode output voltage shall be less than or equal to 12 mV RMS with respect to signal ground. Common-mode output voltage requirements shall be met regardless of the transmit equalizer setting.

5.2.3.5 Transmitter transition (rise/fall) time

Based on OIF-CEI-04.0 13.3.10, © 2017 Optical Internetworking Forum.

Rise and fall time define the limits on the transition time. These limits are intended to bound crosstalk as well as near-end reflections due to channel return loss.

Transition times (rise and fall times) are defined as the time between the 20% and 80% times, or 80% and 20% times, respectively, of isolated edges.

The test pattern is a square wave with eight ones and eight zeros; the 0% level and the 100% level are the average values of the center 20% of the two time intervals of the square wave.

The waveform is observed through a fourth-order Bessel-Thomson response with a bandwidth of 40 GHz.

NOTE. This definition is not the same as the rise and fall times typically reported by an oscilloscope from an eye diagram, which take all the edges into account.

weloifation

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.6 Transmitter output return loss

IEEE Std. 802.3-2018 93.8.1.4. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The differential output return loss of the transmitter shall meet (3). This output impedance requirement applies to all valid output levels. The reference impedance for differential return loss measurements shall be 100 Ω.

$$RL_{DD}(f) \geq \begin{cases} 12.05 - f, & 0.05 \leq f \leq 6 \\ 6.5 - 0.075f, & 6 < f \leq 0.75 \cdot f_b \end{cases} \quad \text{[dB]} \tag{3}$$

where f is the frequency in GHz and f_b is the maximum data rate supported by the device in GHz. Note that (3) is compatible with IEEE Std. 802.3-2018 Equation (93-3). The differential output return loss limit is illustrated by Figure 26.

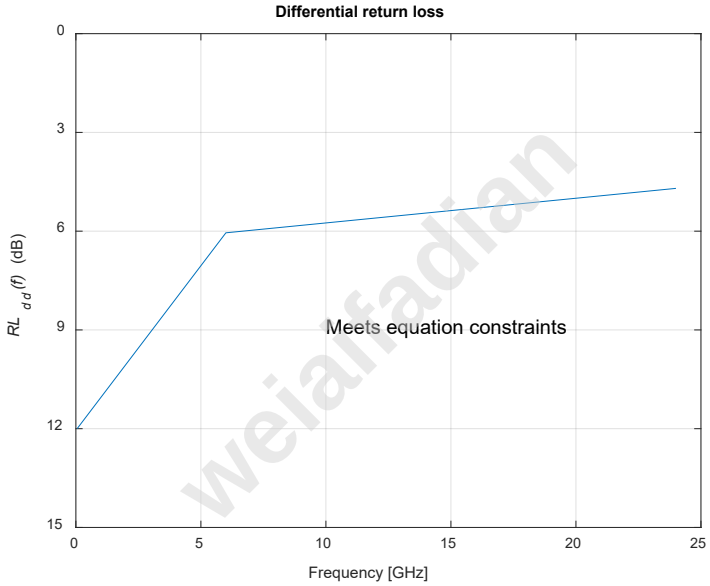


Figure 26 — Transmitter differential return loss

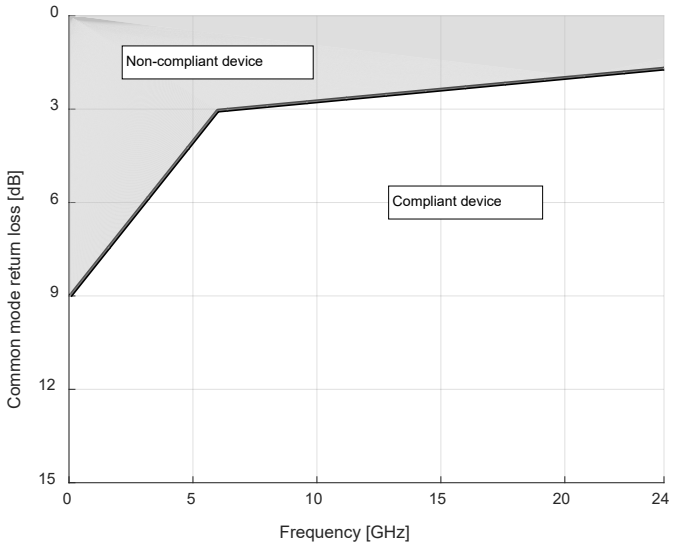
The common-mode output return loss, in dB, of the transmitter shall meet (4) where f is the frequency in GHz. This output impedance requirement applies to all valid output levels. The reference impedance for common-mode return loss measurements shall be 25 Ω.

$$RL_{CM}(f) \geq \begin{cases} 9.05 - 25.78125 \cdot \dfrac{f}{f_b} & 0.05 \leq f \leq 0.233 \cdot f_b \\ 3.5 - 1.93 \cdot \dfrac{f}{f_b} & 0.233 \cdot f_b < f \leq 0.75 \cdot f_b \end{cases} \quad \text{[dB]} \tag{4}$$

where f is the frequency in GHz and f_b is the maximum data rate supported by the device in GHz. The common return loss limit is illustrated by Figure 27.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.6 **Transmitter output return loss (cont'd)**



NOTE IEEE Std. 802.3-2018 Figure 93-8. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Figure 27 — Transmitter common-mode return loss limit

5.2.3.7 Transmitter output waveform

IEEE Std. 802.3-2018 93.8.1.5. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

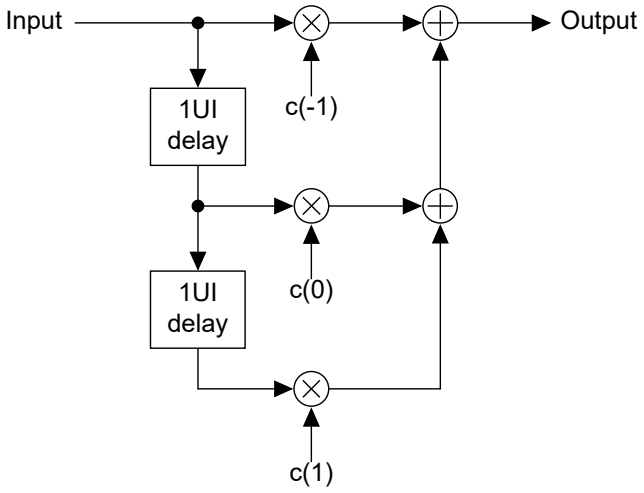
The category C transmit function includes programmable equalization to compensate for the frequency-dependent loss of the channel and facilitate data recovery at the receiver. The functional model for the transmit equalizer is the 3-tap transversal filter shown in Figure 28; the model's mathematical representation is shown in (5).

$$H(z) = c(-1) + c(0) \cdot z^{-1} + c(1) \cdot z^{-2} \quad \textbf{(5)}$$

Weiaidian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.7 **Transmitter output waveform (cont'd)**



NOTE IEEE Std. 802.3-2018 Figure 93-9. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Figure 28 — Transmit equalizer functional model

The coefficients of the transmitter equalizer shall be determined from the measured waveform during transmitter compliance test using the process as follows:

1. The transmitter under test is preset such that $c(0)$ is at its maximum value and all other coefficients are zero.

2. Capture at least one complete cycle of the test pattern PRBS9 at TP0a (TP0a is defined as the test point at the output of transmitter test fixture) per 5.2.3.2. PRBS9 generation is specified in Annex H.

3. Compute the linear fit to the captured waveform per 5.2.3.11.

4. Define t_x to be the time when the rising edge of the linear fit pulse, p, from step 3 crosses 50% of its peak amplitude.

5. Sample the linear fit pulse, p, at bit period-spaced intervals relative to the time $t_0 = t_x + 0.5$ UI, interpolating as necessary to yield the sampled pulse p_i.

6. Use p_i to compute the vector of coefficients, w, of a N_w-tap bit period-spaced transversal filter that equalizes for the transfer function from the transmit function to TP0a per 5.2.3.12.

The parameters of the pulse fit and the equalizing filter are given in Table 22.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.7 **Transmitter output waveform (cont'd)**

Table 22 — Linear fit pulse and equalizing filter parameters

Parameter	Symbol	Value	Units
Linear fit pulse length	N_p	14	UI
Linear fit pulse delay	D_p	2	UI
Equalizer length	N_w	13	UI
Equalizer delay	D_w	1	UI
NOTE IEEE Std. 802.3-2018 Table 85-6. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.			

The steady-state differential output voltage at TP0a with no pre- or post-cursor compensation applied, v_f, is defined to be the sum of the linear fit pulse $p(k)$ divided by the samples per unit interval M:

$$v_f = \frac{1}{M} \cdot \sum_{k=1}^{M \cdot N_p} p(k) \quad [\text{V}] \tag{6}$$

In (6), p is the linear fit pulse from step 3 and M is the number of samples per bit period as defined in 5.2.3.9. The peak value of the linear fit pulse from step 3, p_{max}, and the RMS value of the error between the linear fit and measured waveform from step 3, σ_e, shall satisfy the requirements of the "Output waveform" specification of Table 21.

For each configuration of the transmit equalizer, continue the procedure with the steps as follows:

7. Configure the transmitter under test as required.
8. Capture at least one complete cycle of the test pattern PRBS9 (see Annex H) at TP0a.
9. Compute the linear fit to the captured waveform per 5.2.3.11.
10. Define t_x to be the time when the rising edge of the linear fit pulse, p, from step 3 crosses 50% of its peak amplitude.
11. Sample the linear fit pulse, p, at bit period-spaced intervals relative to the time $t_0 = t_x + 0.5$ UI, interpolating as necessary to yield the sampled pulse p_i.
12. Equalize the sampled pulse, p_i, using the coefficient vector, w, computed in step 6 per 5.2.3.12 to yield the equalized pulse q_i.

The RMS value of the error between the linear fit and measured waveform from step 9, σ_e, shall be equal or less than $0.027 \cdot v_f$.

The coefficient $c(-1)$, if present, is the value of q_i at time $t_0 + (D_w - 1)$ UI.

The coefficient $c(0)$ is the value of q_i at time $t_0 + D_w$ UI.

The coefficient $c(+1)$ is the value of q_i at time $t_0 + (D_w + 1)$ UI.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.7 **Transmitter output waveform (cont'd)**

Valid combinations of the coefficients $c(-1)$, $c(0)$, and $c(+1)$ shall meet the following conditions:

a) $v_f \cdot \sum_{i=-1}^{+1} |c(i)|$, the peak-to-peak output voltage shall not exceed 1200 mV.

b) $v_f \cdot \left|\sum_{i=-1}^{+1} c(i)\right|$, the steady-state output voltage with pre- and post-cursor compensation applied, shall not be below 400 mV.

5.2.3.8 Transmitter output noise and distortion

IEEE Std. 802.3-2018 93.8.1.6. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Signal-to-noise-and-distortion ratio (SNDR) measured at the transmitter output using the method described shall be greater than 27 dB regardless of the transmit equalizer setting. All transmitters on all lanes shall be enabled and transmitting the same pattern with identical transmit equalizer settings.

Given a configuration of the transmit equalizer, capture at least one complete cycle of the test pattern PRBS9 (see Annex H) at TP0a per 5.2.3.9. Compute the linear fit pulse response $p(k)$ and the linear fit error waveform $e(k)$ from the captured waveform per 5.2.3.11 using $N_p = 14$ and $D_p = 2$. Denote the standard deviation of $e(k)$ as σ_e.

Given the same configuration of the transmit equalizer, measure the RMS deviation from the mean voltage at a fixed point in a run of at least 8 consecutive identical bits in a suitable pattern. PRBS9 is an example of a pattern that includes runs suitable to perform the measurement. It is recommended that the deviation is measured within the flattest portion of the waveform at a point where the slope is closest to zero. The RMS deviation is measured for a run of zeros and also a run of ones. The average of the two measurements is denoted as σ_n.

SNDR is defined by (7) where p_{max} is the maximum value of $p(k)$.

$$SNDR = 10 \cdot \log_{10}\left(\frac{p_{max}^2}{\sigma_e^2 + \sigma_n^2}\right) \text{ [dB]} \quad (7)$$

5.2.3.9 Waveform acquisition

IEEE Std. 802.3-2018 85.8.3.3.4. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The transmitter under test repetitively transmits the specified test pattern (see 5.2.3.10). The waveform shall be captured with an effective sample rate that is M times the signaling rate of the transmitter under test. The value of M shall be 32. Averaging multiple waveform captures is recommended.

The captured waveform shall represent an integer number of repetitions of the test pattern totaling N bits. Hence the length of the captured waveform should be $M \cdot N$ samples. The waveform should be aligned such that the first M samples of waveform correspond to the first bit of the test pattern, the second M samples to the second bit, and so on.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.10 Test pattern

To avoid correlated crosstalk, it is highly recommended that the PRBS9 patterns generated on each lane be generated from independent, random seeds or a minimum offset of 20 UI between the PRBS9 sequence on any lane and any other lane. The PRBS9 pattern shall be generated as specified in Annex H.

5.2.3.11 Linear fit to the waveform measured at TP0a

IEEE Std. 802.3-2018 85.8.3.3.5. Reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Given the captured waveform $y(k)$ and corresponding aligned bits $x(n)$ derived from the procedure defined in 5.2.3.9, define the M-by-N waveform matrix $\boldsymbol{Y}$ as shown in (8).

$$\boldsymbol{Y} = \begin{bmatrix} y(1) & y(M+1) & \cdots & y(M\cdot(N-1)+1) \\ y(2) & y(M+2) & \cdots & y(M\cdot(N-1)+2) \\ \vdots & \vdots & \cdots & \vdots \\ y(M) & y(2\cdot M) & \cdots & y(M\cdot N) \end{bmatrix} \tag{8}$$

Rotate the bits vector x by the specified pulse delay D_p to yield x_r as shown in (9).

$$x_r = \begin{bmatrix} x(D_p+1) & x(D_p+2) & \cdots & x(N) & x(1) & \cdots & x(D_p) \end{bmatrix} \tag{9}$$

Define the matrix $\boldsymbol{X}$ to be an N-by-N matrix derived from x_r as shown in (10).

$$\boldsymbol{X} = \begin{bmatrix} x_r(1) & x_r(2) & \cdots & x_r(N) \\ x_r(N) & x_r(1) & \cdots & x_r(N-1) \\ \vdots & \vdots & \cdots & \vdots \\ x_r(2) & x_r(3) & \cdots & x_r(1) \end{bmatrix} \tag{10}$$

Define the matrix $\boldsymbol{X}_1$ to be the first N_p rows of $\boldsymbol{X}$ concatenated with a row of 1's of length N. The M-by-(N_p+1) coefficient matrix, $\boldsymbol{P}$, corresponding to the linear fit is then defined by (11).

$$\boldsymbol{P} = \boldsymbol{Y}\cdot\boldsymbol{X}_1^T\cdot\left(\boldsymbol{X}_1\cdot\boldsymbol{X}_1^T\right)^{-1} \tag{11}$$

In (11) the superscript "T" denotes the matrix transpose operator. The error waveform $e(k)$, in then read column-wise from the elements of E.

$$\boldsymbol{E} = \boldsymbol{P}\cdot\boldsymbol{X}_1 - \boldsymbol{Y} = \begin{bmatrix} e(1) & e(M+1) & \cdots & e(M\cdot(N-1)+1) \\ e(2) & e(M+2) & \cdots & e(M\cdot(N-1)+2) \\ \vdots & \vdots & \cdots & \vdots \\ e(M) & e(2\cdot M) & \cdots & e(M\cdot N) \end{bmatrix} \tag{12}$$

Define $\boldsymbol{P}_1$ to be a matrix consisting of the first N_p columns of the matrix $\boldsymbol{P}$ as shown in (13).

$$\boldsymbol{P}_1 = \begin{bmatrix} p(1) & p(M+1) & \cdots & p\left(M\cdot\left(N_p-1\right)+1\right) \\ p(2) & p(M+2) & \cdots & p\left(M\cdot\left(N_p-1\right)+2\right) \\ \vdots & \vdots & \cdots & \vdots \\ p(M) & p(2\cdot M) & \cdots & p\left(M\cdot N_p\right) \end{bmatrix} \tag{13}$$

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.11 **Linear fit to the waveform measured at TP0a (cont'd)**

The linear fit pulse response, $p(k)$, is then read column-wise from the elements of $\boldsymbol{P}_1$.

5.2.3.12 Removal of the transfer function between the transmit function and TP0a

IEEE Std. 802.3-2018 85.8.3.3.6. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Rotated sampled pulse response p_i by the specified equalizer delay D_w to yield p_r as shown in (14).

$$p_r = \begin{bmatrix} p_i(D_w + 1) & p_i(D_w + 2) & \cdots & p_i(N_p) & p_i(1) & \cdots & p_i(D_w) \end{bmatrix} \quad (14)$$

Define the matrix $\boldsymbol{P}_2$ to be a N_p-by-N_p matrix delivered from p_r as shown in (15).

$$\boldsymbol{P}_2 = \begin{bmatrix} p_r(1) & p_r(N_p) & \cdots & p_r(2) \\ p_r(2) & p_r(1) & \cdots & p_r(3) \\ \vdots & \vdots & \cdots & \vdots \\ p_r(N_p) & p_r(N_p - 1) & \cdots & p_r(1) \end{bmatrix} \quad (15)$$

Define the matrix $\boldsymbol{P}_3$ to be the first N_w columns of $\boldsymbol{P}_2$. Define a unit pulse column vector x_p of length N_p. The value of element $x_p(D_p + 1)$ is 1 and all other elements have a value of 0. The vector of filter coefficients w that equalizer p_i is then defined by (16).

$$w = \left(\boldsymbol{P}_3^T \cdot \boldsymbol{P}_3\right)^{-1} \cdot \boldsymbol{P}_3^T \cdot x_p \quad (16)$$

Given the column vector of equalizer coefficients, w, the equalized pulse response q_i is determined by (17).

$$q_i = \boldsymbol{P}_3 \cdot w \quad (17)$$

5.2.3.13 Transmitter output jitter

5.2.3.13.1 Overview

IEEE Std. 802.3-2018 92.8.3.8. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Three components of the transmitter output jitter are specified: even-odd jitter (DCD), effective bounded uncorrelated jitter, and effective total uncorrelated jitter. A high-resolution oscilloscope, time interval analyzer, or other instrument with equivalent capability may be used to measure jitter.

The effect of a single-pole high-pass filter with a 3 dB frequency of $f_b/1667$ is applied to the jitter. The voltage threshold for the measurement of BER or crossing times is the mid-point (0 V) of the AC-coupled differential signal.

Jitter measurements are performed with transmitters on all lanes enabled and transmitting the same pattern with identical transmit equalizer settings.

weiqiaqian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.13.2 Even-odd jitter (DCD)

IEEE Std. 802.3-2018 92.8.3.8.1. Reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Even-odd jitter is measured using two repetitions of a PRBS9 pattern (see Annex H). The deviation of the time of each transition from an ideal clock at the signaling rate is measured. Even-odd jitter is defined as the magnitude of the difference between the average deviation of all even-numbered transitions and the average deviation of all odd-numbered transitions, where determining if a transition is even or odd is based on possible transitions but only actual transitions are measured and averaged.

Even-odd jitter shall be less than or equal to 0.035 UI regardless of the transmit equalization setting.

5.2.3.13.3 Effective bounded uncorrelated jitter (EBUJ) and effective random jitter (ERJ)

IEEE Std. 802.3-2018 92.8.3.8.2. Reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Effective bounded uncorrelated jitter (EBUJ) and effective random jitter (ERJ) are measured on each of two specific transitions in a PRBS9 pattern (see Annex H). The two transitions occur in the sequence of five zeros and four ones and nine ones and five zeros, respectively. The sequences are located at bits 10 to 18 and 1 to 14, respectively, where bits 1 to 9 are the run of nine ones.

a) The jitter components are determined according to the following method. Acquire a horizontal histogram of a transition around the zero-crossing point. The number of acquired samples should be sufficiently large to yield consistent measurement results. Designate the total number of samples as NS, the number of bins as NB, the number of samples in each bin as N_i where i is the bin number from 1 to NB, and the sample time corresponding with the center of each bin as t_i.

b) Create two cumulative distribution curves $CDFL_i$ and $CDFR_i$ according to (18) and (19), and two corresponding curves QR_i and QL_i according to (20) and (21), where $erfc^{-1}(x)$ is the inverse of the complementary error function $erfc(x)$ defined by (22).

c) Determine the parameters m_{left} and b_{left} of (23) that best fit QL_i as a function of t_i for bins with $CDFL_i$ in the range of 10^{-6} to $2.5 \cdot 10^{-4}$. Similarly determine the parameters of m_{right} and b_{right} of that best fit QR_i as a function of t_i for bins with $CDFR_i$ in the range of 10^{-6} to 10^{-4}.

d) Calculate the values of effective bounded uncorrelated jitter and effective total uncorrelated jitter according to (25) and (26), respectively. The peak-to-peak contribution of the effective random jitter in the effective total uncorrelated jitter is related to BER $= 10^{-15}$.

$$CDFL_i = \sum_{k=1}^{i} \frac{N_k}{NS} \tag{18}$$

$$CDFR_i = \sum_{k=i}^{NB} \frac{N_k}{NS} \tag{19}$$

$$QL_i = \sqrt{2} \cdot erfc^{-1}(2 \cdot CDFL_i) \tag{20}$$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.3.13.3 **Effective bounded uncorrelated jitter (EBUJ) and effective random jitter (ERJ) (cont'd)**

$$QR_i = \sqrt{2} \cdot erfc^{-1}(2 \cdot CDFR_i) \tag{21}$$

$$erfc(x) = \frac{2}{\sqrt{\pi}} \cdot \int_x^{\infty} e^{-t^2} \cdot dt \tag{22}$$

$$Q_{left} = m_{left} \cdot t + b_{left} \tag{23}$$

$$Q_{right} = m_{right} \cdot t + b_{right} \tag{24}$$

$$EBUJ = \frac{b_{left}}{m_{left}} - \frac{b_{right}}{m_{right}} \tag{25}$$

$$ERJ = \frac{m_{left} - m_{right}}{2 \cdot m_{right} \cdot m_{left}} \tag{26}$$

$$TUJ = 15.9 \cdot ERJ + EBUJ \tag{27}$$

Effective bounded uncorrelated jitter (EBUJ) shall be less than or equal to 0.15 UI peak-to-peak regardless of the transmit equalization setting. The effective total uncorrelated jitter (TUJ) shall be less than or equal to 0.26 UI peak-to-peak regardless of the transmit equalization setting.

5.2.4 Receiver characteristics

Receiver electrical characteristics are specified at TP5a. The receiver shall meet electrical specification in 5.2.4.1. Meet the return loss specified in 5.2.4.4 using the text fixture specified in 5.2.4.2. In addition, the requirements in 5.2.4.3 apply. The receiver shall meet the interference tolerance specification in 5.2.4.5 and jitter tolerance specification in 5.2.4.6.

5.2.4.1 Receiver electrical specifications

IEEE Std. 802.3-2018 93.8.2. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Unless otherwise noted, the receiver electrical characteristics shall be measured at test point TP5a (Figure 29) and shall meet the specifications in Table 23.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.4.1 **Receiver electrical specifications (cont'd)**

Table 23 — Category C receiver electrical specification

Characteristic	Symbol	Ref.	Min.	Max.	Unit
Data rate	f_b	5.2.4.3	6.375	32.45	Gbps
Operable input voltage level			400	1200	p-p mV
Differential input return loss	RL_{DD}	5.2.4.4	See (30)		dB
Input common-mode voltage	v_{cm}		0	1,900	mV
Jitter tolerance		5.2.4.6			
Sinusoidal					
Low frequency (20 kHz)	SJ_{LF}			5.00	p-p UI
High frequency (20 MHz)	SJ_{HF}			0.05	p-p UI
Bounded high-probability uncorrelated	UBHPJ			0.25	p-p UI
Bounded high-probability correlated	CBHPJ			0.20	p-p UI
Total [a,b]	TJ			0.70	p-p UI

NOTE IEEE Std. 802.3-2018 Table 93-5. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

[a] After data measured at TP5a is processed through a COM reference equalizer. The optimal transmitter FFE and reference equalizer CTLE and DFE settings, as applicable for the device class, shall be used. The optimal values shall be determined by a COM simulation of the reference system.

[b] Includes high-frequency sinusoidal jitter; the Gaussian jitter (GJ) portion is defined with respect to a BER of 10^{-15}. Compliance may be verified at a BER of 10^{-12} by applying adjustments to the Gaussian Jitter (GJ) portion of the total jitter to compensate for the lower test populations (consult OIF-CEI-04.0 Clause 2 [OIF-CEI-04.0, Common Electrical I/O (CEI) - Electrical and Jitter Interoperability agreements for 6G+ bps and 11G+ bps and 25G+ bps I/O, Optical Internetworking Forum, February 18,]).

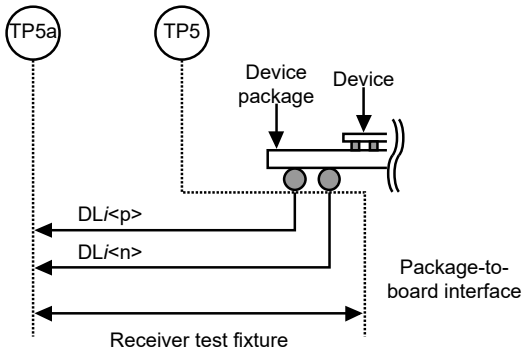
A category C receiver shall have the ability to receive a PRBS9 or PRBS15 pattern (see Annex H) and report bit errors, if any, over a given observation period. As with category C transmitters, it is recommended that a receiver should also have the ability to receive other test patterns: PRBS7, SSPS-64 (see OIF-CEI-04.0 2.D.4 [OIF-CEI-04.0, Common Electrical I/O (CEI) - Electrical and Jitter Interoperability agreements for 6G+ bps and 11G+ bps and 25G+ bps I/O, Optical Internetworking Forum, February 18,]), programmable test patterns, etc. A category C receiver shall also have the ability to measure the horizontal (time) data eye opening at the input to the comparator that determines whether each received bit is a logic one or logic zero (typically called "the slicer"). There shall also be a methodology to evaluate whether the measured eye opening is wide enough to support the required 10^{-12} bit error rate at the operating data rate. The specific implementation of this methodology is not mandated, but may be done as a simple comparison with a provided reference table, a flag bit that can be read via the control interface (see Subclause 4.4), etc. The ability to measure the vertical data eye opening (volts) at the slicer input and at the bit sampling time, as well as the ability to measure the horizontal and/or vertical data eye opening at other relevant points within the receiver (input pins, output of the continuous-time linear equalizer, etc.) is advantageous and suggested but not mandatory. The control interface shall be used to enable the PRBS pattern error detection and to make the receiver measure the data eye opening, output the result(s), etc.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.4.2 Receiver test fixture

IEEE Std. 802.3-2018 93.8.2.1. Adapted and reprinted with permission from IEEE.
© 2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Unless otherwise noted, measurements of the receiver are made at the input to a test fixture (TP5a) as shown in Figure 29.



NOTE IEEE Std. 802.3-2018 Figure 93-10. Reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Figure 29 — Receiver test fixture and test points

The differential insertion loss of the test fixture (in dB) at half of the receiver maximum data, $\frac{f_b}{2}$, shall not deviate by more than 15% from the reference insertion loss of the test fixture (in dB, see (28)). Additionally, the magnitude of the differential insertion loss deviation of the test fixture shall be less than or equal to 0.1 dB from 0.05 GHz to half of the receiver maximum data rate, $\frac{f_b}{2}$, (see Annex F.2).

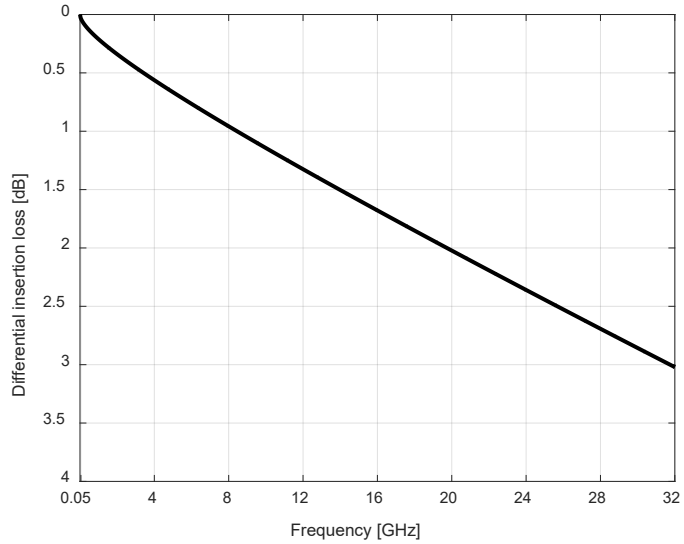
The reference insertion loss of the test fixture is defined by (28).

$$IL_{RXF_{ref}}(f) = -0.0015 + 0.144 \cdot \sqrt{f} + 0.069 \cdot f \quad 0.05 \text{ GHz} \le f \le f_b \quad \text{[dB]} \tag{28}$$

where f is the frequency in GHz. The effects of differences between the insertion loss of an actual test fixture and the reference insertion loss are to be accounted for in the measurements. The reference insertion loss is illustrated in Figure 30.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.4.2 **Receiver test fixture (cont'd)**



NOTE IEEE Std. 802.3-2018 Figure 93-3. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Figure 30 — Receiver test fixture reference differential insertion loss with f_b = 32 Gbps

The differential return loss of the test fixture, in dB, shall meet (29) where f_b is the receiver data rate in GHz. The return loss limit is illustrated by Figure 31.

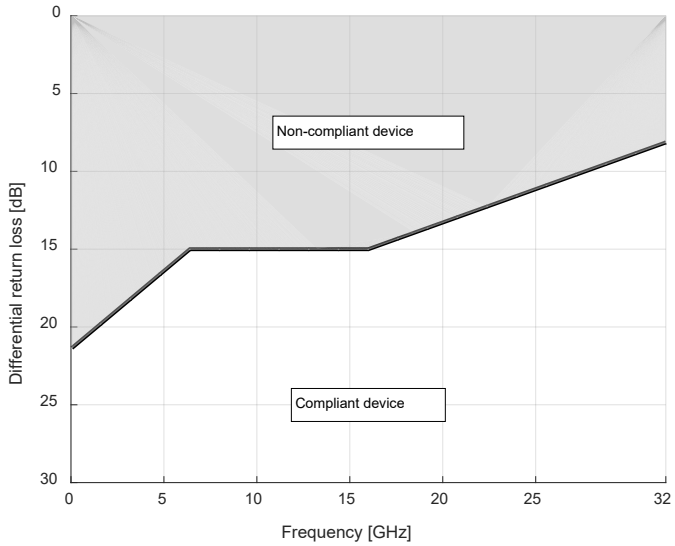
$$RL_{RXF_{ref}}(f, f_b) \geq \begin{cases} 20 - 25 \cdot \frac{f}{f_b} & 0.05 \leq f \leq 0.2 \cdot f_b \\ 15 & 0.2 \cdot f_b < f \leq 0.52 \cdot f_b \\ 20.57 - 10.715 \cdot \frac{f}{f_b} & 0.52 \cdot f_b < f \leq f_b \end{cases} \text{ [dB]} \quad (29)$$

The common-mode return loss of the test fixture shall be greater than or equal to 10 dB from 0.05 GHz to half of the maximum receiver data rate, $\frac{f_b}{2}$.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.4.2 **Receiver test fixture (cont'd)**



NOTE IEEE Std. 802.3-2018 Figure 93-4. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

Figure 31 — Receiver test fixture differential return loss limit with f_b = 32 Gbps

weifaifadian

5.2.4.3 Signaling rate and range

IEEE Std. 802.3-2018 93.8.1.2. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The category C receiver signaling range shall be between 6.375 Gbps±100 ppm and 32.45 Gbps±100 ppm; devices are allowed to operate in a subset of this range.

5.2.4.4 Receiver input return loss

IEEE Std. 802.3-2018 93.8.2.2. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The differential input return loss of the receiver shall meet (30), where f_b is the data rate. This input impedance requirement applies to all valid input levels. The reference impedance for differential return loss measurements shall be 100 Ω.

$$RL_{DD}(f) \geq \begin{cases} 12.05\text{-}f & 0.05 \leq f \leq 6 \\ 6.5 - 0.075 \cdot f & 6 < f \leq 0.75 \cdot f_b \end{cases} \text{ [dB]} \qquad (30)$$

where f is the frequency in GHz and f_b is the maximum data rate supported by the device in GHz. Note that (30) is compatible with IEEE Std. 802.3-2018 Equation (93-3). The differential input return loss limit is illustrated by Figure 32.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.4.4 **Receiver input return loss (cont'd)**

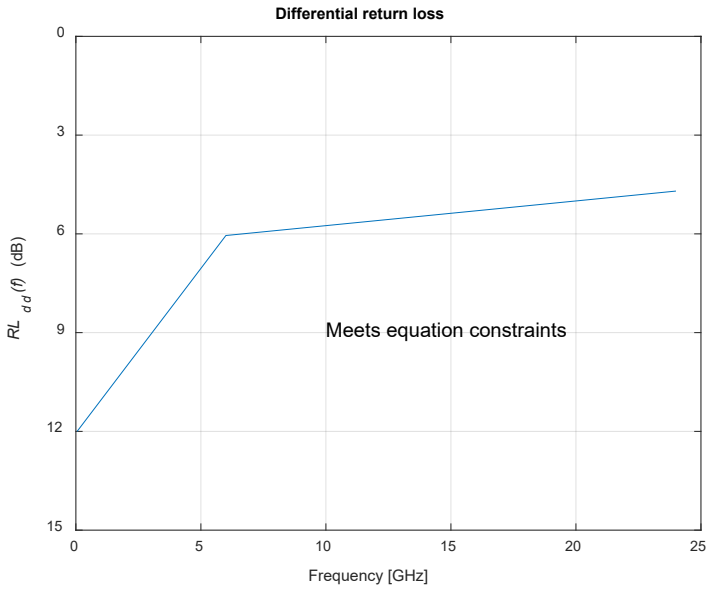


Figure 32 — Receiver differential return loss with f_b = 32 Gbps

The differential-to-common-mode return loss of the receiver shall meet (31), where f_b is the maximum device data rate. This input impedance requirement applies to all valid output levels. The reference impedance for differential return loss measurements shall be 100 Ω.

$$RL_{CD}(f) \geq \begin{cases} 25 - 37.125 \cdot \frac{f}{f_b} & 0.05 \leq f \leq 0.270 \cdot f_b \\ 15 & 0.270 \cdot f_b \leq f \leq 0.737 \cdot f_b \end{cases} \text{ [dB]} \qquad (31)$$

where f is the frequency in GHz and f_b is the maximum data rate supported by the device in GHz.

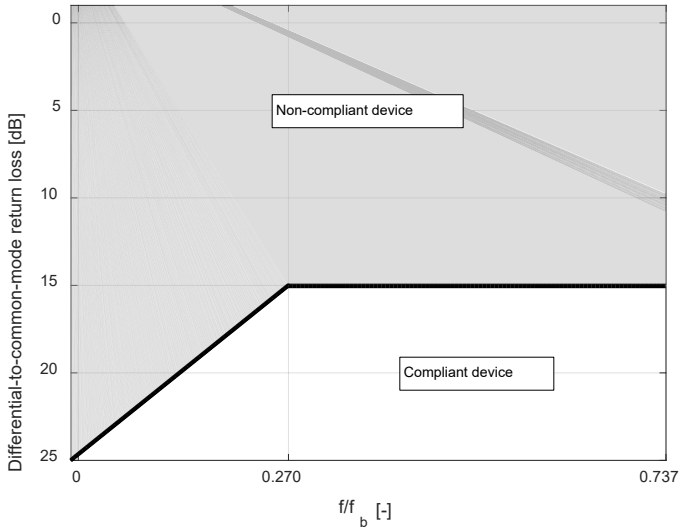


Figure 33 — Receiver differential-to-common-mode return loss limit

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.4.5 Receiver interference tolerance

IEEE Std. 802.3-2018 93.8.2.3 and Annex 93C. Adapted and reprinted with permission from IEEE. ©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The receiver shall satisfy the requirements for interference tolerance defined in Table 24 for Class-CS, Table 25 for Class-CM and Table 26 for Class-CR. The interference tolerance test uses the method described in IEEE Std. 802.3 Annex 93C as specified by IEEE Std. 802.3 93.8.2.3, with the following exceptions:

e) The test requirements in this subclause, given in Table 24, Table 25 and Table 26 for each class respectively, overrides the test requirements given in 93.8.2.3. For Class-CM and Class-CR the test requirements for RSS-DFE value are provided in Table 25 and Table 26 respectively. The equation to calculate the RSS-DFE value is described in (93A-50).

f) The test transmitter meets the specifications in 5.2.3 as measured at TP0a (see Figure 22). The test transmitter is constrained such that for any transmit equalizer setting the differential peak-to-peak voltage (see 5.2.3.4) is less than or equal to 800 mV and the pre- and post-cursor equalization ratios (see 93.8.1.5.5) are less than or equal to 4.6. The lowest frequency for constraints on the noise spectral density is 1 GHz. The return loss of the test setup in Figure 93C–4 measured at TP5 replica meets the requirements of Equation (**2**).

g) COM is calculated using both Test 1 and Test 2 device package model transmission line lengths listed in Table 24, Table 25 and Table 26 for each class respectively on the receiver side. The value of COM is taken as the lower of the two calculated values.

h) The transmitter device package model $S(tp)$ is omitted from Equation (93A–3) in the calculation of COM. The filtered voltage transfer function $H(k)(f)$ calculated in Equation (93A–19) uses the filter $H_t(f)$ defined by Equation (93A–46), where T_r is the measured 20% to 80% transition time of the signal at TP0. T_r might be obtained by de-embedding the signal from TP0a to TP0.

i) The additive broadband noise characteristics is defined in 5.2.4.5.1 and overrides values given in Annex 93C.1.

It is recommended to adjust the test transmitter jitter such that the effective bounded uncorrelated jitter and the effective total uncorrelated jitter are as close as practical to their limits in Table 21 Table 21

All combinations of package length are tested (see Table 27 for Class-CS, Table 28 for Class-CM and Table 29 for Class-CR). The insertion loss at $f_b/2$ is defined in (32) for Class-CS, Equation (33) for Class-CM and Equation (34) for class-CR. The data rate in the interference test is stepped from minimum data rate for Category C, 6.375 Gbps, to the receiver maximum working data rate with a maximum data rate step of 8.5 Gbps.

Class-CS $$IL_{S\frac{f_b}{2}} = 12.5 \quad \text{(dB)} \tag{32}$$

Class-CM $$IL_{M\frac{f_b}{2}} = 22.5 \quad \text{(dB)} \tag{33}$$

Class-CR $$IL_{R\frac{f_b}{2}} = 27 \quad \text{(dB)} \tag{34}$$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.4.5 **Receiver interference tolerance (cont'd)**

Table 24 — Receiver interference tolerance parameters, Class-CS

Parameter	Test 1 values		Test 2 values		Units
	Min	Max	Min	Max	
Symbol error ratio		10^{-15}		10^{-15}	-
Insertion loss at $f_b/2$[a]	5.75	6.75	12	13	dB
Coefficients of fitted insertion loss[b]					
a_o	-1	1.200	-1	1.200	dB
a_1	0	5.684	0	5.684	dB
a_2	0	27.319	0	27.319	dB
a_3	0	12.265	0	12.265	dB
COM, including effects of broadband noise	2		2		dB

[a]Measured between TPt and TP5 (see IEEE Std. 802.3 Figure 93C-4).
[b]Coefficients are calculated from the insertion loss measured between TPt and TP5 (see IEEE Std. 802.3 Figure 93C-4) using the method in Annex F with f_{min} = 0.05 GHz, up to maximum data rate of the RX, and maximum Δf = 0.01 GHz.

Table 25 — Receiver interference tolerance parameters, Class-CM

Parameter	Test 1 values		Test 2 values		Units
	Min	Max	Min	Max	
Symbol error ratio		10^{-15}		10^{-15}	-
Insertion loss at $f_b/2$[a]	10.75	11.75	22	23	dB
Coefficients of fitted insertion loss[b]					
a_o	-1	2.000	-1	2.000	dB
a_1	0	17.513	0	17.513	dB
a_2	0	48.997	0	48.997	dB
a_3	0	23.897	0	23.897	dB
RSS-DFE2	0.05		0.05		-
COM, including effects of broadband noise	2		2		dB

[a]Measured between TPt and TP5 (see IEEE Std. 802.3 Figure 93C-4).
[b]Coefficients are calculated from the insertion loss measured between TPt and TP5 (see IEEE Std. 802.3 Figure 93C-4) using the method in Annex F with f_{min} = 0.05 GHz, up to maximum data rate of the RX, and maximum Δf = 0.01 GHz.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.4.5 **Receiver interference tolerance (cont'd)**

Table 26 — Receiver interference tolerance parameters, Class-CR

Parameter	Test 1 values		Test 2 values		Units
	Min	Max	Min	Max	
Symbol error ratio		10^{-15}		10^{-15}	-
Insertion loss at $f_b/2$[a]	13	14	26.5	27.5	dB
Coefficients of fitted insertion loss[b]					
a_o	-1	2.000	-1	2.000	dB
a_1	0	21.590	0	21.590	dB
a_2	0	54.034	0	54.034	dB
a_3	0	26.600	0	26.600	dB
RSS-DFE4	0.05		0.05		-
COM, including effects of broadband noise	2		2		dB

[a]Measured between TPt and TP5 (see IEEE Std. 802.3 Figure 93C-4).
[b]Coefficients are calculated from the insertion loss measured between TPt and TP5 (see IEEE Std. 802.3 Figure 93C-4) using the method in Annex F with f_{min} = 0.05 GHz, up to maximum data rate of the RX, and maximum Δf = 0.01 GHz.

weijiafatian

5.2.4.5.1 Additive broadband noise characteristics

IEEE Std. 802.3-2018 93C.1. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The broadband noise used for receiver calibration is Gaussian with a crest factor of at least 5. The noise spectral density, $NSD(f)$, is normalized and constrained according to the relations in (35), where f_b is the maximum data rate supported by the receiver under test and $f_{NSD1} = 1\text{GHz}$. $NSD(f)$ is in units of V^2/Hz. The average noise spectral density, $NSD_{average}$ is determined according to (36).

$$\left.\begin{aligned} 10 \cdot log_{10}\left(\frac{NSD(f)}{NSD_{average}}\right) &< 3 \\ 10 \cdot log_{10}\left(\frac{NSD(f)}{NSD_{average}}\right) &> -3 \cdot \left(1 - 1.2 \cdot \frac{f}{f_b}\right) \end{aligned}\right\} f_{NSD1} \le f \le \frac{f_b}{2} \quad (35)$$

$$NSD_{average} = \frac{\int_{f_{NSD1}}^{\frac{f_b}{2}} NSD(f) \cdot df}{\frac{f_b}{2} - f_{NSD1}} \quad (36)$$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

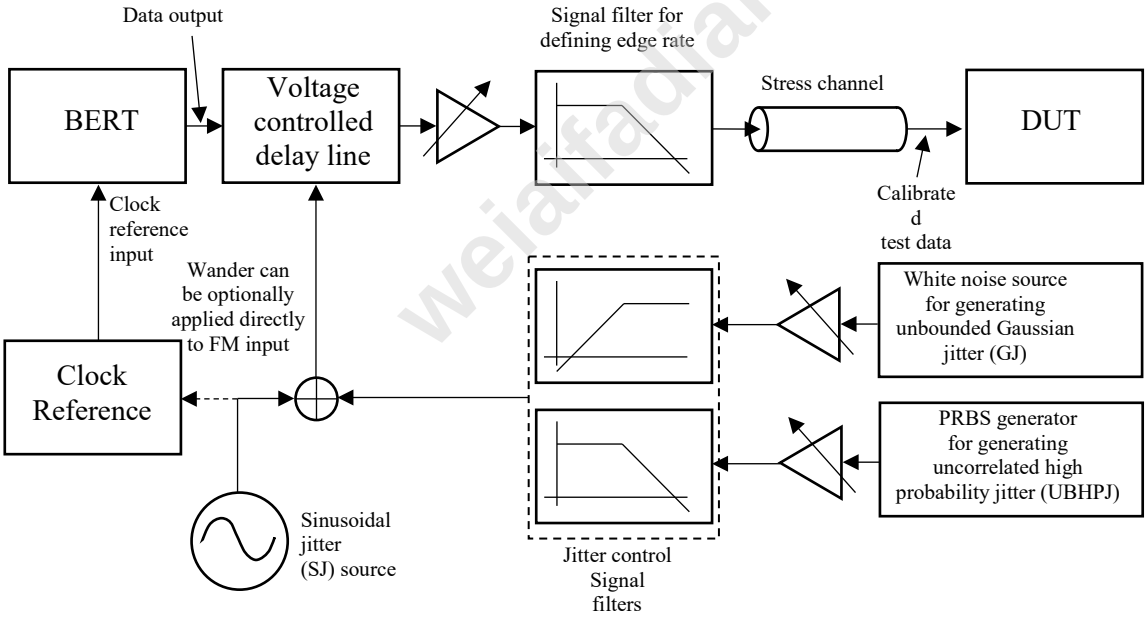
5.2.4.6 Receiver jitter tolerance

Based on OIF-CEI-04.0 2.5.4 and 2.A.1, © 2017 Optical Internetworking Forum.

Receiver jitter tolerance is measured using the test setup shown in Figure 34 or its equivalent.

The test transmitter meets the specifications of 5.2.3 as measured at TP0a (see Figure 25). The test transmitter is constrained so that its differential peak-to-peak output voltage does not exceed 800 mV at TP0a regardless of the transmitter equalizer setting. In the system represented in Figure 34 the appropriate transmitter feed-forward equalizer setting and differential output amplitude level are adjusted in the bit error rate tester ("BERT") while the appropriate transmitter transition time is adjusted via the bandwidth of the "Signal filter for defining edge rate." Different BERTs may have any or all of these capabilities built-in.

The test channel meets the requirements of the interference tolerance test channel using Test 2 values (see 5.2.4.5) for respective class. Note that the values measured for EBUJ include the effects of this added sinusoidal jitter. Broadband noise is added to obtain a COM of 2 dB with these measured jitter values as for the interference tolerance test. The total wonder mask for the sinusoidal jitter is given in Figure 35. The symbol error ratio, BER, shall be less than or equal to 10^{-15}.



NOTE Based on OIF-CEI-04.0 Figure 2-33, ©2017 Optical Internetworking Forum.

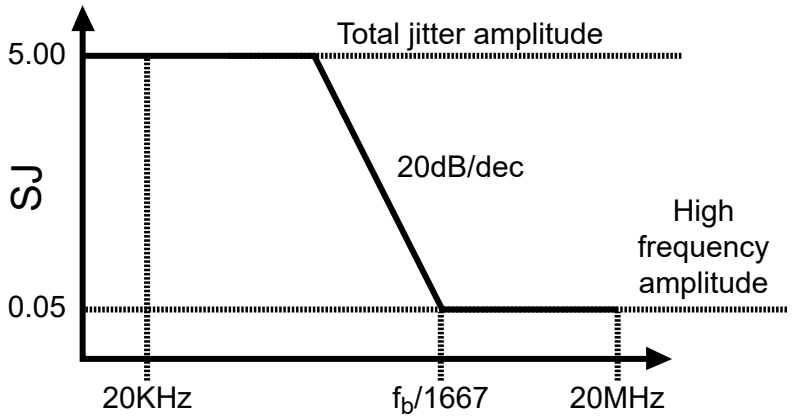
Figure 34 — Setup to test receiver jitter tolerance with no relative wander

At higher frequencies, this jitter source is used to ensure margin in the high frequency jitter tolerance of the receiver. At lower frequencies, the higher sinusoidal jitter should then be tracked by the receiver.

weijiafadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.4.6 Receiver jitter tolerance (cont'd)



NOTE Based on OIF-CEI-04.0 Figure 2-3, ©2017 Optical Internetworking Forum.

Figure 35 — Total wander mask

5.2.5 Channel characteristics

5.2.5.1 Insertion loss

The insertion loss, in dB, of the channel is recommended to meet Equation (37) for Class-CS, Equation (38) for Class-CM and Equation (39) for Class-CR.

Class-CS
$$IL(f) \le \begin{Bmatrix} 0.676 + 8.068\sqrt{\frac{f}{f_b}} + 12.248 \cdot \frac{f}{f_b} & 0.01 \le f < \frac{f_b}{2} \\ -11.149 + 47.309 \cdot \frac{f}{f_b} & \frac{f_b}{2} \le f < f_b \end{Bmatrix} \text{ (dB)} \quad (37)$$

Class-CM
$$IL(f) \le \begin{Bmatrix} 1.217 + 14.523\sqrt{\frac{f}{f_b}} + 22.049 \cdot \frac{f}{f_b} & 0.01 \le f < \frac{f_b}{2} \\ -20.068 + 85.155 \cdot \frac{f}{f_b} & \frac{f_b}{2} \le f < f_b \end{Bmatrix} \text{ (dB)} \quad (38)$$

Class-CR
$$IL(f) \le \begin{Bmatrix} 1.461 + 17.427\sqrt{\frac{f}{f_b}} + 26.455 \cdot \frac{f}{f_b} & 0.01 \le f < \frac{f_b}{2} \\ -24.082 + 102.186 \cdot \frac{f}{f_b} & \frac{f_b}{2} \le f < f_b \end{Bmatrix} \text{ (dB)} \quad (39)$$

where

f is the frequency in GHz
f_b is the signaling rate in GHz
$IL(f)$ is the insertion loss at frequency f

The insertion loss limit is illustrated by Figure 36.

weijiafadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.5.1 Insertion loss (cont'd)

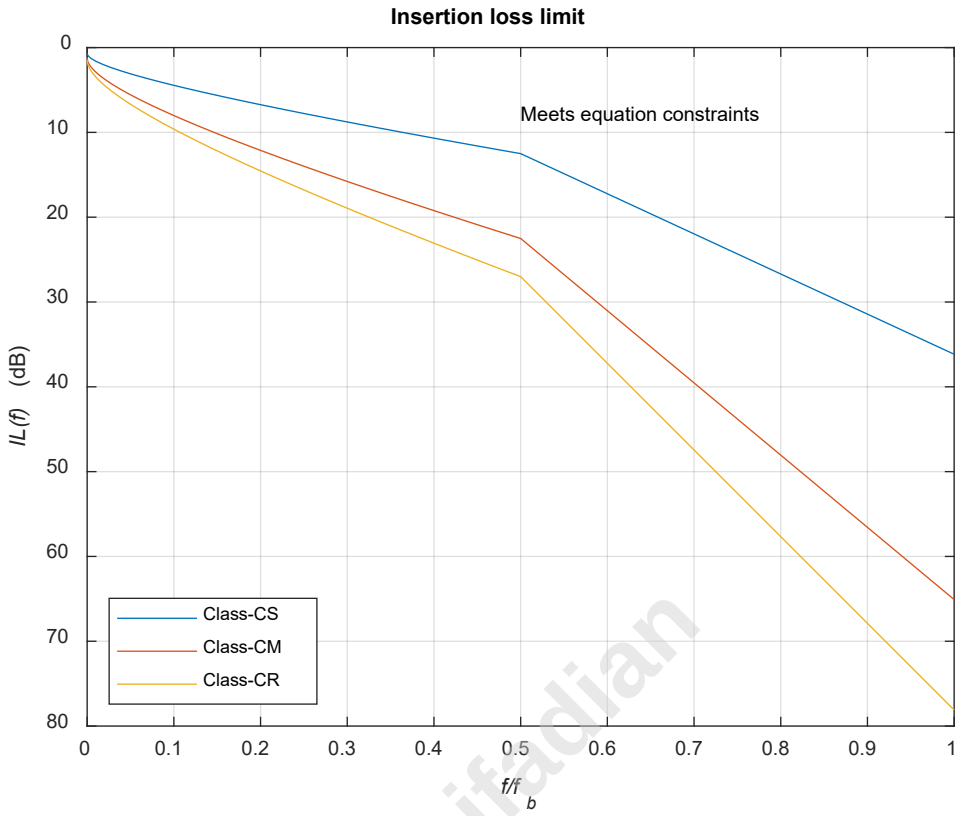


Figure 36 — Insertion loss limit

5.2.5.2 Isolation in package and footprint area

Crosstalk in the package and footprint vias degrades signal-to-noise ratio (SNR) and thus reduces link performance. Additionally, it may be impractical and/or unfeasible to remove the effects of package crosstalk via equalization techniques at the receiver. The isolation requirement is for the area in the proximity of the package or connector etc. the area where the majority of the crosstalk take place. The isolation requirement, in this subclause, will be placed on different parts of a channel, for example, at the TX-package break out, RX-package break out or a connector pin break out. The channel will be short an then the channel insertion loss will be low for each of the isolation test regions. Crosstalk between channels will lower the COM margin. With an isolation as the requirements limit given in Equations (40) and (41) below might lower the COM margin with a few tenth of dB. If the isolation is close to the requirements as given in Equations (40) and (41) in several points in a channel the isolation might be increased to not influence the COM value too much.

It is recommended that the differential far-end crosstalk power sum (*PSFEXT* (*f*)) and the differential near-end crosstalk power sum (*PSNEXT* (*f*)) shall meet or exceed the limits specified in (40) for class C-S devices, and (41) for class C-M and class C-R devices. *PSFEXT* (*f*) and *PSNEXT* (*f*) are define in (42) and (43), respectively.

$$\text{Class} - \text{CS} \qquad \begin{Bmatrix} PSNEXT(f) \geq 47\ (\text{dB}) & f \leq 0.75 \cdot f_b \\ PSFEXT(f) \geq 37\ (\text{dB}) & f \leq 0.75 \cdot f_b \end{Bmatrix} \qquad (40)$$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.5.2 **Isolation in package and footprint area (cont'd)**

$$\text{Class} - \text{CM} \text{ and } \text{Class} - \text{CR} \quad \begin{cases} PSNEXT(f) \geq 60\ (\text{dB}) & f \leq 0.178 \cdot f_b \\ PSNEXT(f) \geq 45\ \text{dB} - 20 \cdot log_{10}\left(\frac{f}{f_b}\right) & 0.178 \cdot f_b < f < 0.75 \cdot f_b \\ PSFEXT(f) \geq 50\ (\text{dB}) & f \leq 0.178 \cdot f_b \\ PSFEXT(f) \geq 35\ \text{dB} - 20 \cdot log_{10}\left(\frac{f}{f_b}\right) & 0.178 \cdot f_b < f < 0.75 \cdot f_b \end{cases} \quad (41)$$

$$PSNEXT(f) = -10 \cdot log_{10}\left(\sum_{n=1}^{NUM_{NEXT}} 10^{-\frac{NEXT_n(f)}{10}}\right) \quad (42)$$

$$PSFEXT(f) = -10 \cdot log_{10}\left(\sum_{n=1}^{NUM_{FEXT}} 10^{-\frac{FEXT_n(f)}{10}}\right) \quad (43)$$

NUM_{FEXT} is the number of FEXT aggressors in the system, NUM_{NEXT} is the number of NEXT aggressors in the system, $FEXT_n(f)$ is the differential crosstalk, in decibels, of FEXT aggressor $n \in [1, \cdots, NUM_{FEXT}]$ and $NEXT_n(f)$ is the differential crosstalk, in decibels, of NEXT aggressor $n \in [1, \cdots, NUM_{NEXT}]$.

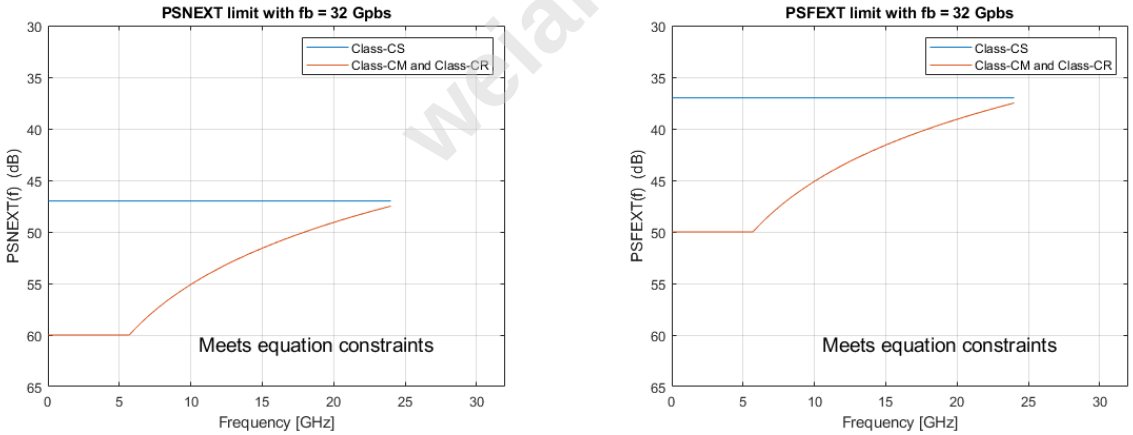


Figure 37 — *PSNEXT* and *PSFEXT* limit lines with f_b = 32 Gbps.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.5.3 Channel Operating Margin

IEEE Std. 802.3-2018 93.A.1. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

A channel shall be considered class C-S, class C-R, or class C-M-compliant at a tested data rate $f_{b_{ch}}$ and maximum transmitter output voltage $v_{d_{ref}}$ if when placed in the system described below its COM value equals or exceeds 2 dB.

The Channel Operating Margin (COM) is computed using the procedure in IEEE Std. 802.3 93A.1 with the Test 1 and Test 2 values in Table 27 for Class-CS, Table 28 for Class-CM and Table 29 for Class-CR. Test 1 and Test 2 differ in the value of the device package model transmission line length z_p.

COM shall be greater than or equal to 2 dB for each test. This minimum value allocates margin for practical limitations on the receiver implementation as well as the largest step size allowed for transmitter equalizer coefficients.

Table 27 — COM parameters for class C-S

Parameter	Symbol	Value	Units
Signal rate	f_b	6.375 to 32.45	GBd
Maximum start frequency	$f_{\min}$	0.05	GHz
Maximum frequency step	Δf	0.01	GHz
Device package model Single-ended device capacitance Transmission line length, Test 1 Transmission line length, Test 2 Single-ended package capacitance at package-to-board interface Package transmission line nominal characteristic impedance	 C_d z_p z_p C_p Z_c	 1.8e-4 12 30 1.1e-4 95	 nF mm mm nF Ω
Single-ended reference resistance	R_0	50	Ω
Single-ended termination resistance	R_d	55	Ω
Receiver 3 dB bandwidth	f_r	$0.75 \cdot f_b$	GHz
Transmitter equalizer, minimum cursor coefficient	$c(0)$	0.5	—
Transmitter equalizer, pre-cursor coefficient Minimum value Maximum value Step size	$c(-1)$	 -0.333 0 0.08325	 — — —
Transmitter equalizer, post-cursor coefficient Minimum value Maximum value Step size	$c(1)$	 -0.333 0 0.08325	 — — —
Continuous time filter, DC gain Minimum value Maximum value Step size	g_{DC}	 -6 0 3	 dB dB dB

welafradjam

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Continuous time filter, zero frequency for $g_{DC} = 0, f_Z$ Minimum value Maximum value Step size	 Setting 1 Setting 2 Setting 3	 $16 \cdot f_b$ $0.2825 \cdot f_b$ $0.2295 \cdot f_b$	GHz
Continuous time filter, pole frequencies, f_{p1} Setting 1 is with g_{DC} = 0 dB Setting 2 is with g_{DC} = -3 dB Setting 3 is with g_{DC} = -6 dB	 Setting 1 Setting 2 Setting 3	 $1.3 \cdot f_b$ $0.3594 \cdot f_b$ $0.29 \cdot f_b$	GHz
Continuous time filter, pole frequencies, f_{p2} Setting 1 is with g_{DC} = 0 dB Setting 2 is with g_{DC} = -3 dB Setting 3 is with g_{DC} = -6 dB	 Setting 1 Setting 2 Setting 3	 $16 \cdot f_b$ f_b f_b	GHz
Transmitter differential peak output voltage Victim Far-end aggressor Near-end aggressor	 A_v A_{fe} A_{ne}	 0.4 0.4 0.6	 V V V
Number of signal levels	L	2	—
Level separation mismatch ratio	R_{LM}	1	—
Transmitter signal-to-noise ratio	SNR_{TX}	27	dB
Number of samples per unit interval	M	32	—
Decision feedback equalizer (DFE) length	N_b	0	UI
Normalized DFE coefficient magnitude limit, for n = 1 to N_b	$b_{max}(n)$	—	—
Random jitter, RMS	σ_{RJ}	0.01	UI
Dual-Dirac jitter, peak	A_{DD}	0.05	UI
One-sided noise spectral density	η_0	$5.2 \cdot 10^{-8}$	V^2/GHz
Target detector error ratio	DER_0	10^{-15}	—
COM Pass threshold	COM	2	dB

weinfadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.5.4 Channel Operating Margin (cont'd)

Table 28 — COM parameters for class C-M

Parameter	Symbol	Value	Units
Signal rate	f_b	6.375 to 32.45	GBd
Maximum start frequency	f_{min}	0.05	GHz
Maximum frequency step	Δf	0.01	GHz
Device package model Single-ended device capacitance Transmission line length, Test 1 Transmission line length, Test 2 Single-ended package capacitance at package-to-board interface Package transmission line nominal characteristic impedance	 C_d z_p z_p C_p Z_c	 1.8e-412 30 1.1e-4 95	 nF mm mm nF Ω
Single-ended reference resistance	R_0	50	Ω
Single-ended termination resistance	R_d	55	Ω
Receiver 3 dB bandwidth	f_r	$0.75 \cdot f_b$	GHz
Transmitter equalizer, minimum cursor coefficient	$c(0)$	0.5	—
Transmitter equalizer, pre-cursor coefficient Minimum value Maximum value Step size	$c(-1)$	 -0.333 0 0.08325	 — — —
Transmitter equalizer, post-cursor coefficient Minimum value Maximum value Step size	$c(1)$	 -0.333 0 0.08325	 — — —
Continuous time filter, DC gain Minimum value Maximum value Step size	g_{DC}	 -9 0 3	 dB dB dB
Continuous time filter, zero frequency for $g_{DC} = 0$ f_Z Setting 1 is with g_{DC} = 0 dB Setting 2 is with g_{DC} = -3 dB Setting 3 is with g_{DC} = -6 dB Setting 4 is with g_{DC} = -9 dB	 Setting 1 Setting 2 Setting 3 Setting 4	 $16 \cdot f_b$ $0.2825 \cdot f_b$ $0.2295 \cdot f_b$ $0.2114 \cdot f_b$	 GHz
Continuous time filter, pole frequencies, f_{p1} Setting 1 is with g_{DC} = 0 dB Setting 2 is with g_{DC} = -3 dB Setting 3 is with g_{DC} = -6 dB Setting 4 is with g_{DC} = -9 dB	 Setting 1 Setting 2 Setting 3 Setting 4	 $1.3 \cdot f_b$ $0.3594 \cdot f_b$ $0.29 \cdot f_b$ $0.2656 \cdot f_b$	 GHz
Continuous time filter, pole frequencies, f_{p2} Setting 1 is with g_{DC} = 0 dB Setting 2 is with g_{DC} = -3 dB Setting 3 is with g_{DC} = -6 dB Setting 4 is with g_{DC} = -9 dB	 Setting 1 Setting 2 Setting 3 Setting 4	 $16 \cdot f_b$ f_b f_b f_b	 GHz

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Transmitter differential peak output voltage Victim Far-end aggressor Near-end aggressor	 A_v A_{fe} A_{ne}	 0.4 0.4 0.6	 V V V
Number of signal levels	L	2	—
Level separation mismatch ratio	R_{LM}	1	—
Transmitter signal-to-noise ratio	SNR_{TX}	27	dB
Number of samples per unit interval	M	32	—
Decision feedback equalizer (DFE) length	N_b	3	UI
Normalized DFE coefficient magnitude limit, for n = 1 to N_b	$b_{\max}(n)$	0.35	—
Random jitter, RMS	σ_{RJ}	0.01	UI
Dual-Dirac jitter, peak	A_{DD}	0.05	UI
One-sided noise spectral density	η_0	$5.2 \cdot 10^{-8}$	V^2/GHz
Target detector error ratio	DER_0	10^{-15}	—
COM Pass threshold	COM	2	dB

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.5.4 Channel Operating Margin (cont'd)

Table 29 — COM parameters for class C-R

Parameter	Symbol	Value	Units
Signal rate	f_b	6.375 to 32.45	GBd
Maximum start frequency	f_{min}	0.05	GHz
Maximum frequency step	Δf	0.01	GHz
Device package model Single-ended device capacitance Transmission line length, Test 1 Transmission line length, Test 2 Single-ended package capacitance at package-to-board interface Package transmission line nominal characteristic impedance	 C_d z_p z_p C_p Z_c	 1.8e-4 12 30 1.1e-4 95	 nF mm mm nF Ω
Single-ended reference resistance	R_0	50	Ω
Single-ended termination resistance	R_d	55	Ω
Receiver 3 dB bandwidth	f_r	$0.75 \cdot f_b$	GHz
Transmitter equalizer, minimum cursor coefficient	$c(0)$	0.5	—
Transmitter equalizer, pre-cursor coefficient Minimum value Maximum value Step size	$c(-1)$	 -0.333 0 0.08325	 — — —
Transmitter equalizer, post-cursor coefficient Minimum value Maximum value Step size	$c(1)$	 -0.333 0 0.08325	 — — —
Continuous time filter, DC gain Minimum value Maximum value Step size	g_{DC}	 -12 0 3	 dB dB dB
Continuous time filter, zero frequency for $g_{DC} = 0$, f_Z Setting 1 is with g_{DC} = 0 dB Setting 2 is with g_{DC} = -3 dB Setting 3 is with g_{DC} = -6 dB Setting 4 is with g_{DC} = -9 dB Setting 5 is with g_{DC} = -12 dB	 Setting 1 Setting 2 Setting 3 Setting 4 Setting 5	 $16 \cdot f_b$ $0.2825 \cdot f_b$ $0.2295 \cdot f_b$ $0.2114 \cdot f_b$ $0.2070 \cdot f_b$	 GHz
Continuous time filter, pole frequencies f_{p1} Setting 1 is with g_{DC} = 0 dB Setting 2 is with g_{DC} = -3 dB Setting 3 is with g_{DC} = -6 dB Setting 4 is with g_{DC} = -9 dB Setting 5 is with g_{DC} = -12 dB	 Setting 1 Setting 2 Setting 3 Setting 4 Setting 5	 $1.3 \cdot f_b$ $0.3594 \cdot f_b$ $0.29 \cdot f_b$ $0.2656 \cdot f_b$ $0.263 \cdot f_b$	 GHz

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Continuous time filter, pole frequencies, f_{p2} Setting 1 is with g_{DC} = 0 dB Setting 2 is with g_{DC} = -3 dB Setting 3 is with g_{DC} = -6 dB Setting 4 is with g_{DC} = -9 dB Setting 5 is with g_{DC} = -12 dB	 Setting 1 Setting 2 Setting 3 Setting 4 Setting 5	 16· f_b f_b f_b f_b f_b	GHz
Transmitter differential peak output voltage Victim Far-end aggressor Near-end aggressor	 A_v A_{fe} A_{ne}	 0.4 0.4 0.6	 V V V
Number of signal levels	L	2	—
Level separation mismatch ratio	R_{LM}	1	—
Transmitter signal-to-noise ratio	SNR_{TX}	27	dB
Number of samples per unit interval	M	32	—
Decision feedback equalizer (DFE) length	N_b	14	UI
Normalized DFE coefficient magnitude limit, for n = 1 to N_b	$b_{max}(n)$	0.35	—
Random jitter, RMS	σ_{RJ}	0.01	UI
Dual-Dirac jitter, peak	A_{DD}	0.05	UI
One-sided noise spectral density	η_0	$5.2 \cdot 10^{-8}$	V^2/GHz
Target detector error ratio	DER_0	10^{-15}	—
COM Pass threshold	COM	2	dB

5.2.5.4 Transmitter and receiver device package models

The transmitter and receiver device package models, used in the COM simulation, is defined in IEEE Std. 802.3 93A.1.2.

5.2.5.5 Filters

IEEE Std. 802.3-2018 93.A.1.4. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The voltage transfer function for each signal path $H_{21}^{(k)}(f)$ (see IEEE Std. 802.3 93A.1.3) is multiplied by a set of filter transfer functions to yield $H^{(k)}(f)$ as shown in Equation (44).

$$H^{(k)}(f) = H_{ffe}(f) \cdot H_t(f) \cdot H_{21}^{(k)}(f) \cdot H_r(f) \cdot H_{ctf}(f) \quad (44)$$

The receiver noise filter $H_r(f)$ is defined in 5.2.5.5.1, the transmitter transition time filter, $H_t(f)$, is defined by Equation (93A–46) where T_r is the 20% to 80% transition time. T_r is 0 unless defined otherwise for the Physical Layer specification that invokes this method, the transmitter equalizer, $H_{ffe}(f)$, is defined in (46), and the receiver equalizer, $H_{ffe}(f)$, is defined in (47).
The filtered voltage transfer function $H^{(k)}(f)$ is used to compute the pulse response (see IEEE Std. 802.3 93A.1.5).

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

5.2.5.5.1 Receiver noise filter

IEEE Std. 802.3-2018 93.A.1.4.1. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

$H_r(f)$ is a noise filter defined by Equation (45).

$$H_r(f) = \frac{1}{1 - 3.414214(f/f_r)^2 + (f/f_r)^4 + j2.613126(f/f_r - (f/f_r)^3)} \quad (45)$$

5.2.5.5.2 Transmitter equalizer

IEEE Std. 802.3-2018 93. A.1.4.2. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

$H_{ffe}(f)$ is defined by Equation (46) and is intended to represent the transmitter equalizer. If *k* corresponds to a near-end crosstalk path, then *c*(–1) and *c*(1) are zero regardless of the values used for the other paths. The value of the "cursor" coefficient *c*(0) is set to $1 - |c(-1)| - |c(1)|$ for any value of *c*(–1) and *c*(1). If the value of *c*(0) is less than the specified minimum value, the corresponding combination of *c*(–1) and *c*(1) is considered invalid and is not used to calculate COM.

$$H_{ffe}(f) = \sum_{i=-1}^{1} c(i)\, e^{(-j2\pi(i+1)\cdot(f/f_r))} \quad (46)$$

5.2.5.5.3 Receiver equalizer

IEEE Std. 802.3-2018 93. A.1.4.3. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

$H_{ctf}(f)$ is defined by Equation (47). Where *f* is the frequency and the coefficients g_{DC}, f_z, f_{p1} and f_{p2} are given in Table 27, Table 28 and Table 29 for respective class.

$$H_{ctf}(f) = \frac{10^{\frac{g_{DC}}{20}} + j\frac{f}{f_z}}{\left(1 + j\frac{f}{f_{p1}}\right)\left(1 + j\frac{f}{f_{p2}}\right)} \quad (47)$$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6 Transport layer

6.1 Overview

The transport layer maps the conversion samples to non-scrambled octets. JESD204 provides several options for this mapping:

- A single converter to a single-lane link
- Multiple converters in the same device to a single-lane link
- A single converter to a multi-lane link
- Multiple converters in the same device to a multi-lane link

In addition, it is possible to combine the interfaces of multiple converter devices on a multipoint link. However, this is not a special mapping case. The transport layer mapping applies only to the samples of a single converter device, i.e., transmitted over a single link.

A set of samples and/or partial samples is grouped into a frame of *F* octets. In many applications, the frame duration will be equal to the sampling interval. However, JESD204 allows more than one sample per converter to be transmitted per frame duration. The number *S* of samples per converter per frame duration shall always be an integer. This is necessary in order to minimize cross-talk between the SERDES circuits and sensitive analog parts. Each sample is transmitted as a group of *N'* bits, consisting of *N* data bits, optional control bits and optional tail bits. Additional tail bits at the end of the frame may be necessary to fill a whole number of octets per lane per frame cycle.

The user data mapping is parameterized according to the descriptions in the next subclauses. The limit for these parameters is only set by the needs of a particular converter device.

6.2 User data format for an independent lane

6.2.1 General

This subclause specifies the mapping of samples from one or more converters in the same device to octets for transmission over a single lane. This specification is backward-compatible with JESD204A and JESD204B. However, the position of tail bits may differ from JESD204 version April 2006.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6.2.2 User data mapping without oversampling

One device contains *M* converters, each producing *N* data bits per sample. The numbering scheme for all items in the below figures will start from 0. Within a sample, the leftmost bit is the most significant bit (MSB) and the rightmost bit is the least significant bit (LSB). The process of mapping the samples to octets is described in the following steps:

1. The samples are mapped to a linear axis, starting with converter 0, followed by converter 1, etc. until all samples have been mapped.
2. The samples are mapped to words. When the samples contain no control bits (e.g., overrange indication), the words are identical to the samples. When sample-specific control bits are available, there are two options:
   a. A conversion word is formed by appending the relevant control bits after the LSB of each conversion sample. The number of control bits per sample is indicated by parameter *CS*.
   b. A conversion word is identical to the corresponding sample. The control bits are grouped into a separate control word, which is appended after the words containing the 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 bits of converter 1, etc.

      If *CF* is the number of control words in the frame, with *CF*=0 or *CF*=1 for an independent lane, the total number of words transmitted per frame cycle is thus *M*+*CF*.
3. It is recommended, for words not containing a whole multiple of 4 bits, to extend each word to the smallest possible nibble group (group of half octets) using tail bits. The extended words are indicated by “NG” in Figure 38. This step is optional and may be disregarded in cases where highest line efficiency is prioritized against easier reconfigurability in mapping. A conversion word may thus be extended by control and/or tail bits to a length of $N' \geq N$ bits, where N' is recommended to be a whole multiple of 4. Note that for *CF*=0, control bits are considered part of the data word and there will be no tail bits between data and control bits, but one or more tail bits could be necessary after the control bit(s). For *CF*=1, data and control bits are in different words and one or more tail bits could be necessary after the data bits of each sample
4. If necessary, tail bits are appended to make the total number of bits after the last step a whole multiple of 8.
5. The sequence obtained in the previous step is regrouped into *F* octets.

Preliminary

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6.2.2 User data mapping without oversampling (cont'd)

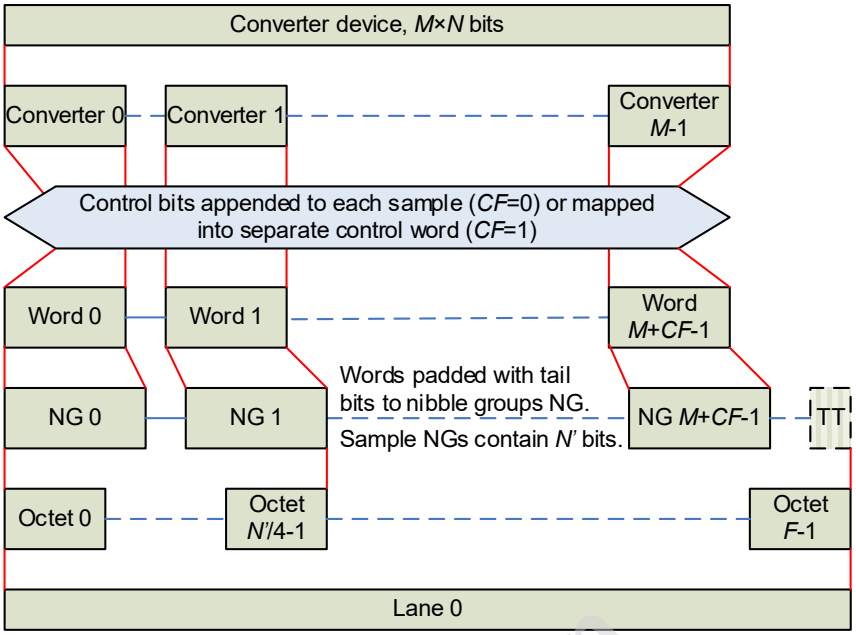


Figure 38 — User data format for an independent lane without oversampling

EXAMPLE Difference in tail bit position between JESD204 version 2006 and newer versions.

Figure 54 illustrates the difference in the recommended location of tail bits in version 2006 of JESD204 and later versions. In version 2006, tail bits were only inserted at the end of a frame. The newer versions recommend that tail bits be inserted at the end of each sample, if needed to fill a nibble group. In this example the new mapping method needs one extra octet per frame.

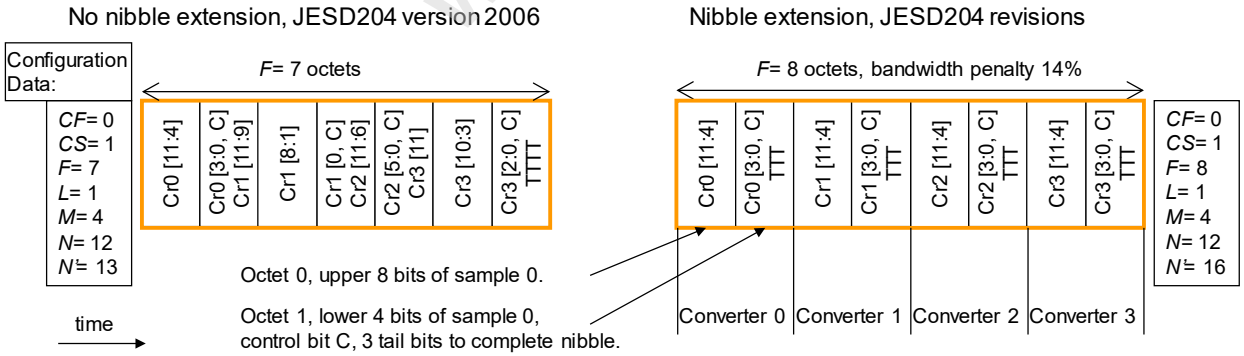


Figure 39 — JESD204 version 2006 and newer versions compared for a 4×12-bit converter with control bit over a single lane, no control words

EXAMPLE Compacting the frame structure by using a control word.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6.2.2 User data mapping without oversampling (cont'd)

Figure 55 illustrates how grouping the control bits into a control word can reduce the amount of tail bits and enable a shorter frame.

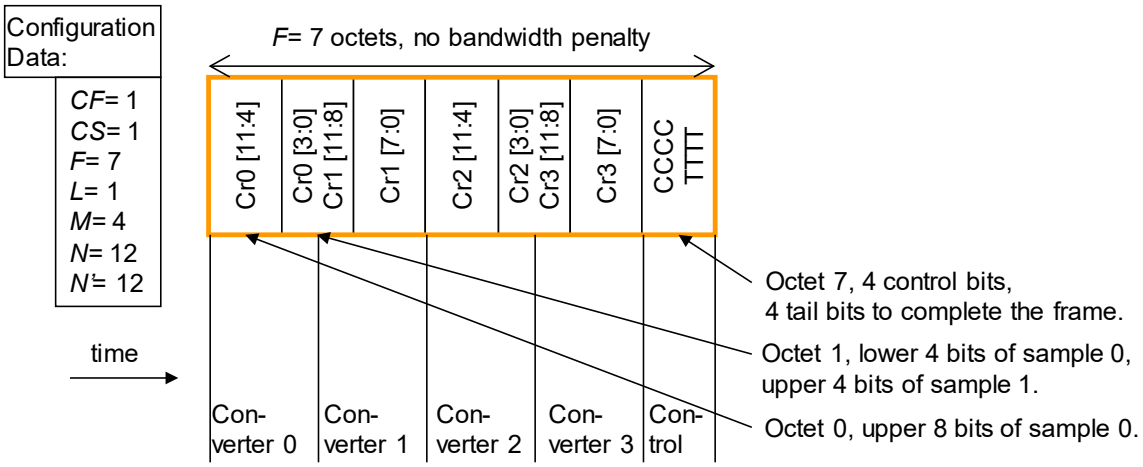


Figure 40 — User data format with control word (Example for 4×12-bit converter)

6.2.3 User data mapping with oversampling

The mapping used with oversampling is similar to the mapping without oversampling. The general principle is illustrated in the below figure. Instead of one sample per converter, *S* samples are cascaded before mapping the data of the next converter.

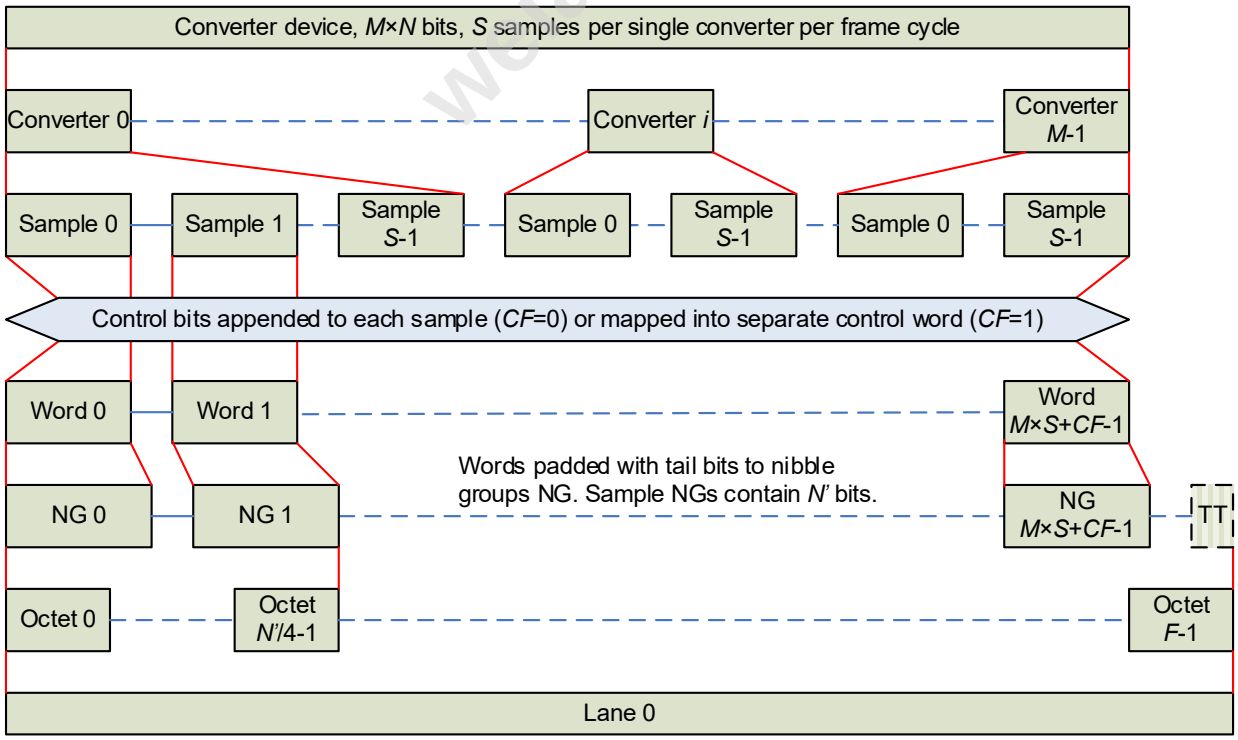


Figure 41 — User data format for an independent lane with oversampling

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6.2.3 User data mapping with oversampling (cont'd)

EXAMPLE Mapping for a 4×12-bit converter with two times oversampling and control word.

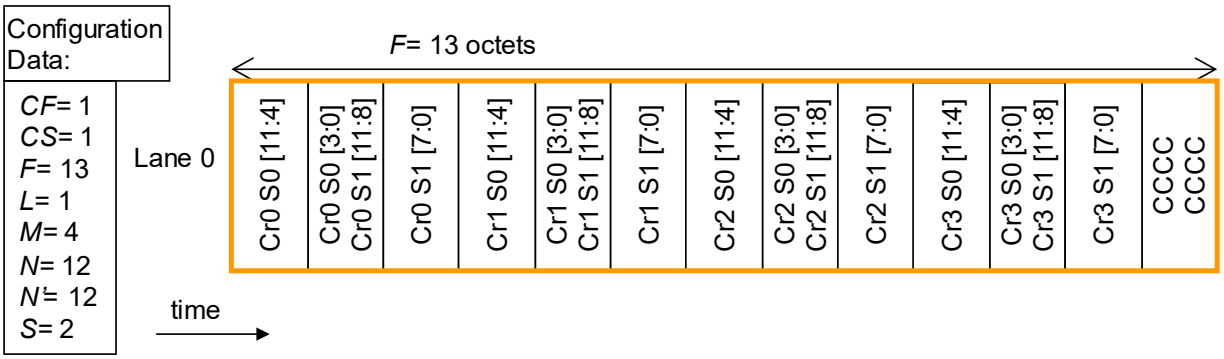


Figure 42 — User data format with oversampling
(Example for 4×12-bit converter, 2× oversampling and control word)

6.3 User data format for multiple lanes

For a link consisting of L lanes, the same mapping method is used as for a single lane. However, in the last step, a row of $L \cdot F$ octets is obtained. The first F octets are transmitted over lane 0, the next F octets over lane 1, etc. The last F octets are transmitted over lane L-1. This is illustrated in Figure 58.

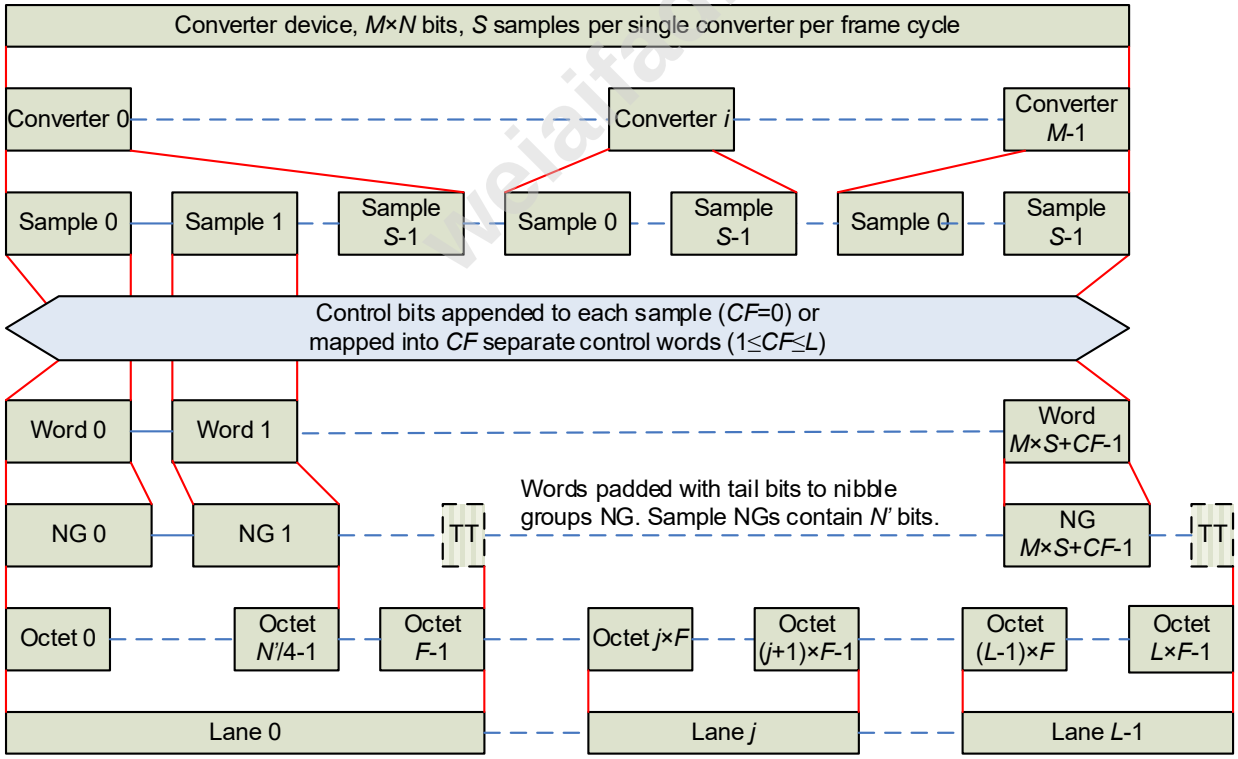


Figure 43 — User data format for multiple lanes

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6.3 User data format for multiple lanes (cont'd)

Compared to the independent lane format, the following additional specifications are needed.

6. Parameter *HD* controls whether a sample may be divided over more lanes. In the Low Density mode (*HD*=0), partial conversion words at the end of a group of *F* octets are avoided by adding more tail bits (TT) after the last full nibble group (NG) in the group if necessary. In the High Density mode (*HD*=1), the conversion words may break at the lane boundary.
7. Parameter *CF*, the total number of control words per frame period per link, controls which lanes will carry control words. *CF*=0 means that no control words are used. Other allowed *CF* values are common divisors of *L* and *M*. The *L* lanes are divided into *CF* groups of *L*/*CF* lanes. Each group of lanes transmits the samples of *M*/*CF* converters. After these samples a control word is inserted, containing in successive order the control bits belonging to these samples. If the control word fits on a single lane, it is not allowed to break it over a lane boundary.

EXAMPLE, Mapping of a 16×(11+2)-bit converter with and without control words

Figure 44 shows the mapping of a 16×11-bit converter with two control bits per sample without using control words. Note that the control bits extend each 11-bit sample to a 13-bit word, which is further extended to four nibbles by means of tail bits. In total 32 octets must be transmitted during one frame period, e.g., using 8 lanes of 4 octets per frame.

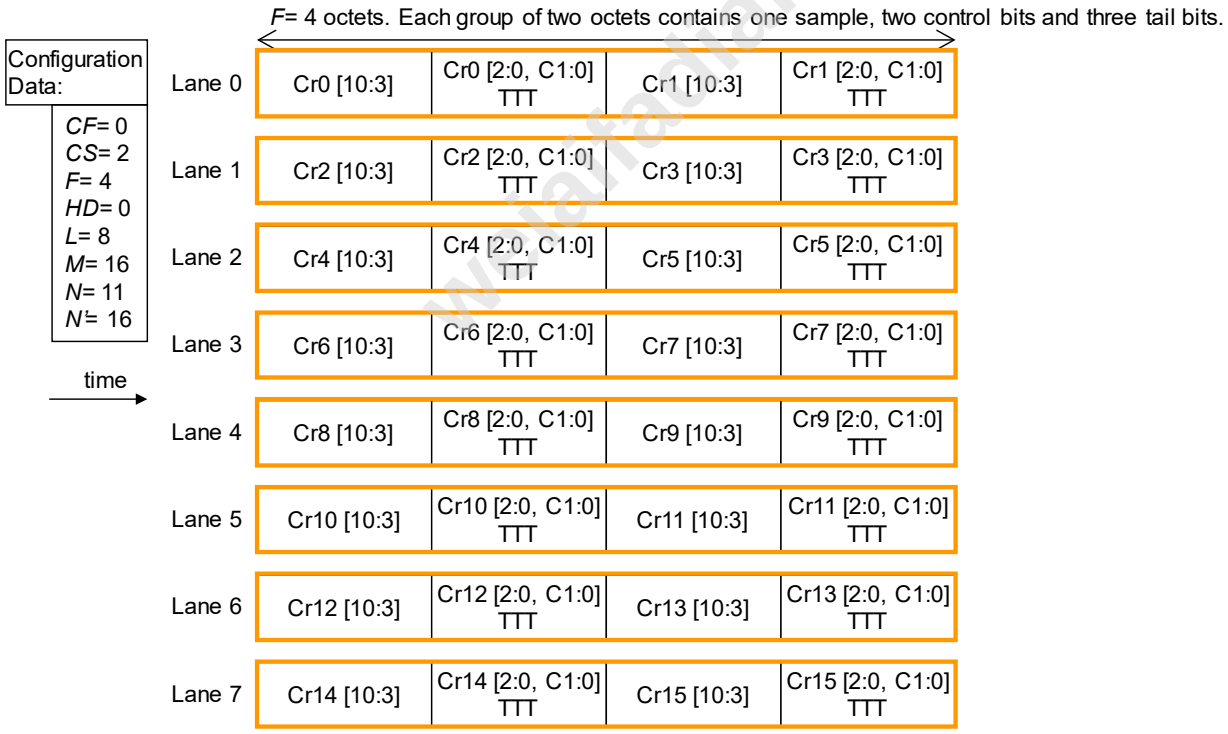


Figure 44 — Multilane user data format for 16×(11+2)-bit converter without control word

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6.3 User data format for multiple lanes (cont'd)

Figure 60 shows how the total amount of octets to transmit per frame period can be reduced to 28, by grouping all control bits into one control word, which is transmitted on the last lane. This way one lane can be saved. However, if only a fraction of the converters is active, there may be no saving in the amount of lanes that need to be powered on. For instance, to transmit only the data of converter 2, it is enough to activate lane 1 in the mapping of Figure 44, while lanes 0, 1 and 6 must be active in the mapping of Figure 45.

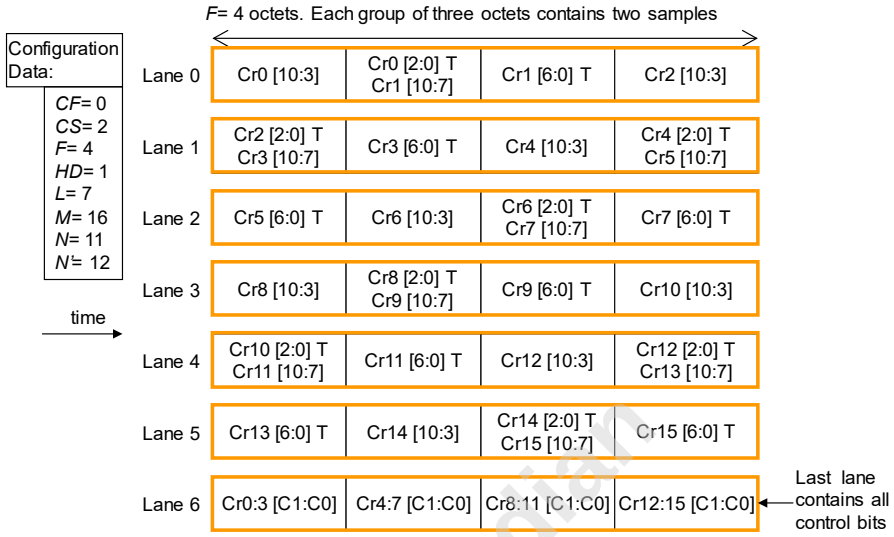


Figure 45 — Multilane user data format for 16×(11+2)-bit converter with single control word

Figure 61 shows the mapping of the same converter when using two control words. No octets are saved in this configuration. There would be room for an extra sample to transmit on lane 3, but that is prohibited by the requirement to transmit the data of an equal amount of converters in each group.

Figure 46 — Multilane user data format for 16×(11+2)-bit converter with two control words

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6.4 Tail bits

The tail bits (T) are specified for the unscrambled frame and are fed through the scrambler with the data bits. To avoid situations in which the tail bits prevent or significantly reduce the generation of frame synchronization symbols in the 8B/10B link layer (see 8.4.4.2), they shall be compliant with one of the following requirements:

- either the sequence of tail bits is the same from frame to frame, or
- the sequence is generated by a pseudo-random generator based on a polynomial of a degree of at least 9.

It should be noted that constant tail bits can potentially cause spurious spectral lines if scrambling is not used.

6.5 Idle mode

6.5.1 General

An idle mode is a state in which one or more converters that are connected to the same link are inactive, but the interface is kept alive and the frame structure is not changed.

In systems having multiple converters per link, it is possible that a converter is sharing parts of all its octets with other converters. Therefore, an inactive converter cannot be marked by a special octet code, e.g., a control symbol on the 8B10B link layer. Instead, a sample specific control bit could be used for this purpose. It is also possible to pass information on inactive converters via a control interface.

6.5.2 Dummy Samples

The samples of an inactive converter are replaced by dummy samples. There is no other requirement for the dummy samples than that they shall not prevent the generation of alignment characters in the 8B/10B link layer (see 8.4.4) irrespective of whether scrambling is enabled or not.

Dummy samples might be generated in the application layer, where it may not be known which bits in a dummy sample will be mapped to the last octet of a frame. Therefore, to avoid the possibility of interference with alignment character generation, it is recommended that all dummy bits comply with the same requirement as the tail bits (see 6.4). Pseudo-random bits are a good choice because they will avoid peaks in the transmitted spectrum when scrambling is disabled. Another option is to replace the samples of an inactive converter by a transport layer test sequence (see 6.6). However, if control bits are toggled as part of the test sequence a possible control bit for marking the inactive converter shall remain constant throughout the test sequence.

Weiaifantian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6.6 Test modes

6.6.1 General

A transport layer test mode is a state where the data samples from or to all converters connected to the same link are replaced by predetermined test samples. If the user data contains control bits, the control bits are replaced by test control bits. The test samples and control bits are mapped according to the current user data format and scrambled if scrambling is enabled. A JESD204 device is put into a test mode via a control interface.

NOTE Generation and detection of test samples is carried out in the application layer. The JESD204 link itself does not need a special mode for transport layer testing.

JESD204 specifies a long transport layer test pattern with a periodicity of multiple frames as well as a short transport layer test pattern with a periodicity of a single frame period. The use of other test patterns is optional. In general, test samples for transport layer testing shall comply with the following requirements:

- The pattern of test samples and possible control bits shall be repetitive. The periodicity shall be a whole number of frame periods, at the shortest one frame period.
- The pattern shall be such that the receiver is able to find the boundaries between successive periods of the pattern.

A transport layer test sequence will allow users to verify that the mappings between samples and octets and lanes in transmitter and receiver transport layers match. Although at the converter side the samples are also available in the analog domain, JESD204 devices shall have the possibility for transport layer testing entirely in the digital domain. This allows for testing the functionalities of the link and the data conversion independently from each other. A JESD204 TX shall support a test mode in which it replaces conversion samples and possible control bits by predetermined test samples and control bits. A JESD204 receiver shall support a test mode in which it compares the received test samples and control bits to the expected ones. The receiver shall have a means to communicate the test result to an upper layer, e.g., via a control interface.

6.6.2 Short transport layer test pattern

Support for the short test pattern is mandatory for all logic devices and for those converter devices that do not support the long test pattern. The short test pattern has a duration of one frame period and is repeated continuously for the duration of the test. Each sample shall have a unique value, which can be identified with the position of the sample in the user data format. The sample values shall be such that correct sample values will never be decoded at the receiver if there is a mismatch between the mapping formats being used at the transmitter and receiver devices. This can generally be accomplished by ensuring there are no repeating subpatterns within the stream of samples being transmitted. The converter manufacturer shall specify the test samples transmitted by an ADC device or expected by a DAC device, or shall give the device user the possibility to program these sample values via a control interface.

Not for distribution

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

6.6.3 Long transport layer test pattern

Support for the long test pattern is mandatory for all logic devices and for those converter devices that support transmission of control bits over the JESD204 interface. Also, the long test pattern is repeated continuously for the duration of the test. For a link connected to *M* converters, with *S* samples transmitted per converter per frame period, the test cycle has a length of max(M·S+2, 4) frames rounded up to the lowest number of full multiframes (when using 8B/10B encoding) or extended multiblocks (when using 64B/66B or 64B/80B encoding). For subclass 0 converters using 8B/10B encoding and without support of multi-lane alignment, the length of the test cycle is rounded up to the lowest whole multiple of 4 frames. With frame number *i* in the test cycle starting from 0, the data in the sequence is defined as follows:

- Frame 0: Each sample is the binary representation of the converter identification number *CID*+1 mod 2^N, with *CID* running from 0 to *M-1*
- Frame 1: Each sample is the binary representation of the sample identification number *SID*+1 mod 2^N, with *SID* running from 0 to *S-1*
- Frame $i \geq 2$: Each sample has the highest bit set to "1" and all other bits are set to "0".
- Frame $0 \leq i \leq M \cdot S-1$: If the converter device supports control bits, the highest control bit belonging to sample mod(i,S) of converter floor(i/S) is set to "1", while the remaining control bits are set to "0".
- Frame $i \geq \max(M \cdot S,2)$: Data bits like $i \geq 2$, all control bits set to "0"
- If tail bits are used, they shall be the same from frame to frame.

EXAMPLE Long transport layer test pattern for a 2× oversampling 2×14-bit converter with one control bit.

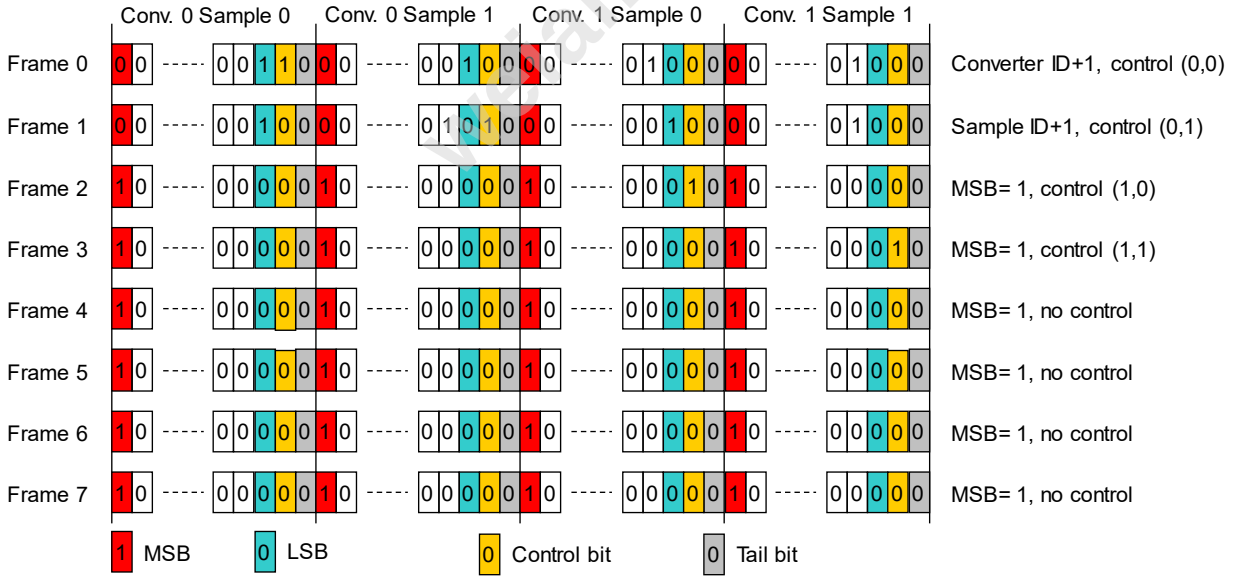


Figure 47 — Example of transport layer test sequence for a two-times oversampling dual 14-bits converter with one control bit

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7 64B/66B and 64B/80B link layer

7.1 Overview

7.1.1 64B/66B and 64B/80B encoding

The link layer described in this clause is built from the 64B/66B encoding concept where a 2-bit sync header is concatenated with 64 bits of data. Thus, the base level data structure is a block that starts with the 2-bit sync header. Multiblocks are built with concatenated blocks, and extended multiblocks are built with concatenated multiblocks.

While 64 bits of data within a block are always filled with scrambled user data, the 2-bit sync headers are simultaneously useful for the following:

- Receiver synchronization
- Constraining physical run length
- Extra bandwidth for features such as CRC, FEC, or generic user commands

In addition to 64B/66B encoding, this clause defines 64B/80B encoding. The purpose of 64B/80B encoding is to facilitate an integer relationship between the SERDES line rate and converter sample rate. Compared to 64B/66B encoding, the 64B/80B encoding increases the block size by inserting fill bits between the data octets.

7.1.2 Block structure

The block is a container consisting of a 2-bit unscrambled sync header followed in time by 8 octets of scrambled user data. The total width of a block (*BkW*) may be 66 or 80 depending on the PCS encoding mode. When the block width is 80, there are fill bits between the octets (see 7.2.5).

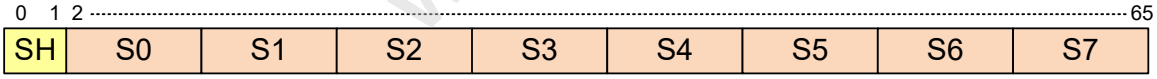


Figure 48 — 64B/66B block format

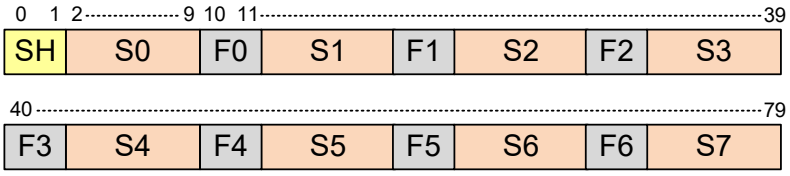
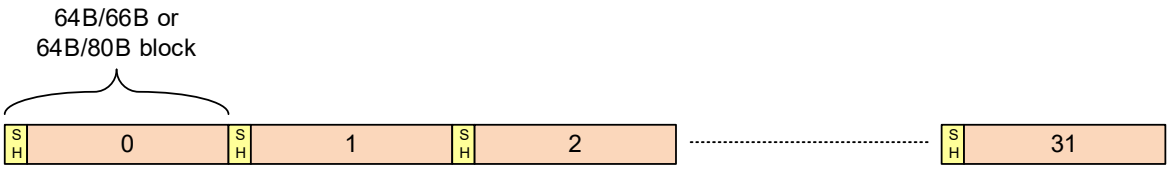


Figure 49 — 64B/80B block format

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.1.3 Multiblock structure

The multiblock is a 32-block container.



JESD204C MultiBlock Format

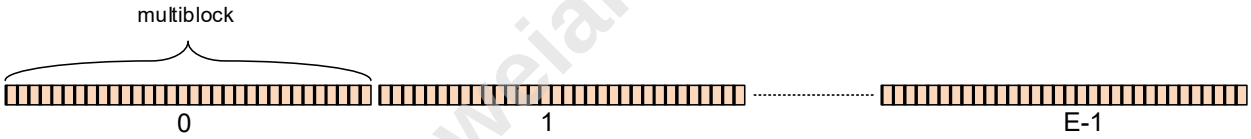
Figure 50 — Multiblock format

7.1.4 Extended multiblock structure

The extended multiblock is a container of *E* multiblocks where *E* is greater than or equal to one. An extended multiblock shall contain an integer number of frames, which is denoted by *K*. Therefore, a value of *E* greater than one must be used for applications where individual multiblocks do not contain an integer number of frames (i.e., when (256 mod *F*) != 0).

NOTE Frames are defined in the transport layer specification, see clause 6.

In subclass 1 (see 4.2.3), the length of an extended multiblock shall be larger than the maximum possible delay variation across any two lanes of a link or multipoint link, measured at skew point 5 (SP5 – see 4.3.8).



JESD204C Extended MultiBlock Format

Figure 51 — Extended Multiblock Format

7.1.5 Data channel

The data channel transmits the user data, generated by the transport layer, using the scrambled data bits of the 64B/66B or 64B/80B block (see Figure 53).

7.1.6 Sync header stream

The sync header stream provides transmission of information parallel to the user data using the transitions in the synchronization headers. This information contains the pilot signal, which is used to mark the borders of the multiblocks and extended multiblocks, and in addition at least one of the following:

- The CRC-12 or optional CRC-3 signal, used for error detection.
- The FEC signal, used for error detection and correction.
- The command channel, for transmitting commands and status information.

weiaifatian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.2 Physical coding sublayer

7.2.1 Overview

The PCS or “physical coding sublayer” is the part of the link layer responsible for the final digital transference of blocks to the physical layer. This includes sync header encoding/decoding as well as scrambling/descrambling. There are two formats for the PCS, one is 64B/66B and the other is 64B/80B. The purpose of 64B/80B encoding is to maintain an integer relationship between the SERDES line rate and converter sample rate. A device must implement at least one of these two encoding schemes, neither being mandatory. Both may be implemented in the same device for compatibility with different operating modes.

7.2.2 Transmit process

The transmitter scrambles the user data from the transport layer and encodes the sync header stream. The sync headers are then appended to the scrambled data and sent to the PMA. For the 64B/80B encoding, fill bits are added according to a PRBS sequence. For the 64B/66B encoding, these fill positions are not transmitted. The encoding of the sync headers is defined in 7.3.2. The scrambler is defined in 7.2.4.2. The fill bits for 64B/80B encoding are defined in 7.2.5.

The block bit transmission order shall be as illustrated in Figure 52.

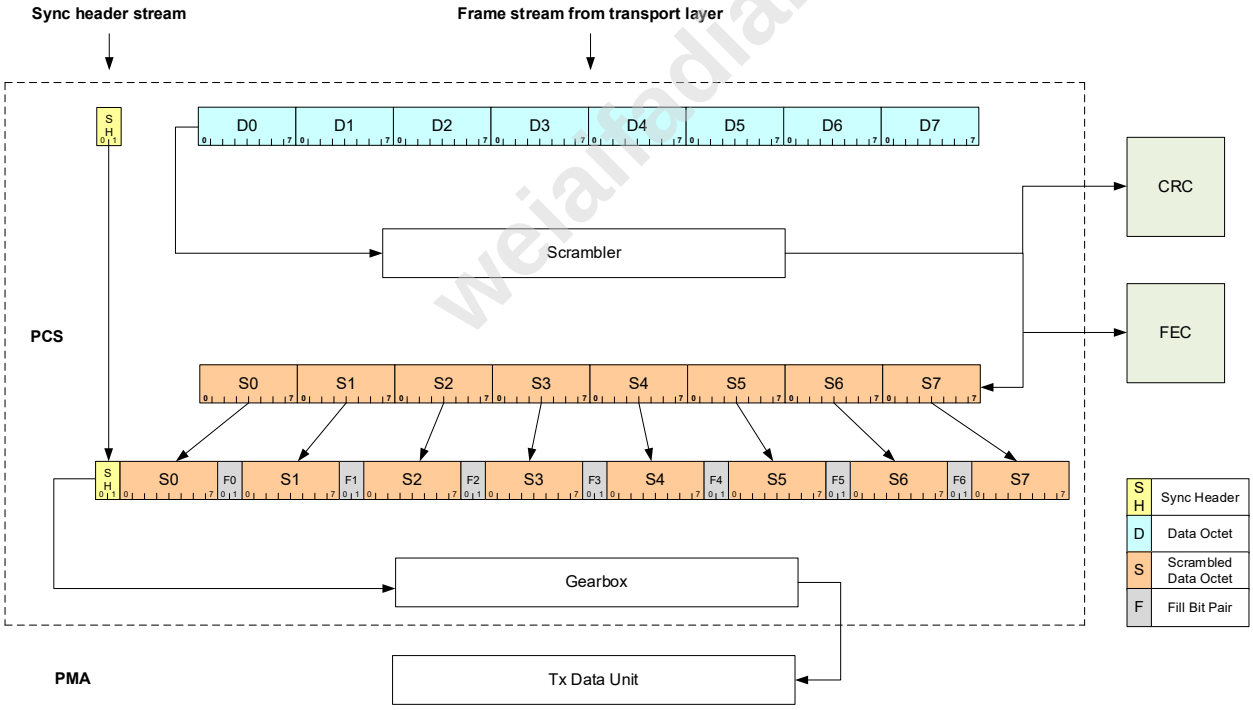


Figure 52 — PCS transmit bit ordering

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.2.2 Transmit process (cont'd)

The mapping between the transport layer (clause 0) and link layer is defined via a serial interface between these layers. The mapping in the transmit direction is illustrated in Figure 53. The leftmost bit of the frame, i.e., the MSB, is shifted in first. The actual implementation shall produce the same result as the serial definition.

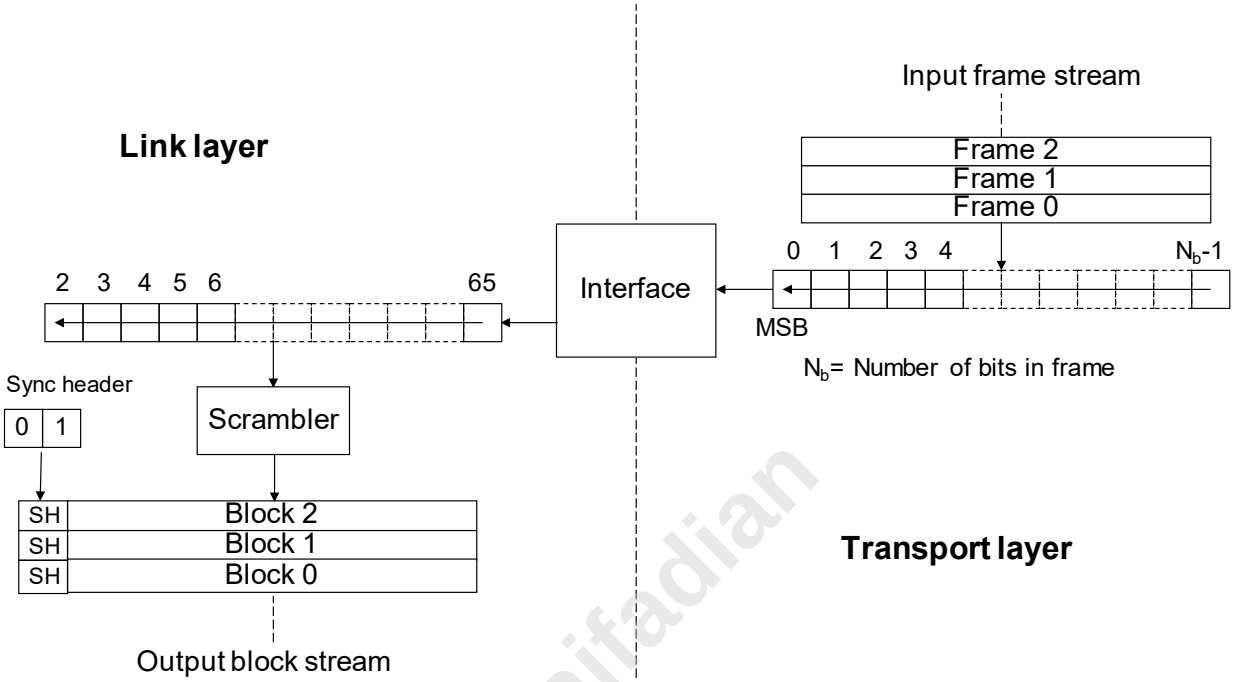


Figure 53 — Mapping from transport layer to link layer

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.2.3 Receive process

The receiver first determines the sync header boundaries. Once this has been achieved, it descrambles the user data for the data link and decodes the sync header stream. Fill bits are not received in the 64B/66B encoding mode, and they are always ignored in the 64B/80B encoding mode. The decoding of the sync headers is defined in 7.3.2. The descrambler is defined in 7.2.4.3.

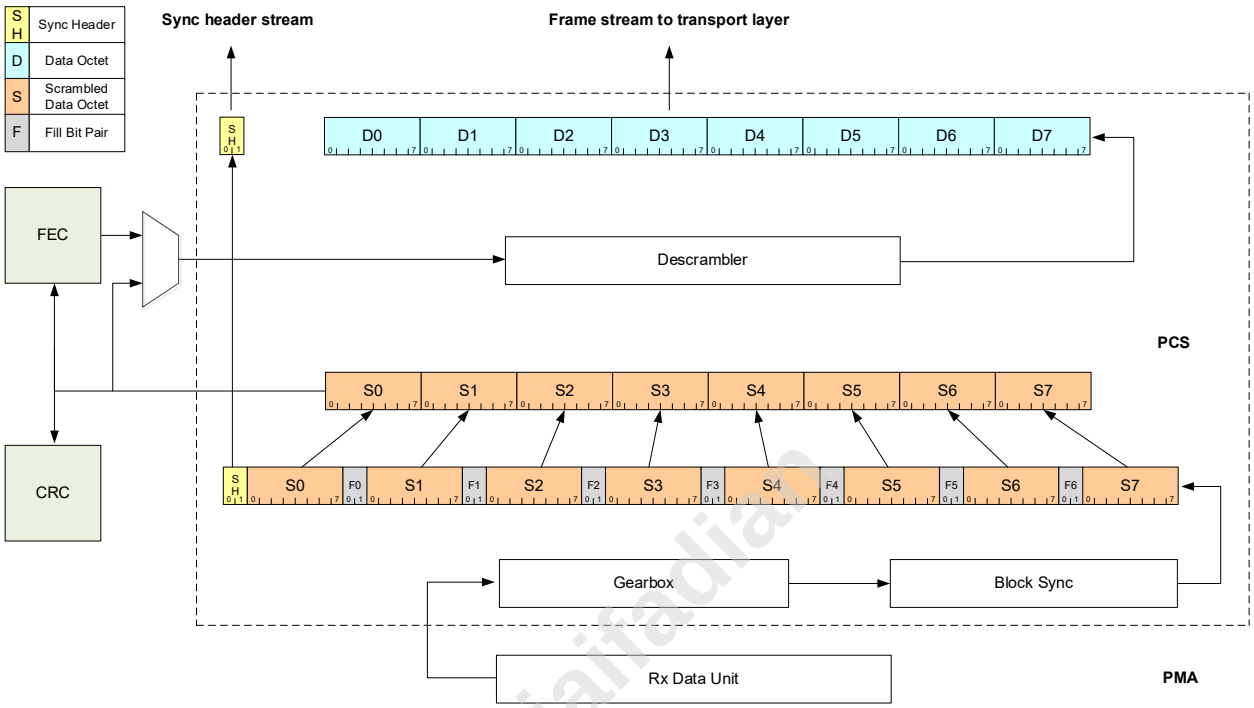


Figure 54 — PCS receive bit ordering

7.2.4 Scrambling and descrambling

7.2.4.1 Overview

The purpose of scrambling is as follows:

- Ensure a proper balance of ones and zeros over the physical link.
- Avoid the spectral peaks that may be produced when the same data octet repeats from frame to frame.
- Eliminate data-dependent bit errors.

Scrambling is required for all encoding modes and shall not be disabled for normal operation.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.2.4.2 Scrambler

The payload of the block is scrambled with a self-synchronizing scrambler. The scrambler shall produce the same result as the implementation shown in Figure 55. While Figure 55 and Figure 56 implement Galois style polynomials, they are compatible with the Fibonacci polynomial implemented in IEEE 802.3 [8]. Figure 55 implements the scrambler polynomial:

$$x^{58} + x^{39} + 1$$

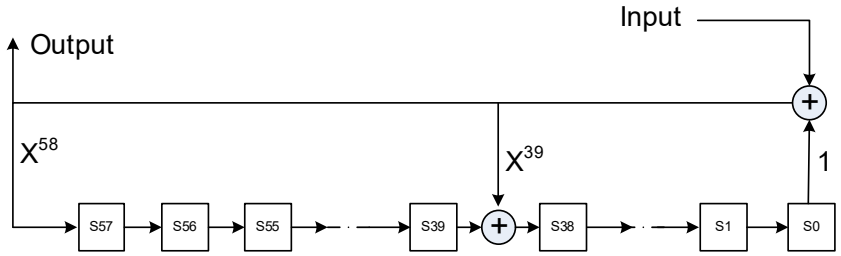


Figure 55 — Serial scrambler implementation

The recommended initial state for the storage elements is a non-zero number such as 0x01.

EXAMPLE Scrambler output for given input data and initial state

Table 30 — Scrambler example; initial state 0x01.

Block	Source data[63:0]	Scrambled data[63:0]
0	0x00_01_02_03_04_05_06_07	0x80_01_02_03_05_05_04_23
1	0x08_09_0A_0B_0C_0D_0E_0F	0x0E_43_80_C2_0B_50_81_CD
2	0x10_11_12_13_14_15_16_17	0x04_E7_83_92_5A_3C_AA_51
3	0x18_19_1A_1B_1C_1D_1E_1F	0x05_4D_87_D9_31_3d_11_51

7.2.4.3 Descrambler

The descrambler processes the payload to reverse the effect of the scrambler using the same polynomial. It shall produce the same result as the implementation shown in Figure 56.

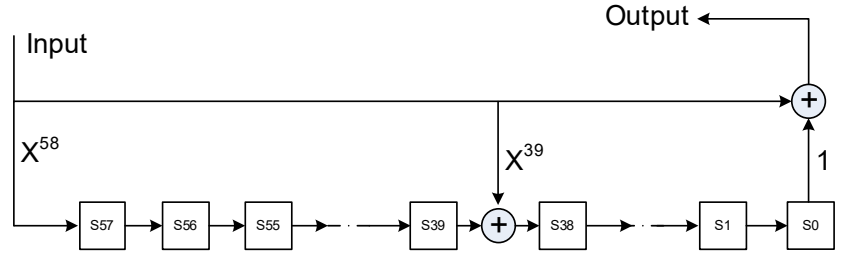


Figure 56 — Serial descrambler implementation

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.2.5 Fill bits for 64B/80B encoding

Fill bits are transmitted in the 64B/80B encoding mode only, and these bits are never scrambled. The fill bits shall be generated based on a 17 bit PRBS sequence. An exception to this shall occur, depending on the value of PRBS[2:0], in which one of four fill bit pairs (F1, F2, F4, or F5) is replaced with a “01” or “10” value instead of the normal PRBS value. This ensures a transition to prevent the maximum run length from increasing beyond 66. The specific details of the PRBS sequence, including the polynomial and seed value, are left to the device implementer.

Table 31 — Fill bit encoding

Field	Normally filled with	Replacement	Replaced if
F0[0:1]	PRBS[4:3]	-	Never
F1[0:1]	PRBS[6:5]	{!PRBS[2], PRBS[2]}	PRBS[1:0] == 00
F2[0:1]	PRBS[8:7]	{!PRBS[2], PRBS[2]}	PRBS[1:0] == 01
F3[0:1]	PRBS[10:9]	-	Never
F4[0:1]	PRBS[12:11]	{!PRBS[2], PRBS[2]}	PRBS[1:0] == 10
F5[0:1]	PRBS[14:13]	{!PRBS[2], PRBS[2]}	PRBS[1:0] == 11
F6[0:1]	PRBS[16:15]	-	Never

EXAMPLE

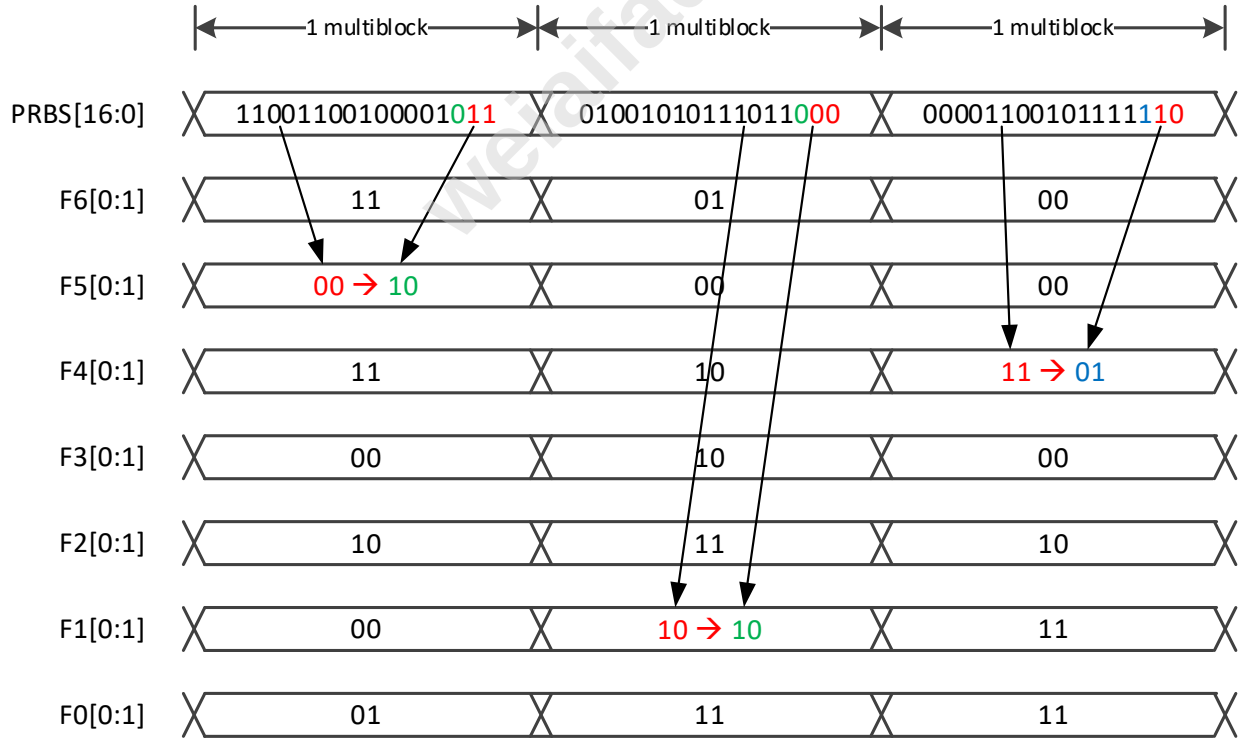


Figure 57 — 64B/80B fill bit mapping

Weifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.2.6 Gearbox

The purpose of a ‘gearbox’ is to translate data from an input data width and clock period to a different output data width and clock period. The total data rate at the input must always equal the data rate at the output. A gearbox may not be required if the physical layer interface matches the link layer input or output.

7.3 Sync header stream

7.3.1 Overview

The sync header stream provides transmission of information parallel to the user data, using the transitions within the sync headers in the PCS. A sync word of 32 bits is formed from the 32 sync transition bits per multiblock (see 7.3.2). The sync word consists of bits 0 to 31, of which bit 0 is transmitted first.

The sync word can contain the following information:

- pilot signal (always);
- CRC-3 signal;
- CRC-12 signal;
- FEC signal;
- command channel.

The above signals and command channel are described in the following subclauses. Support for the CRC-12 signal and associated sync word format (see Table 34) is mandatory. All other signal types except the pilot signal may be ignored by the receiver. Transmitters may fill the command channel with idle headers if desired (see 7.3.7.6). If an application does not require the use of a CRC or FEC signal, the stand-alone command channel (see 7.3.7.2) shall be transmitted.

7.3.2 Sync header encoding and decoding

The sync headers are encoded according to Table 32 and decoded according to Table 33.

Table 32 — SYNC Header Encoding

Sync transition bit	Sync header [0:1]
0	10
1	01

Table 33 — SYNC header decoding

Sync header [0:1]	Sync transition bit
00	Invalid
01	1
10	0
11	Invalid

NOTE Sync header bit 0 in the tables above is the first bit to be transmitted.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.3.3 Pilot signal

The purpose of the pilot signal is to allow the receiver to find the boundaries of the multiblocks and extended multiblocks. The format of the pilot signal depends on whether CRC-3, CRC-12, FEC, or only the command channel is in use. The pilot signal utilizes the fixed '1' and '0' bits as well as the EoEMB identifier shown in the mapping tables in the following subclauses. EoEMB in the pilot signal is defined as follows: '1' indicates the current multiblock is the last multiblock of the current extended multiblock, and '0' indicates the current multiblock is not the last multiblock of the current extended multiblock.

7.3.4 CRC-12 signal

The purpose of the CRC-12 signal is to allow error detection in the receiver with much higher precision than the CRC-3 signal. The mapping of the CRC-12 signal to the sync word is specified in Table 34.

Table 34 — Sync word mapping with CRC-12 signal

Bit	Function	Bit	Function	Bit	Function	Bit	Function
0	CRC[11]	8	CRC[5]	16	Cmd[6]	24	Cmd[2]
1	CRC[10]	9	CRC[4]	17	Cmd[5]	25	Cmd[1]
2	CRC[9]	10	CRC[3]	18	Cmd[4]	26	Cmd[0]
3	1	11	1	19	1	27	0
4	CRC[8]	12	CRC[2]	20	Cmd[3]	28	0
5	CRC[7]	13	CRC[1]	21	1	29	0
6	CRC[6]	14	CRC[0]	22	EoEMB	30	0
7	1	15	1	23	1	31	1

Table 35 — Meaning of the CRC-12 sync word fields

Content	Description
CRC	CRC-12 signal (see 7.4.2)
Cmd	Command channel (see 7.3.7.3)
1, 0, EoEMB	Pilot signal (see 7.3.3)

weiangjian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.3.5 CRC-3 signal

The purpose of the CRC-3 signal is to allow error detection in the receiver with much lower latency than the CRC-12 signal. The mapping of the sync word with a CRC-3 signal is specified in Table 36.

Table 36 — Sync word mapping with CRC-3 signal

Bit	Function	Bit	Function	Bit	Function	Bit	Function
0	CRC-A[2]	8	CRC-B[2]	16	CRC-C[2]	24	CRC-D[2]
1	CRC-A[1]	9	CRC-B[1]	17	CRC-C[1]	25	CRC-D[1]
2	CRC-A[0]	10	CRC-B[0]	18	CRC-C[0]	26	CRC-D[0]
3	1	11	1	19	1	27	0
4	Cmd[6]	12	Cmd[3]	20	Cmd[0]	28	0
5	Cmd[5]	13	Cmd[2]	21	1	29	0
6	Cmd[4]	14	Cmd[1]	22	EoEMB	30	0
7	1	15	1	23	1	31	1

Table 37 — Meaning of the CRC-3 sync word fields

Content	Description
CRC-A	CRC-3 signal for blocks 24 through 31 of the previous multiblock (see 7.4.3)
CRC-B	CRC-3 signal for blocks 0 through 7 of the current multiblock (see 7.4.3)
CRC-C	CRC-3 signal for blocks 8 through 15 of the current multiblock (see 7.4.3)
CRC-D	CRC-3 signal for blocks 16 through 23 of the current multiblock (see 7.4.3)
Cmd	Command channel (see 7.3.7.3)
1, 0, EoEMB	Pilot signal (see 7.3.3)

Vetaruan

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.3.6 FEC signal

The purpose of the pilot signal is to allow enhanced error detection in the receiver at the cost of longer detection latency. The mapping of the FEC signal to the sync word is specified in Table 38.

Table 38 — Sync word mapping with FEC signal

Bit	Function	Bit	Function	Bit	Function	Bit	Function
0	FEC[25]	8	FEC[17]	16	FEC[9]	24	FEC[2]
1	FEC[24]	9	FEC[16]	17	FEC[8]	25	FEC[1]
2	FEC[23]	10	FEC[15]	18	FEC[7]	26	FEC[0]
3	FEC[22]	11	FEC[14]	19	FEC[6]	27	0
4	FEC[21]	12	FEC[13]	20	FEC[5]	28	0
5	FEC[20]	13	FEC[12]	21	FEC[4]	29	0
6	FEC[19]	14	FEC[11]	22	EoEMB	30	0
7	FEC[18]	15	FEC[10]	23	FEC[3]	31	1

Table 39 — Meaning of the FEC sync word fields

Content	Description
FEC	FEC signal (see 7.5.2)
1, 0, EoEMB	Pilot signal (see 7.3.3)

NOTE The pilot signal for FEC does not contain fixed “1” bit locations before the last bit in the sync word, so the EoMB sequence “00001” may appear within the multiblock as well as at the end of the multiblock. This out-of-place EoMB sequence may result in the receiver requiring extra time for multiblock alignment and false-positive errors potentially being reported in the process. The user may avoid extra synchronization time by transmitting all ones in place of the FEC parity for a period of time after startup.

Do not violate
Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.3.7 Command channel

7.3.7.1 Overview

The command channel provides a mechanism for the transmitter to send various commands and data to the receiver over the sync headers without taking space from the data channel. This channel can only exist when the device is not transmitting the FEC signal.

7.3.7.2 Stand-alone command channel

The stand-alone channel may be used when extra bandwidth is needed for command words. It is not possible to send a CRC signal simultaneously with the stand-alone command signal. The mapping of the stand-alone command channel to the sync word is specified in Table 40.

Table 40 — Sync word mapping with stand-alone command channel

Bit	Function	Bit	Function	Bit	Function	Bit	Function
0	Cmd[18]	8	Cmd[12]	16	Cmd[6]	24	Cmd[2]
1	Cmd[17]	9	Cmd[11]	17	Cmd[5]	25	Cmd[1]
2	Cmd[16]	10	Cmd[10]	18	Cmd[4]	26	Cmd[0]
3	1	11	1	19	1	27	0
4	Cmd[15]	12	Cmd[9]	20	Cmd[3]	28	0
5	Cmd[14]	13	Cmd[8]	21	1	29	0
6	Cmd[13]	14	Cmd[7]	22	EoEMB	30	0
7	1	15	1	23	1	31	1

Table 41 — Meaning of the Cmd sync word fields

Content	Description
Cmd	Command channel (see 7.3.7.3)
1, 0, EoEMB	Pilot signal (see 7.3.3)

weiangjian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.3.7.3 Command word encoding

A command word is made up of 6 command bits from the command channel. When coupled with a CRC signal, the command word is formed from bits Cmd[6:1]. When the command channel is alone in the sync word, three command words will be transmitted: Cmd[18:13], Cmd[12:7] and Cmd[6:1]. The MSB of each command word shall specify whether the following 5 bits form a header or payload. Command bit Cmd[0] shall contain parity for all other Cmd bits in each sync word. This parity bit is formed using XOR logic. The encoding is illustrated in Table 42 and Table 43.

Table 42 — Command word encoding with CRC utilized

Cmd bit	Encoding	Cmd bit	Encoding
		3	Header[2] | Payload[2]
6	0: Header | 1: Payload	2	Header[3] | Payload[3]
5	Header[0] | Payload[0]	1	Header[4] | Payload[4]
4	Header[1] | Payload[1]	0	Parity

Table 43 — Command word encoding for stand-alone command channel

Cmd bit	Encoding	Cmd bit	Encoding	Cmd bit	Encoding	Cmd Bit	Encoding
		14	Header[3] | Payload[3]	9	Header[2] | Payload[2]	4	Header[1] | Payload[1]
18	0: Header | 1: Payload	13	Header[4] | Payload[4]	8	Header[3] | Payload[3]	3	Header[2] | Payload[2]
17	Header[0] | Payload[0]	12	0: Header | 1: Payload	7	Header[4] | Payload[4]	2	Header[3] | Payload[3]
16	Header[1] | Payload[1]	11	Header[0] | Payload[0]	6	0: Header | 1: Payload	1	Header[4] | Payload[4]
15	Header[2] | Payload[2]	10	Header[1] | Payload[1]	5	Header[0] | Payload[0]	0	Parity

Verification

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.3.7.4 Header and payload sequence

When a header is received, it can be followed by payload bits in multiples of 5. There shall be no more than 35 payload bits following a header. The number of payload bits transmitted depends on the function code defined by the header (see Table 44) and how much payload pertains to that particular function.

EXAMPLE Figure 58 and Figure 59 show examples of the command bit structure for payloads of 0, 5 and 10 bits.

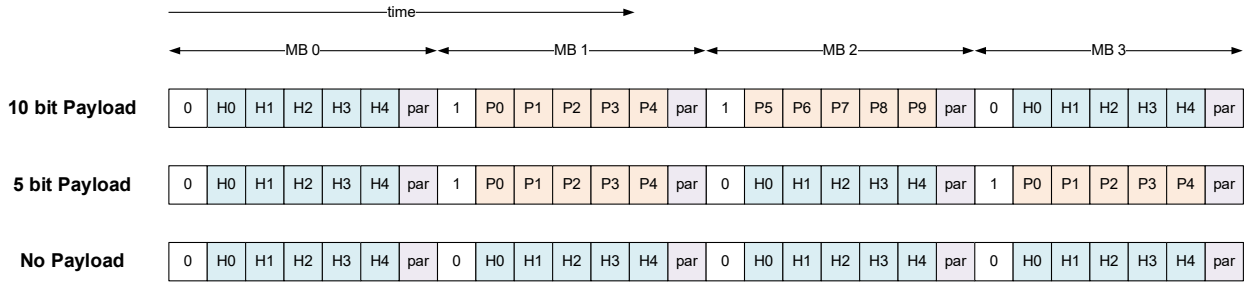


Figure 58 — Payload size examples (CRC utilized)

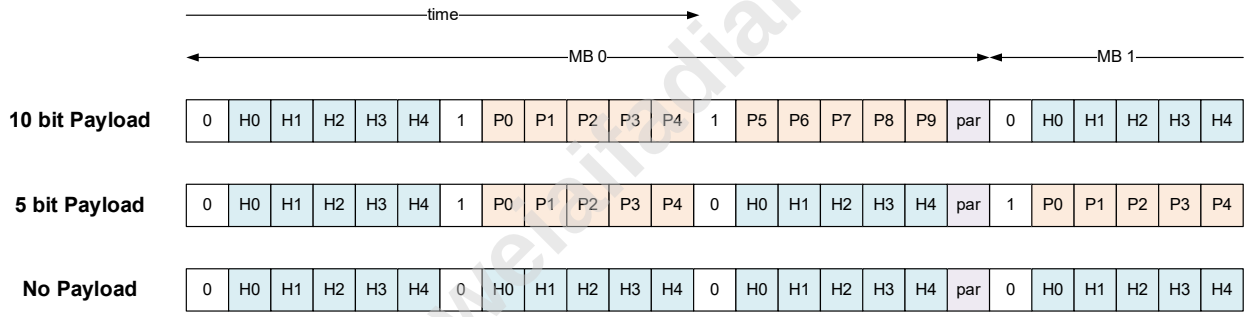


Figure 59 — Payload Size Examples (stand-alone command channel)

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.3.7.5 Operating modes

Since separate sync header streams (and thus separate command bits) exist for each lane, the command channel has two operating modes. The single-lane mode ignores all but one lane, and the multi-lane mode utilizes all lanes in unison to form the command channel. The receiver and transmitter must be programmed to use the same mode via their separate control interfaces. When a single lane is utilized, all command bits except those on lane 0 are ignored. When multiple lanes are utilized, the sequence shall begin with lane 0 for the first command word and continue with consecutive command words up to the last lane before starting again at lane 0 in the next command word.

EXAMPLE

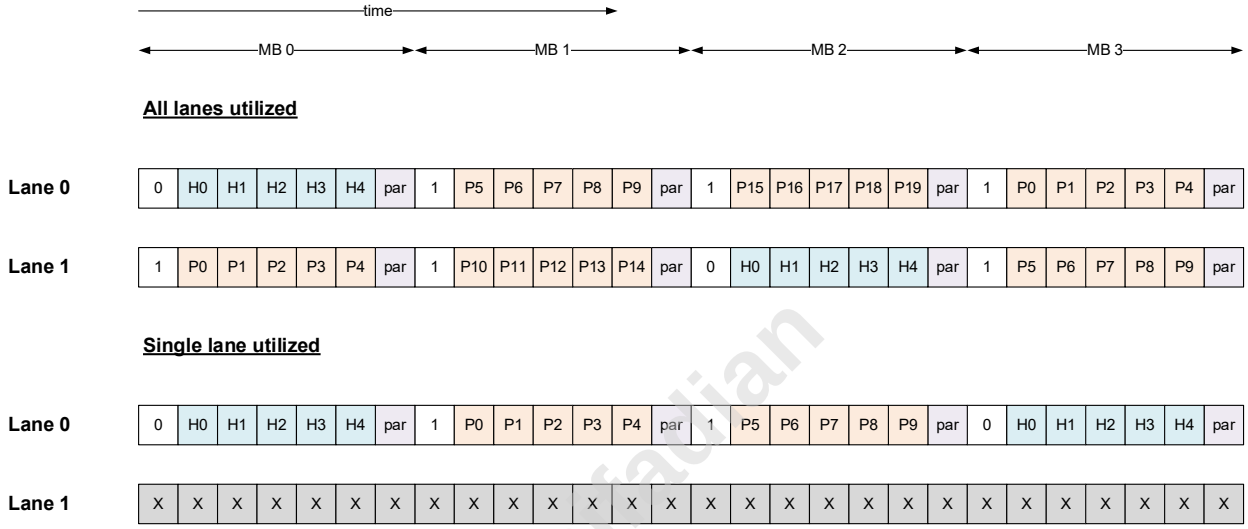


Figure 60 — Command channel operating modes (CRC utilized)

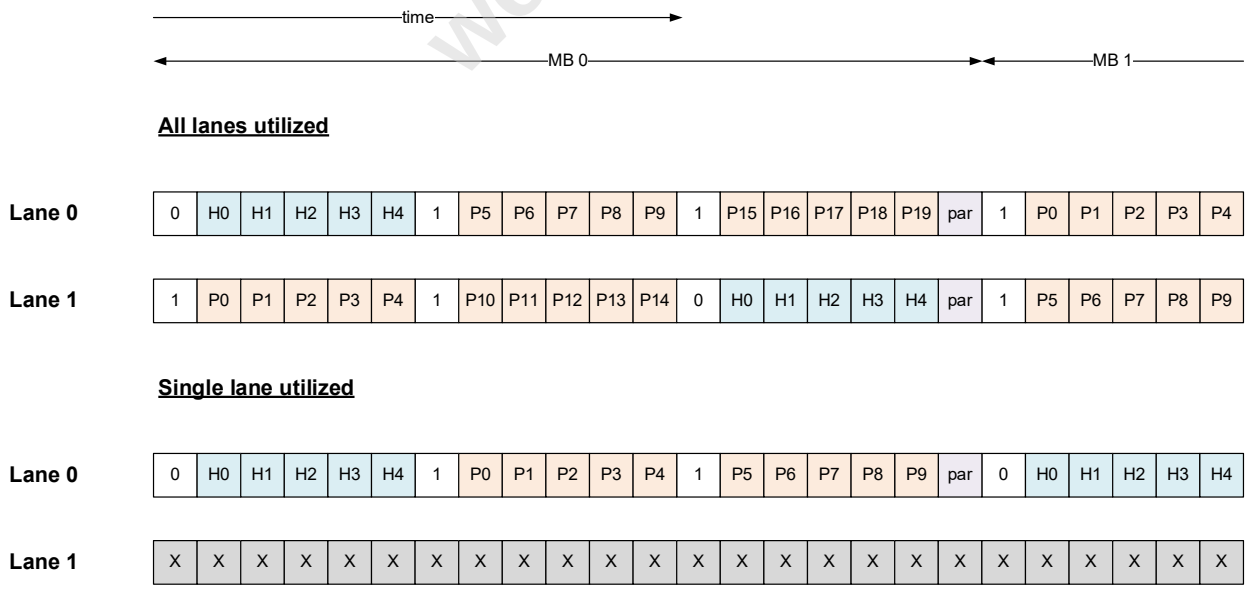


Figure 61 — Command channel operating modes (stand-alone command channel)

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.3.7.6 Function codes

Function codes are sent via the command channel headers. Some header values are reserved for specific functions to be implemented in hardware. Others are user defined and may be implemented by higher layer software. The predefined function codes assume that a logic device is interfacing with a coupled transmitter and receiver such that read requests and buffer control requests may be useful. These function codes are ignored for devices with a unidirectional link.

Table 44 — Command word function definitions

Code	Function	Payload Bits	Description
0x00	noOP	0	Idle header
0x01	Full Register Write	25	Byte-addressable register write interface. 17 MSBs used for address, eight LSBs used for data.
0x02	Streaming Register Write	10	Register write using a byte-addressable 17-bit address of 1 + the address from the latest 0x01 or 0x02 function code received. If no 0x01 or 0x02 function code has yet been received then an address of 0 shall be used. The two MSBs of the payload are unused.
0x03	Register Read Request	20	Byte-addressable register the read request interface. 17 MSBs used for address, three LSBs unused.
0x04	Register Read Acknowledge	25	Read data returned from previous request. 17 MSBs used for address acknowledgement, eight LSBs used for read data.
0x05	Buffer Full	0	Indicates to transmitter that the receiver payload buffer is full (or almost full depending on application). This shall trigger an interrupt in the transmitting device to halt all further transactions on the command channel until a “Buffer Clear” command is received.
0x06	Buffer Clear	0	Indicates to transmitter that receiver payload buffer is ready to accept more data (though it may not be empty).
0x07 – 0x0F	Reserved	-	Not to be used.
0x10 – 0x1F	User Defined	-	Function to be decided by user.

7.4 CRC encoding

7.4.1 General

CRC is available to detect potential bit errors during transmission. JESD204C supports two CRC methods: CRC-3 and CRC-12. CRC-3 has lower latency, but worse error detection capabilities. Support for CRC-12 encoding and error checking is mandatory. Support for CRC-3 encoding and error checking is optional.

The transmitter computes the CRC parity bits from the scrambled data bits within groups of blocks. It encodes these parity bits on the sync header stream aligned with the next group of blocks. The receiver computes the CRC parity bits of the received scrambled data bits in the same group of blocks and compares them to the parity bits that have been received over the link. If the parity bits do not match, there is at least one error in the received data bits or in the received parity bits.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.4.2 CRC-12 encoding

The CRC-12 encoder takes in the 2048 scrambled data bits of each multiblock and computes 12 parity bits, defined by the following generating polynomial:

$$0x987 == x^{12} + x^9 + x^8 + x^3 + x^2 + x + 1$$

This polynomial will detect all 2-bit errors, spanning any distance, in a multiblock and burst error patterns with length up to 12 bits. The probability of not detecting a 3-bit error spanning any distance in a multiblock is approximately 0.004%.

The CRC-12 calculation shall produce the same result as the serial implementation shown in Figure 62. Before processing a new multiblock, the shift register shall be initialized to all zeros.

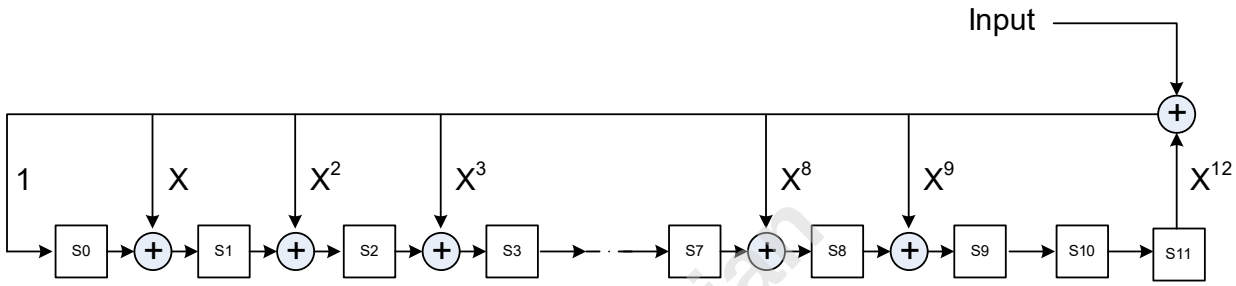


Figure 62 — CRC-12 field generation

After all data bits have been shifted in, S11, S10, … S0 will hold the CRC[11], CRC[10], … CRC[0] parity bits respectively.

Because of the mapping of the CRC bits to the sync header stream (see 7.3.4), CRC-12 has a minimum latency of 46 blocks in the detection of a bit error in the first scrambled data bit of a multiblock.

EXAMPLE CRC-12 result for given input data

Table 45 shows an example of the CRC-12 parity bits being computed as each block is processed. The LFSR has a starting value of all zeros before block 0 is processed.

Table 45 — CRC12 example

Block	Source data[63:0]	CRC[11:0]
0	0x80_01_02_03_05_05_04_23	0xD00
1	0x0e_43_80_c2_0b_50_81_cd	0x11C
2	0x04_e7_83_92_5a_3c_aa_51	0xFEA
3	0x05_4d_87_d9_31_3d_11_51	0x5FE

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.4.3 CRC-3 encoding

For the computation of the CRC-3 parity bits, the multiblock is split into four groups of eight blocks. The CRC-3 encoder takes in the 512 scrambled data bits within these groups and computes three parity bits, defined by the following generator polynomial:

$$0x5 == x^3 + x + 1$$

This polynomial will detect all single-bit errors in a group of eight blocks and all burst error patterns with length up to three bits.

The CRC-3 calculation shall produce the same result as the serial implementation shown in Figure 63. Before processing a new group of eight blocks, the shift register shall be initialized to all zeros.

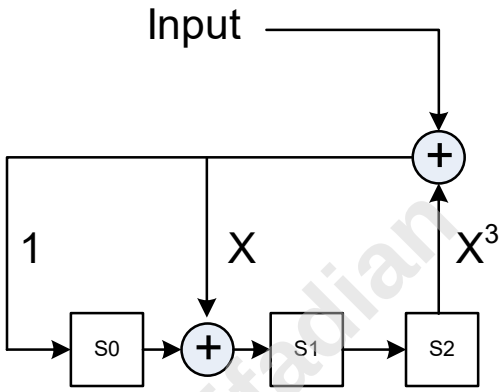


Figure 63 — CRC-3 field generation

After all data bits have been shifted in, S2, S1 and S0 will hold the CRC[2], CRC[1] and CRC[0] parity bits respectively.

Because of the mapping of the CRC bits to the sync header stream (see 7.3.5), CRC-3 has a minimum latency of 10 blocks in the detection of a bit error in the first scrambled data bit of a group of eight blocks.

EXAMPLE CRC-3 result for given input data

Table 46 shows an example of the CRC-3 parity bits being computed as each block is processed. The LFSR has a starting value of all zeros before block 0 is processed.

Table 46 — CRC-3 example

Block	Source data[63:0]	CRC[2:0]
0	0x80_01_02_03_05_05_04_23	011
1	0x0e_43_80_c2_0b_50_81_cd	111
2	0x04_e7_83_92_5a_3c_aa_51	010
3	0x05_4d_87_d9_31_3d_11_51	001

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.5 FEC encoding

7.5.1 General

Forward error correction (FEC) is an optional feature, which can be used to improve BER for error-sensitive applications. While CRC can only detect errors, FEC can also correct errors.

The transmitter computes the FEC parity bits of the scrambled data bits in a multiblock and encodes these parity bits on the sync header stream of the next multiblock. The receiver calculates the syndrome of the received bits, which is the difference between the locally generated and the received parity. If the syndrome is zero, the received data bits are assumed to be correct. If the syndrome is non-zero, it can be used to determine the most likely error.

Because of the mapping of the FEC parity bits to the sync header stream (see 7.3.6), FEC has a minimum latency of 58 blocks in the detection and correction of a bit error in the first scrambled data bit of a multiblock.

7.5.2 FEC encoding

The FEC parity bits are calculated in a similar way to CRC. The FEC encoder takes in the 2048 scrambled data bits of the multiblock and adds 26 parity bits to construct a shortened (2074, 2048) binary cyclic code. The (2074, 2048) binary cyclic code is shortened from the cyclic Fire code (8687, 8661) The generator polynomial for this code is:

$$g(x) = (x^{17} + 1)(x^9 + x^4 + 1) == x^{26} + x^{21} + x^{17} + x^9 + x^4 + 1$$

This polynomial can correct up to a 9-bit burst error per multiblock.

In detail, the parity bits are generated as follows:

- The 2048 message bits are encoded as a polynomial $m(x) = m_{2047}\, x^{2047} + m_{2046}\, x^{2046} + m_{2045}\, x^{2045} + \ldots m_1\, x + m_0$, where $m_{2047}\ m_{2046}\ m_{2045} \ldots m_0$ are the 2048 data bits to be transmitted
- The message polynomial m(x) is then multiplied by x^{26} and then divided by the generator polynomial g(x) to obtain the parity bits: $rem(x) = x^{26}\, m(x) \bmod g(x)$
- The cyclic codeword c(x) is then formed by appending rem(x) to $x^{26}\, m(x)$: $c(x) = x^{26}\, m(x) + rem(x)$

Figure 64 shows a systematic encoder for computing $x^{26}\, m(x) \bmod g(x)$. The state of the registers S25…S0 shall be cleared before processing a new multiblock. The encoding process is then started by serially shifting in information data starting with the coefficient of the term with the highest degree. Information data is shifted in from the front end of the division circuit, which is equivalent to multiplying m(x) by x^{26}. The cyclic shift register encoder, with tap coefficients matching the generator polynomial, then performs the division to obtain the parity bits. After all data bits are shifted in, S25 S24 …S0 will hold the FEC25 FEC24…FEC0 parity bits.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.5.2 FEC encoding (cont'd)

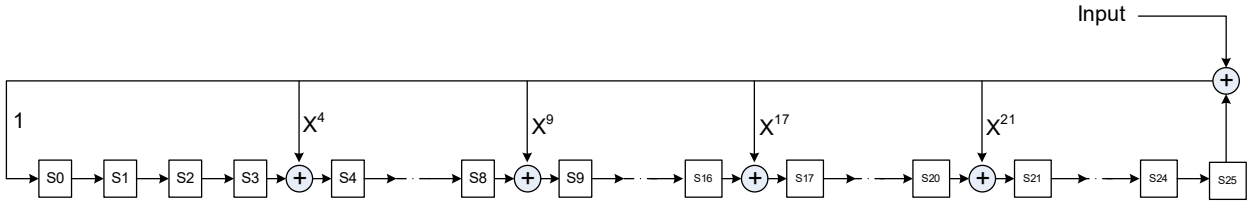


Figure 64 — Serial FEC Encoder

EXAMPLE

Table 47 — FEC Block Boundary Example

Block	Source Data[63:0]	FEC[25:0]
0	0x80_01_02_03_05_05_04_23	0x22A43C3
1	0x0e_43_80_c2_0b_50_81_cd	0x05DFF66
2	0x04_e7_83_92_5a_3c_aa_51	0x07171AB
3	0x05_4d_87_d9_31_3d_11_51	0x01957FD

Table 48 — FEC MultiBlock Boundary Example

MultiBlock	Source Data [2047:0]	FEC [25:0]
0	0x80_00_00_..._00_00_00	0x2_35_01_1A
1	0xFF_FF_FF_..._FF_FF_FF	0x3_C7_FF_E3
2	0x00_00_00_..._00_00_01	0x0_22_02_11

7.5.3 FEC decoding

7.5.3.1 FEC code reconstruction

The FEC bits of a multiblock are transmitted as part of the next multiblock. The FEC reconstruction block, therefore, has to create the cyclic codeword by appending the FEC bits transmitted as part of the current multiblock to the data that was transmitted in the previous multiblock. Figure 65 below illustrates this process where the data from the current multiblock is shifted into the upper shift register before being loaded as part of the codeword register at the start of the next multiblock. The FEC data, on the other hand, is shifted into the lower register so that it can be appended to the data received as part of the previous multiblock. When the last FEC bit from the current multiblock is received, the codeword is formed and the decoder can start the decoding process. Note that if filler bits are present on the incoming stream, then they are simply ignored.

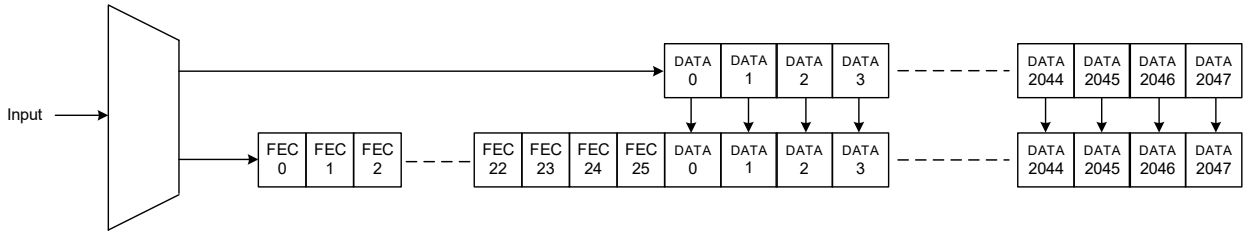


Figure 65 — Codeword reconstruction at the receiver

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.5.3.2 FEC code decoding

The decoding process (see Annex J for more details) consists of computing the syndromes $s^j(x)$ of the rotated words $r^j(x)$ ($r^j(x)$ is the received word r(x) rotated by j bits, $0 \leq j \leq 2047$), trapping the error and correcting the error once it is trapped. A serial decoder for the cyclic (2074, 2048) shortened code is shown Figure 81.

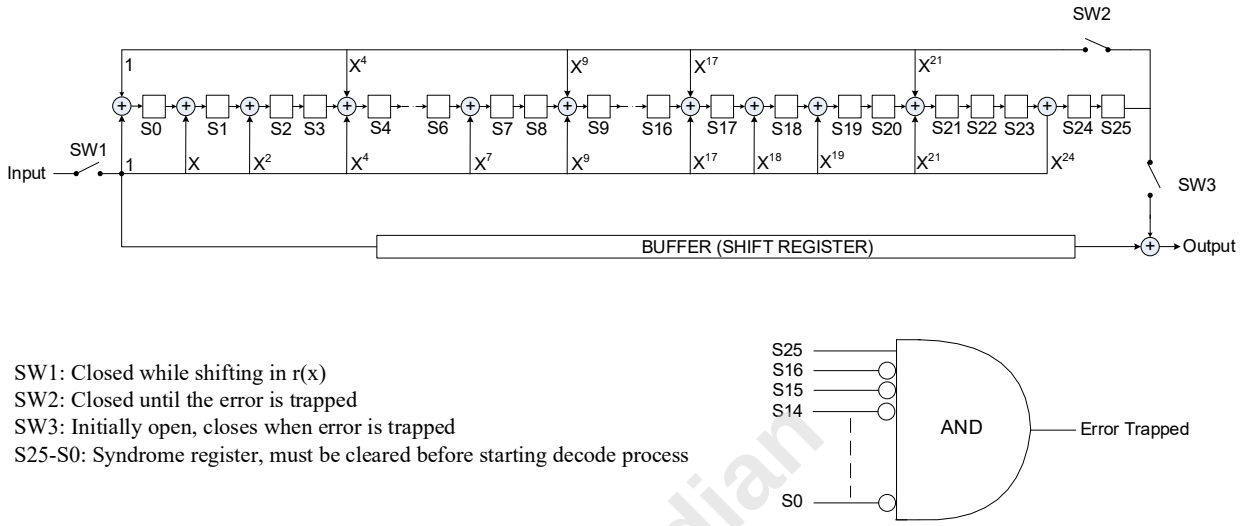


Figure 66 — (2074, 2048) fire decoder

The above decoder computes the syndrome s(x) of the word r(x) after the last input bit of the received word is shifted in. It then looks for a trapped error in the syndrome register (S25-S0) by checking if the MSB of the computed syndrome, S25, is a '1' and if the computed syndrome (S25-S0) contains at most one burst error of 9 bits or less. Checking bit S25 ensures that the trapped error is right-aligned and checking that bits (S16-S0) of the computed syndrome are all '0' ensures that if an error is trapped, it is located within (S25-S17), the 9 upper MSB bits of the syndrome register.

If the error is trapped after computing s(x), then the decoder circuit proceeds to correcting the received word. The above decoder arranges the syndrome bits such that when an error is trapped within the syndrome register, the erroneous data bits line up with the non-zero bits of the syndrome register. Hence, correction can proceed by a simple XOR operation that inverts the erroneous bits (SW3 in the figure above closes after an error is trapped).

If the error is not trapped after computing s(x), then the above decoder proceeds to computing the syndrome $s^1(x)$ of the word $r^1(x)$ (r(x) rotated once) and simultaneously shifting out one bit out of the shift register. The decoder circuit, then, checks if an error is trapped and proceeds with correction if it is. Otherwise, it again shifts out the received word and computes the syndrome $s^2(x)$ of $r^2(x)$ ($r^1(x)$ rotated once). The data shift and syndrome computation of the rotated words continues until the error is trapped and the received word is corrected. If after shifting out all received bits, the error is never trapped and the syndrome is not all zeros, then an uncorrectable error condition is detected (the circuitry for detecting this condition is not shown in the above decoder).

Preliminary
Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.5.3.2 FEC code decoding (cont'd)

It should be noted that the above decoder not only arranges the syndrome bits such that when an error is trapped, the error lines up with the erroneous bits on the shift register, but it also takes into account that r(x) is a shortened word. The syndrome s(x) of the shortened word r(x) is computed as follows (see Annex J for more details):

- Calculate the polynomial $p(x) = x^{(8687-8661)+(8687-2074)} \bmod g(x) = x^{6639} \bmod g(x)$
  $= x^{24} + x^{21} + x^{19} + x^{18} + x^{17} + x^9 + x^7 + x^4 + x^2 + x + 1$

- The syndrome of the received shortened word is then: $s(x) = r(x)\, p(x) \bmod g(x)$

The decoder operation/control is illustrated in Figure 67.

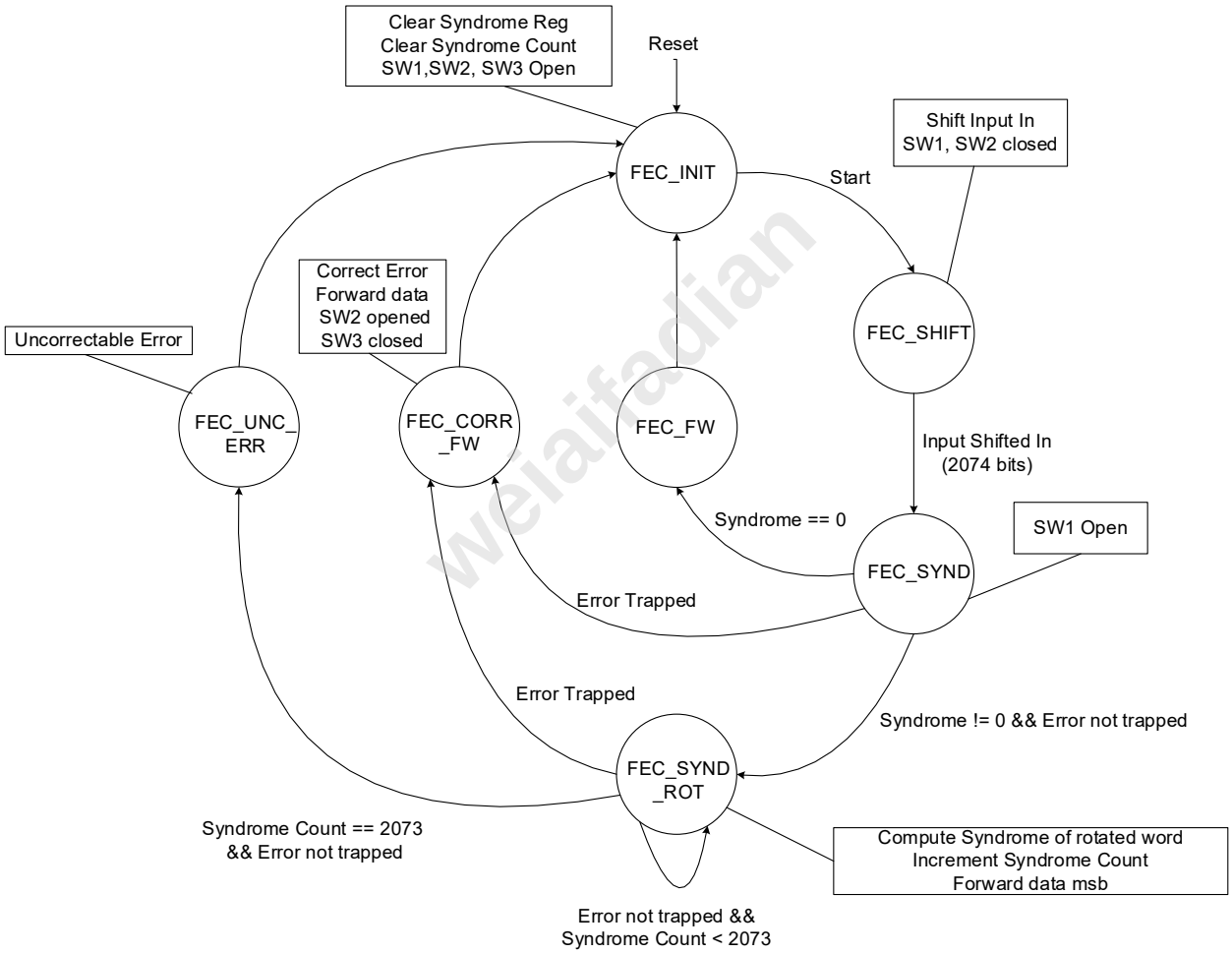


Figure 67 — Decoder sequencing state machine

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.5.3.2 FEC code decoding (cont'd)

Figure 82 state definitions as follows:

- FEC_INIT: Clear the syndrome register, the syndrome count (j), and open switches SW1, SW2 and SW3. Proceed to FEC_SHIFT when encoded data is available.
- FEC_SHIFT: The reconstructed word (data and parity bits) is shifted serially into the buffer and the syndrome registers; SW1 and SW2 are closed during this step. Proceed to FEC_SYND when the last input bit is being shifted into the buffer and syndrome registers.
- FEC_SYND: The syndrome $s^0(x) = s(x)$ of the received word r(x) is now available in the syndrome register (S25-S0). Open SW1 and examine the calculated syndrome:
  - If the calculated syndrome is all 0s, then the received word is error-free. Proceed to FEC_FW state where the buffered word is serially shifted out one bit at a time.
  - If the calculated syndrome is not all 0s and the error trapped signal is asserted, then proceed to FEC_CORR_FW where the error can be corrected and the data forwarded.
  - If the calculated syndrome is not all 0s and the error is not trapped, then proceed to FEC_SYND_ROT.
- FEC_SYND_ROT: The buffer register msb is forwarded to the output, $S^{j+1}(x)$ the syndrome of $r^{j+1}(x)$ is computed, and the syndrome count (j) is incremented. Note that $r^{j+1}(x)$ is $r^j(x)$ cyclically shifted by one bit and that $S^{j+1}(x) = x\ S^j(x) \bmod g(x)$ (See Annex J for more details).
  - If the "Error Trapped" signal is asserted, then the error is trapped. Proceed to FEC_CORR_FW to correct the error and forward the data.
  - If the error is not trapped and the input buffer is not fully shifted out, execute FEC_SYND_ROT again.
  - If the error is not trapped and the buffer is fully shifted out, then we have an uncorrectable error. Proceed to FEC_UNC_ERR
- FEC_FW: The received word is error-free. This state, then, forwards the received data to the output without any modifications.
- FEC_CORR_FW: The received word is erroneous and the error was trapped. SW3 is closed to correct the error. SW2 is open to clear the error out of the syndrome register.
- FEC_UNC_ERR: Flag that an uncorrectable error was detected.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.6 Receiver Operation

7.6.1 Sync header alignment

Receiver sync header alignment can be achieved using the following algorithm. The use of other algorithms, e.g., as described in [8] is not precluded.

- The receiver contains a bit-counter that counts from 0-*BkW-1*. It starts by holding the counter at 0.
- The receiver then waits for a transition in the incoming bit stream, at which point it releases the counter.
- The receiver continues to examine every BkW^{th} incoming bit to ensure that it always contains a transition
- If 64 consecutive valid sync headers are detected, then sync header alignment is achieved (SH_LOCK == 1)
- Before the locked state, any single invalid sync header shall return the algorithm to the first step
- Within the locked state, the receiver continues to monitor incoming sync headers. A threshold (Rx_Thresh_SH_ERR) is specified for determining the number of erroneous sync headers required to force the algorithm back to the initial SH_INIT state. This threshold may be fixed at 16 or user programmable with the option of selecting the value 16.
- A transition from SH_LOCK 1→0 on any lane shall force all lanes to the SH_INIT state.

The process of sync header alignment is illustrated in Figure 68.

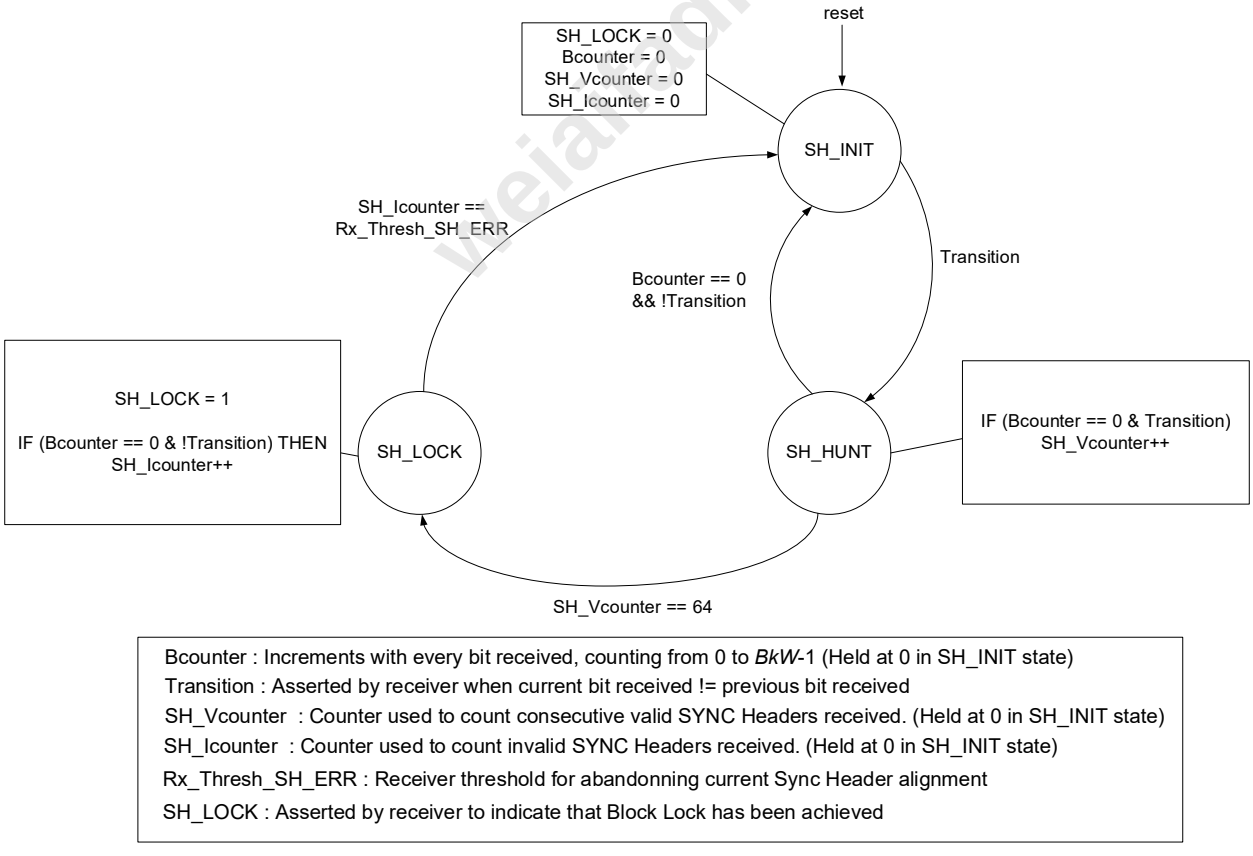


Figure 68 — Receiver state machine for sync header alignment

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.6.2 Extended multiblock alignment

Extended multiblock alignment shall first require that SH_LOCK == 1. Receiver extended multiblock alignment can be achieved using the following algorithm:

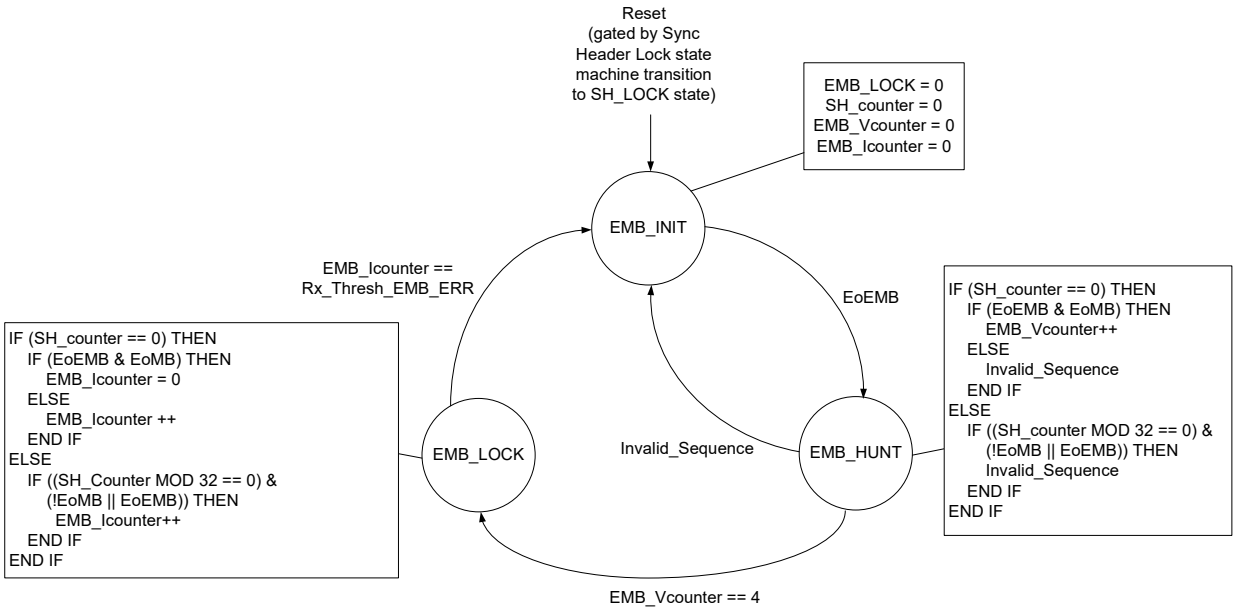
- The receiver contains a counter that increments on each received sync header / sync transition bit. The counter shall count from 0 to (*E**32-1), but is initially held at 0.
- The receiver waits for an end-of-extended-multiblock sequence, which is a completed pilot signal transmission ending in {0, 0, 0, 0, 1}, with EoEMB == 1 in the sync header stream. The counter is started when this sequence is found.
- The receiver continues to examine every 32nd incoming sync transition bit, ensuring it always aligns to the end of a correct pilot signal transmission.
    - For cases where *E* == 1, every 32nd sync transition bit shall correspond to the end of a pilot signal transmission with EoEMB == 1 (since EoEMB is always set to '1' when *E* == 1).
    - For cases where *E* > 1, every 32nd sync transition bit shall correspond to the end of a pilot signal transmission (00001 sequence). For every *E**32nd sync transition bit, the pilot signal will have contained an EoEMB set to '1'. In addition, for every 32nd sync transition bit except for the *E**32nd, the pilot signal will have contained an EoEMB set to '0'.
    - If 4 consecutive valid sequences are detected, extended multiblock alignment is asserted (EMB_LOCK == 1). A valid sequence is defined as correct EoEMB & EoMB values for a full *E**32-bit sync transition stream.
    - Any single invalid sequence (i.e., an invalid EoEMB value or 00001 sequence) returns the algorithm to the initial EMB_INIT state.
- Within the locked state (EMB_LOCK), the receiver still monitors the pilot signal. A threshold (Rx_Thresh_EMB_ERR) is specified for determining the number of consecutive, erroneous sequences required to force the algorithm back to the initial EMB_INIT state. This threshold may be fixed at 8 or user programmable with the option of selecting the value 8.

The process of extended multiblock alignment is illustrated in Figure 69.

.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.6.2 Extended multiblock alignment (cont'd)



SH_counter : Increments with every decoded Sync Header bit received, counting from 0 to (*E**32 - 1). (Held at 0 in EMB_INIT state)

EoMB : Asserted by the receiver when the 00001 pattern is detected in the last bits of the incoming stream

EoEMB : Asserted by the receiver when bit 22 of the pilot signal is asserted (0 otherwise)

EMB_Vcounter : Counter used to count consecutive valid EndOfMultiBlock sequences received. (Held at 0 in EMB_INIT state)

EMB_Icounter : Counter used to count invalid EndOfMultiBlock sequences received. (Held at 0 in EMB_INIT state)

Rx_Thresh_EMB_ERR : Receiver error threshold for abandoning current alignment phase

EMB_LOCK : Asserted by receiver to indicate that extended multiblock alignment has been achieved

Figure 69 — Receiver state machine for extended multiblock alignment

weidadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.6.3 Error handling

7.6.3.1 Error kinds

Table 49 shows the minimum set of errors to be detected in each receiver:

Table 49 — Minimum set of errors to detect per receiver

Error	Description
Invalid Sync Header	“11” or “00” received in expected sync header location.
Unexpected EoMB sequence	The “00001” sequence in the pilot signal is not received at the expected location in the sync word.
Incorrect EoEMB	The EoEMB identifier in the pilot signal has an unexpected value.
Parity error in command channel	The final parity bit in the command channel data for a given sync word does not match the calculated parity for the received command channel bits.
Invalid command channel header	A header received in the command channel does not correspond to a recognized value in the receiver.
Smaller than expected payload in command channel	Less payloads than expected are received in the command channel based on the previous header.
CRC error	The receive CRC generator has calculated a parity which does not match the parity received in the sync word.
Uncorrectable FEC error	The FEC decoder has detected an error that cannot be corrected.

weiaifadian

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

- “0” received in a “fixed 1” pilot bit location
- Larger than expected payload received in command channel

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

7.6.3.2 Action taken on error

Table 50 describes the action that shall be taken by the receiver when a particular error type is detected. More actions than listed here may be available depending on the receiver implementation.

Table 50 — Action taken on detected error

Error	Action
Invalid Sync Header	See 7.6.1
Unexpected EoMB sequence	See 7.6.2
Unexpected EoEMB	See 7.6.2
Parity error in command channel*	The previously received header and all subsequent payload is ignored until a new header is received. In the case of the stand-alone command channel, all headers received in the erroneous sync word are ignored.
Invalid command channel header*	The header and all subsequent payload are ignored.
Smaller than expected payload in command channel*	The corresponding header and all subsequent payload are ignored.
CRC error*	The error is reported via the control interface
Uncorrectable FEC error*	The error is reported via the control interface

7.6.3.3 Error reporting via control interface

A receiver in a logic device shall have the ability to report errors via a control interface to higher application layers. The details of this reporting are application-dependent 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.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8 8B/10B link layer

8.1 8B/10B encoding

The link layer described in this clause encodes the data as 8B/10B symbols before being transmitted. The 8B/10B encoding is specified in IEEE Std. 802.3™-2018 [2]. The 8B/10B codes have the following properties:

- Sufficient bit transition density (three to eight transitions per 10-bit symbol) to allow clock recovery by the receiver.
- DC balance (can be AC-coupled).
- Detection of single bit errors.
- Control symbols used for
  - establishing receiver synchronization to the 10-bit symbol boundaries,
  - marking the start and end of frames or other sequences of data, and
  - enabling alignment between serial lanes.

8.2 Transmission order

The frame contents are processed from left to right, i.e., from MSB to LSB. After serialization, the leftmost bit of the 8B/10B code group, i.e., bit "a" (see IEEE Std. 802.3™-2018, 36.2.4 [2]) is transmitted first.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.3 Scrambling

8.3.1 Introduction

Although it is not mandatory to enable scrambling, JESD204C TX and RX devices shall support scrambled data octets, as shown below. The scramblers and descramblers are one per lane. Functionally they are located between the transport layer and the link layer, as illustrated in Figure 70. Enabling scrambling/descrambling for a link involves activating the individual scramblers/descramblers on each lane belonging to the link. Mixed mode operation where only some lanes in a link contain scrambled data is not permitted.

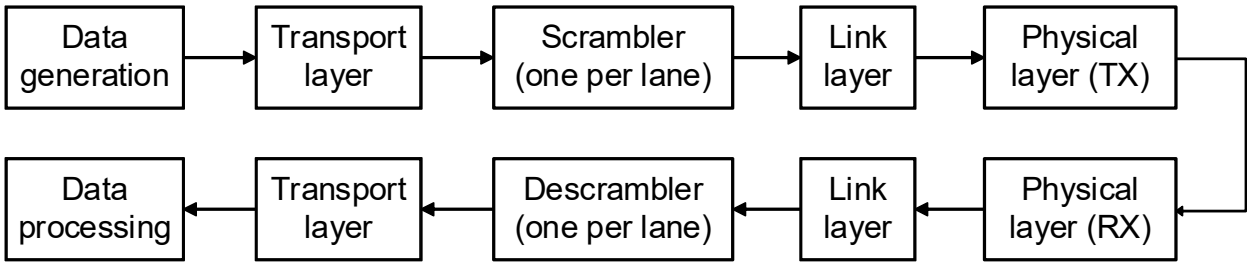


Figure 70 — Functional location of scrambler and descrambler

NOTE The main purpose of scrambling is to avoid the spectral peaks that would be produced when the same data octet repeats from frame to frame. Spectral peaks can cause electromagnetic compatibility or interference problems in sensitive applications. Via aliasing, they also cause code-dependent DC offsets in the data converters. Another advantage of scrambling is that it makes the spectrum data-independent, so that possible frequency-selective effects on the electrical interface will not cause data-dependent errors. The use of scrambled data may also be necessary for successful adaptation of a decision feedback equalizer (DFE) in the receiver. However, all digital operations in converters (including scrambling) cause some amount of switching noise, so there may be applications where it is of advantage to disable the scrambling.

8.3.2 Scrambler polynomial

The scrambler polynomial shall be

$$1 - x^{14} + x^{15}$$

The period of this polynomial is long enough (32,767 bits) to meet the spectral requirements of sensitive radio applications while allowing the descrambler to self-synchronize in two octets.

weiaotian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.3.3 Scrambler bit order

The scrambler and descrambler are defined via their serial implementations, processing the transmitted / received data frame by frame. The leftmost bit of the frame is shifted in first, as illustrated in Figure 71. The actual implementation shall produce the same result as the serial definition.

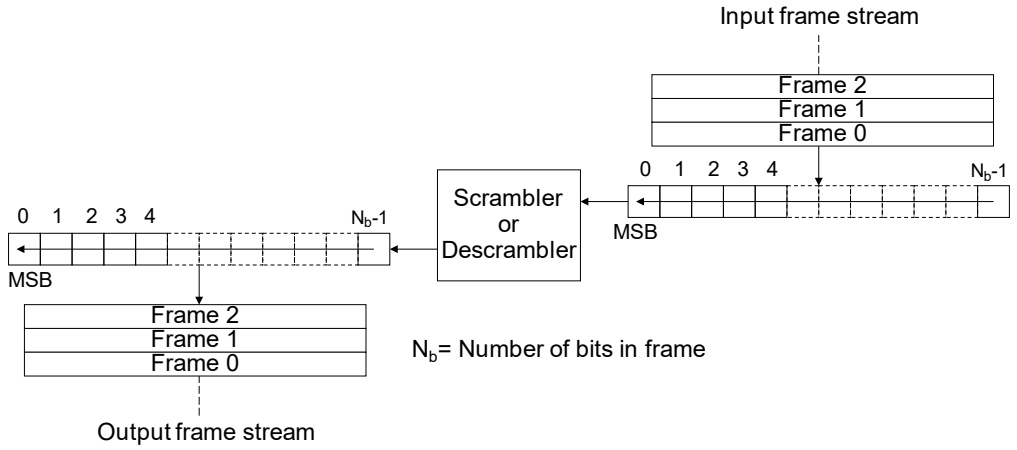


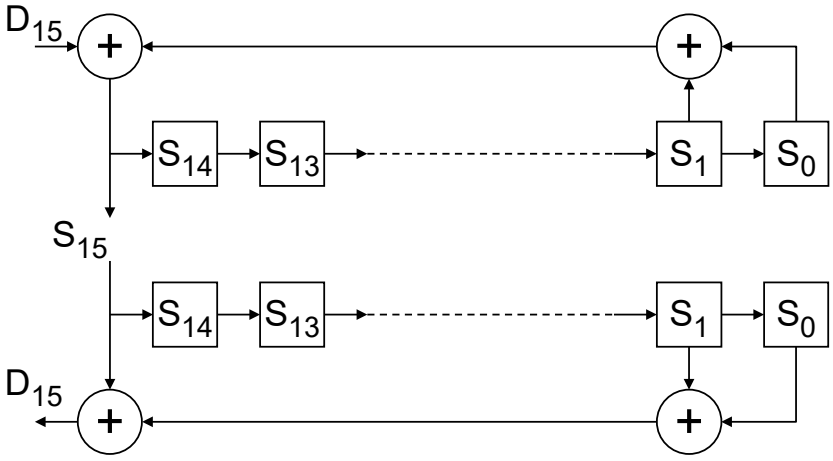
Figure 71 — Serial scrambling

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.3.4 Scrambler type

The scrambler shall be of the self-synchronous type. The serial implementation and the equations for parallel implementation are shown in Figure 72. An alternative scrambler is defined in 8.3.5 and Figure 73. Parallel implementation examples for scramblers and descramblers are shown in Annex O.



$$D = D_0, D_1, D_2, \cdots = unscrambled\ data\ in \quad (MSB\ first)$$

$$S = S_0, S_1, S_2, \cdots = scrambler\ state = scrambled\ data\ out$$

$$serial\ update: S_{15} = D_{15} + S_1 + S_0\ ;\ D_{15} = S_{15} + S_1 + S_0$$

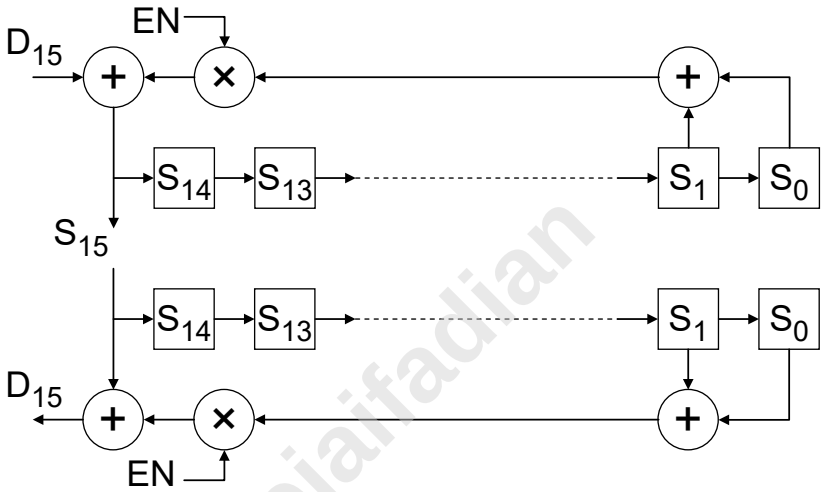
$$parallel\ update: \begin{cases} S_{31} = D_{31} + S_{17} + S_{16}\ ;\ D_{31} = S_{31} + S_{17} + S_{16} & (LSB\text{-}16b) \\ S_{30} = D_{30} + S_{16} + S_{15}\ ;\ D_{30} = S_{30} + S_{16} + S_{15} & \\ \cdots & \\ S_{23} = D_{23} + S_9 + S_8\ ;\ D_{23} = S_{23} + S_9 + S_8 & (LSB\text{-}8b) \\ S_{22} = D_{22} + S_8 + S_7\ ;\ D_{22} = S_{22} + S_8 + S_7 & \\ \cdots & \\ S_{17} = D_{17} + S_3 + S_2\ ;\ D_{17} = S_{17} + S_3 + S_2 & \\ S_{16} = D_{16} + S_2 + S_1\ ;\ D_{16} = S_{16} + S_2 + S_1 & (MSB) \end{cases}$$

Figure 72 — Serial scrambler and descrambler implementation and equations for parallel implementation

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.3.5 Early synchronization option

If used, scrambling is enabled at the start of user data. A scrambler implemented per Figure 72 would be bypassed during code group synchronization and transmission of an initial lane alignment sequence. After enabling the scrambler, two octets must be received before the state registers in the scrambler and descrambler have synchronized and the descrambler starts producing correct data. To avoid loss of user data at start up, an alternative scrambler may be implemented, in which unscrambled octets are also flowing through the state registers. The selection between scrambled and unscrambled operation is made using an enable signal to the scrambler logic. At the receiver, the descrambler input can always be connected to the 8B/10B decoder output, while the selection between original and descrambled data is made at the descrambler output. The serial implementation and the equations for parallel implementation for the modified scrambler are shown in Figure 73. A parallel implementation example for the alternative scrambler is shown in Figure O.117.



weiaifadian

$D = D_0, D_1, D_2, \cdots = unscrambled\ data\ in \quad (MSB\ first)$

$S = S_0, S_1, S_2, \cdots = scrambler\ state = scrambled\ data\ out$

$EN = Enable\ scrambling\ /\ descrambling$

$$serial\ update: S_{15} = D_{15} + EN \cdot (S_1 + S_0);\ D_{15} = S_{15} + EN \cdot (S_1 + S_0)$$

$$parallel\ update: \begin{cases} S_{31} = D_{31} + EN \cdot (S_{17} + S_{16});\ D_{31} = S_{31} + EN \cdot (S_{17} + S_{16}) & (LSB\text{-}16b) \\ S_{30} = D_{30} + EN \cdot (S_{16} + S_{15});\ D_{30} = S_{30} + EN \cdot (S_{16} + S_{15}) & \\ \ldots & \\ S_{23} = D_{23} + EN \cdot (S_9 + S_8);\ D_{23} = S_{23} + EN \cdot (S_9 + S_8) & (LSB\text{-}8b) \\ S_{22} = D_{22} + EN \cdot (S_8 + S_7);\ D_{22} = S_{22} + EN \cdot (S_8 + S_7) & \\ \ldots & \\ S_{17} = D_{17} + EN \cdot (S_3 + S_2);\ D_{17} = S_{17} + EN \cdot (S_3 + S_2) & \\ S_{16} = D_{16} + EN \cdot (S_2 + S_1);\ D_{16} = S_{16} + EN \cdot (S_2 + S_1) & (MSB) \end{cases}$$

Figure 73 — Alternative serial scrambler and descrambler implementation and equations for parallel implementation

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.3.6 Initial state

The scramblers described in 8.3.4 and 8.3.5, as well as many other commonly used self-synchronous scramblers, will produce repetitive output when the input data is a repeating copy of the initial state. Such repetitive output causes peaks in the spectral domain, which could lead to electromagnetic interference (EMI) or a maladaptation of the tap weights in a decision feedback equalizer (DFE). To minimize the occurrence of repetitive output, the scrambler must be initialized to a state that is unlikely to repeat continuously in the octet data produced by the transport layer. The recommended initial state is "1" for the eight storage elements with the highest indices and "0" for the seven remaining ones. No preset is needed in the descrambler, because it is self-synchronized. In the alternative scrambler defined in 8.3.5 there is also no need for preset, because the initial state at the start of scrambling will be determined by the last two unscrambled octets.

8.3.7 Scrambling disable

In some applications, the disadvantages of scrambling outweigh the benefits. In such cases, the converter manufacturer may at his discretion provide a means to disable the (de)scrambler. Consequently, the logic device shall have the option of disabling the (de)scrambling.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4 Link operation

8.4.1 Code group synchronization

Code group synchronization is achieved by the following process. Although described for multiple receivers and transmitters, the same process is applicable to a single receiver and transmitter:

- The receivers issue a synchronization request via the synchronization interface.
- The transmitters emit a stream of /K/= /K28.5/ symbols.
- The receivers synchronize, and then wait for the correct reception of at least four consecutive /K/symbols.
- The receivers deactivate the synchronization request in accordance with the guidelines outlined in 8.6.1.

The next steps in the process are dependent on the deterministic latency subclass of the transmitter device, see 8.5.1.

Subclass 0 transmitters shall implement the following behavior:

- Upon detecting that all receivers have deactivated their synchronization requests, the transmitters continue emitting /K/ symbols until the start of the next frame. See also 8.7.2.
- From the start of the next frame, the transmitters emit either an initial lane alignment sequence or encoded user data.

NOTE Per 8.4.5.3, sending an initial lane alignment is mandatory on multi-lane links and on single-lane links between subclass 1 or subclass 2 devices. The ability to start emitting an ILAS on any frame boundary implies that the subclass 0 TX must be able to shift the multiframe boundaries to any frame boundary. If the subclass 0 TX contains an LMFC, this means that either the multiframe boundaries can be offset from the active edges of the LMFC, or that the timing of the LMFC can be adjusted to match the frame in which the ILAS starts.

For Subclass 1 or 2 transmitters:

- Upon detecting that all receivers have deactivated their synchronization requests, the transmitters continue emitting /K/ symbols until the next multiframe boundary or a programmable later multiframe boundary. A minimum duration of the emission of /K/ symbols is necessary, see 8.7.2.
- On the first frame following the chosen multiframe boundary, the transmitters emit an initial lane alignment sequence.

www.erfadian
Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.1 Code group synchronization (cont'd)

The synchronization process is shown in the flow diagrams of Figure 89 and Figure 90. The 'SYNC' transitions indicate a change of state in the SYNC~ signal generated by the RX on the hard-wired SYNC interface (see 8.8.2) or a message over the soft SYNC interface (see 8.8.3). The 'DATA' transitions indicate a change of state in the data generated by the TX.

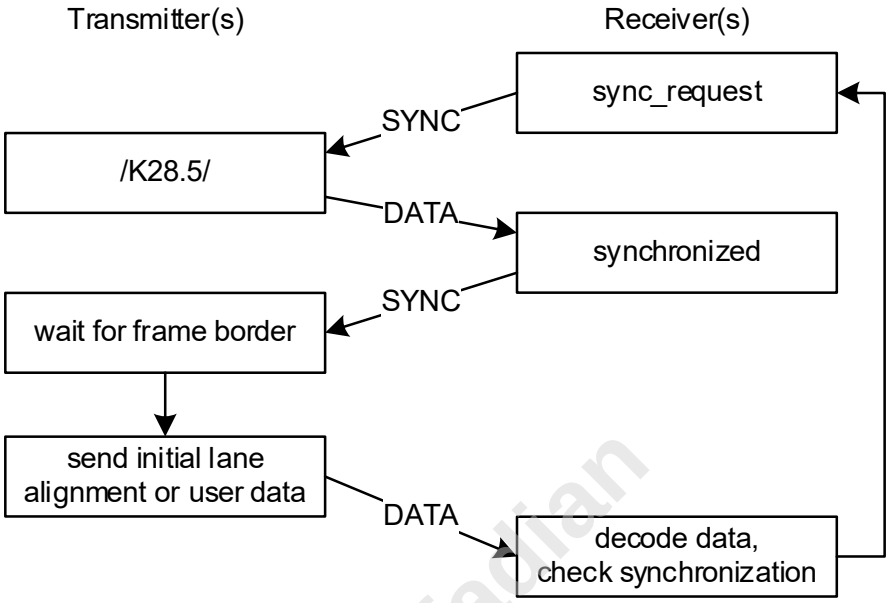


Figure 74 — Synchronization process for subclass 0

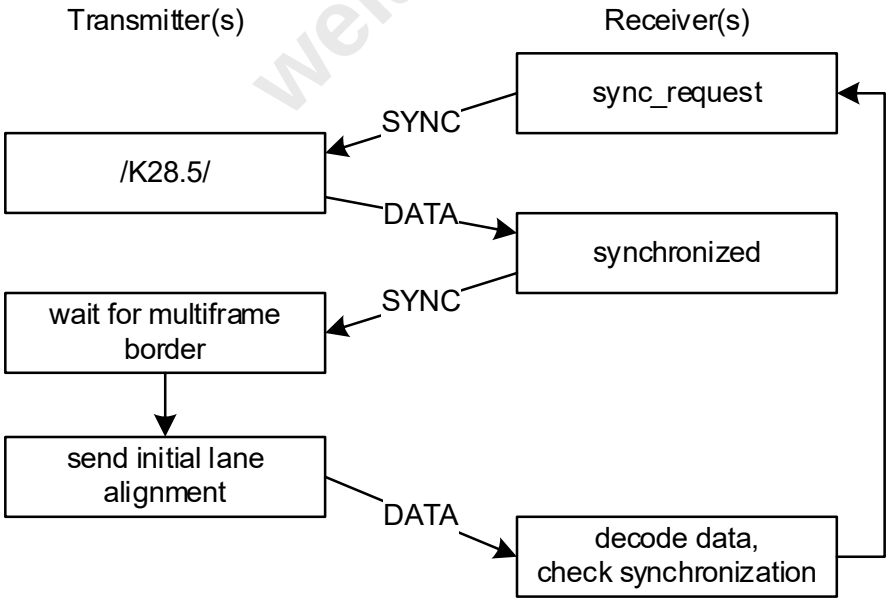


Figure 75 — Synchronization process for subclasses 1 and 2

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.2 Combination of synchronization requests

With multiple lanes, the synchronization requests of all receivers belonging to the same link are combined into one signal or message and presented simultaneously to the transmitter device. On multipoint links, it is permitted, but not mandatory to combine the synchronization requests of the individual links. The options for combining synchronization requests in multipoint links are outlined below:

- Inside a receiver logic device, the synchronization requests from each link in the logic device shall either be combined and distributed to all ADCs, or distributed to each ADC as individual per-link synchronization requests.
- Inside a transmit logic device, the synchronization requests from all DAC devices may either be 1) first decoded and then combined in the transmit logic device, or 2) treated as individual per-link synchronization requests.

Figure 76 provides an example of combination of synchronization requests on the hard-wired SYNC interface. Figure 77 provides an example of non-combined synchronization requests on the hard-wired SYNC interface.

With combined synchronization requests, as long as a single receiver requests code group synchronization, all transmitters connected to the multipoint link will send /K28.5/ symbols.

Without combined synchronization requests, only the specific link requesting code group synchronization will be affected.

On multipoint subclass 0 ADC links, combination of synchronization requests in the way of the left half of Figure 76 will approximately align the generation of the ILAS across the ADC devices. However, the total timing uncertainty in the detection of a deactivated synchronization request and the alignment of the ILAS to the moment of detection is likely to be larger than the uncertainty in the latency of a subclass 0 link (JESD204C, 7.8). Therefore, it is generally not recommended to align the received ILASs across the receivers in the logic device.

weakMedian

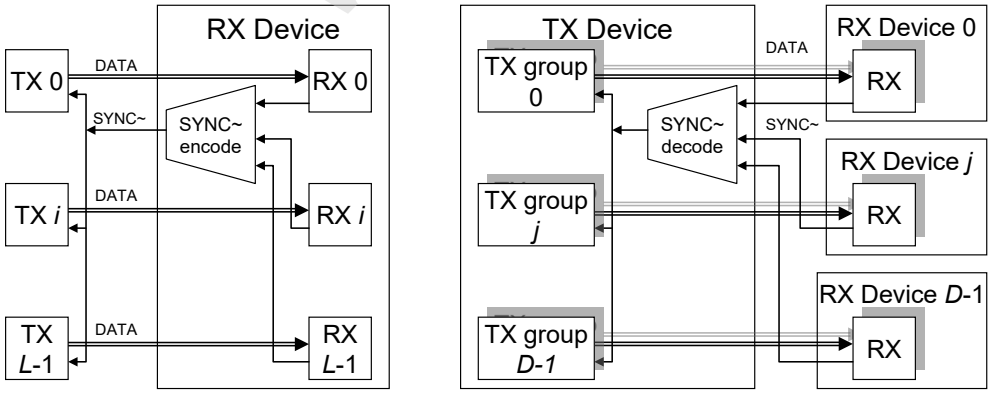


Figure 76 — Examples of SYNC~ signal combination

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.2 Combination of synchronization requests (cont'd)

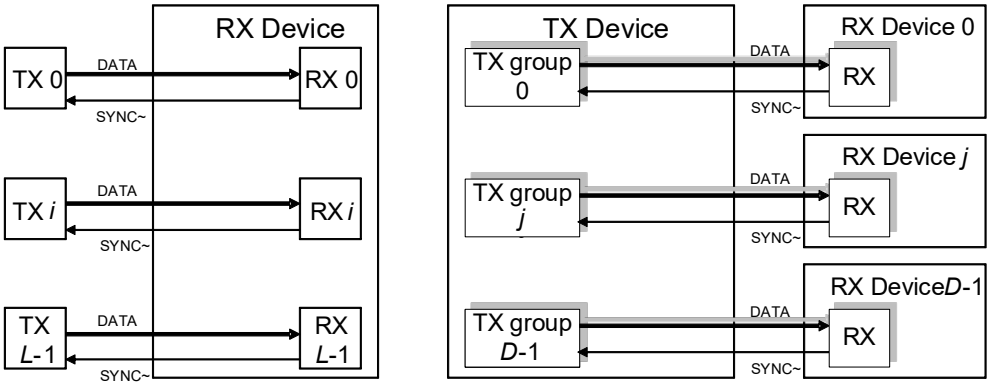


Figure 77 — Examples of non-combined SYNC~ signaling

8.4.3 Initial frame synchronization

At link startup, frame synchronization is achieved in the following way:

- During code group synchronization, the transmitter always sends full frames of /K28.5/ comma symbols.
- After code group synchronization, the receiver assumes that the first non-/K28.5/ symbol marks the start of a frame. If the transmitter emits an initial lane alignment sequence (see 8.4.5), the first non-/K28.5/ symbol will always be /K28.0/.
- The receiver assumes that a new frame starts every *F* octets.

8.4.4 Frame alignment monitoring and correction

8.4.4.1 Alignment characters

Frame alignment is monitored via alignment characters, which are inserted by the transmitter under certain conditions at the end of a frame. The receiver resynchronizes its frame to the alignment characters after checking that their reception is not likely to have been caused by a bit error on the lane. Resynchronization will require repeated reception of a valid alignment character at the same unexpected position in the frame.

The alignment character shall be a frame alignment character /F/= /K28.7/. However, if both sides of the lane support lane alignment, the lane alignment character /A/= /K28.3/ shall be used in the last frame of a multiframe. Multiframes are defined in 8.4.5.2.

NOTE /F/ is encoded from octet value 0xFC and /A/ from octet value 0x7C.

The character replacement depends on whether scrambling has been enabled or disabled and on whether lane alignment is supported. Support of lane alignment is required for all devices links except for subclass 0 single-lane DAC devices.

weiaifactian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.4.2 Character replacement without scrambling

On multilane links and on single-lane links between devices supporting lane alignment, character replacement in the transmitter and receiver during transmission of data from the transport layer (clause 0) shall be as follows:

- When the last octet in the current frame, not coinciding with the end of a multiframe, equals the last octet in the previous frame, the transmitter shall replace the current last octet and encode it as control character /F/= /K28.7/. However, if an alignment character was already transmitted in the previous frame, the original octet shall be encoded.
- When the last octet in the current frame at the end of a multiframe equals the last octet in the previous frame, the transmitter shall replace the current last octet and encode it as control character /A/= /K28.3/, even if a control character was already transmitted in the previous frame.
- Upon receiving an /F/ or /A/ symbol, the receiver shall replace it with the value of the octet decoded or used at the same position in the previous frame.

If at least one side of the lane does not support lane alignment (this support is not required in single-lane subclass 0 converter devices), character replacement in the transmitter and receiver during transmission of data from the transport layer (clause 0) shall be as follows:

- When the last octet in the current frame equals the last octet in the previous frame, the transmitter shall replace the current last octet with /K28.7/. However, if a /K28.7/ symbol was already transmitted in the previous frame, the actual octet shall be transmitted.
- Upon receiving a /K28.7/ symbol, the receiver shall replace it with the value of the data octet decoded at the same position in the previous frame.

NOTE The "last octet in a frame or multiframe" means the last octet in the frame or multiframe transmitted on a given lane, hence the character replacement functions in each lane are independent.

weiaiaaron

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.4.3 Character replacement with scrambling

On multilane links and on single-lane links between devices supporting lane alignment, character replacement in the transmitter and receiver during transmission of data from the transport layer (see clause 0) shall be as follows:

- When the last scrambled octet in a frame, but not at the end of a multiframe, equals 0xFC, the transmitter shall encode it as a control character /F/.
- When the last scrambled octet in a multiframe equals 0x7C, the transmitter shall encode it as a control character /A/.
- Upon receiving an /F/ or /A/ symbol, the receiver shall input the corresponding data octet 0xFC or 0x7C to the descrambler.

If at least one side of the lane does not support lane alignment (this support is not required in single-lane subclass 0 converter devices), character replacement in the transmitter and receiver during transmission of data from the transport layer (see clause 0) shall be as follows:

- When the last scrambled octet in a frame equals 0xFC, the transmitter shall encode it as a control character /F/.
- Upon receiving an /F/ symbol, the receiver shall input the corresponding data octet 0xFC to the descrambler.

NOTE The "last octet in a frame or multiframe" means the last octet in the frame multiframe transmitted on a given lane, hence the character replacement functions in each lane are independent.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.4.4 Frame alignment correction in the RX

When enabled, the alignment correction shall be carried out in the following way:

- If two successive valid alignment characters are detected at the same position in the frame, other than the assumed end of the frame, without receiving a (valid or invalid) alignment character at the expected position between the two alignment characters, the receiver will realign its frame to the position of the received alignment characters.
- The receiver shall have an option to disable the realignment described above, because without scrambling, certain types of periodic data may not produce enough alignment characters for reliable detection of frame misalignment, see NOTE 1. Frame realignment can also be undesirable because it can cause an error in the lane alignment or latency, see NOTE 2.

NOTE 1 When scrambling is enabled, on the average one out of 256 frames will end in an alignment character. Without scrambling, the frequency of alignment characters depends on the sampled data and the mapping of the samples to the frame. Most practical signals will map to random or quasi random octets at the end of the frame and generate alignment characters with a frequency of about one per 256 frames. However, noise-free periodic signals with a harmonic frequency relation to the sample rate of the ADC may not generate alignment characters at all for certain values of initial phase. Usually this problem is corrected with a small amount of random noise on the signal. There may however be singular cases in which alignment monitoring cannot be performed reliably in the non-scrambled mode.

NOTE 2 Frame alignment correction is in the first place a shift of the markers indicating the frame borders in the data stream. However, to align these markers to a local timing reference, in addition an adjustment to an elastic buffer is necessary. The direction of this adjustment does not matter for the correction of the frame alignment error. However, if the wrong direction is chosen, an error in the lane alignment and lane delay will be created. Unless the frame is sufficiently long compared to the maximum possible alignment error, it is not possible to decide the correct direction only from the position of the frame alignment character.

8.4.5 Initial lane alignment

8.4.5.1 General principles

The initial lane alignment is carried out before the start of user payload data. The initial lane alignment procedure follows the principles of other standards such as the 10 Gigabit Ethernet [7] subclause 48.2. At a well-defined point in time, all the transmitters issue a dedicated lane alignment character /A/= /K28.3/. Due to different lane delays, these alignment characters may be received at different times by the receivers. With the reception of an /A/, each receiver stores subsequent data in a buffer memory and indicates a flag (‘ready’) to the other receivers, indicating that the buffer contains a valid alignment start point. When all receivers have raised their “alignment received” flags, they start propagating the received data to subsequent data processing logic/function at the same point in time, where alignment is based on a common signal (‘start’).

For JESD204 devices supporting deterministic latency, this ‘start’ signal is the moment of RX buffer release, as defined in 4.2.1. The specification also permits this alignment procedure at a specified time after the reception of the alignment character in a main receiver, with the other receivers issuing an error signal if they do not find the alignment character in their own buffer at the same time.

weiaipian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.5.2 Multiframes

The 8B/10B link layer uses multiframes for the following purposes:

- to enable lane alignment monitoring, by comparing the positions of the multiframes on different lanes;
- to enable monitoring and adjustment of the latency on the link;
- to hold information in the initial lane alignment sequence (see 8.4.5.3).

A multiframe is defined as a group of K successive frames, where K is between 1 and 256 and such that the number of octets per multiframe is between 17 and 1024:

$\text{ceil}(17/F) \leq K \leq \min(256, \text{floor}(1024/F))$

In JESD204 transmitter devices, the value of K shall be programmable. In JESD204 receiver devices, the value of K is recommended to be programmable. JESD204 receiver devices shall clearly specify their requirements or recommendations for the setting of the factor K in the transmitter device(s).

NOTE 1 The data in the initial lane alignment sequence requires at least 17 octets per multiframe. The clocking implementation or the receiver capability to unambiguously decide the direction of the latency adjustment will generally require a longer multiframe.

NOTE 2 The use of K values above 32 is not backwards compatible with JESD204A and JESD204B.

8.4.5.3 Initial lane alignment sequence

Initial lane synchronization is achieved by means of an initial lane alignment sequence, starting immediately after code group synchronization. All logic device transmitters, all subclass 1 and 2 ADC devices as well as all multi-lane subclass 0 ADC devices shall be capable of transmitting this sequence. The transmission of this sequence is mandatory on all multi-lane links and on all single-lane links between subclass 1 or subclass 2 devices. It shall be possible to skip this sequence on a link to a single-lane subclass 0 DAC device.

The initial lane alignment sequence consists of exactly four multiframes and shall never be scrambled.

The structure of the initial lane alignment sequence is illustrated in Figure 78. Each multiframe starts with an /R/= /K28.0/ and ends with an /A/. The /R/ is an indication to the receiver that the multiframe is part of an initial lane alignment sequence. The /A/ marks the end of the multiframe and is used for both lane and frame synchronization. The second multiframe shall contain configuration information about the JESD204 link from the transmitter to the receiver, starting from the third symbol. The fixed transmission code of /K28.4/ in the second symbol is an extra confirmation to the receiver that configuration data is going to start.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.5.3 Initial lane alignment sequence (cont'd)

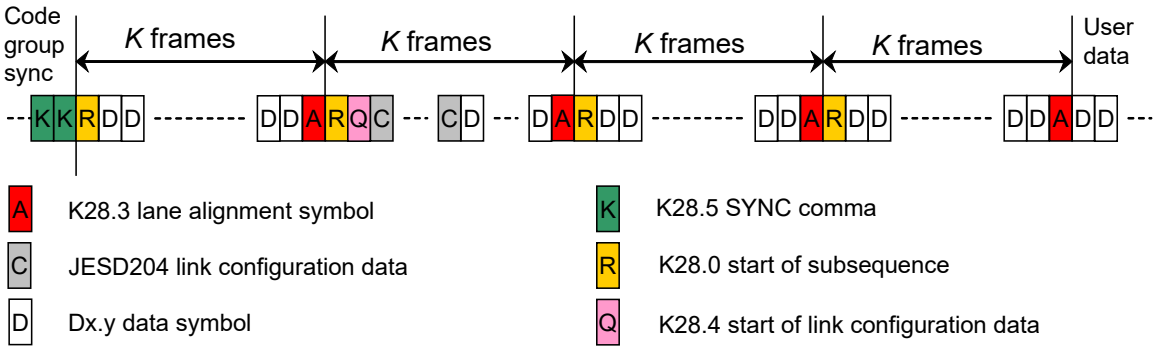


Figure 78 — Initial lane alignment sequence with four multiframes

The data octets 'D' are specified by means of the flow diagram of Figure 79. Note that this diagram defines the octets at the input of the 8B/10B encoder.

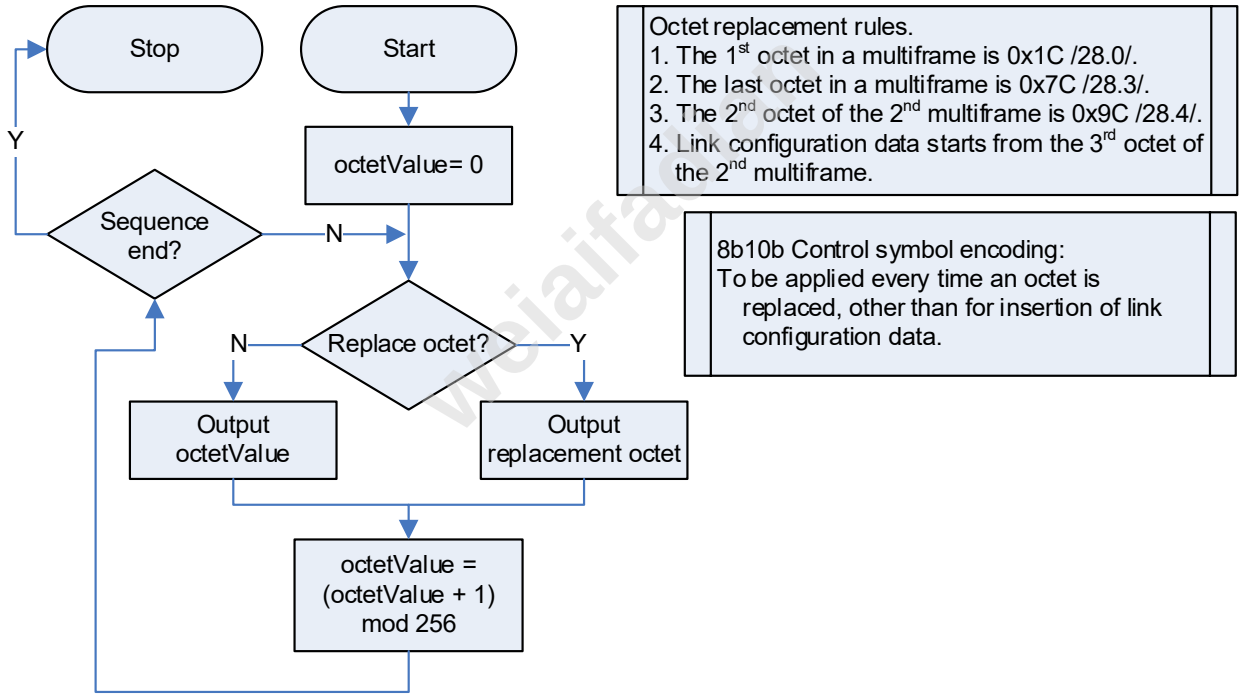


Figure 79 — Character specification for initial lane alignment sequence

The format of the configuration data is defined in clause 8.9.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.6 Lane alignment monitoring and correction

After initial frame and lane alignment, the channel switches to the alignment monitoring mode. Lane alignment is monitored via /A/= /K28.3/ characters, which are inserted by the transmitter under certain conditions at the end of a multiframe. The insertion is further specified under the frame alignment monitoring, see 8.4.4. Except for single-lane subclass 0 ADC devices, all transmitters shall be able to insert /A/ characters. All logic device receivers, all subclass 1 and 2 DAC devices as well as all multilane subclass 0 DAC devices shall be able to detect and replace /A/ characters (see 8.4.4). The transmitter shall not insert /A/ characters if the receiver is not able to detect and replace /A/ characters.

In general, not all lanes will transmit an /A/ simultaneously. However, each receiver separately can check the arrival of an /A/ against a local timing reference and make necessary corrections in alignment.

Each receiver in a multi-lane environment can be authorized from a higher-level application layer to perform a dynamic (on-the-fly) realignment when all the following three conditions are met:

- misalignment has been reliably detected;
- a reliable new alignment position has been found;
- realignment is possible based on the conditions of the data that is maintained in the alignment data buffer.

The authorization of realignment is given to the receiver via the control interface defined in 4.4. The realignment rules for lanes are similar to the rules for frames (see 8.4.4.4):

- If two successive, valid /A/ symbols are detected at the same position, other than at the assumed end of the multiframe, without receiving a (valid or invalid) /A/ symbol at the expected position between the two /A/ symbols, the receiver shall realign the multiframes to the position of the newly received /A/ symbols.
- If a recent frame realignment is the most likely cause of loss of lane alignment (see NOTE 2 in 8.4.4.4), the receiver shall realign the multiframes already to the position of the first received /A/ symbol at an unexpected position.

NOTE 1 Lane alignment correction is essentially a shift of the markers indicating the multiframe borders in the data stream. If the shift is a non-integer number of frames, also the markers indicating the frame borders must be shifted. To align the markers to a local timing reference and to align the markers on different lanes to each other, in addition adjustments to elastic buffers are necessary.

NOTE 2 Dynamic realignment will correctly restore the lane delay only if the local timing reference is correct. If lane misalignment has been detected because of a phase change in the local timing reference, dynamic realignment will align all lanes to the same reference, but the latency of the link will change. To avoid the possibility of a latency change, subclass 1 and 2 receivers can initiate an LMFC realignment via SYSREF (subclass 1) or SYNC~ (subclass 2) in addition to dynamic realignment. Receivers of all subclasses may also request link resynchronization as an alternative to dynamic realignment.

Weidradian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.7 Link reinitialization

Under certain error conditions (see 8.6.5.3), a receiver can request reinitialization of the link by asserting a synchronization request. If all receivers are within the same device, all synchronization requests can be made visible to all receivers in the device. This way device-internal synchronization requests can be used to reinitialize the frame and lane alignment in all receivers in the device.

In addition, receivers can see each other's synchronization requests from the received characters on the data interface. Because synchronization requests from all receivers on a link and optionally on a multipoint link are combined into a single synchronization request, which is presented simultaneously to all transmitters (see 8.4.2), all transmitters will start sending /K/= /K28.5/ symbols based on any synchronization request from any receiver.

A receiver shall, upon reception of a stream of /K/ symbols, return to the initial frame and lane alignment states as shown in Figure 86 (in 8.6.2). This behavior is mandatory in all devices, except for subclass 0 DAC devices. To guarantee reliable detection of the stream of /K/ symbols, a minimum duration this stream is specified in 8.7.2.1 for subclasses 1 and 2. In subclass 0, this minimum duration of the stream of /K/ characters is guaranteed by the minimum duration of the synchronization request over the hard-wired SYNC interface (see 8.8.2).

Transmitters with support for lane alignment (see 8.4.5.3) shall follow the stream of /K/ characters by an initial lane alignment sequence, unless the receiver does not support lane alignment. Transmitters without support for lane alignment may follow the stream of /K/ characters directly with converter data.

Without an active synchronization request from a receiver, a transmitter has the possibility to request reinitialization of the link by moving its state machine (Figure 90 in 8.7.1) to the SYNC state and issuing a stream of /K/ symbols as above.

NOTE In the RX, reinitialization over the data interface can be implemented outside the state machine for code group synchronization. However, the receiver will interpret a /K/ symbol during data transmission as an unexpected control character (see 8.6.5.1). Depending on how the error handling in the receiver has been configured, this may lead to an error report or synchronization request by the receiver.

8.4.8 Test modes

8.4.8.1 General

A link layer test mode is a state where a continuous sequence of predetermined 8B/10B characters is transmitted on all lanes on the multipoint link. A JESD204 system is put into a test mode via an external control interface (see 4.4).

NOTE Link layer test characters are specified as if injected directly to the 8B/10B encoder. They are never scrambled.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.4.8.2 Test sequences

All TX devices shall be able to transmit the following test sequences:

- a continuous sequence of /K28.7/ characters (low frequency pattern, useful for characterization of rise- and fall times;
- a continuous sequence of /K28.5/ characters (mixed frequency pattern or code group synchronization);
- repeated transmission of the initial lane alignment sequence, with sensitivity to synchronization requests from the receiver. If this test mode is enabled in the transmitter, the transmitter shall initiate code group synchronization whenever the receiver issues a synchronization request. Upon completion of code group synchronization, the transmitter shall repeatedly transmit the ILAS. If there is no active code group synchronization request when the transmitter enters the test mode, the transmitter shall behave as if it received one. When entering the test mode, the transmitter shall transmit at least four /K28.5/ symbols in succession as part of the code group synchronization sequence. This test sequence is not required in single-lane subclass 0 ADC devices.
- At least one of the following two patterns:
  - a continuous sequence of a modified random pattern (modified RPAT);
  - a continuous sequence of a scrambled jitter pattern (JSPAT).

The 12-octet modified RPAT pattern is fully defined in INCITS TR-35-2004 (reference 6). It is also defined in annex 48A of IEEE Std. 802.3 (reference 3), but only as a sequence of octets. The first octet of the modified RPAT patterns shall be encoded with positive running disparity. The 50-octet JSPAT pattern is defined in INCITS 450-2009 (reference 7) and in INCITS TR-46-2011 (reference 8). JSPAT has a negative starting and ending disparity.

Optionally, JESD204 transmitter devices may support other standardized link layer test patterns to be injected before the 8B/10B encoder, such as:

- the high frequency pattern /D21.5/, or
- a pseudo-random bit sequence (PRBS).

NOTE A PRBS15-like sequence will be generated in the short and long transport layer tests when scrambling is enabled, see 8.3.

All RX devices shall be able to verify the first of the following sequences. The second sequence shall be supported by all devices other than single-lane subclass 0 DAC devices (see 4.2.2 and 8.4.5.3):

- a continuous sequence of /K28.5/ characters for code group synchronization;
- a code group synchronization sequence, followed by repeated transmission of a lane alignment sequence.

In addition, RX devices shall have the capability to suppress error reports due to a missing ILAS. This makes it possible to perform BER measurements using 8B/10B encoded test patterns from a standard pattern generator, after initial synchronization with a /K28.5/ sequence. Most of the bit errors will lead to a running disparity or not-in-table error, which will be indicated by the RX with an error report on the SYNC~ interface. Using the SYNC~ as a bit error indicator will be sufficiently accurate for BER testing. BER compliance testing shall be performed with a repetitive JTSPAT sequence, which is defined for instance in INCITS 450-2009 (Reference 7).

NOTE The RX is not required to verify other sequences than /K28.5/ and ILAS. Errors in other sequences can be detected via the regular error detection functionality in the 8B/10B decoder.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5 Deterministic latency using SYNC~ detection (device subclass 2)

8.5.1 Introduction

In general, JESD204 achieves deterministic latency in two steps:

1. align local reference clocks across the JESD204 devices;
2. align the data to these local reference clocks.

In case of the 8B/10B link layer, the local reference clocks used for alignment of the data are called local multiframe clocks or LMFCs (see 4.3.4). The method for aligning the LMFCs depends on the device subclass. Subclass 2 uses the SYNC~ signal (see 8.8.2) to estimate the timing difference between the LMFCs in the logic device and the converter device(s). In subclass 2, the misalignment between LMFCs is minimized by adjusting the timing of the LMFC in the converter device(s). The method for obtaining the LMFC alignment is specified in the following subclauses. The method for obtaining data alignment and general requirements for obtaining data alignment are independent of the device subclass and are specified in 4.2.1.

8.5.2 Accuracy limitations

The SYNC~ signal carries the phase of the LMFC in the RX device, see 8.8.2. The SYNC~ signal must be accurately captured by the TX device to determine the relative alignment of TX and RX multiframe periods. With knowledge of the relative phase difference between RX and TX multiframe periods, correct alignment must be achieved by:

- aligning a TX converter device to the RX logic device; or
- aligning an RX converter device to the TX logic device.

The accuracy of the deterministic latency in subclass 2 is based on the accuracy of the transfer and the sampling of the SYNC~ de-assertion on the hard-wired SYNC interface. At both the RX and TX ends of the link, the devices will introduce a variable amount of delay to the launch and capture phase of SYNC~ respectively. The extent to which the variability of the delay can be controlled will limit the maximum speed at which deterministic latency can be achieved in subclass 2.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.2 Accuracy limitations (cont'd)

Figure 80 gives an idea of the variable delays in the system and the effects they have on the available timing margin. The figure shows representative timing effects when SYNC~ is detected with an internal clock in the link layer. For simplicity, the figure is shown for the case of an internal clock at the device clock rate. As can be seen, if SYNC~ is either launched from or detected by the device clock, less PVT variation is incurred than if the launch or detection comes directly from an internal adjustment or detection clock. All possible means to minimize variation of delay due to PVT in SYNC~ launch and detection must be used at both ends of the link to maximize speed of operation. The variable delays are specified by the manufacturer through the parameters outlined above Figure 91 in 8.8.2. The maximum speed of operation is achieved when the excess timing margins labeled "PVT margin" in Figure 80 reach zero.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.2 Accuracy limitations (cont'd)

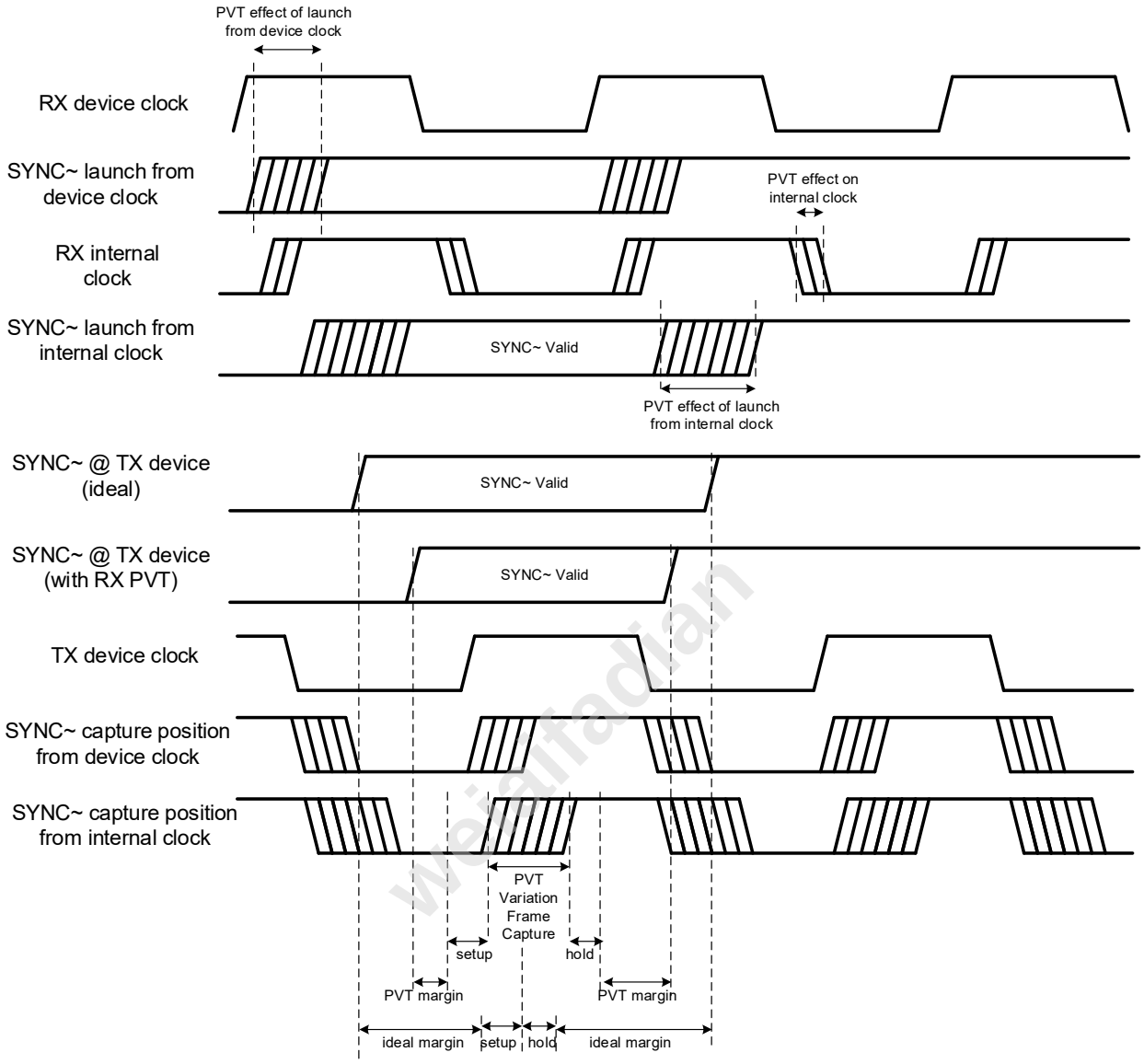


Figure 80 — PVT effects on SYNC~ detection and generation

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.3 Principles of SYNC~ sampling

8.5.3.1 General

The following section outlines concepts that are important to understand when developing a subclass 2 deterministic latency system.

8.5.3.2 SYNC~ generation at the RX device

The SYNC~ signal from the RX shall be generated from a SYNC~ generation clock that is phase aligned to the LMFC and has, depending on the data rate, a period of no more than a specified number of 8B/10B character durations (see 4.3.7 and 8.8.2). If the multiframe duration is a whole number of link layer clock periods, the link layer clock can be used as SYNC~ generation clock. The de-assertion of the SYNC~ signal shall be aligned to the active edge of the LMFC. The de-assertion carries the phase information of the RX LMFC to the TX device via the SYNC~ interface. Generally, the SYNC~ generation clock will be used to directly launch SYNC~ to the TX device. When the ratios between the clock frequencies of a device allow it, the SYNC~ should be realigned to the device clock before launch.

8.5.3.3 Adjustment resolution and adjustment clock

In converter devices, the timing of the LMFC is adjusted in increments of the adjustment resolution. The adjustment resolution is defined as the minimum time-step in which the LMFC can be adjusted. The adjustment is specified as the alignment of the converter LMFC to an active edge of an adjustment clock. This adjustment clock has the following properties:

- The adjustment clock period is equal to the adjustment resolution of the link.
- The adjustment clock is phase aligned to the device clock for purposes of timing measurement.

The adjustment clock will generally be a dedicated clock that is derived from the device clock, in which case it shall have a programmable frequency relation to the device clock. In special cases it may be possible to use the converter device clock or link layer clock as an adjustment clock.

NOTE Only in cases where the adjustment resolution is a whole multiple of the converter device clock period, the converter LMFC can be aligned to the device clock at each adjustment point.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.3.4 Detection resolution at the TX device

In TX devices, the de-assertion of SYNC~ is detected at the SYNC~ decoder. As the de-assertion of SYNC~ carries the phase of the RX LMFC, the phase is detected and acted upon. The detection of the de-assertion is performed based on a detection clock (see 4.3.7 and 8.8.2), with period equal to the detection resolution of the link. The accuracy of this subclass's latency relies on the relationship that the detection resolution is finer than or equal to the adjustment resolution. There is a possibility that the detection resolution of a TX logic device may differ from the adjustment resolution of its links. To develop fine detection resolution from a slower device clock, it may be necessary to develop a higher speed detection clock inside the TX device.

8.5.3.5 SYNC~ de-assertion detection and the detection interval

There is a possibility that the adjustment resolution of the link will be coarser than the detection resolution. Due to this, there may be multiple detection points within one adjustment step. If this is the case, an interval in detection resolution steps can be defined to equal the adjustment resolution. The detection interval now will allow the TX device to detect based on the converter's ability to adjust itself. A detection interval has the following properties:

- The detection interval exists only when the adjustment resolution of the link is coarser than the detection resolution.
- The detection interval is set equal to the adjustment resolution of the system.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.3.5 SYNC~ de-assertion detection and the detection interval (cont'd)

Figure 81 shows how a detection interval would function for a DAC link. In this example, the logic device samples SYNC~ de-assertion with its link layer clock. The vertical arrows indicate valid detection positions for interval '1'. When SYNC~ is sampled at either position, it will be considered detected within interval '1'.

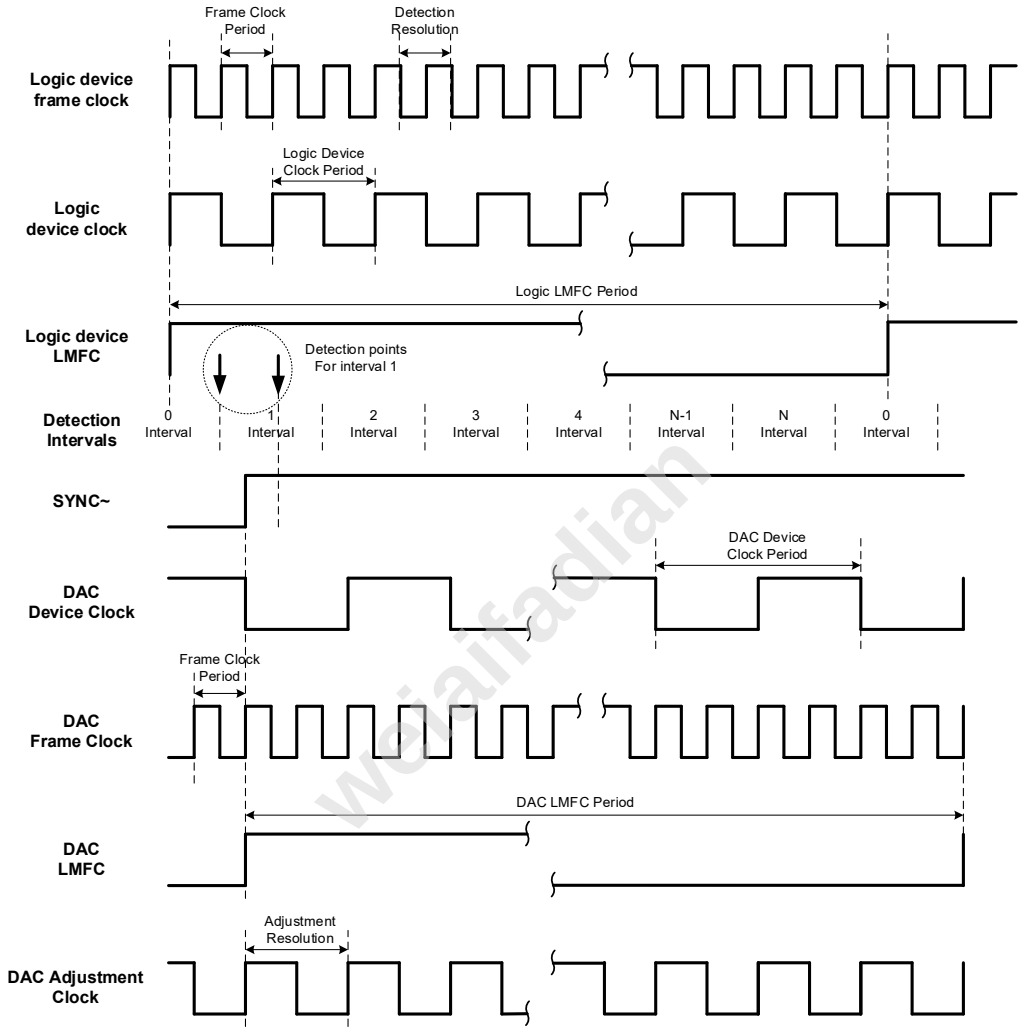


Figure 81 — Relationship of detection interval to system signals (DAC link)

8.5.4 Initiator and target configurations

8.5.4.1 Introduction

Subclass 2 is a initiator and target configuration between two devices. In multiple converter configurations, the logic device is the only device in the system that can share a common LMFC across all links. For this reason, the LMFC of the logic device shall be used as the main timing reference of the link and the converter will always synchronice from the main reference. The operation of the main and secondary configurations for ADC devices and DAC devices is outlined in 8.5.4.2 and 8.5.4.3, respectively.

weijaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.4.2 ADC initiator and target configurations

In a subclass 2 ADC device, the de-assertion of SYNC~ input is detected based on the ADC detection resolution. The LMFC phase reset must occur a deterministic number of adjustment clock periods following the detection of SYNC~ de-assertion. This delay shall be defined by the device manufacturer. The ADC can perform LMFC phase alignment operations based on one-shot or repetitive SYNC~ de-assertions. The usage of SYNC~ for latency monitoring is left to the implementer. Examples 1 and 2 give a few implementation possibilities.

NOTE A phase adjustment of the LMFC may also require a phase adjustment of a higher frequency clock from which the LMFC is generated.

EXAMPLE 1 Synchronization request based SYNC~ usage

In this usage model, the ADC will assume that each synchronization request from the RX logic device also be considered a request for phase reset. Upon initial ADC link startup, the RX logic device will de-assert the SYNC~ after code group synchronization. The ADC will reset its LMFC and, if necessary for this purpose, other internal clocks to that SYNC~ de-assertion location. The ADC will issue at least a minimum duration of /K28.5/ characters on its data lanes to reinitialize the link (see 8.4.7) if a clock phase adjustment is made. It will continue to send /K28.5/ characters until internal timing adjustment has settled. After this point, on the next active LMFC edge in the ADC, the ADC will send a new ILAS followed by resumption of data transmission. This mode can may also be used if deterministic latency accuracy is not particularly critical and SYNC~ detection can violate setup/hold. Figure 82 shows a timing diagram of the procedure outlined above.

EXAMPLE 2 All event based SYNC~ usage

In this usage model, the SYNC~ de-assertion may be used at each synchronization request or error report. Both events will be considered equally. Upon the first de-assertion seen by the ADC, it will reset its phases as outlined above in Example 1. Subsequent SYNC~ de-assertions can be used to either reset the phases or to monitor the latency of the link. If constant reset of the link is required, it is important that SYNC~ de-assertion detection meet proper setup/hold requirements to prevent unnecessary phase adjustment during each reset operation. Each detection of a SYNC~ de-assertion will reset with behavior similar to the “one shot SYNC~ usage” mode. Subsequent de-assertions can alternatively be used to monitor whether the alignment of the TX and RX LMFCs is correct. This can be done by simply detecting each de-assertion and comparing its phase to the current expected phase for SYNC~ de-assertion. If the phases are different, then latency has been lost. In the event that the detection resolution is finer than the time period required to meet the total timing margin the SYNC interface, a programmable “window” of expected phases can be used for this comparison (see 8.5.3.5). Finally, if the link latency has been lost, the ADC device can always reset its LMFC from a future SYNC~ de-assertion by either waiting for an error report or forcing one via a reinitialization over the data interface (see 8.4.7).

NOTE The TX initiates link reinitialization by sending a stream of /K/ characters. The receiver will interpret a /K/ character during data transmission as an unexpected control character (see 8.6.5.1). Normally a logic device is not required to generate an error report over the SYNC interface, but in the latter usage model the logic device must be configured to do so.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.4.2 ADC initiator and target configurations (cont'd)

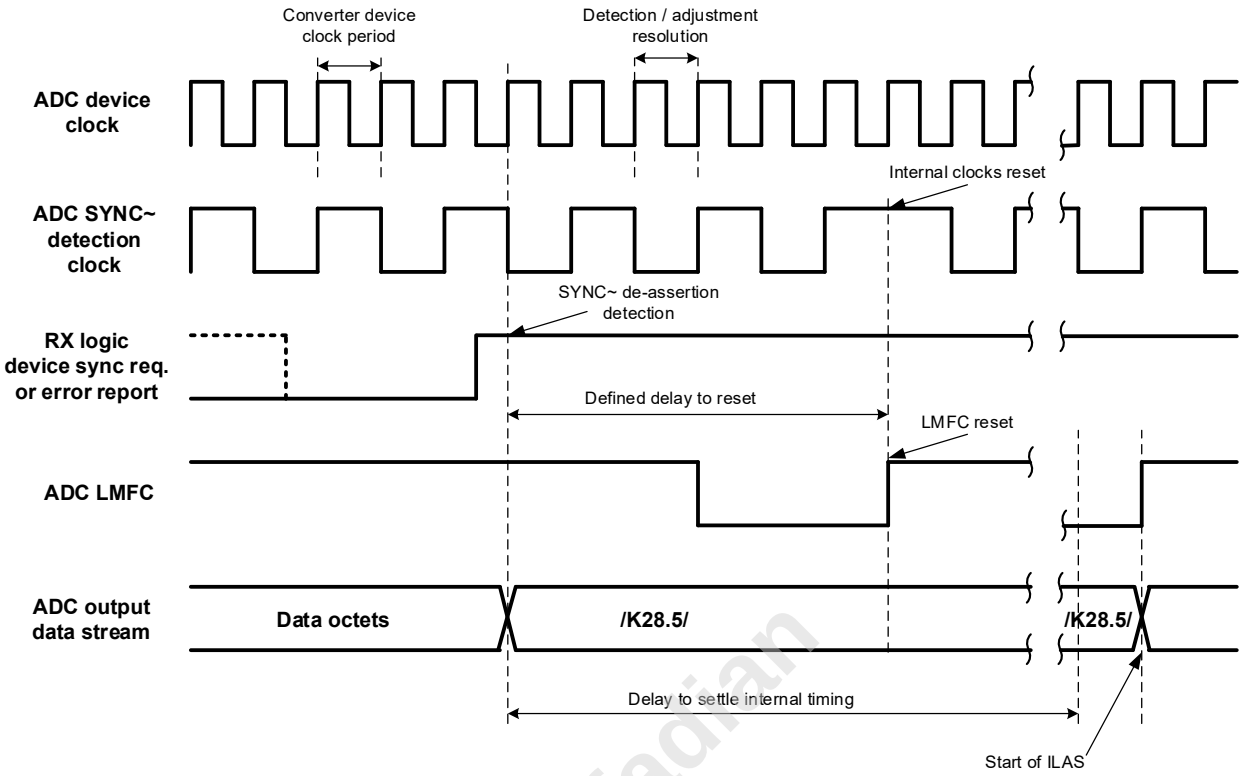


Figure 82 — ADC reset as a result of SYNC~ de-assertion detection

8.5.4.3 DAC initiator and target configurations

A subclass 2 DAC device must support adjustment of the phase of its LMFC and, if necessary for this purpose, other internal clocks in response to requests by a logic device. At the DAC, de-assertion of the SYNC~ signal will always be launched at the active LMFC edge. The SYNC~ de-assertion for each individual DAC will be detected, to the accuracy of the detection resolution, at the TX logic device. The detection phase will be compared to the LMFC phase in the TX logic device. If adjustment of the DAC LMFC phase is required, the logic device will signal the adjustment to the DAC by embedding the request, PHADJ, an adjustment step count, ADJCNT, and an adjustment direction, ADJDIR, in configuration bytes 1 and 2 during an ILAS. Table 65 defines the locations of PHADJ, ADJCNT, and ADJDIR in the configuration data for a DAC link.

If the need to adjust the LMFC was detected from a synchronization request, the ILAS will occur as part of initial lane synchronization (see 8.4.5.3). Otherwise the logic device logic device will provide the ILAS as part of a reinitialization over the data interface (see 8.4.7). If SYNC~ combining is used in the system, SYNC~ de-assertion detection must take place prior to SYNC~ combination, allowing for individual phase adjustment of each DAC device.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.4.3 DAC initiator and target configurations (cont'd)

One bit, PHADJ, is defined for the purposes of adjustment request. ADJCNT<3:0> is defined as the adjustment resolution step count. ADJDIR is defined as the direction of adjustment. ADJDIR equal to 1 is a delay while ADJDIR equal to 0 is an advance. When the DAC sees PHADJ set while processing an ILAS, it shall respond by adjusting its LMFC by the number of adjustment resolution steps sent in ADJCNT<3:0> and in a direction defined by ADJDIR (see NOTE 1). After adjustment is done, the DAC shall issue an error report to the TX logic device with the new LMFC phase information as an acknowledgement of adjustment request (see NOTE 2). The use of PHADJ, ADJCNT, and ADJDIR enables an iterative adjustment of the DAC LMFC until which point it is phase aligned to the TX logic device LMFC.

NOTE 1 Not all DACs will provide the full ability to adjust up to 15 steps. The number of steps that can be adjusted in any DAC must be defined and, if less than 15, will be LSB justified in ADJCNT<3:0>.

NOTE 2 To prevent invalid LMFC phase information passing from DAC to TX logic device, the reporting of reception errors must be suspended during loop operation.

Figure 83 shows an example state diagram used by the TX logic device to handle alignment of the LMFC of a subclass 2 DAC. During the detect SYNC~ de-assertion state, the TX logic device samples SYNC~ continually using its detection clock. When a de-assertion is detected, the TX logic device determines whether an adjustment is needed relative to its LMFC. An adjustment is needed if a discrepancy exists between the active LMFC edge and the SYNC~ detection based on either direct detection resolution or detection interval. If an adjustment is not required and the SYNC~ de-assertion did not happen as a result of a synchronization request from the DAC, an adjustment need not be requested, and the TX logic device goes back to sampling the SYNC~. If either an adjustment is required or the SYNC~ de-assertion happened as a result of a synchronization request by the DAC, then a link initialization will be performed as follows. If an adjustment is required, it will be requested by setting PHADJ high, setting ADJCNT<3:0> to the number of adjustment steps requested, and ADJDIR to the direction requested.

If no adjustment is required and the SYNC~ de-assertion is the result of a synchronization request from the DAC, PHADJ will be set to zero. After these parameters are set, the TX logic device will send a stream of /K/ characters (see 8.4.1) and an ILAS, in which the three parameters above will be embedded (see 8.9). The checksum for the configuration data will need to be recalculated to take into account the new values for configuration bytes 1 and 2. Once ILAS has been transmitted, the TX will return to sampling SYNC~ until a new de-assertion is detected. The DAC will acknowledge the phase change with an error report containing the new DAC LMFC phase information. This error report will allow the loop to iterate.

Verification Only

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.4.3 DAC initiator and target configurations (cont'd)

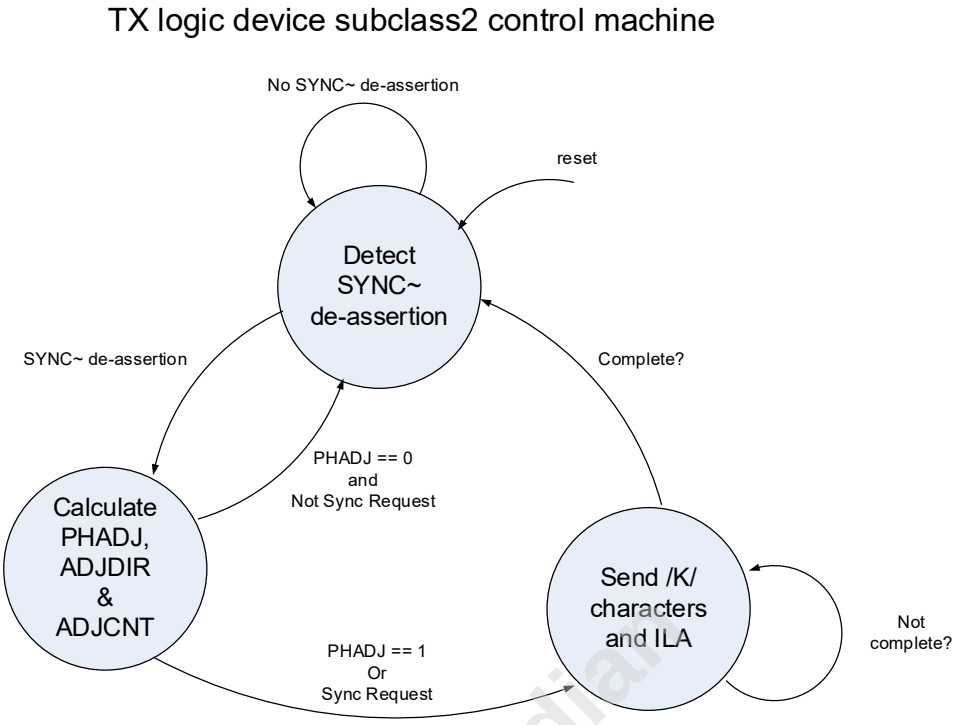


Figure 83 — Example state diagram of TX logic device supporting subclass 2 deterministic latency

The DAC will respond to the requests from the logic device. Figure 84 shows an example state diagram of the adjustment machine used by the DAC device to perform its end of the loop. After reset, the adjustment machine will monitor the interface for a link initialization condition. This condition will be the reception of multiple /K/ characters followed by an ILAS. During the ILAS, the machine will evaluate what is transmitted in PHADJ, ADJDIR, and ADJCNT. If PHADJ is not set, then no adjustment is needed and the DAC will proceed to propagate data. The machine will go back to monitoring the interface for the next link initialization condition. If PHADJ is set, then the machine will make a timing adjustment to the LMFC. The adjustment is equal to the number of adjustment resolution steps in ADJCNT<3:0> and in the direction set by ADJDIR. In order that the TX logic device can accurately calculate ADJCNT, the DAC manufacturer must define the DAC adjustment resolution in units of DAC device clock periods.

If ADJCNT equals zero, a zero adjustment is performed. Once the LMFC adjustment is complete, the DAC will issue an error report to the TX logic device through the SYNC~ interface. Once the error report is issued, the machine will go back to waiting for the next link initialization event.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.4.3 DAC initiator and target configurations (cont'd)

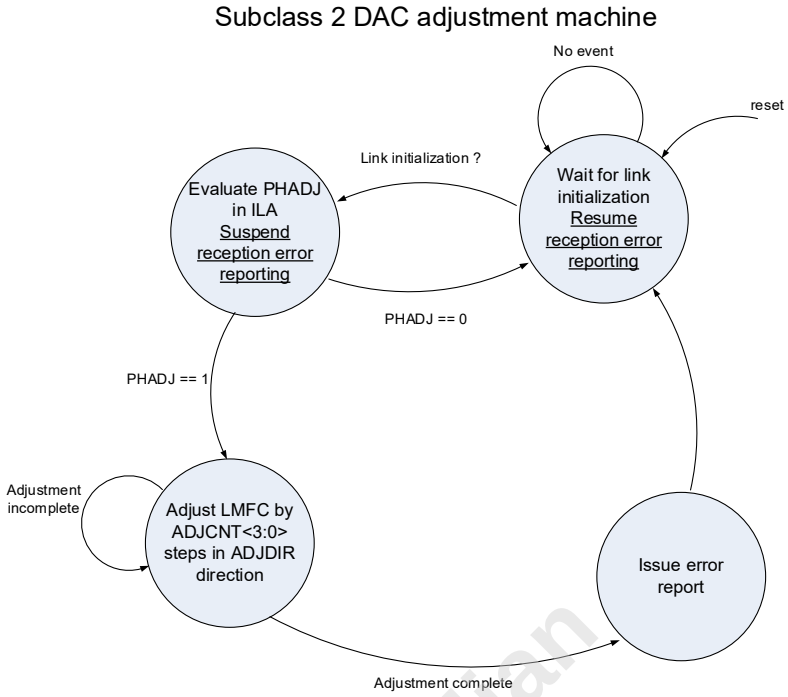


Figure 84 — Example state diagram of DAC device supporting deterministic latency

Loop initialization:

For the loop above to be set in motion, a synchronization event must be developed. Upon startup, the DAC will assert the SYNC~ signal to request synchronization, and this event will get the loop started. At a future point, the user may want to start the loop again. This can be done in one of two ways. The first way involves the TX logic device performing a reinitialization over the data interface and set PHADJ while setting ADJCNT to zero. By doing so, the TX logic device requests an adjustment of zero counts, and the DAC will respond with an error report. The second way is to program the DAC, via the control interface if so equipped, to issue a synchronization request to the transmitter.

Latency monitoring:

Like the ADC case, SYNC~ detection can be used to monitor the alignment of the RX and TX LMFCs. During an alignment, the TX logic device will adjust the DAC LMFC to the proper position. After initial alignment, a monitoring phase may be entered, and an expected detection phase (based on the detection resolution) can be defined. If future detected SYNC~ de-assertions differ from the expected detection phase, then latency has been lost. It is up to the implementer, as with the ADC, whether a realignment be requested or not.

In the case that the detection resolution is finer than the time period required to meet the total timing margin the SYNC interface, a programmable window can be applied to the SYNC~ detection during the monitoring phase. Its width can be set to a programmable number of detection resolution steps. If SYNC~ de-assertion falls outside this detection window, the logic device can treat this as a loss of latency and optionally treat it as above.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.5.5 Summary of requirements for subclass 2 deterministic latency

General requirements for support of subclass 2 protocol.

- Per 8.8.2, a subclass 2 RX device shall de-assert SYNC~ at its active LMFC edge, therefore transmitting its LMFC phase information to a TX device through the SYNC~ interface.
- Per 4.3.1, the TX device clock period shall be a whole number of RX device clock periods, or the RX device clock period shall be a whole number of TX device clocks periods.

Requirements for subclass 2 ADC devices.

- The ADC device shall be able to adjust the phase of its LMFC and, if necessary for this purpose, other internal clocks in increments of the adjustment resolution, relative to the detection of the SYNC~ de-assertion.
- The ADC device shall specify the functional delay, in units of adjustment clock periods, from the detection of SYNC~ de-assertion to the start of the TX LMFC.
    - When detection and adjustment are done by the device clock period, this value is referenced to the device clock coincident to the detection of de-assertion.
    - When the detection is done by an internal clock, this value is referenced to the adjustment clock active edge coincident or following the internal clock when the detection was done.

Requirement for subclass 2 DAC devices.

- The DAC device shall be able to adjust the phase of its internal LMFC signal in increments of its adjustment resolution.
- The DAC device shall be able to adjust its LMFC phase when PHADJ is equal to 1 and based on information resident in the ADJDIR, and ADJCNT<3:0>. See Table 57 and Table 58.
- The DAC device shall issue an error report each time an adjustment has been performed (PHADJ = 1).
- The adjustment resolution of the DAC device shall be specified in its datasheet for use in the calculation of ADJCNT by the TX logic device. It shall be defined in units of DAC device clock periods.

Requirement for a TX logic device for subclass 2 support.

- The TX logic device shall be able to detect the phase of the captured SYNC~ de-assertion relative to its LMFC based on its detection resolution.
- The TX logic device shall accept a programmed DAC adjustment resolution, and be able to use this adjustment resolution to calculate ADJCNT based on the relative detection required above.
- The TX logic device shall provide correction information to the DAC device through an ILAS (8.4.5.3) as a result of a synchronization request by the receiver (8.4.1) or following link reinitialization over the data interface by the transmitter (8.4.7). The correction information shall be contained in PHADJ, ADJDIR, and ADJCNT. See also Table 57 and Table 58 as well as 8.5.4.3.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6 Receiver operation

8.6.1 Code group synchronization

Code group synchronization in receiver devices is obtained and maintained in the following way:

- On link start-up, the receiver issues a synchronization request and the transmitter emits comma characters /K/= /K28.5/
- The receiver de-asserts the synchronization request after correct reception of four successive /K/ characters.
- After correct reception of another four 8B/10B characters, the receiver assumes full code group synchronization.
- Upon receiving an invalid code, the receiver enters a check state.
- If three additional invalid codes are received while the receiver is in the check state, loss of synchronization is declared.
- The receiver exits the check state and returns to normal operation after receiving four valid codes in succession.

The activation and de-activation of the synchronization request are transferred to the transmitter via the SYNC interface, further specified in clause 8.8. In multi-lane receivers, the synchronization requests of all lanes shall be combined, so that each transmitter will continue to send /K/ characters until each receiver has achieved code group synchronization. If the hard-wired SYNC interface is used, the timings specified in 8.8.2 shall be observed.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6.1 Code group synchronization (cont'd)

The code group synchronization is illustrated in the receiver state machine of Figure 85. In each of the states, the decoder processes one code group. The meaning of the variables is explained in Table 51.

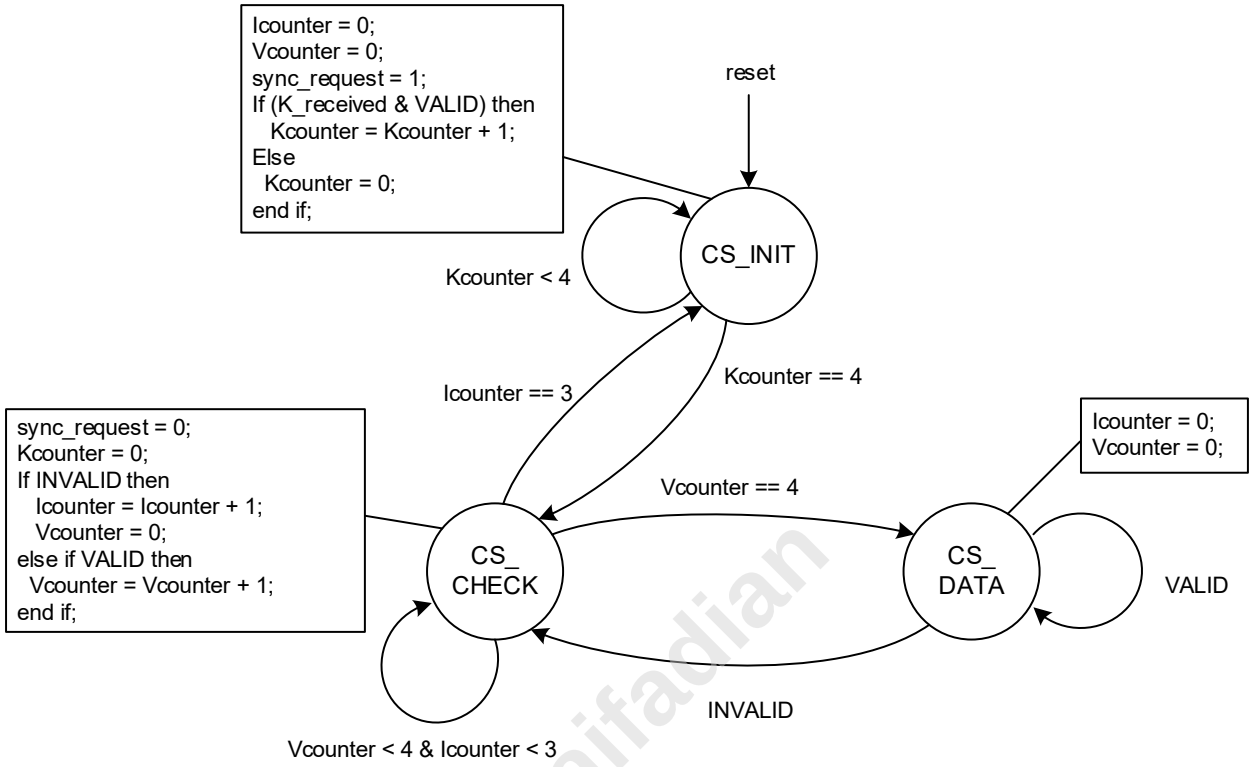


Figure 85 — Receiver state machine for code group synchronization

Table 51 — 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 the SYNC interface of multi-lane receivers will combine the synchronization requests of each lane.
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.

weifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6.2 Initial frame synchronization

The initial frame synchronization process is illustrated in the state machine shown in Figure 86 and Figure 87. The meaning of the variables used in these figures is explained in Table 52. At reset, the state machine enters its initial state and the octet counter is cleared (zeroed). At reset, the code group synchronization machine activates a synchronization request, which will keep the frame synchronization in its initial state for the duration of code group synchronization.

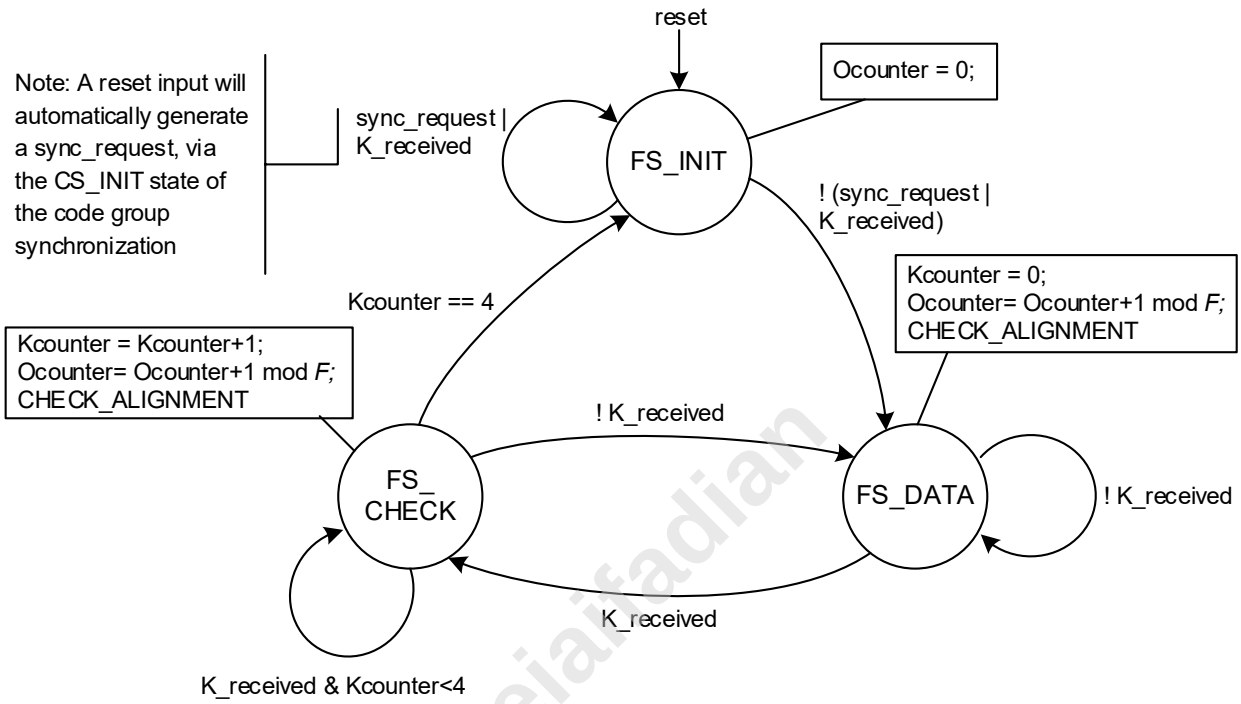


Figure 86 — State machine for frame synchronization in receivers supporting reinitialization over the data interface

The state machine moves to the FS_DATA state when the receiver has de-asserted its synchronization request and the transmitter has stopped sending /K/ symbols. In the FS_DATA state, the octet counter counts the position of the received octet in the frame, between 0 and *F*-1. A receiver shall return to the FS_INIT state if any receiver on the link issues a synchronization request. In a configuration having multiple receiver devices, this is only possible by monitoring the reception of /K/= /K28.5/ symbols, as shown in Figure 86.

If a K28.5 symbol is received, the state machine moves to the FS_CHECK state. In this state, the octet counter keeps running. However, if four K28.5 symbols are received in succession, the frame synchronization returns to its initial state.

Subclass 0 DAC devices may also implement the state machine of Figure 87. This state machine will only reinitialize the frame synchronization when one or more receivers in the same device and connected to the same link issue a synchronization request.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6.2 Initial frame synchronization (cont'd)

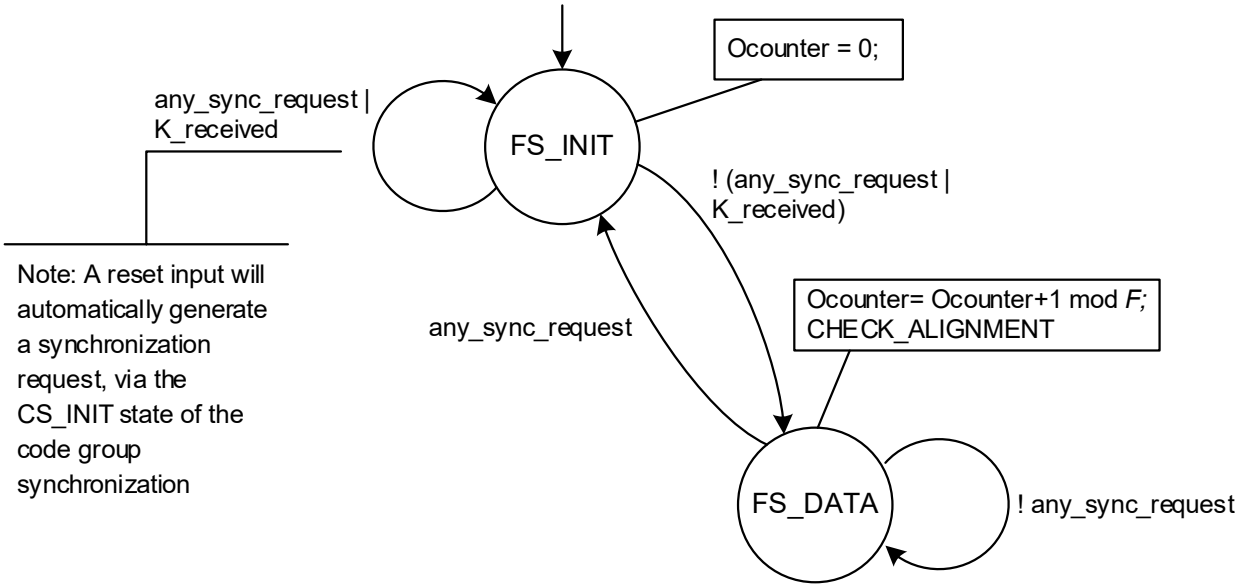


Figure 87 — Alternative state machine for frame synchronization in receivers *not* supporting reinitialization over the data interface

Table 52 — Variables used in initial frame synchronization

Variable	Meaning
any_sync_request	A synchronization request asserted by any receiver connected to the link.
CHECK_ALIGNMENT	Perform frame alignment monitoring (see 8.6.4)
F	Number of octets per frame
Kcounter	Counter used in the FS_CHECK phase to count the number of K28.5 symbols
K_received	Asserted when the current symbol corresponds to control character K28.5 (valid or invalid)
Ocounter	Counter used to mark the position of the current octet in the frame.
sync_request	Asserted by receiver when loss of code group synchronization has been detected or if another error requires reinitialization.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6.3 Initial lane alignment

Receivers inside a single device shall be able to achieve mutual lane alignment within four multiframes of the initial lane alignment sequence specified in 8.4.5.3. Subclass 1 and subclass 2 devices will further align incoming alignment characters to their internal LMFCs within four multiframes.

Each receiver in a multi-lane configuration shall prepare for a new initial lane alignment if any receiver on the link or multipoint link issues a synchronization request. On a multipoint link, logic device receivers and subclass 1 and 2 DAC devices shall detect such a request from the reception of 4 successive K28.5 characters (see 8.4.7 and 8.6.2). Subclass 0 DAC devices and receivers connected to a single link may alternatively monitor the synchronization requests of the other receivers in the device that are connected to the link.

8.6.4 Monitoring and correction of frame and lane alignment

The detection and correction process is illustrated by means of the following examples. As stated in 8.4.4.4 and 8.4.6, it shall be possible to disable the frame and / or lane alignment correction (RESET_FRAME_COUNTER and RESET_OCTET_COUNTER in the below examples) in case the user data is such that not sufficient alignment characters are produced for reliable detection of alignment errors or when realignment via a synchronization request followed by an initial lane alignment sequence is preferred. Frame and lane alignment monitoring and corrections are illustrated in separate examples, but because of their interdependencies, the implementation could also be combined.

EXAMPLE 1 Implementation of frame alignment monitoring and correction

The detection and correction process is illustrated in the pseudo-code of Figure 88. The meaning of the variables and functions is explained in Table 53.

```
if (A_received | F_received)
  REPLACE_ALIGNMENT_CHARACTER; /* See Table 53 */
  if ((Ocounter == previous_AF_position) & VALID)

    RESET_OCTET_COUNTER; /* Only if enabled */

    if Ocounter!=F-1

      INFORM_LANE_ALIGNMENT_ENGINE

    end if;

  end if;
```

Figure 88 — Pseudo-code for frame alignment monitoring and correction in receiver

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6.4 Monitoring and correction of frame and lane alignment (cont'd)

Table 53 — Variables and functions in frame alignment monitoring and correction

Variable	Meaning
A_received	Asserted when the current symbol, before possible substitution in lane alignment monitoring, corresponds to control character K28.3 Note: detection of K28.3 is not required in single-lane subclass 0 DAC devices.
F	Number of octets per frame
F_received	Asserted when the current symbol corresponds to control character K28.7
INFORM_LANE_ALIGNMENT_ENGINE	Inform lane alignment engine that frame reference has changed.
Ocounter	Counter used to mark the position of the current octet in the frame. Octet indexing starts from 0.
previous_AF_position	Variable into which to store the position in the frame of a K28.3 or K28.7 symbol
REPLACE_ALIGNMENT_CHARACTER	Replace the alignment character at the decoder output by: • The data character decoded or used at the same position in the previous frame when scrambling is disabled • The data character with the same value when scrambling is enabled If initial lane alignment or lane alignment monitoring is implemented after frame alignment monitoring, the position of a replaced K28.3 should be marked.
RESET_OCTET_COUNTER	Reset octet counter to zero at reception of next octet.
VALID	Asserted by receiver to indicate that the current symbol is a valid symbol given the current running disparity.

weiaiyuan

EXAMPLE 2 Implementation of lane alignment monitoring and correction

The detection and correction process is illustrated in the pseudo-code shown in Figure 89. In this example, the lane alignment engine automatically also adjusts the frame markers. Because a lane alignment error could have been caused by an alignment error in the LMFCs, the lane alignment engine in this example also initiates a synchronization check between the LMFCs via the method supported by the device subclass. In subclass 1, this means sending a SYSREF generation request to the clock generator (see 4.2.3), and in subclass 2 the generation of an error report or synchronization request over the hard-wired SYNC interface (see 8.5.4). The meaning of the variables and functions is explained in Table 54.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6.4 Monitoring and correction of frame and lane alignment (cont'd)

```
if A_received
  REPLACE_A; /* See Table 54 */
  if (((Fcounter == previous_A_position) | FRAME_REALIGNED) & VALID)
    RESET_FRAME_COUNTER; /* Only if enabled */
    RESET_OCTET_COUNTER; /* Only if enabled */
    if (Fcounter!=K-1)
      INITIATE_SYNC_CHECK;
    end if;
  end if;
  if (VALID | (Fcounter==K-1))
    previous_A_posit     ion = Fcounter;

    previous_AF_position = Ocounter;
  end if;
end if;
```

Figure 89 — Pseudo-code for lane alignment monitoring and correction in receiver

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6.4 Monitoring and correction of frame and lane alignment (cont'd)

Table 54 — Variables and functions in lane alignment monitoring and correction

Variable	Meaning
A_received	Asserted when the current symbol, before possible substitution in frame alignment monitoring, corresponds to control character K28.3.
Fcounter	Counter used to mark the position of the current frame in the multiframe. Frame indexing starts from 0.
FRAME_REALIGNED	A lane misalignment is expected because a recent frame realignment.
K	Number of frames in multiframe.
previous_A_position	Variable into which to store the position in the multiframe of a K28.3 symbol.
previous_AF_position	Variable into which to store. the position in the frame of a K28.3 or K28.7 symbol (see Table 53).
REPLACE_A	Replace the K28.3 at the decoder output by: • the data character decoded or used at the same position in the previous frame when scrambling is disabled; • D28.3 when scrambling is enabled. However, if frame alignment monitoring is implemented after lane alignment monitoring, the K28.3 shall not be replaced or it shall be marked.
RESET_FRAME_COUNTER	Reset frame counter to zero at reception of next frame
RESET_OCTET_COUNTER	Reset octet counter for frame alignment to zero at reception of next octet (see Table 53)
INITIATE_SYNC_CHECK	In subclass 1 and 2 receivers, if authorized via the control interface, initiate a synchronization check between the LMFCs via one of the methods supported by the device subclass (see the text in the first paragraph of this example).
VALID	Asserted by receiver to indicate that the current symbol is a valid symbol given the current running disparity.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6.5 Error handling

8.6.5.1 Error kinds

Table 55 shows the minimum set of errors to be detected in each receiver.

Table 55 — 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;
- error in link configuration data (parameters in TX and RX do not match);
- decoding error in lane alignment sequence (wrong octets decoded).

weiaiadian

8.6.5.2 Data output on error

This subclause applies only to the reception of data to the transport layer (see clause 0). Figure 85 specifies how to deal with errors occurring during code group synchronization. During initial lane alignment, the reaction on errors will depend on the implementation of lane alignment.

In response to a not-in-table error, the decoder shall repeat the previously received non-error frame. A not-in-table error with scrambling enabled can corrupt the following two octets, so in cases where the corrupted octets span a frame boundary both the frame containing the not-in-table error and the following frame shall be replaced by the frame preceding the not-in-table error.

In case of a disparity error, the output shall be the decoded symbol according to the running disparity indicated by the received code group.

NOTE Detection of an invalid code-group in the 8B/10B transmission code does not necessarily indicate that the code-group in which the error was detected was the one in which the error occurred. Invalid code-groups may result from a prior error that altered the running disparity of the bit stream but that did not result in a detectable error at the code-group in which the error occurred. See the examples in [6].

If an unexpected control character is received, the action depends on its value. In case of an /A/ or /F/, octet replacement shall take place according to the rules for frame and lane alignment monitoring (see 8.4.4 and 8.4.6). Other unexpected control characters shall be treated like not-in-table characters.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.6.5.3 Errors requiring reinitialization

Certain errors cannot be corrected other than by reinitialization of the whole JESD204 link. Because some of these errors could be tolerated in certain applications, it is recommended that the receiver implementer provide the option to specify via a control interface which errors will lead to reinitialization.

8.6.5.4 Error reporting via the SYNC interface

On detection of an error requiring reinitialization, the receiver shall send a synchronization request over the SYNC interface (see clause 8.8). On detection of an error not requiring reinitialization, a receiver in a DAC device shall send an error report over the SYNC interface. If additional errors are detected before completion of the error report, these are not required to be reported, unless they would require link reinitialization.

A receiver in a logic device is allowed, but not required to send and error report over the SYNC interface, in the same way as a DAC device. However, certain usage models in subclass 2 assume also the logic device to send an error report over the (hard-wired) SYNC interface (see 8.5.4.2).

Subclass 2 DAC devices must be able to suspend reporting of errors not requiring reinitialization over the hard-wired SYNC~ interface. As the DAC uses this interface to transmit its LMFC phase, it must control when error reports occur during its initiator target alignment operation. Please see 8.5.4.3 for further information.

Receiver devices for which the vendor has declared backward compatibility with JESD204A or JESD204B (which assume the existence of a frame clock for timing the SYNC~ signal) shall be programmable to suppress reporting errors not requiring reinitialization over the hard-wired SYNC interface, unless the timing of the error report generation over the hard-wired SYNC interface is compatible with those versions of the standard. See Table M.1 in Annex M for more information.

8.6.5.5 Error reporting via a control interface

A receiver in a logic device shall be able 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.

weiantian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.7 Transmitter operation

8.7.1 Synchronization

The transmitter state diagram is shown in Figure 90. The variables and functions used are explained in Table 56. The INIT_LANE state may be skipped if at least one side of the lane does not support lane alignment.

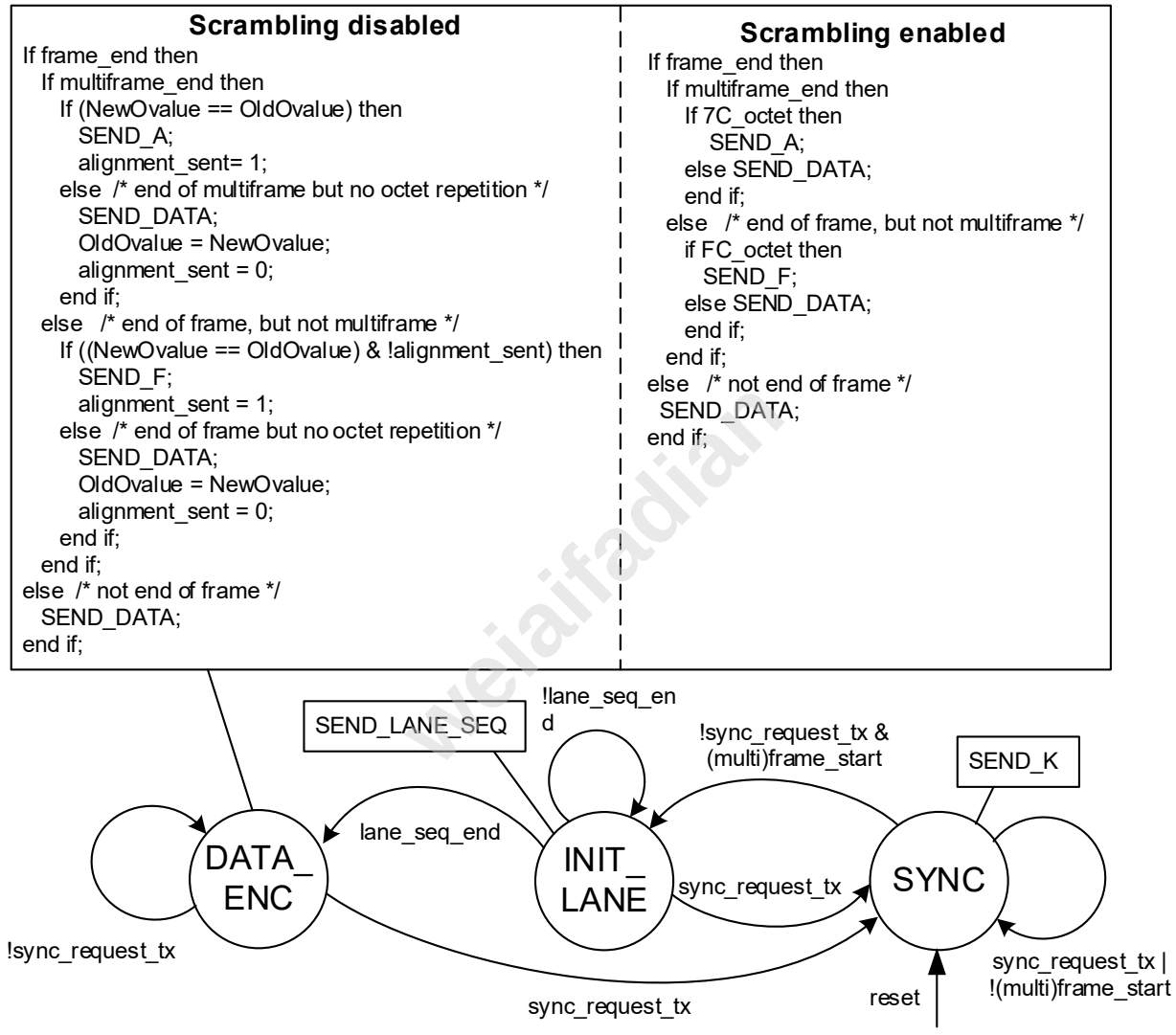


Figure 90 — Transmitter state machine

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.7.1 Synchronization (cont'd)

Table 56 — Variables and functions in transmitter state machine

Variable	Meaning
7C_octet	Asserted when the scrambler outputs a 0x7C (28.3) octet.
alignment_sent	Used to indicate a /K28.3/ or /K28.7/ has been sent in the previous frame.
FC_octet	Asserted when the scrambler outputs a 0xFC (28.7) octet.
frame_end	Asserted by transmitter to indicate end of frame.
(multi)frame_start	Asserted by transmitter to indicate start of multiframe in subclasses 1 and 2, and start of frame in subclass 0.
lane_seq_end	Asserted by transmitter to indicate end of initial lane alignment sequence.
multiframe_end	Asserted by transmitter to indicate end of multiframe. Only to be asserted if both sides of the lane support lane alignment.
NewOvalue	Value of last octet in current frame.
OldOvalue	Used for storage of last octet in frame.
SEND_A	Send /K28.3/ symbol.
SEND_DATA	Send code group belonging to current data octet.
SEND_F	Send /K28.7/ symbol.
SEND_K	Send /K28.5/ symbol.
SEND_LANE_SEQ	Send initial lane alignment sequence. Only if the TX is able and configured to send an ILAS (see 8.4.5.3).
sync_request_tx	Asserted when transmitter detects a synchronization request.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.7.2 Handling of error reports and synchronization requests

8.7.2.1 Hard-wired SYNC interface

Any high to low transition of the SYNC~ signal shall be interpreted as a reported error from the receiver (see 8.6.5.4). The handling of an error report is application-dependent.

The sampling of the SYNC~ signal shall occur on the active edge of the SYNC~ detection clock, which is further specified in 4.3.7. If the TX device detects a low SYNC~ signal for at least the minimum duration specified in 8.8.2, it shall additionally interpret the SYNC~ pulse as a synchronization request. Once the SYNC~ signal has been interpreted as a synchronization request, the required behavior of the TX device depends on the deterministic latency subclass of the device:

- Subclass 0 transmitters shall send exactly one frame of /K28.5/ symbols for each additional frame duration that the detected SYNC~ signal remains low. Once the detected SYNC~ signal goes high, subclass 0 transmitters shall cease sending /K28.5/ symbols at the next frame boundary after that point.
- Subclass 1 and 2 transmitters shall send continuous /K28.5/ symbols until the detected SYNC~ signal deactivates (i.e., goes high). At this point, subclass 1 and 2 transmitters shall continue to send continuous /K28.5/ symbols until both following conditions are met:
  - The /K28.5/ symbol generation has lasted for at least 17 octets.
  - The end of a local multiframe period is reached.

The transition from /K28.5/ generation to ILAS generation shall occur at a multiframe boundary, but may be a programmable number of multiframes after SYNC~ signal de-activation is detected. For devices supporting a programmable number of multiframes, the device must be able to support generation of ILAS on the first possible multiframe boundary after SYNC~ de-activation.

A high SYNC~ signal shall always be interpreted as a deactivated resynchronization request.

8.7.2.2 Soft SYNC interface

The handling of an error report is application-dependent. After detection of a synchronization request on the soft SYNC interface, the behavior shall be as specified for device subclasses 1 and 2 using the hard-wired SYNC interface (see 8.7.2.1).

wenjiadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.7.3 SYNC~ detection in device subclass 2

The SYNC~ signal is used to adjust the LMFC and, if necessary for this purpose, other internal clocks of a subclass 2 deterministic latency converter device. A SYNC~ de-assertion is detected by using a detection clock to detect the high side of a low to high transition of the SYNC~ signal at the TX device. The accuracy of the detection is governed by the detection resolution of the TX device.

In an ADC device, SYNC~ and the LMFC boundaries are adjusted such that they occur a deterministic number of adjustment clock cycles after the detection event. See 8.5.3.3 and 0 for clock and timing definitions, as well as Figure 82.

In a TX logic device, SYNC~ de-assertion is detected and the calculated phase discrepancy between the detection phase and the TX logic device LMFC phase is used to adjust the DAC frame and LMFC boundaries. The required adjustment in the DAC is embedded in parameters PHADJ, ADJDIR, and ADJCNT found in ILAS configuration registers 1 and 2, (see Table 58) and passed from logic device to DAC through the ILAS. This ILAS is a result of either link reinitialization over the data interface (see 8.4.7) or initial lane alignment (see 8.6.3). Please see 8.5.4.3 for functional usage of measured information.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.8 SYNC interface

8.8.1 Introduction

The SYNC interface is used in the 8b/10b link layer for the following purposes:

- to communicate synchronization requests from the receiver device to the transmitter device;
- on DAC links, and optionally on ADC links, to report reception errors from the receiver device to the transmitter device;
- in subclass 2, to align the LMFCs in the receiver and transmitter devices.

This part of JESD204 defines two types of SYNC interfaces. A hard-wired SYNC interface and a "soft" SYNC interface, using commands over the control interface. Alignment of the LMFCs in subclass 2 requires a hard-wired SYNC interface. Subclasses 0 and 1 may alternatively use a soft SYNC interface.

8.8.2 Hard-wired SYNC interface

The hard-wired SYNC interface contains exactly one signal, which is denoted by SYNC~. The tilde indicates that the signal is active low. In case of a differential interface, the true part of the signal is active low.

Implementation of the hard-wired SYNC interface is mandatory in subclass 2 devices and in devices for which the manufacturer has declared support for interoperability with devices according to JESD204B or earlier versions of this standard (see 4.5.3). In other devices, the hard-wired SYNC interface may be replaced by the soft SYNC interface (see 8.8.3).

In the RX device, the activation and de-activation of the SYNC~ signal shall take place on an active edge of a SYNC~ generation clock. In the TX device, the SYNC~ signal shall be sampled on the active edges of a SYNC~ detection clock. The SYNC~ generation and detection clocks are further specified in 4.3.7.

In subclass 1 and subclass 2 RX devices, de-activation of the SYNC~ signal shall always take place on an active edge of the LMFC.

NOTE In subclass 2, the TX device estimates the timing of the LMFC in the RX device from the moment of de-activation of the SYNC~ signal. A common specification of the SYNC~ signal timing for subclasses 1 and 2 makes it possible that also a subclass 1 TX device can estimate the timing of the LMFC in the RX device. The latter functionality is not mandatory in subclass 1, but it may be useful for finding the cause of reception errors.

The duration of the activation of the SYNC~ signal determines whether the SYNC~ signal carries an error report or a synchronization request. A long activation means a synchronization request and a short activation an error report (see 8.6.5.4). To allow for different frequencies of the SYNC~ generation and detection clock in the RX and TX device and possible violation of the setup and hold times in the TX device, the RX device shall have a programmable duration of an error report and the TX device shall have a programmable longest duration of the detected SYNC activation to be interpreted as an error report.

In the RX device, the duration of an error report shall be programmable over at least the range from 8 to 16 character durations. Backward compatibility with a previous version of this standard may require the capability to program also error report durations outside this range (see Annex M). If the SYNC~ generation clock period is one character duration or longer, the granularity shall be one SYNC~ generation clock cycle. Otherwise, the granularity shall be no more than the smallest number of SYNC~ generation clock cycles having a duration equal to or exceeding one character duration. In subclasses 1 and 2, the error report shall start at the programmed duration before an active edge of the LMFC.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.8.2 Hard-wired SYNC interface (cont'd)

Between consecutive error reports, the SYNC~ signal shall be deactivated for at least the programmed duration of an error report.

A synchronization request can start at any active edge of the SYNC~ generation clock and shall last at least 49 character durations.

In the TX device, the maximum detected duration of SYNC~ activation for interpretation as an error report shall be programmable over at least the range from three detection clock periods to 31 character durations. Backward compatibility with a previous version of this standard may require the capability to program also duration limits outside this range (see Annex M). If the SYNC~ detection clock period is at least one character duration, the resolution shall be one SYNC~ detection clock cycle. Otherwise, the resolution shall be no more than the ceiling of the number of SYNC~ detection clock cycles per character. The minimum detected duration of SYNC activation for interpretation as a synchronization request shall be programmable with the same resolution over at least the range from four detection clock periods to 32 character durations. Again, backward compatibility with a previous version of this standard may require the capability to program detection thresholds outside this range.

NOTE Violation of setup and hold times can cause an error in the detected duration of the SYNC activation of up to one cycle of the TX detection clock in either direction. Therefore, the RX must activate SYNC for at least two cycles of the TX detection clock. If the error report has this duration, its detected duration will be between one and three cycles of the TX detection clock. If the detected duration is at least four cycles of the TX detection clock, the TX can be sure that the RX transmitted a synchronization request. To make sure that the detected duration is at least four cycles of the TX detection clock, the RX must activate SYNC for at least one more cycle of the TX detection clock. The longest allowed period of the SYNC~ generation and detection clock is eight character durations. Therefore, the duration of the error report must be programmable up to at least sixteen character durations and the minimum duration of a synchronization request up to at least 40 character durations. This minimum duration is extended by nine character durations in order that a subclass 0 TX will always send enough /K/ characters for reliable link reinitialization over the data interface (see 8.4.7).

In subclass 2 operation, the timing of the SYNC interface relative to the device clock is critical. Therefore, it is strongly recommended to use a similar interface for the SYNC interface and the device clock in subclass 2 (with the exception that SYNC~ should never be AC-coupled).

Figure 91 shows the critical timing specifications relating to the SYNC~ signal that are necessary for subclass 0 devices claiming backward compatibility to JESD204A and for subclass 2 deterministic latency devices. The values for these parameters are not specified here, but the transmitter and receiver device specifications shall specify these values.

$\mathbf{t}_{DS\_R}$ (min/max): Device-clock-to-SYNC~ delay at the receiver device pins. JESD204A compatible subclass 0 and all subclass 2 receiver devices shall specify this parameter.

$\mathbf{t}_{SU\_T}$ (min) and $\mathbf{t}_{H\_T}$ (min): JESD204A compatible subclass 0 and all subclass 2 transmitter devices shall specify these parameters.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.8.2 Hard-wired SYNC interface (cont'd)

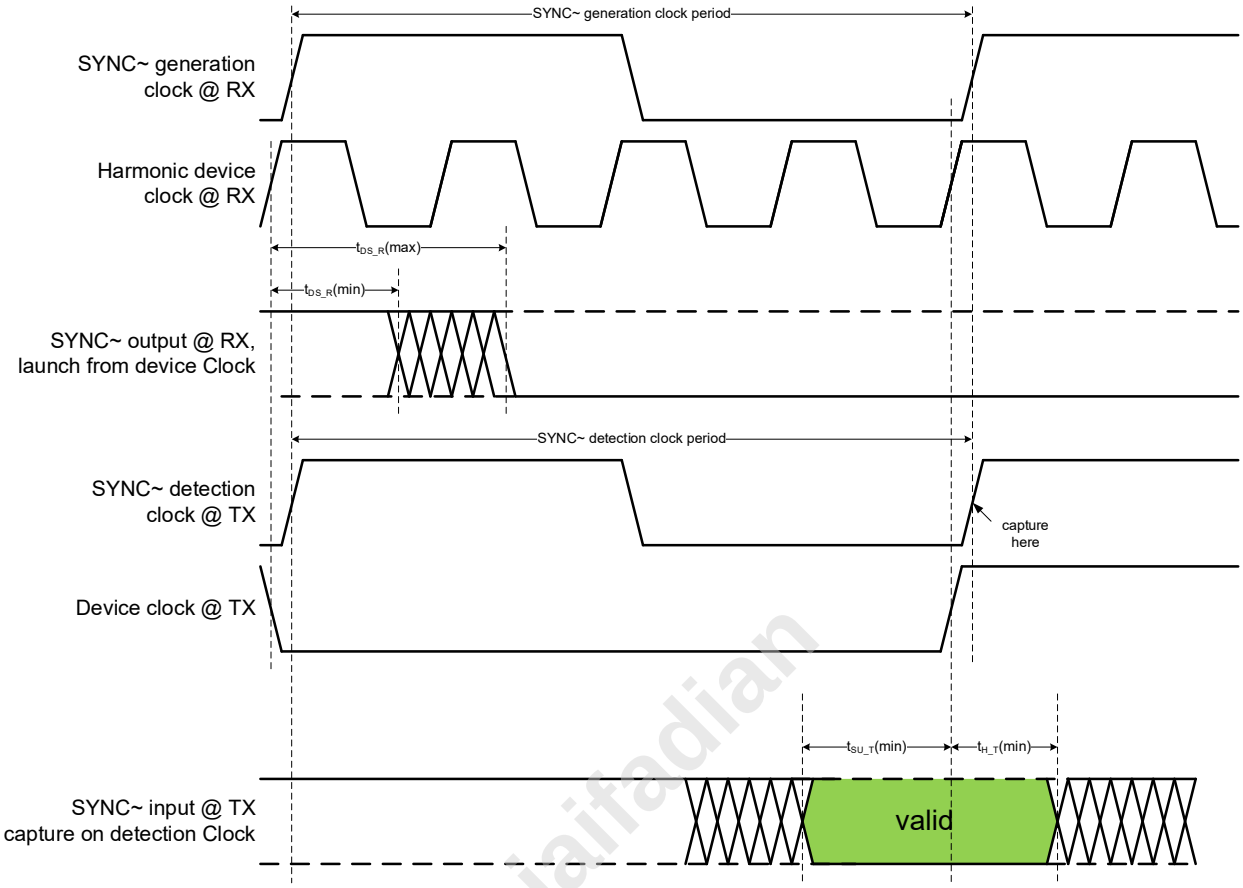


Figure 91 — SYNC~ signal timing for JESD204A compatible subclass 0 and for subclass 2 devices

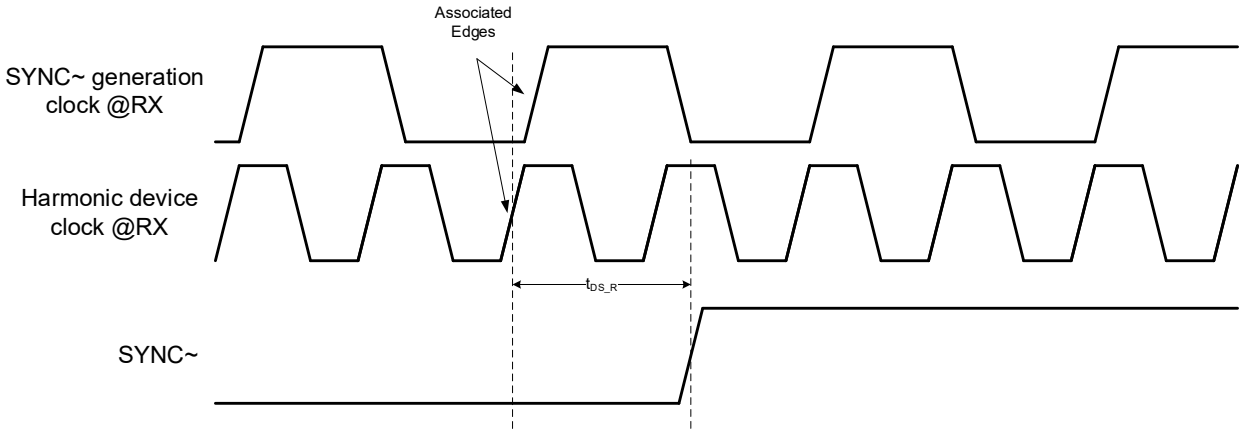
In devices where the device clock is faster than or as fast as the SYNC~ generation or detection clock, the timing can be related to the device clock. In a device where the SYNC~ generation or detection clock is faster than the device clock, SYNC~ may be launched or captured by the SYNC~ generation or detection clock and the timing can only be measured relative to an associated edge. In JESD204A and in subclass 2, the SYNC~ generation and detection clocks are aligned to the device clock for at least one edge per multi-frame. The measurements of the timing parameters can be taken relative to this edge. All other timing is relative to this basis and will differ by some number of periods of the SYNC~ generation or detection clock.

Figure 92 shows edge associations for two RX SYNC~ launch cases. In the first case, the SYNC~ generation clock is slower than the device clock. In the second, the device clock is slower than the SYNC~ generation clock.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.8.2 Hard-wired SYNC interface (cont'd)

Associated edges for SYNC~ launch when device clock is faster than SYNC~ generation clock.



Associated edges for SYNC~ launch when device clock is slower than SYNC~ generation clock.

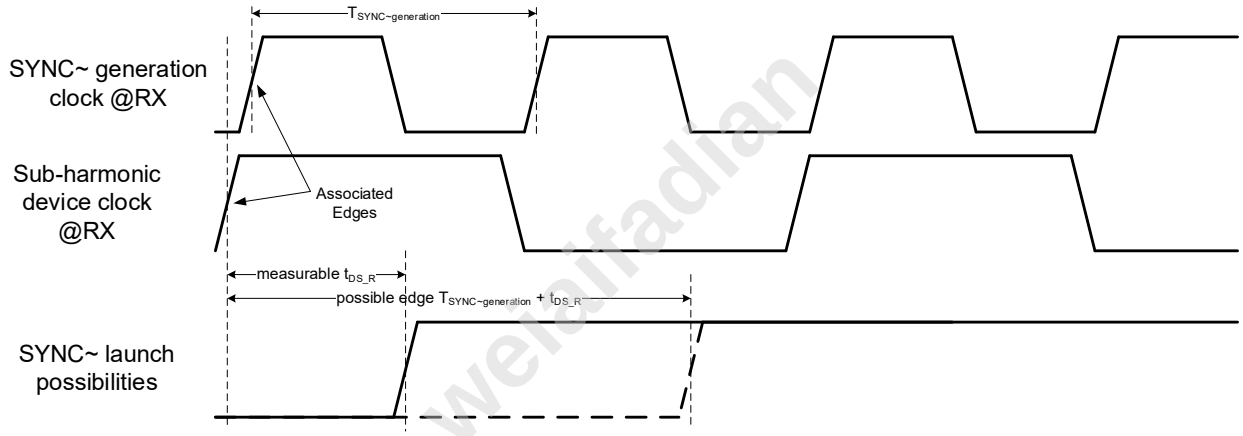


Figure 92 — Associated edges for JESD204A compatible subclass 0 and subclass 2 devices

weifafadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.8.3 Soft SYNC interface

If the link uses no hard-wired SYNC interface, the receiver shall make synchronization requests and error reports available to the TX over one of the control interfaces mentioned in 4.4. If the link uses a hard-wired SYNC interface, then synchronizations requests shall always be sent over the hard-wired SYNC interface. Error reports are allowed to be sent over the control interface instead of or in addition to the hard-wired SYNC interface.

NOTE 1 Transmission of error reports over the SYNC interface is generally not required for logic devices (see 8.6.5.4).

NOTE 2 Certain usage models in subclass 2 do not allow the hard-wired SYNC interface to be replaced by the soft SYNC interface for the transmission of error reports (see 8.5.4.2).

If the RX is the initiator of the control interface, the following requirement applies:

- The RX shall encode the synchronization request or error report as a message over the control interface to the TX as soon as it is available.

If the TX is the initiator of the control interface, the following requirements apply:

- The soft SYNC interface shall include a hard-wired interrupt signal.
- The RX shall send an interrupt to the TX as soon as an error report or synchronization request is available. The TX shall retrieve the information from the RX over the control interface as soon as is practically possible after detecting the interrupt.

8.9 Link configuration parameters and their encoding

The link configuration parameters and their encoding in the initial lane alignment sequence are summarized in Table 57. The link configuration data in the ILAS is meant to specify the parameter values defining the user data formats used by the transmitter device, as described in clause 0, and to specify information on the transmitter configuration and capabilities. The purpose of the configuration data is solely to give the receiver the confirmation that it is connected to the right TX device and that both devices have been programmed to the correct mode. On subclass 2 DAC links (see 8.5.4.3), the link configuration data can also contain commands for realignment of the receiver LMFC. Table 58 shows how the link configuration fields shall be mapped to octets. It is impossible to foresee all future applications and configurations of the JESD204 interface. Therefore, bits marked by “X” in Table 56 may be used to pass information not specified here or to extend existing fields if they are turn out to be too narrow. When device vendors reach consensus about use of currently unspecified bits, this may be added to the standard in a later phase.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.9 Link configuration parameters and their encoding (cont'd)

Table 57 — Link configuration parameters and their encoding in the ILAS

Parameter	Description	Encoding 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[a]	BID<3:0>	Binary value
CF	No. of control words per frame duration per link	0 ... 32[a]	CF<4:0>	Binary value[b]
CS	No. of control bits per sample	0 ... 3[a]	CS<1:0>	Binary value
DID[c]	Device (= link) identification no.	0 ... 255[a]	DID<7:0>	Binary value
F	No. of octets per frame	1 ... 256[a]	F<7:0>	Binary value minus 1
HD	High Density format	0 ... 1	HD<0>	Binary value
JESDV[d]	JESD204 version 000 – JESD204A 001 – JESD204B 010 – JESD204C	0 … 7	JESDV<2:0>	Binary Value
K	No. of frames per multiframe	1 ... 256[a]	K<7:0>	Binary value minus 1
L	No. of lanes per converter device (link)	1 ... 32[a]	L<4:0>	Binary value minus 1
LID[e]	Lane identification no. (within link)	0 ... 31[a]	LID<4:0>	Binary value
M	No. of converters per device	1 ... 256[a]	M<7:0>	Binary value minus 1
N	Converter resolution	1 ... 32[a]	N<4:0>	Binary value minus 1
N'	Total no. of bits per sample	1 ... 32[a]	N'<4:0>	Binary value minus 1

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.9 Link configuration parameters and their encoding (cont'd)

Table 57 — Link configuration parameters and their encoding in the ILAS (cont'd)

Parameter	Description	Encoding Range	Field	Encoding
PHADJ	Phase adjustment request to DAC Subclass 2 only	0 … 1	PHADJ<0>	Binary value
S[a]	No. of samples per converter per frame cycle	1 ... 32[a]	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[f]	0 ... 255	FCHK<7:0>	Binary value

[a] The device is not required to support all values in the encoding range. The device may also support parameter values outside the encoding range, unless prohibited elsewhere in the standard.

[b] *CF*==*L* shall always be encoded as 31: control words on all lanes. *CF*==31 can only occur when *L*==31, see clause 0.

[c] A device with a single SYNC~ interface shall be assigned a unique DID. A device with multiple SYNC~ interfaces shall be assigned a unique DID for each SYNC~ interface.

[d] The JESDV is intended to tell the receiver according to which version of the standard the SYNC~ detection in the transmitter has been configured.

[e] Each lane within a given DID shall be assigned a unique LID, with values ranging from 0-L-1. Each lane within a JESD204 system shall have a unique DID / LID combination. For single lane interfaces, LID shall be 0. For multiple lane interfaces, LID of 0 shall be used for the lane carrying the first F octets mapped to the linear axis (as defined in figure 7 in clause 0), with LID of 1 being used for the next F octets, etc. Provided these requirements are met, there is no restriction on which physical lane is assigned a value of LID = 0.

[f] The checksum shall be calculated based on the 21 fields (not including the FCHK field) contained within the link configuration octets, and all bits not belonging to one of the 21 fields shall be ignored when calculating the checksum.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

8.9 Link configuration parameters and their encoding (cont'd)

Table 58 — Mapping of link configuration fields to octets

Configuration octet no.	Bits							
	MSB	**6**	**5**	**4**	**3**	**2**	**1**	**LSB**
0	DID<7:0>							
1	ADJCNT<3:0>				BID<3:0>			
2	X	ADJDIR <0>	PHADJ <0>	LID<4:0>				
3	SCR<0>	X	X	L<4:0>				
4	F<7:0>							
5	K<7:0>							
6	M<7:0>							
7	CS<1:0>		X	N<4:0>				
8	SUBCLASSV<2:0>			N'<4:0>				
9	JESDV<2:0>			S<4:0>				
10	HD<0>	X	X	CF<4:0>				
11	RES1<7:0> - Set to all X							
12	RES2<7:0> - Set to all X							
13	FCHK<7:0>							

weiaifactran

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex A (informative) Differences between revisions

A.1 Differences between JESD204C and JESD204C.01

This subclause briefly describes most of the changes made to entries that appear in this revision of the standard, JESD204C.01, compared to its predecessor, JESD204C (December 2017). In addition to the differences listed, there were various formatting and wording changes made to improve clarity.

- JCOM is removed and replaced with COM and is computed using the procedure in IEEE Std. 802.3 93A.1.
- Maximum data rate increased from 32 Gbps to 32.45 Gbps.
- The jitter parameters BHPCJ and BHPUJ were replaced with CBHPJ and UBHPJ, respectively. See, for example, Table 23 — Category C receiver electrical specification.
- The receiver differential input return loss limit was missing from JESD204C, this was corrected by adding Figure 32 to JESD204C.1. The receiver input common mode return loss has been removed to make receiver specifications compatible with IEEE Std. 802.3-2018.
- In Table 23 — Category C receiver electrical specification, TUJ was replaced with TJ, to reflect that the measured jitter includes correlated jitter.
- "Annex E (informative) Reference JCOM implementation" has been removed, it is now published as a separate document and distributed with the JCOM reference implementation. The reference channel insertion loss vs. data rate and channel length vs. data rate approximation and Figure 24 have also been removed.
- The Class C differential return loss specification, described in 5.2.3.6 and 5.2.4.4, has been modified to be compatible with IEEE Std. 802.3.
- Receiver interference tolerance test uses is introduced with the method described in IEEE Std. 802.3 Annex 93C as specified by IEEE Std. 802.3 93.8.2.3.
- Removed division of Category C transmitters into classes.
- Several scattering parameter computations (subclauses 5.2.12.6.1, 5.2.12.6.2, 5.2.12.6.3, and 5.2.12.6.4 have been removed.
- Description of reference systems throughout clause 5 have been moved to subclause 5.2.3, 5.2.4 and 5.2.5.
- Test fixtures insertion and return loss were made dependent on the maximum device data rate.
- Annex D (normative for compliance) Transmission line model is removed.
- Annex F (normative) JCOM device models interface) is removed.
- Annex G (informative) Conversion of scattering parameters representation is removed.

weiantdian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

A.2 Differences between JESD204B and JESD204C

This subclause briefly describes most of the changes made to entries that appear in this revision of the standard, JESD204C (December 2017), compared to its predecessor, JESD204B.01 (January 2012). In addition to the differences listed, there were various formatting and wording changes made to improve clarity.

3 Terminology

- The terms and definitions were updated to reflect the features that are new to JESD204C.
- Terms that are no longer used within the standard were removed.

4.1.2 Physical layer overview

- Added to JESD204C. Provides an overview of the data interface classes.

4.1.3 Transport and link layer overview

- Updated to reflect the additional link layers defined in JESD204C.
- Added Table 3 to describe the supported link layers versus data rate.
- Replaced occurrences of “LMFC” with “LEMC/LMFC”.
- Abbreviated the configuration examples (subclause 4.1.5).

4.2 Deterministic latency

- Bug correction: The length of a multiframe or extended multiblock size must be larger than the maximum delay *variation* across the link.
- Improved description of the buffer release opportunity.
- Wording changes related to the removal of the frame clock.
- Use of the LEMC instead of the LMFC in the 64B/66B and 64B/80B link layers.
- The granularity in the deterministic latency is now an “adjustment step” instead of a “frame cycle.” The size of the “adjustment step” can, within certain limitations, be chosen by the device implementer.

4.3 Physical timing

- Requirements added for the device clock rate and the SYSREF rate when used with 64B/66B or 64B/80B encoding.
- The existence of a conversion clock, sample clock, character clock and frame clock is no longer assumed. However, a physical frame clock may still be necessary for interfacing to a device belonging to a previous version of the standard.
- Transport layer and link layer clocks added. The standard does not require specific rates for these clocks or specific relationships between their rates and the frame rate.
- SYNC~ generation and SYNC~ detection clocks added (8B/10B link layer only). These may be separate clocks, but it may also be possible to reuse another clock (e.g., the link layer clock).
- Local extended multiblock clock (LEMC) added. This replaces the LMFC in the 64B/66B and 64B/80B link layers.
- No longer requirements for the LMFC in subclass 0, because subclass 0 devices can be designed without LMFC.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

A.2 Differences between JESD204B and JESD204C (cont'd)

- Complete revision of the skew budget, to remove the dependency on the frame clock. The skew budget is now based on the assumed minimum rates of the transport layer, link layer, SYNC~ generation and SYNC~ detection clocks. The assumptions behind the skew budget are documented in Annex C.
- The optional lane-to-lane inter-device synchronization interface for subclass 0 DAC devices has been replaced by the optional MULTIREF signal for subclass 1 converter devices. Device subclass 1 using the MULTIREF instead of the SYSREF signal realizes lane alignment across converter devices, but without deterministic latency.

4.5 Device classification

- Updated classes listed in subclause 4.5.1 to reflect the new data interface classes.
- Added subclause 4.5.3 (Features to be declared).

5 Physical layer specification

Revision A of the standard was expanded to support serial data interfaces consisting of single or multiple lanes per converter device. In addition, converter functionality (ADC or DAC) can be distributed over multiple devices:

- All parallel running devices are implemented or specified to run synchronously with each other using the same data format.
- Normally this means that they are part of the same product family.

Revision B of the standard added the following additional functions:

- Mechanism for achieving repeatable, programmable deterministic delay across the link.
- Support for serial data rates up to 12.5 Gbps.
- Transition from using frame clock as the main clock source to using device clock as the main clock source. Device clock frequency requirements offer much more flexibility compared to requiring a frame clock input.

Revision C of the standard implements the following functionality changes:

- Subsumes Revision B physical layer specification to category B devices, and (re)names Revision B LV-OIF-SxI5 operation as class B-3, Revision B LV-OIF-6G-SR operation as class B-6, and Revision B LV-OIF-11G-SR operation as class B-12. The differential insertion loss linear fit used for compliance of the transmission medium was changed to be scalable with data rate.
- Support for serial data rates up to 32.45 Gbps with raw[3] BER $\leq 10^{-15}$.
- Addition of three new receiver device classes to minimize link power dissipation.
- Link compliance based on channel operating margin (JCOM) methodology for the new device classes in JESD204C. Link compliance based on channel operating margin (COM) methodology in JESD204C.1

[3] Absent error correcting codes.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

A.2 Differences between JESD204B and JESD204C (cont'd)

6 Transport layer

This clause contains minor wording and stylistic updates compared with the JESD204B transport layer (clause 5.1). There is a single functional difference related to the test cycle length defined in clause 6.6.3. When using the 64B/66B or 64B/80B link layer, this length is changed compared with JESD204B clause 5.1.6.3.

7 64B/66B and 64B/80B link layer

The 64B/66B and 64B/80B link layer is a new clause in JESD204C.

8 8B/10B link layer

Clocking
The frame clock has essentially been removed from the link layer as well as from other parts of the standard. The frame clock is only mentioned in the context of backwards compatibility with earlier versions of the standard. In the link layer, the frame clock has been replaced by the link layer clock and the SYNC~ generation and detection clocks. These clocks are specified in more detail in 4.3.2 and 4.3.7. The standard does not require specific rates for these clocks or specific relationships between their rates and the frame rate.

The SYNC interface
Whereas previous versions of the standard only specified the hard-wired SYNC interface with the SYNC~ signal, 8.8.3 specifies the alternative soft SYNC interface. The soft SYNC interface makes use of one of the control interfaces mentioned in 4.4 to send synchronization requests and optionally error reports to the transmitter. Only in device subclass 2 the use of the SYNC~ signal is still mandatory.

In previous versions of the standard, the SYNC~ signal was generated on and sampled on the edges of the frame clock, which has the same rate in the TX and RX. In this version of the standard, the SYNC~ signal is generated on the edges of the SYNC~ generation clock and sampled on the edges of the SYNC~ detection clock. These rates of these clocks can, within some limitations, be chosen by the device implementer and they are not necessarily the same in the TX and the RX device. This means increased uncertainty in the moments when the TX detects activation and deactivation of the SYNC~ signal. Because the decision between interpreting an active SYNC~ signal as an error report or as a synchronization request is based on the detected duration of the activation, 8.8.2 requires a programmable duration of the error report generated in the RX and a programmable decision threshold in the detected duration in the TX in order to accommodate the uncertainty in the detection moments.

Other substantial changes

- A clarification added to table with link configuration parameters that only the encoding range is specified, not the range of supported parameter values in the device.
- Maximum number of frames *K* in a multiframe increased from 32 to 256, to have more freedom in the creation of multiframes with a duration longer than the maximum skew on fast links.
- /D21.5/ changed to an optional test sequence and /K28.7/ to a mandatory test sequence.
- There is no longer a requirement for a programmable length of the ILAS in logic devices. In previous versions, this requirement was thought to be necessary for the lane-to-lane inter-device synchronization interface, but this interface has been removed from the standard. The length of the ILAS is now always four multiframes.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

A.2 Differences between JESD204B and JESD204C (cont'd)

Device classification and sub-classification

JESD204B based the device classification on the capability to align lanes on links between single devices and on links between multiple devices. Starting from JES204C, the device classification is based on the capabilities of the physical layer (see clause 5). Therefore, all requirements from JESD204B that referred to a specific device class have been rewritten.

JESD204B defined device subclasses 0, 1 and 2, which specified the method by which deterministic latency was achieved on the link. Subclass 0 supports no deterministic latency. Subclass 1 uses the SYSREF signal and subclass 2 the SYNC~ signal to achieve deterministic latency. JESD204C made the several changes to the device subclasses. Most of these changes are related to the change in clocking (see "Clocking" in this Annex). Subclass 2 is specific to the 8B/10B link layer and is therefore defined in 0. Device subclasses 0 and 1 are applicable to all link layers and are therefore define in clause 4. JESD204C makes the following changes to the device subclasses:

Subclass 0 no longer supports lane alignment across ADC devices on a multipoint link using SYNC~ signal combination (see 8.4.2). This is because of the increased uncertainty in the moment that the TX detects deactivation of the SYNC~ signal (see "The Sync Interface" in this Annex). Also, the lane-to-lane inter-device synchronization interface for subclass 0 DAC devices has been removed from the standard. This means that subclass 0 no longer supports lane alignment across converter devices. Device classes 1 and 2 always support lane alignment across converter devices. Hence, the NMCDA (No Multiple-Converter Device Alignment) type of classes in JESD204B corresponds uniquely to subclass 0 in JESD204C.

Subclass 1 adds the option to generate the SYSREF signal in the converter device(s) instead of in the clock generator. If this option is used, the SYSREF signal is renamed to MULTIREF. If this option is used, there is no longer deterministic latency in subclass 1, but there is still lane alignment across converter devices. This option replaces the lane-to-lane inter-device synchronization interface for subclass 0 DAC devices and makes it also possible to align lanes across ADC devices without implementing deterministic latency. Otherwise, the changes to subclass 1 are related to the change in clocking and the use in combination with the 64B/66B and 64B/80B link layers, which are defined in clause 0.

Subclass 2 has remained functionally unchanged. All changes are related to the change in clocking.

Incompatibilities with previous revisions

The major changes to JESD204C that could result in incompatibilities with previous revisions of this standard are (see Annex M):

- The 64B/66B and 64B/80B link layers are not supported in JESD204A and JESD204B.
- JESD204A only supported the B-3 data interface, JESD204B supported all B- classes, and JESD204C supports all B- and C-classes.
- The timing of the SYNC~ signal in the 8B/10B link layer has been modified. Please refer to "The SYNC interface" in Annex A.
- JESD204C has increased skew limits, compared to JESD204A/B. A JESD204A/B receiver may not be able to align the lanes from JESD204C transmitter(s).

Differences between JESD204C and JESD204C.1

JESD204C.1 includes the following changes from JESD204C:

- The jitter parameters BHPCJ and BHPUJ were replaced with CBHPJ and UBHPJ, respectively. See, Table 23.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

A.2 Differences between JESD204B and JESD204C (cont'd)

- The receiver differential input return loss limit was missing from JESD204C, this was corrected by adding Figure 32 to JESD204C.1. The receiver input common mode return loss has been removed to make receiver specifications compatible with IEEE Std. 802.3-2018.

- In Table 23, TUJ was replaced with TJ, to reflect that the measured jitter includes correlated jitter.

- "Annex E (informative) Reference JCOM implementation" has been removed.

- The Class C differential return loss specification, described in 5.2.3.6 and 5.2.4.4, has been modified to be compatible with IEEE Std. 802.3.

- Removed division of Category C transmitters into classes.

- Several scattering parameter computations (subclauses 5.2.12.6.1, 5.2.12.6.2, 5.2.12.6.3, and 5.2.12.6.4 have been moved to "Annex D (informative) Scattering parameters computations".

- The JCOM device model has been removed.

- Various formatting and wording changes made throughout the document to improve clarity.

- Fixed broken cross-references throughout the document.

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex B (informative) Example of device clock and SYSREF generation

In practical systems there may be several JESD204 links connected to the same logic device. All these links could operate at different device clock, frame and multiframe rates. The clock generation and distribution circuit has the task of supplying the system with device clocks and SYSREF signals in such a way that the device clock edges on which the different devices capture SYSREF have predictable timing relations relative to each other.

Figure B.93 illustrates a typical implementation for generating the required clocks. The timing of the whole system is based on a main clock, which could be for instance a crystal oscillator operating in the 10 MHz range. It is often not convenient to derive the device clocks directly from such a low frequency signal. Instead, a higher frequency source clock may be used from which all device clocks are derived by frequency division. The source clock is phase locked to the main clock and would typically operate somewhere in the range of several hundred MHz to a few GHz. In addition to the device clocks, a timing reference for SYSREF is also derived from the source clock. This reference is designated as the SYSREF clock. In principle, the SYSREF clock could be permanently distributed directly to all devices in the system, but this is not always desirable from the perspective of electromagnetic interference. Therefore, JESD204 offers the possibility to generate a SYSREF pulse only on request of one of the JESD204 devices. Such a SYSREF request can be used to enable the output of the SYSREF clock from the clock generator circuit.

When the first device clock rising edge after each rising edge of the SYSREF clock can be compared directly to a rising edge of each LEMC/LMFC in the system, and they are aligned within some margin, no realignment will occur at repeated applications of a SYSREF pulse. When all LEMC/LMFC are at the same frequency, the SYSREF clock rate must be equal to the LEMC/LMFC rate or a subharmonic of it. When there are multiple LEMC/LMFC frequencies, the SYSREF clock must be a common subharmonic of all LEMC/LMFCs. This requirement determines the highest possible SYSREF clock frequency that can be used in the system. The lowest possible frequency is determined by the maximum supported division ratio in the clock distribution circuit.

Phase alignment between all clocks is ensured by a common reset signal to all frequency dividers in the clock distribution circuit, which is labelled as “Clock SYNC request” in Figure B.93. This signal will typically be generated at system start-up only. If a system uses multiple device clock frequencies, this signal can be used to properly align the SYSREF pulses with each other and to the device clock pulses.

Many commercial clock distribution devices contain programmable delays, by means of which the phases of the device clocks and SYSREF can be fine adjusted to match the setup and hold time requirements of the converter devices and logic device. It is also possible to split the enabled SYSREF clock output into multiple SYSREF signals, each with its own delay.

The timings of the circuit in Figure B.93 are illustrated in Figure B.94. For simplicity, no optional delays have been assumed. All dividers start counting at the first rising edge of the source clock after release of the Clock SYNC signal. All divider outputs change state at the rising edge of the source clock. All other clocks have their low-to-high transitions at the rising edge of the source clock.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex B (informative) Example of device clock and SYSREF generation (cont'd)

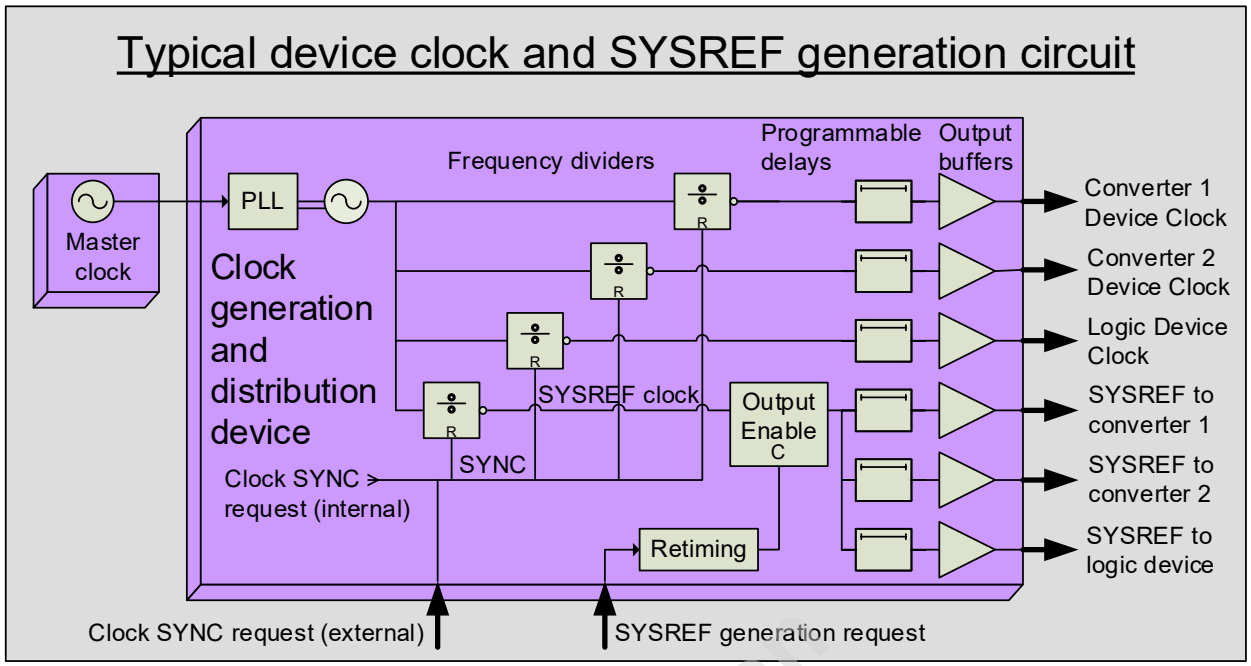


Figure B.93 — Typical implementation of device clock and SYSREF generation

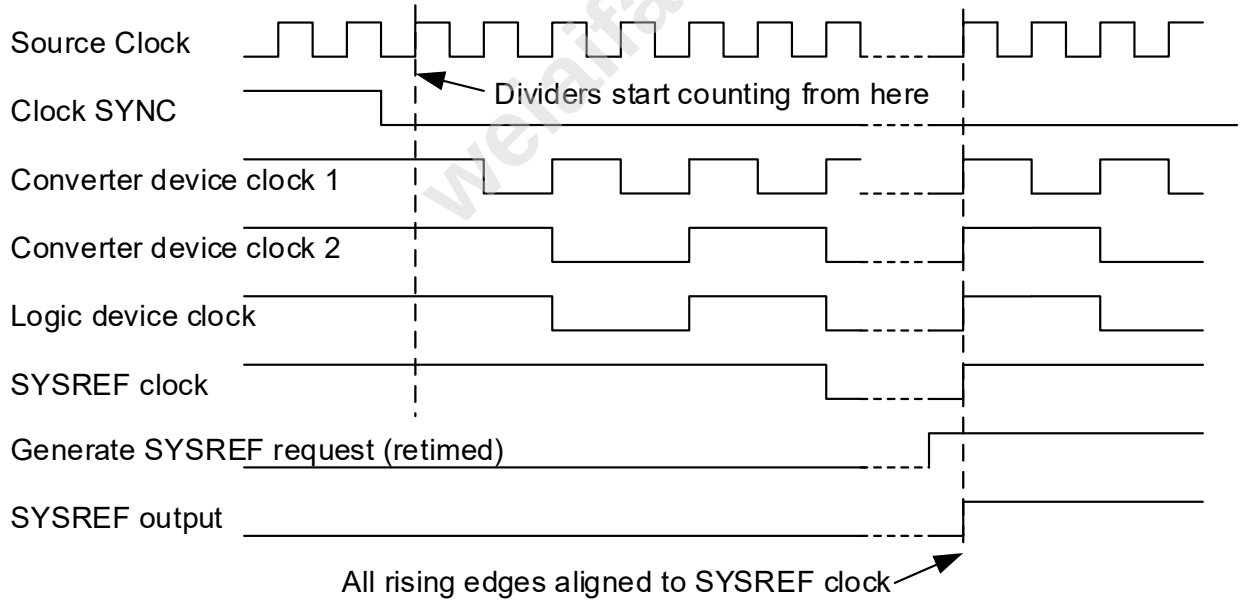


Figure B.94 — Timing diagram

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex C **(informative) Background of the values in the skew budget**

C.1 Transmission skew

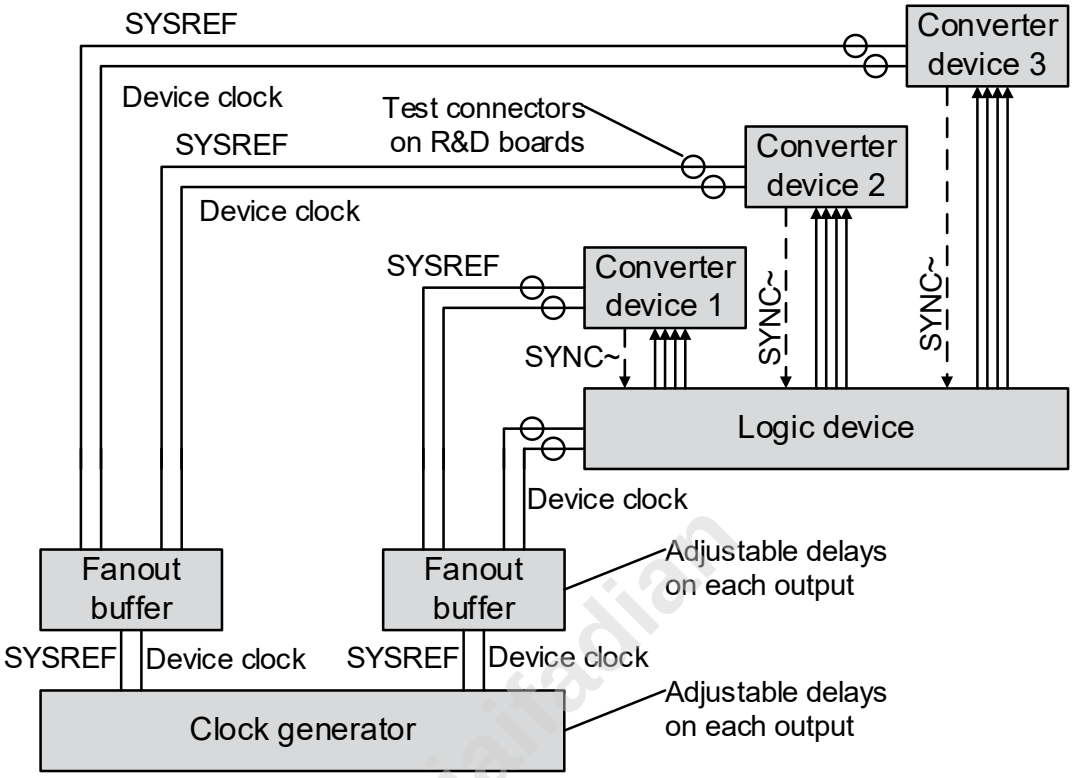


Figure C.95 — JESD204 system for illustration of transmission and timing reference skews

Figure C.95 shows a typical JESD204 system consisting of a clock generation and distribution system, one logic device and three converter devices (DACs in this example). The transmission skew between the data lanes is determined by the lane length differences. On a single link, these are not likely to be more than few centimeters. Therefore, it is enough to allocate 1 ns of transmission skew in the budget of a single link. However, on a multipoint link the lane length differences between devices could be close to the maximum supported length of the link. Allocation of 1 ns for the skew on a single link and 3 ns for the added skew on a multipoint link will support a maximum of 4 ns propagation delay difference between lanes, which corresponds e.g., to a lane length difference of 56 cm for stripline on an FR-4 board with ER=4.5. Larger lane length differences will not be likely in a JESD204 system.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

C.2 Timing reference skew

C.2.1 SYSREF and device clock skew

The routing of the device clock and SYSREF signals will generally not follow the data lanes and there may be significant differences in the propagation delays to the different JESD204 devices. However, the clock system designer will generally aim to minimize the skew variation in the system. For instance, each path will have the same number of fanout buffers, if used. More than one fanout buffer per path is not likely, because these buffers add jitter and increase cost. Device clock and SYSREF will be routed close to each other, to keep their mutual timing as predictable as possible. Clock generators and fanout buffers intended for JESD204 applications generally allow individual delay adjustments to each output, which can be used to compensate the delay differences in the clock and SYSREF distribution system. On an R&D board, it is possible to calibrate the whole system, using test connectors at the converter devices, so that the skew can be kept below 1 ns or less. However, it may not be practical to calibrate production boards or to compensate for environmental parameters. For instance, if the length difference between the longest and shortest SYSREF trace is 50 cm and the dielectric constant can vary between 4 and 5 over boards, humidity and temperature, the board will add a skew uncertainty of 400 ps. Several hundreds of ps may also be added by the PVT effects on the tracking between the delays of the output signals in the individual devices and between the devices in the clocking system. All in all, a SYSREF skew below 2 ns may not be a realistic goal. In addition, if meeting the setup and hold times of SYSREF versus device clock cannot be guaranteed, there will be (at least) one device clock cycle of added uncertainty in the timing of the internal clocks in the JESD204 devices relative to SYSREF. Consistently meeting the setup and hold time requirements gets challenging at device clock frequencies above 500 MHz and almost impossible at device clock frequencies above 1 GHz. Therefore, the limit for device clock and SYSREF skew has been increased by 2 ns at each side of the link, to a total of 6 ns.

C.2.2 Device clock and MULTIREF skew

The optional MULTIREF signal runs between the converter devices. It is generated on the edges of the device clock and detected on the edges of the device clock. The MULTIREF transmission skew is determined by the physical distances between these devices and could be up to 4 ns, like the multipoint transmission skew (see C.1). However, it is possible to compensate part of this skew by advancing the timing of the MULTIREF signal in the MULTIREF transmitting device. In case of a daisy chain distribution topology, each device in the chain will create extra skew. It is in principle possible to regenerate a periodic or gapped periodic MULTIREF signal with essentially zero-delay, but there will be an uncertainty of one device clock cycle if the setup and hold times at the MULTIREF input are violated. Like in the SYSREF skew budget (see C.2.1), one can assume a maximum uncertainty of 2 ns per MULTIREF input for violation of setup and hold times. Further, the position of the rising edge of the MULTIREF signal relative to the LMFC or LEMC may differ slightly between the devices. A maximum value of 2 ns for this skew is a reasonable assumption. Assuming a daisy chain topology with no more than eight converter devices per multipoint link and compensation of the MULTIREF transmission delays by advancing the MULTIREF output signals, the above leads to a maximum MULTIREF skew of 16 ns. If there are fewer converter devices on the multipoint link or the devices are located very close to each other, it may be possible to achieve this maximum skew without compensation of the MULTIREF transmission delays.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

C.2.3 SYNC~ skew

The SYNC~ signals, which may be used with 8B/10B encoding and affect the deterministic latency in device subclass 2, will generally follow the paths of the data lanes, but in opposite directions. Thus, the SYNC~ transmission skew will have similar magnitude as the transmission skew on a multipoint link (see C.1), in the order of up to 4 ns. However, the skew in generation and detection of the SYNC~ signals must be added.

EXAMPLE 1: Multipoint subclass 2 DAC link.

On a subclass 2 DAC link, the rising edge of the SYNC~ signal indicates the position of the active edge of the LMFC in the DAC device (see 8.8.2 for details). The logic device instructs the DAC device to align its LMFC to the LMFC in the logic device, based on the detected position of the rising edge of the received SYNC~ signal. Assume a multipoint DAC link with:

- SYNC~ generation on the DAC device clock edges,
- adjustment resolution in the DAC LMFC timing equal to one DAC device clock cycle,
- SYNC~ detection on the edges of link layer clock in the logic device, and
- the link layer clock in the logic device slower than the device clock of the DAC devices.

Because it is generally not possible to guarantee compliance to the setup and hold times at the SYNC~ inputs, the uncertainty at the detection side is one cycle of the link layer clock. Because the resolution in SYNC~ generation and LMFC adjustment in the DAC devices is finer than the resolution in SYNC~ detection in the logic device, the logic device can steer the timing of the LMFCs in the DAC devices such that the rising edges of all received SYNC~ signals are detected on the same active edge of the link layer clock. However, because of the detection uncertainty, the actual positions of the rising edges of the SYNC~ signals may differ by up to one link layer clock cycle. Further, the position of the rising edge of the SYNC~ signal relative to the LMFC may differ slightly between DAC devices. This SYNC~ generation skew is assumed to be no more than 2 ns. The maximum SYNC~ and device clock skew in this example is thus 2 ns (SYNC~ generation skew) + 4 ns (SYNC~ transmission skew) + T_{LLC} (SYNC~ detection skew).

EXAMPLE 2: Multipoint subclass 2 ADC link.

On a subclass 2 ADC link, the rising edge of the SYNC~ signal indicates the position of the active edge of the LMFC in the logic device. The ADC device adjusts the timing of its LMFC to the detected position of the rising edge of the received SYNC~ signal. Assume a multipoint ADC link with:

- SYNC~ detection on the edges of the link layer clock in the ADC device, and
- a whole number of ADC LMFC adjustment steps in one link layer clock cycle.

The adjustment resolution of the ADC LMFCs is such that the ADCs can exactly align their LMFCs to the detected position of the rising edge of the received SYNC~ signals. However, because the link layer clocks are not synchronized between the ADCs, the detected positions may differ by up to one link layer clock period. In addition, it is possible that the setup and hold times are not met at all SYNC~ inputs. If so, the rising edge of the SYNC~ signal will be detected on the next following active edge of the link layer clock. However, non-compliance to the setup and hold times will only happen in those ADCs whose active link layer clock edge is "early" with respect of the rising SYNC~ edge. Thus, a delay of the detection moment by one link layer clock cycle will only happen in the ADCs with an "early" detection clock, not in the ADCs with a "late" detection clock. Hence, not meeting the setup and hold times will not increase the maximum difference in detected positions of the rising SYNC~ edge between ADC devices.

In the logic device, the position of the rising edge of the SYNC~ signal relative to the LMFC may differ slightly between the links. This SYNC~ generation skew is assumed to be no more than 2 ns. The maximum SYNC~ and device clock skew in this example is thus 2 ns (SYNC~ generation skew) + 4 ns (SYNC~ transmission skew) + T_{LLC} (SYNC~ detection skew).

Withdrawn
Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

C.3 Processing skew in logic device

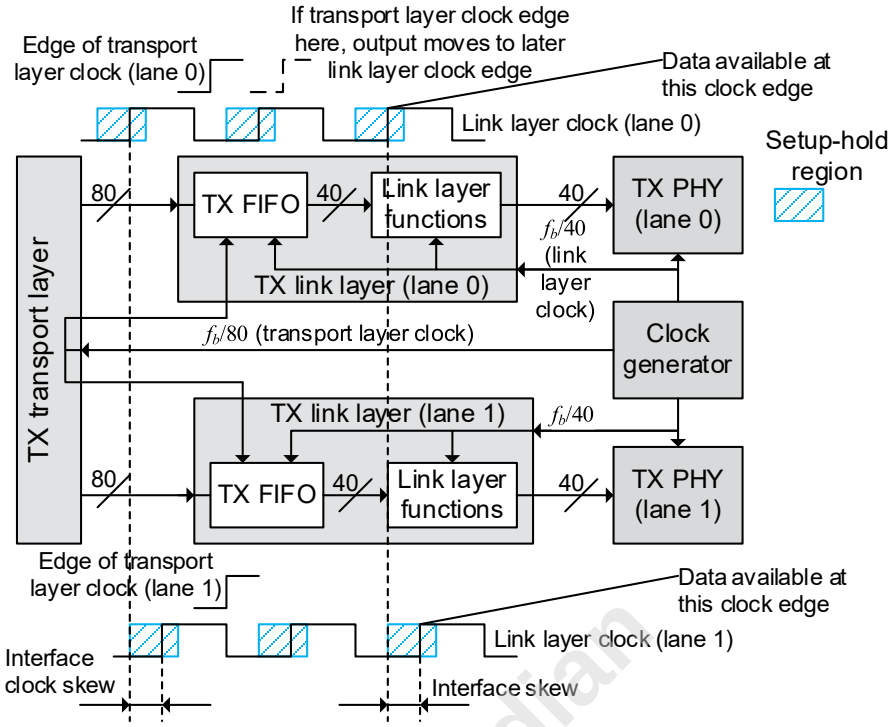


Figure C.96 — Typical implementation of an 8B/10B or 64B/80B TX in a logic device

Figure C.96 illustrates a typical implementation of an 8B/10B or 64B/80B TX in a logic device. The physical layer supports only specific ratios between the serial clock and the parallel clock. Commonly supported ratios are 16, 20, 32, 40, 64 and 80. For 8B/10B and 64B/80B encoding, it only makes sense to use ratios of 20, 40 or 80. In this example, a ratio of 40 is used. Hence, the link layer clock is $f_b/40$, where f_b is the serial bit rate. The link layer and the PHY of a single lane form usually one IP block and have a common clock. Therefore, it is possible to avoid the delay uncertainty involved with a clock domain crossing on this interface.

The transport layer is part of the user logic and is clocked separately from the link layer. Further, as shown in this example, the transport layer clock and link layer clock could even operate at a different frequency. Therefore, the link layer input contains a FIFO to mitigate the effects of the inaccurate timing relation between the bit transitions in the input word and the edges of the link layer clock. At the FIFO input, the data is sampled at the rising edges of the transport layer clock. By careful routing of the transport layer clock and the data signals, it is possible to ensure there are no setup and hold issues at the input of the FIFO. However, the timing relationship between the transport layer clock and the link layer clock is uncertain. If the edge of the transport layer clock moves to another side of the setup-hold region of the link layer clock, the timing of the data at the output of the FIFO changes by one period of the link layer clock. As demonstrated by the figure, if there is no setup-hold region between the arrival times of the data words in the different link layers, then the skew caused by the interface equals the skew between the interface clocks. However, if the input data on one lane is read in just before a setup-hold region of the link layer clock and the input data on another lane just after a setup-hold region, the data on the latter lane will become available on a later interface clock edge and the resulting skew is almost a full link layer clock cycle.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

C.3 Processing skew in logic device (cont'd)

Hence, with an unknown timing relation between the transport layer and link layer clock inputs of the FIFOs, the clock domain crossing causes a delay uncertainty of one cycle of the of the link layer clock. This uncertainty comes on top of the skew between the arrivals of the transport layer clocks at the FIFOs. However, if a sufficiently accurate timing relationship can be guaranteed between the transport layer clock and the link layer clock to the FIFOs, the total skew caused by the interface equals the skew of the link layer clock and there is no uncertainty in the delay caused by the FIFOs. The latter assumption was made in the JESD204A standard and this was copied into JESD204B. However, as the data and clock rates go up and the size of the logic devices is increasing, this assumption may no longer be safe. Therefore, JESD204C allows a delay uncertainty of one link layer clock cycle due to the clock domain crossing in the TX link layer.

The TX PHY also contains a clock domain crossing, from the link layer clock to the serial clock domain (not shown). Because of the high speed of the serial clock, it is generally not possible to achieve deterministic latency on this clock domain crossing. Therefore, an uncertainty of one cycle in the receiving clock domain is assumed, i.e., 1 UI.

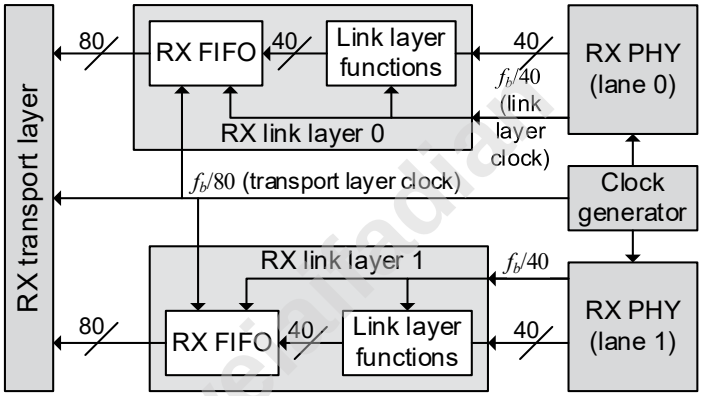


Figure C.97 — Typical implementation of an 8B/10B or 64B/80B RX in a logic device

Figure C.97 shows a typical implementation of an 8B/10B or 64B/80B receiver in a logic device. From the clocking perspective, the implementation is similar to the transmitter, except for the following differences:

- The data at the output of the clock domain crossing in the PHY is sampled by the link layer clock.
- The RX link layer is clocked by the recovered clock from the PHY.
- The data at the output of the clock domain crossing in the link layer is sampled by the transport layer clock.

Because the link layer clock of the receiver is generated by the physical layer, the timing relationship between the link layer clock and the transport layer clock is unpredictable. Therefore, the clock domain crossing in the link layer causes an inherent delay uncertainty of one transport layer clock cycle. On the clock domain crossing from the physical layer to the link layer, the delay uncertainty is one link layer clock cycle, because the delay of the first serial bit of a data frame to the next edge of the link layer clock is unpredictable.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

C.3 Processing skew in logic device (cont'd)

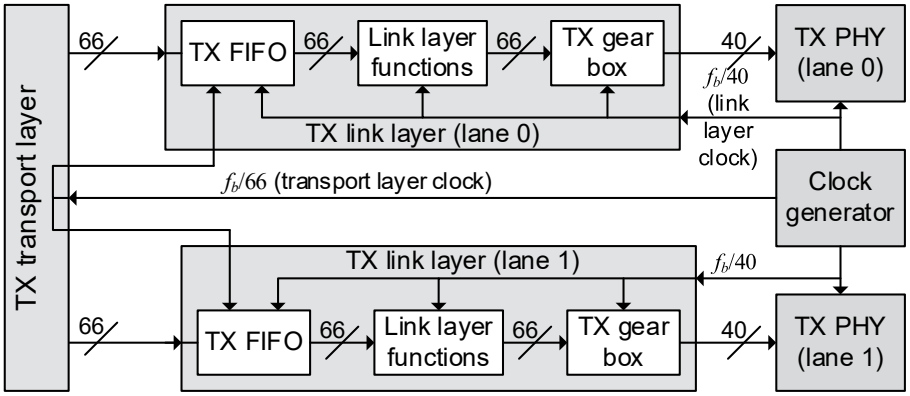


Figure C.98 — Typical implementation of a 64B/66B TX in a logic device

Figure C.98 shows a typical implementation of a 64B/66B TX in a logic device. Because the link layer clock and transport layer clock have no integer frequency relationship, the timing relationship between the edges of these clocks is variable. Hence, the delay uncertainty in the TX FIFOs is one cycle of the link layer clock, which is 40 UI in this example.

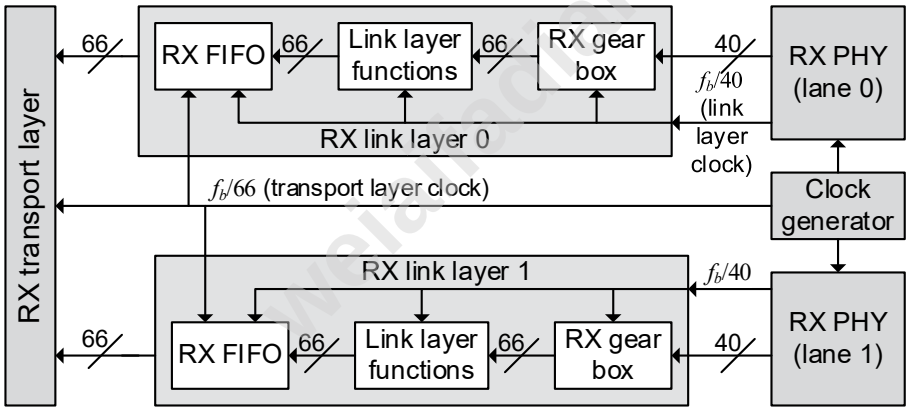


Figure C.99 — Typical implementation of a 64B/66B RX in a logic device

Figure C.99 shows a typical implementation of a 64B/66B RX in a logic device. From the clocking perspective, the differences with the TX are the same as for the 8B/10B RX:

- The data at the output of the clock domain crossing in the PHY is sampled by the link layer clock.
- The RX link layer is clocked by the recovered clock from the PHY.
- The data at the output of the clock domain crossing is sampled by the transport layer clock.

Between link layers of different links, the analysis is the same. However, because the link layers belonging to different links may be far away from each other, it will be even more challenging to accurately align the arrival of the data words and transport layer clocks across different links. The timing of the transport layer clock relative to the interface clock may be as shown with the solid line for lane 0 in Figure C.96 on one link and as shown with the dashed line on another link. Therefore, on multipoint links always the worst-case skew must be assumed.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

C.3 Processing skew in logic device (cont'd)

The skew limits must be based on assumptions on the periods of the link layer and the transport layer clocks. The 10GBASE-X specification of the Ethernet standard [7], on which JESD204A has been based, assumes a link layer clock of $f_b/20$= 156.25 MHz at f_b = 3.125 GHz. JESD204B scaled the skew limit in UI proportionally to the line rate at line rates above 3.125 Gbps. This would imply that any ratio between line rate and link layer clock rate were possible. However, in practice only a few discrete ratios are supported. To keep the specification simple and applicable to all encoding types, the skew budget assumes that only serial-to-link clock ratios of U_{LLC}= 20, 40 and 80 are possible. Further it is assumed that all devices that will be developed for this standard will support a link layer clock of up to 200 MHz if the serial bit rate is up to 16 Gbps. Consequently, a link layer clock period of 20 UI will be possible at data rates up to 4 Gbps, a link layer clock period of 40 UI must be assumed between rates of 4 and 8 Gbps and 80 UI between 8 and 16 Gbps. Devices supporting faster rates than 16 Gbps will use faster technology and can therefore be assumed to support higher link layer clock frequencies, so that the link layer clock period can be kept at 80 UI. The transport layer clock rate is assumed to be equal to the link layer clock rate for 8B/10B and 64B/80B encoding and to $f_b/66$ for 64B/66B encoding. The transport layer clock rate is only relevant in the clock domain crossing from the link layer to the transport layer in the receiver, where the delay uncertainty in the budget is informative.

Table C.59 and Table C.60 show how the above assumptions lead to the skew limit in the TX and RX.

Table C.59 — Partitioning of the skew limit in the TX (link layer and PHY)

Source / notes	Maximum added skew	
	Single link	Multipoint link
Sum of delay tolerances on all internal interfaces, without the effect of clock domain crossings. The value is the skew limit of the JESD204A TX, after subtraction of 1 UI uncertainty in the PHY.	1.5 ns	4.8 ns
Clock domain crossing in link layer	U_{LLC} UI	
Clock domain crossing (parallel-to-serial) in physical layer	1 UI	
Total skew limit	(U_{LLC}+1) UI + 1.5 ns	(U_{LLC}+1) UI + 4.8 ns
NOTE U_{LLC} = 20 for $f_b \leq$ 4 GHz, 40 for 4 GHz < $f_b \leq$ 8 GHz, and 80 for f_b > 8 GHz.		

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

C.3 Processing skew in logic device (cont'd)

Table C.60 — Partitioning of the skew limit in the RX (link layer and PHY)

Source / notes	Maximum added skew	
	Single link	Multipoint link
Clock domain crossing (serial-to-parallel) in physical layer	U_{LLC} UI	
Clock domain crossing in link layer	U_{TLC} UI	
Sum of delay tolerances on all internal interfaces, without the effect of clock domain crossings. The value is the skew limit of the JESD204A TX, after subtraction of 1 UI uncertainty in the PHY.	1.5 ns	4.8 ns
Total skew limit	(U_{LLC}+ U_{TLC}) UI + 1.5 ns	(U_{LLC}+ U_{LLC}) UI + 4.8 ns
NOTE 1 U_{LLC} = 20 for $f_b \leq 4$ GHz, 40 for 4 GHz < $f_b \leq 8$ GHz, and 80 for $f_b > 8$ GHz. NOTE 2 $U_{TLC} = U_{LLC}$ for 8B/10B and 64B/80B encoding and 66 for 64B/66B encoding.		

C.4 Processing skew in converter device

The JESD204 link in a converter device has the same clock boundaries as the logic device. In converter devices, it may be possible to have better alignment between clocks in different domains than in logic devices. However, there is no advantage in assuming so. ADC lanes are deskewed in logic devices, in which it is easier to implement long deskew buffers than in converter devices. Further, on deterministic latency links, the multipoint skew sets the requirement for the buffer length. Even when it may be possible to avoid skew on clock domain crossings inside a converter device, it is less likely that the delay on clock domain crossings is equal in different converter devices of the multipoint link or that the delay is reproducible between startups or power resets. In DAC devices, the skew produced by the receiver is only informative. If a DAC device produces less skew than specified by the standard, then the DAC device can use a shorter receive buffer.

For these reasons, the same skew limits for converter devices and logic devices are assumed. Also, the delay tolerance is modelled at the same place as in the logic device, although the mechanisms causing the delay tolerance are different.

C.5 Skew variation

The maximum values of skew variation have been copied from the 40 Gb/s Ethernet specification in [9]. This specification allows a variation of 0.2 ns between successive skew points, except for the transmission skew, which may be larger. The Ethernet standard specifies 0.2 ns skew variation separately for the link layer and physical layer, so that JESD204 allows 0.4 ns skew variation for their combination. In JESD204, a 0.2 ns variation in transmission skews is a reasonable assumption. This value is 5% of the maximum assumed transmission skew on a multipoint link (see C.1) and therefore corresponds to a maximum 10% variation in dielectric constant during link operation. On multipoint and deterministic latency links, there may be a significant effect of the delay variations in the timing reference signals: device clock and SYSREF or SYNC~. The maximum skew variation due to these signals has been adjusted to keep the end value of the skew budget the same as in the Ethernet specification.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex D **(informative) Scattering parameters computations**

D.1 Cascade connection of two-port networks

The connection of a pair of two-port networks x and y such that port 2 of network x is connected to port 1 of network y may be represented by an equivalent two-port network z. Port 1 of network z corresponds to port 1 of network x and port 2 network z corresponds to port 2 of network y. The scattering parameters of network z are given in terms of the scattering parameters of networks x and y using the transfer scattering matrix $\boldsymbol{T}$, (D.48).

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix} \cdot \begin{bmatrix} b_2 \\ a_2 \end{bmatrix} = \boldsymbol{T} \cdot \begin{bmatrix} b_2 \\ a_2 \end{bmatrix} \tag{D.48}$$

Thus the scattering-to-transfer scattering matrix conversion, denoted as $S2T(\cdot)$, and the reverse transfer scattering matrix-to-scattering matrix, denoted as $T2S(\cdot)$, are defines as (D.49) – (D.50).

$$\boldsymbol{S} = \begin{bmatrix} \frac{T_{21}}{T_{11}} & \frac{T_{11} \cdot T_{22} - T_{12} \cdot T_{21}}{T_{11}} \\ \frac{1}{T_{11}} & -\frac{T_{12}}{T_{11}} \end{bmatrix} \tag{D.49}$$

$$\boldsymbol{T} = \begin{bmatrix} \frac{1}{S_{21}} & -\frac{S_{22}}{S_{21}} \\ \frac{S_{11}}{S_{21}} & \frac{S_{12} \cdot S_{21} - S_{11} \cdot S_{22}}{S_{21}} \end{bmatrix} \tag{D.50}$$

The scattering parameters of a block z, $\boldsymbol{S}^{(z)}$, which is the cascade combination of two blocks, x and y, characterized by their scattering parameters $\boldsymbol{S}^{(x)}$ and $\boldsymbol{S}^{(y)}$ can be obtained as in (D.51).

$$\boldsymbol{S}^{(z)} = cascade\left(\boldsymbol{S}^{(x)}, \boldsymbol{S}^{(y)}\right) = T2S\left(S2T\left(\boldsymbol{S}^{(x)}\right) \cdot S2T\left(\boldsymbol{S}^{(y)}\right)\right) \tag{D.51}$$

D.2 Renormalization of reference port impedances

A system with n-ports may be represented by a scattering parameter matrix $\boldsymbol{S}$ using reference port impedances $\boldsymbol{Z} = [Z_1, Z_2, \cdots, Z_n]$. Z_i $(i = 1,2, \cdots, n)$ represents the reference impedance for port i. The same system may be represented by a different scattering parameter matrix $\boldsymbol{S'}$ using different reference port impedances $\boldsymbol{Z'} = [Z_1', Z_2', \cdots Z_n']$. Z_i' $(i = 1,2, \cdots, n)$ represents the different reference impedance for port i. The matix $\boldsymbol{S'}$ can be computed from the matrix $\boldsymbol{S}$ per (D.52).

$$\boldsymbol{S'} = renorm(\boldsymbol{S}, \boldsymbol{Z}, \boldsymbol{Z'}) = \boldsymbol{A}^{-1} \cdot (\boldsymbol{S} - \boldsymbol{R}) \cdot (\boldsymbol{I} - \boldsymbol{R} \cdot \boldsymbol{S})^{-1} \cdot \boldsymbol{A} \tag{D.52}$$

with $\boldsymbol{I}$ is the identity matrix and:

$$\boldsymbol{A} = \begin{pmatrix} \sqrt{\frac{Z_1'}{Z_1}} \cdot \frac{1}{Z_1' + Z_1} & 0 & \cdots & 0 \\ 0 & \sqrt{\frac{Z_2'}{Z_2}} \cdot \frac{1}{Z_2' + Z_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sqrt{\frac{Z_n'}{Z_n}} \cdot \frac{1}{Z_n' + Z_n} \end{pmatrix} \tag{D.53}$$

welefadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

D.2 Renormalization of reference port impedances (cont'd)

$$\boldsymbol{R} = \begin{pmatrix} \frac{Z_1' - Z_1}{Z_1' + Z_1} & 0 & \cdots & 0 \\ 0 & \frac{Z_2' - Z_2}{Z_2' + Z_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{Z_n' - Z_n}{Z_n' + Z_n} \end{pmatrix} \quad \text{(D.54)}$$

D.3 Two-port network for a differential shunt capacitance

IEEE Std. 802.3-2018 93A.1.2.2. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The scattering parameters with a reference impedance Z_0 for a shunt capacitance with value C are defined by (D.55) where $j = \sqrt{-1}$ and $\omega = 2 \cdot \pi \cdot f$.

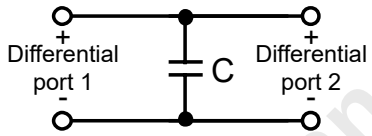


Figure D.100 — Differential shunt capacitor

$$\mathbf{S_C}(C) = \frac{1}{2 + j \cdot \omega \cdot Z_0 \cdot C} \cdot \begin{bmatrix} -j \cdot \omega \cdot Z_0 \cdot C & 2 \\ 2 & -j \cdot \omega \cdot Z_0 \cdot C \end{bmatrix} \quad \text{(D.55)}$$

D.4 Two-port network for a differential serial inductor

The scattering parameters with a reference impedance Z_0 for a series resistance with value R and a series inductance with value L are defined by (D.56) where $j = \sqrt{-1}$ and $\omega = 2 \cdot \pi \cdot f$.

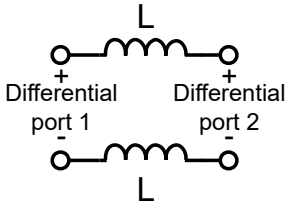


Figure D.101 — Differential series inductance

$$\mathbf{S_L}(L) = \frac{1}{j \cdot \omega \cdot L + Z_0} \cdot \begin{bmatrix} j \cdot \omega \cdot L & Z_0 \\ Z_0 & j \cdot \omega \cdot L \end{bmatrix} \quad \text{(D.56)}$$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

D.5 Two-port network for a differential series resistance-inductance combination

The scattering parameters with a reference impedance Z_0 for a series resistance with value R and a series inductance with value L are defined by (D.57) where $j = \sqrt{-1}$ and $\omega = 2 \cdot \pi \cdot f$.

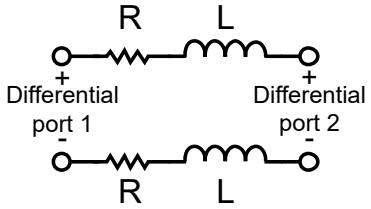


Figure D.102 — Differential series resistance-inductance combination

$$\mathbf{S_{RL}}(R,L) = \frac{1}{(R + j \cdot \omega \cdot L) + Z_0} \cdot \begin{bmatrix} R + j \cdot \omega \cdot L & Z_0 \\ Z_0 & R + j \cdot \omega \cdot L \end{bmatrix} \quad \text{(D.57)}$$

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex E **(normative for category B) Linear insertion loss fit**

The weighted fitted differential insertion loss as a function of frequency is given by (E.58) , f_b is the channel data rate.

$$IL_{fitted}(f) = a_0 + a_2 \cdot \frac{f}{f_b} \quad \text{[dB]} \tag{E.58}$$

Denote the differential insertion loss, in dB, measured at frequency f_n as $IL(f_n)$. Given the differential insertion loss measured at N uniformly-spaced frequencies from start frequency $f_{min} = 50$ MHz to stop frequency $f_{max} = f_b$ with step no larger than $\Delta f = 10$ MHz, the coefficients for the fitted differential insertion loss shall be calculated as follows: define the weighted frequency matrix $\boldsymbol{F}$ using (E.59).

$$\boldsymbol{F} = \begin{bmatrix} 10^{\frac{-IL(f_1)}{20}} & \frac{f_1}{f_b} \cdot 10^{\frac{-IL(f_1)}{20}} \\ 10^{\frac{-IL(f_2)}{20}} & \frac{f_2}{f_b} \cdot 10^{\frac{-IL(f_2)}{20}} \\ \vdots & \vdots \\ 10^{\frac{-IL(f_N)}{20}} & \frac{f_N}{f_b} \cdot 10^{\frac{-IL(f_N)}{20}} \end{bmatrix} \tag{E.59}$$

Define the weighted insertion loss vector $\boldsymbol{L}$ using (F.68).

$$\boldsymbol{L} = \begin{bmatrix} IL(f_1) \cdot 10^{\frac{-IL(f_1)}{20}} \\ IL(f_2) \cdot 10^{\frac{-IL(f_2)}{20}} \\ \vdots \\ IL(f_N) \cdot 10^{\frac{-IL(f_N)}{20}} \end{bmatrix} \tag{E.60}$$

The fitted differential insertion loss coefficients are then given by (E.61), where T denotes the matrix transpose operator.

$$\begin{bmatrix} a_0 \\ a_2 \end{bmatrix} = (\boldsymbol{F}^T \cdot \boldsymbol{F})^{-1} \cdot \boldsymbol{F}^T \cdot \boldsymbol{L} \tag{E.61}$$

This polynomial fit process is expected to yield values for the coefficients a_0 and a_2 that are greater than their minimum and less than their maximum as specified in Table E.61. If a coefficient in the equation is below its minimum allowed value it is forced to its minimum value and the fitting process is iterated. Iteration is done by creating a new IL, IL' by subtracting the coefficient below the minimum allowed value from the original IL, removing the coefficient from matrix $\boldsymbol{F}$ and recalculating the remaining coefficient. At the end of the iteration, limit all coefficients to the maximum allowed, followed by a final iteration on the coefficient not previously limited.

For each iteration only one additional coefficient should be forced to a value. If both coefficients are below their minimum or above their maximum then the coefficients should be forced to a value in the following order: a_2 and then a_0.

weraidian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex E (normative for category B) Linear insertion loss fit (cont'd)

Example iteration: if a_2 needs to be set to zero, but a_0 is within its range, then calculate new IL and solve for a_0 as indicated below.

$$IL' = IL - \left(a_{2_{fixed}} \cdot \frac{f}{f_b}\right) \tag{E.62}$$

Define the frequency matrix $\boldsymbol{F'}$ using (E.63).

$$\boldsymbol{F'} = \begin{bmatrix} 10^{\frac{-IL'(f_1)}{20}} \\ 10^{\frac{-IL'(f_2)}{20}} \\ \vdots \\ 10^{\frac{-IL'(f_N)}{20}} \end{bmatrix} \tag{E.63}$$

And the weighted insertion loss vector $\boldsymbol{L'}$ using (E.64).

$$\boldsymbol{L'} = \begin{bmatrix} IL'(f_1) \cdot 10^{\frac{-IL'(f_1)}{20}} \\ IL'(f_2) \cdot 10^{\frac{-IL'(f_2)}{20}} \\ \vdots \\ IL'(f_N) \cdot 10^{\frac{-IL'(f_N)}{20}} \end{bmatrix} \tag{E.64}$$

The polynomial coefficients a_0 is determined using the equation below.

$$[a_0] = \left(\boldsymbol{F'}^T \cdot \boldsymbol{F'}\right)^{-1} \cdot \boldsymbol{F'}^T \cdot \boldsymbol{L'} \tag{E.65}$$

Table E.61 — Category B linearly-fitted differential insertion loss characteristics

Parameter	Symbol	Min.	Max.	Units
Frequency range	$f_{IL_{min}}, f_{IL_{max}}$	0.05	f_b	GHz
Fitted insertion loss coefficient 0	a_0	-1	1.500	dB
Fitted insertion loss coefficient 2	a_2	0	30.855	dB

weijiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex F **(normative for category C) Quadratic insertion loss fit**

F.1 Differential insertion loss

Based on OIF-CEI-04.0 12.2.1.1, © 2017 Optical Internetworking Forum

The weighted fitted differential insertion loss as a function of frequency is given by (F.66), f_b is the channel data rate.

$$IL_{fitted}(f) = a_0 + a_1 \cdot \sqrt{\frac{f}{f_b}} + a_2 \cdot \frac{f}{f_b} + a_4 \cdot \left(\frac{f}{f_b}\right)^2 \quad \text{[dB]} \tag{F.66}$$

Denote the differential insertion loss, in dB, measured at frequency f_n as $IL(f_n)$. Given the differential insertion loss measured at N uniformly-spaced frequencies from start frequency $f_{min} = 50$ MHz to stop frequency $f_{max} = f_b$ with step no larger than $\Delta f = 10$ MHz, the coefficients for the fitted differential insertion loss shall be calculated as follows: define the weighted frequency matrix $\boldsymbol{F}$ using (F.67).

$$\boldsymbol{F} = \begin{bmatrix} 10^{\frac{-IL(f_1)}{20}} & \sqrt{\frac{f_1}{f_b}} \cdot 10^{\frac{-IL(f_1)}{20}} & \frac{f_1}{f_b} \cdot 10^{\frac{-IL(f_1)}{20}} & \left(\frac{f_1}{f_b}\right)^2 \cdot 10^{\frac{-IL(f_1)}{20}} \\ 10^{\frac{-IL(f_2)}{20}} & \sqrt{\frac{f_2}{f_b}} \cdot 10^{\frac{-IL(f_2)}{20}} & \frac{f_2}{f_b} \cdot 10^{\frac{-IL(f_2)}{20}} & \left(\frac{f_2}{f_b}\right)^2 \cdot 10^{\frac{-IL(f_2)}{20}} \\ \vdots & \vdots & \vdots & \vdots \\ 10^{\frac{-IL(f_N)}{20}} & \sqrt{\frac{f_N}{f_b}} \cdot 10^{\frac{-IL(f_N)}{20}} & \frac{f_N}{f_b} \cdot 10^{\frac{-IL(f_N)}{20}} & \left(\frac{f_N}{f_b}\right)^2 \cdot 10^{\frac{-IL(f_N)}{20}} \end{bmatrix} \tag{F.67}$$

Define the weighted insertion loss vector $\boldsymbol{L}$ using (F.68).

$$\boldsymbol{L} = \begin{bmatrix} IL(f_1) \cdot 10^{\frac{-IL(f_1)}{20}} \\ IL(f_2) \cdot 10^{\frac{-IL(f_2)}{20}} \\ \vdots \\ IL(f_N) \cdot 10^{\frac{-IL(f_N)}{20}} \end{bmatrix} \tag{F.68}$$

The fitted differential insertion loss coefficients are then given by (F.69), where T denotes the matrix transpose operator.

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_4 \end{bmatrix} = (\boldsymbol{F}^T \cdot \boldsymbol{F})^{-1} \cdot \boldsymbol{F}^T \cdot \boldsymbol{L} \tag{F.69}$$

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

F.1 Differential insertion loss (cont'd)

This polynomial fit process is expected to yield values for the coefficients a_0, a_1, a_2, and a_4 that are greater than the minimum and less than the maximum coefficients (minimum and maximum values for class C-S channels specified in Table F.62, for class C-M channels in Table F.63 and for class C-R channels in Table F.64). If any of the coefficients in the equation are below the minimum allowed value they are forced to the minimum value and the fitting process is iterated (see example below). Iteration is done by creating a new IL, IL' by subtracting all coefficients below the minimum allowed value from the original IL, removing those coefficients from matrix $\boldsymbol{F}$ and recalculating the remaining coefficients (see example below). At the end of the iteration, limit all coefficients to the maximum allowed, followed by a final iteration on any coefficients not previously limited.

For each iteration only one additional coefficient shall be forced to a value. If multiple coefficients are below the minimum or above the maximum then the coefficients shall be forced to a value in the following order: a_4 followed by a_1 followed by a_2 and lastly a_0.

Example iteration: if a_2 needs to be set to zero, but all other coefficients are within the range, then calculate new IL and solve for a_0, a_1 and a_4 as indicated below.

$$IL' = IL - \left(a_{2_{fixed}} \cdot \frac{f}{f_b}\right) \tag{F.70}$$

Define the frequency matrix $\boldsymbol{F'}$ using (F.71).

$$\boldsymbol{F'} = \begin{bmatrix} 10^{\frac{-IL'(f_1)}{20}} & \sqrt{\frac{f_1}{f_b}} \cdot 10^{\frac{-IL'(f_1)}{20}} & \left(\frac{f_1}{f_b}\right)^2 \cdot 10^{\frac{-IL'(f_1)}{20}} \\ 10^{\frac{-IL'(f_2)}{20}} & \sqrt{\frac{f_2}{f_b}} \cdot 10^{\frac{-IL'(f_2)}{20}} & \left(\frac{f_2}{f_b}\right)^2 \cdot 10^{\frac{-IL'(f_2)}{20}} \\ \vdots & \vdots & \vdots \\ 10^{\frac{-IL'(f_N)}{20}} & \sqrt{\frac{f_N}{f_b}} \cdot 10^{\frac{-IL'(f_N)}{20}} & \left(\frac{f_N}{f_b}\right)^2 \cdot 10^{\frac{-IL'(f_N)}{20}} \end{bmatrix} \tag{F.71}$$

And the weighted insertion loss vector $\boldsymbol{L'}$ using (F.72).

$$\boldsymbol{L'} = \begin{bmatrix} IL'(f_1) \cdot 10^{\frac{-IL'(f_1)}{20}} \\ IL'(f_2) \cdot 10^{\frac{-IL'(f_2)}{20}} \\ \vdots \\ IL'(f_N) \cdot 10^{\frac{-IL'(f_N)}{20}} \end{bmatrix} \tag{F.72}$$

The polynomial coefficients a_0, a_1, and a_4 are determined using the equation below.

$$\begin{bmatrix} a_0 \\ a_1 \\ a_4 \end{bmatrix} = \left(\boldsymbol{F'}^T \cdot \boldsymbol{F'}\right)^{-1} \cdot \boldsymbol{F'}^T \cdot \boldsymbol{L'} \tag{F.73}$$

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

F.1 Differential insertion loss (cont'd)

Table F.62 — Class C-S channel fitted differential insertion loss characteristics

Parameter	Symbol	Min.	Max.	Units
Frequency range	$f_{IL_{min}}$, $f_{IL_{max}}$	0.05	f_b	GHz
Fitted insertion loss at $\frac{f_b}{2}$, half of the maximum data rate			12.5	dB
Fitted insertion loss coefficient 0	a_0	-1	1.200	dB
Fitted insertion loss coefficient 1	a_1	0	5.684	dB
Fitted insertion loss coefficient 2	a_2	0	27.319	dB
Fitted insertion loss coefficient 4	a_4	0	12.265	dB

Table F.63 — Class C-M channel fitted differential insertion loss characteristics

Parameter	Symbol	Min.	Max.	Units
Frequency range	$f_{IL_{min}}$, $f_{IL_{max}}$	0.05	f_b	GHz
Fitted insertion loss at $\frac{f_b}{2}$, half of the maximum data rate			22.5	dB
Fitted insertion loss coefficient 0	a_0	-1	2.000	dB
Fitted insertion loss coefficient 1	a_1	0	17.513	dB
Fitted insertion loss coefficient 2	a_2	0	48.997	dB
Fitted insertion loss coefficient 4	a_4	0	23.897	dB

Table F.64 — Class C-R channel fitted differential insertion loss characteristics

Parameter	Symbol	Min.	Max.	Units
Frequency range	$f_{IL_{min}}$, $f_{IL_{max}}$	0.05	f_b	GHz
Fitted insertion loss at $\frac{f_b}{2}$, half of the maximum data rate			27.0	dB
Fitted insertion loss coefficient 0	a_0	-1	2.000	dB
Fitted insertion loss coefficient 1	a_1	0	21.590	dB
Fitted insertion loss coefficient 2	a_2	0	54.034	dB
Fitted insertion loss coefficient 4	a_4	0	26.600	dB

Note: The coefficient values in Table F.62, Table F.63 and Table F.64 are based on OIF-CEI-04.0 Table 10-4, Table 14-13 and Table 11-4, ©2017 Optical Internetworking Forum.

weitaidian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

F.2 Differential insertion loss deviation

IEEE Std. 802.3-2018 93A.4. Adapted and reprinted with permission from IEEE.
©2018 The Institute of Electrical and Electronics Engineers, Incorporated. All rights reserved.

The differential insertion loss deviation ILD is the difference between the measured differential insertion loss IL and the fitted differential insertion loss IL_{fitted} as defined in (F.66).

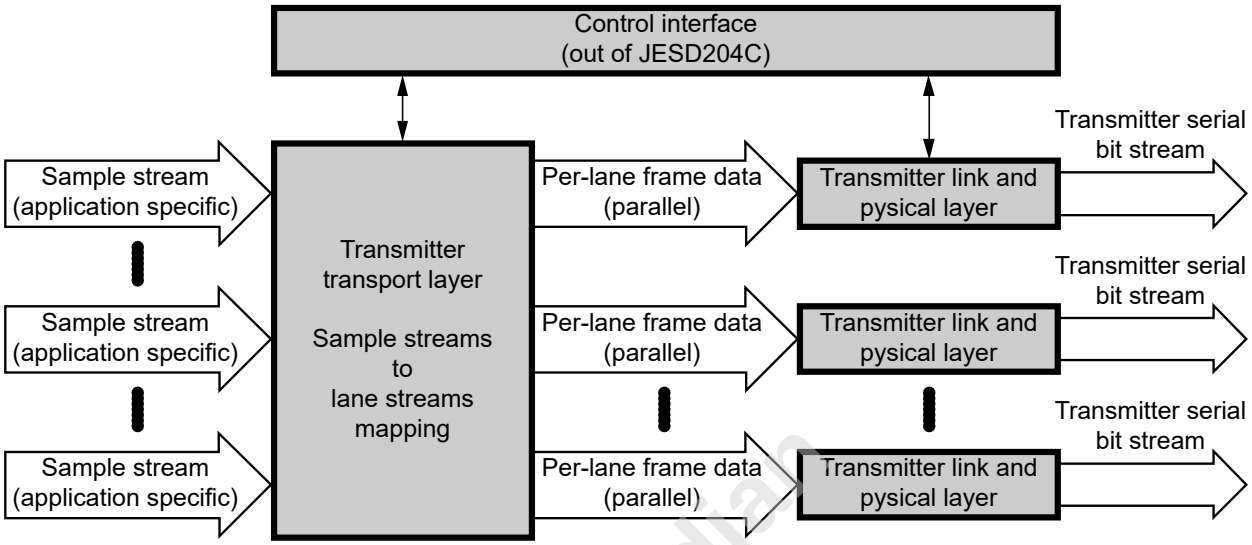
weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex G **(informative) Physical layer implementation**

G.1 Transmitter

The purpose of the transmitter block is to take one or more digital sample streams and convert them to one or more serial streams. Figure J.103 shows the generic transmitter structure of a single link. The same structure can be used for both an ADC converter device and a logic device.



NOTE Transport layer specified in clause 0, link layer specified in clauses 0 and 0.

Figure J.103 — Generic structure of a single-link transmitter block

The functionality of the transmitter device can be divided into a transport layer and one or more link and physical layers. In the transport layer, the incoming stream or streams of samples are mapped to one or more parallel lanes with frame data. The transport layer is specified in clause 0. On the link layer, frames are encoded as a stream using a given encoding scheme (8B/10B, 64/66B, etc.), which are transmitted via the physical layer as a serial bit stream across the transmission medium. The link layer is specified in clause 0 for the 64B/66B and 64B/80B option, and clause 0 for the 8B/10B option. Error reporting and link synchronization requests are handled by higher layers of the communication stack.

The transmitter physical layer interfaces the transmitter link layer to the transmission medium. Its operation is illustrated in Figure J.104. Data leaving the parallel to serial converter is passed to a pulse shaping/pre-emphasis block. After the pulse shaping/pre-emphasis, the signal is then sent to the differential driver circuit that connects to the receiver circuit using a suitable transmission line.

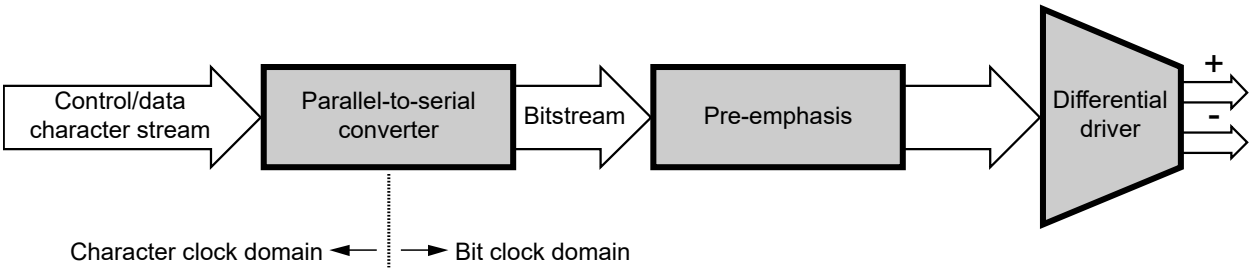


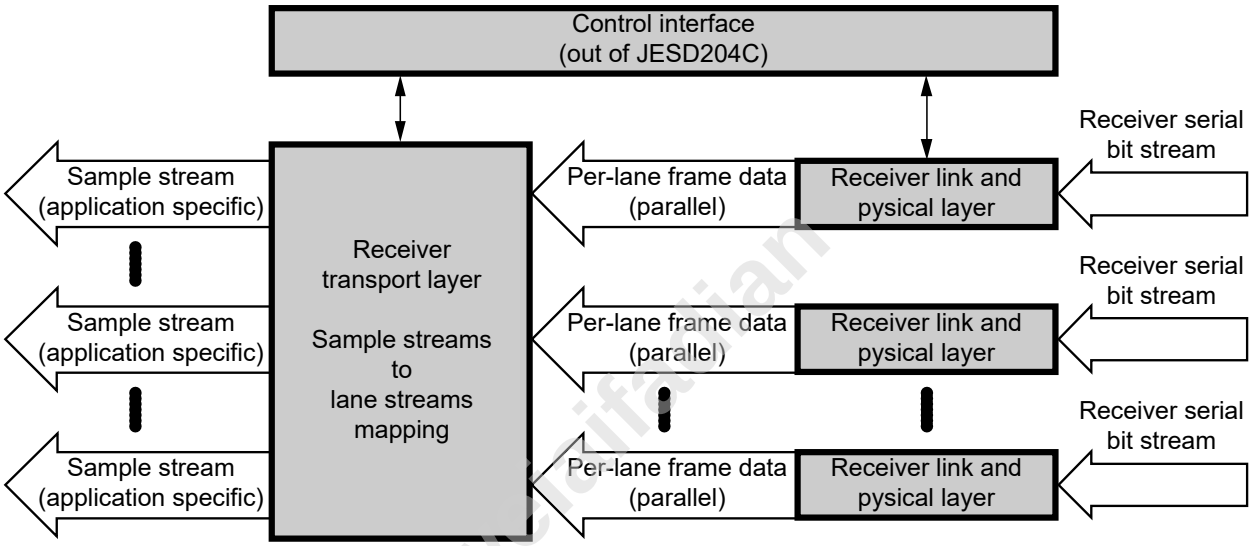
Figure J.104 — Transmitter physical layer

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

G.2 Receiver

The receiver block captures the serial stream(s) from one or more transmitter blocks and converts the stream(s) into one or more sample streams. The same generic receiver structure of Figure G.105 can be used in both a DAC and a logic device.

The functionality of the receiver device can be divided into a transport layer and one or more link and physical layers. In the physical layer, the incoming data stream is used to recover the line clock, which is then used to de-serialize the data. In the link layer the de-serialized data is then decoded and reorganized into frames. Frames are then processed by the receiver transport layer where they are mapped into one or more sample streams. Error reporting and link synchronization requests are handled by higher layers of the communication stack. The link layer is specified in clause 0 for the 64B/66B and 64B/80B option, and clause 0 for the 8B/10B option, while the transport layer is specified in clause 0.



NOTE Transport layer specified in clause 0, link layer specified in clauses 0 and 0.

Figure G.105 — Generic structure of a receiver block

The receiver physical layer interfaces the transmission medium to the receiver link layer. The operation of the RX physical layer is illustrated in Figure G.106. The incoming signal is passed to an equalizer; beyond this, the data stream enters clock recovery and is de-serialized, after which it is passed to the device's link layer for further processing.

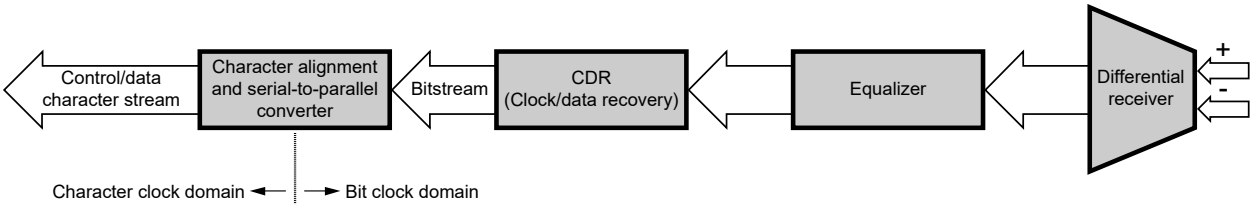


Figure G.106 — Receiver physical layer

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex H **(normative) Pseudo-random binary sequence (PRBS) generation**

A PRBSn pattern, where $n \in [9,15,31]$ shall be generated with an n-stage shift-register whose mth and kth stage outputs are modulo-two added and fed back to the input of the first stage, thus implementing the polynomial $x^{\mathrm{k}} + x^{\mathrm{m}} + 1$. The (m, k) values corresponding to the different values of n are specified in Table G.65.The binary data sequence $d(i)$, $i = 1, \cdots, 2^n - 1$ is given by $d(i) = mod_2\big(d(i - \mathrm{k}) + d(i - \mathrm{m})\big)$. The pattern has a run of n ones and $n - 1$ zeros in its length of $2^n - 1$ bits. The shift registers shall be preloaded with all ones, which makes the pattern start with n consecutive ones.

Table G.65 — PRBS polynomial taps

Length (n)	k	m
9	9	5
15	15	14
31	31	28

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex I **(informative) Transport layer configuration parameters**

Table I.1 summarizes the transport layer configuration parameters.

Table I.1 — Transport layer configuration parameters

Parameter	Description	Defined in
CF	No. of control words per frame duration per link	3.2, 6.2.2
CS	No. of control bits per sample	3.2, 6.2.2
F	No. of octets per frame	3.2, 6.1
HD	High Density format	3.2, 6.3
L	No. of lanes per converter device (link)	3.2, 6.3
M	No. of converters per device	3.2, 6.2.2
N	Converter resolution	3.2, 6.1
N'	Total no. of bits per sample	3.2, 6.1
S	No. of samples per converter per frame cycle	3.2, 6.1

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex J (informative) Forward error correction decoding

This annex briefly describes the basis behind the FEC decoding circuitry, as well as an example FEC decoder.

J.1 Binary cyclic codes decoding

Consider a binary cyclic (n, k) code generated using a polynomial g(x) of degree r. This code consists of k information bits and r = n-k parity bits. Let c(x) be the transmitted codeword and r(x) be the received word. If r(x) = c(x), i.e., the word is received without errors, then s(x), the syndrome of r(x), is:

$$s(x) = r(x) \bmod g(x) = c(x) \bmod g(x) = 0$$

Therefore, an error-free received word has a syndrome of 0 and we can proceed to forwarding the received word without correction. Now, assume that the received word is affected by a burst error e(x), then we have:

$$r(x) = c(x) + e(x),$$

where e(x) is the error burst polynomial with non-zero coefficients for the erroneous bits. We, then, have:

$$s(x) = r(x) \bmod g(x) = (c(x) + e(x)) \bmod g(x)$$

$$= c(x) \bmod g(x) + e(x) \bmod g(x)$$

$$= e(x) \bmod g(x)$$

If the degree of the polynomial e(x) is less than the degree of the generator polynomial g(x), i.e., the burst error is located within the parity bits of r(x), then the above syndrome expression simplifies to s(x) = e(x) and we can correct r(x) by "XOR"ing it with s(x). This is illustrated in Figure J.107:

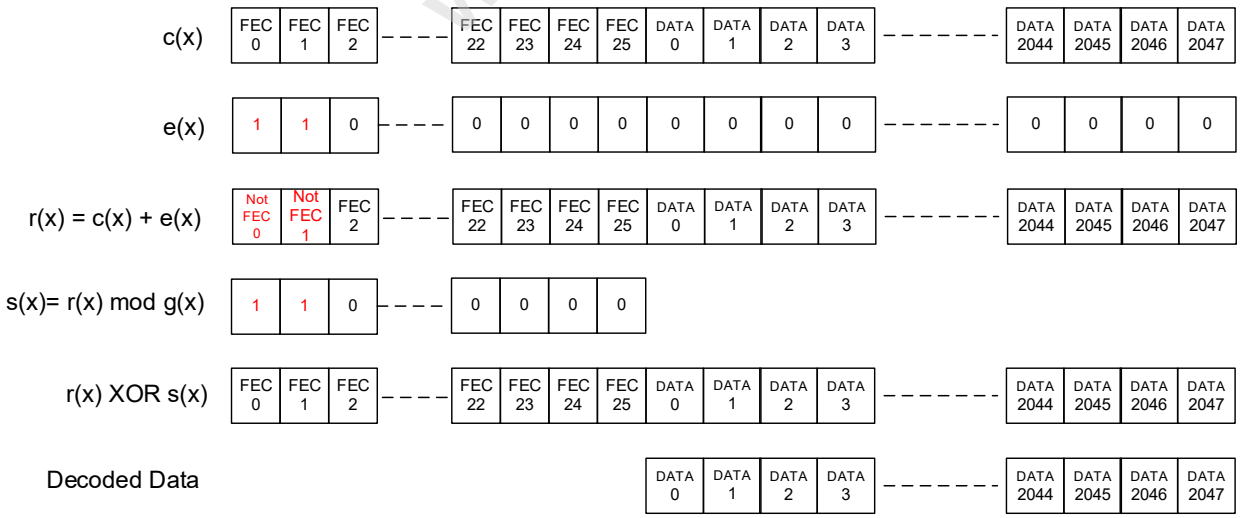


Figure J.107 — Decoding example of a correctable error located within parity bits

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

J.1 Binary cyclic codes decoding (cont'd)

If the degree of the polynomial e(x) is greater than the degree of the generator polynomial g(x), i.e., the burst error is not contained within the parity bits, then s(x) = e(x) mod g(x) ≠ e(x) as shown below:

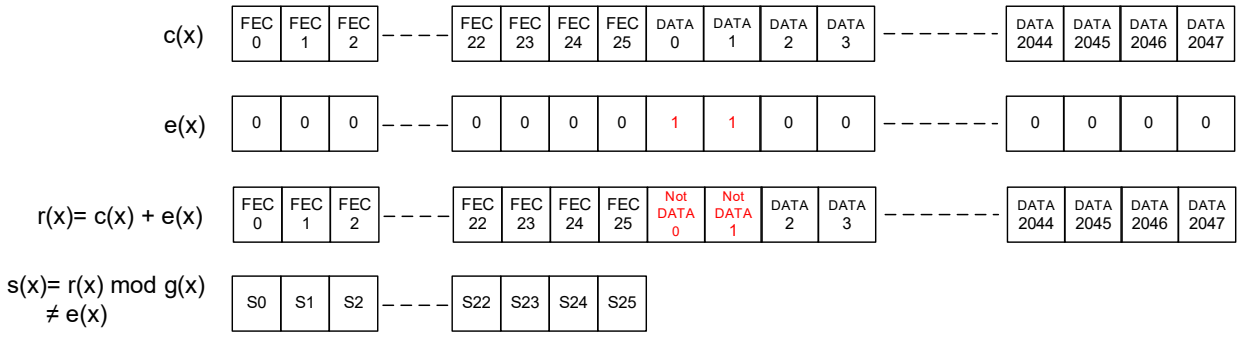


Figure J.108 — s(x) ≠ e(x), when degree(e(x)) > degree (g(x))

We can, however, rotate the received word r(x) until the burst error is trapped within the least significant bits of the rotated word. The syndrome of the rotated word would then simply be equal to the burst error rotated by the number of bits the received word was rotated by, as will be proven below. Once the burst error is "trapped" within the least significant bits of the received word, the bits in error can be corrected using a simple XOR operation, as illustrated in Figure J.109:

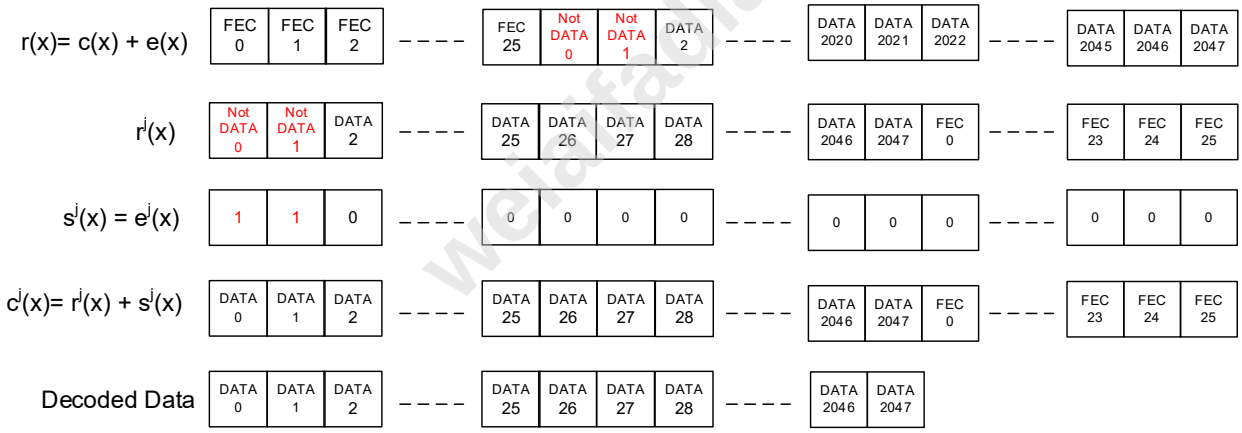


Figure J.109 — Error correction using error trapping

In Figure J.109, define $r^i(x)$ to denote r(x) rotated by i bits and define $s^i(x)$ to be the syndrome of $r^i(x)$. Let j be the number of bits needed to rotate r(x) such that all bits in error are located within the least significant bits of $r^j(x)$ (we assume that the error is a single burst error whose length falls within the correction capabilities of the selected FEC). The syndrome $s^j(x)$ of the rotated word $r^j(x)$ can then be calculated as follows:

$$s^j(x) = r^j(x) \bmod g(x) = (c^j(x) + e^j(x)) \bmod g(x)$$
$$= c^j(x) \bmod g(x) + e^j(x) \bmod g(x)$$
$$= e^j(x) \bmod g(x)$$
$$= e^j(x)$$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

J.1 Binary cyclic codes decoding (cont'd)

Note that $c^j(x) \bmod g(x) = 0$ by definition since c(x) is a cyclic code and $e^j(x) \bmod g(x) = e^j(x)$, since the degree of the polynomial $e^j(x)$ is less than the degree of the polynomial g(x).

The above description suggests then that the decoding/correction process can then be achieved by computing the syndrome $s^0(x)$, $s^1(x)\ldots s^{n-1}(x)$, trapping the error and then correcting the received word when the error is trapped. Computing the syndromes $s^0(x)$, $s^1(x)\ldots s^{n-1}(x)$ can be simplified by using a Meggitt decoder, which makes use of the following relationship:

$$s^{j+1}(x) = r^{j+1}(x) \bmod g(x)$$

$$= x\, r^j(x) \bmod x^n\text{-}1 \bmod g(x)$$

$$= x\, r^j(x) \bmod g(x), \text{ since } g(x) \text{ divides } x^n\text{-}1$$

$$= x\, (r^j(x) \bmod g(x)) \bmod g(x)$$

$$= x\, s^j(x) \bmod g(x)$$

Otherwise stated, to compute $s^{j+1}(x)$, the syndrome of $r^{j+1}(x)$, right shift the syndrome $s^j(x)$ of $r^j(x)$ and then divide it by g(x); The remainder of this operation is the syndrome $s^{j+1}(x)$. The circuit of Figure J.110 shows a hardware implementation for computing the syndrome $s^{j+1}(x)$ from $s^j(x)$ assuming the generator polynomial is $g(x) = x^{26} + x^{21} + x^{17} + x^9 + x^4 + 1$.

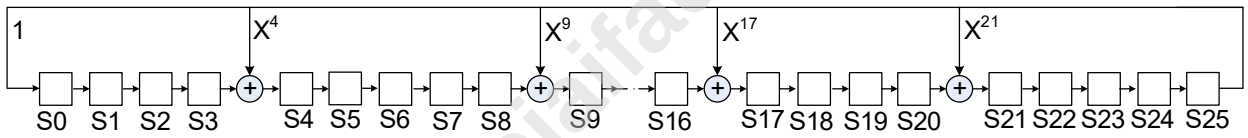


Figure J.110 — Syndrome Computation for rotated word

J.2 Shortened cyclic codes decoding

A specific application may only have k-l information bits to encode, but may require the error correction capability of an (n, k) code. This application can benefit from shortening the cyclic (n, k) code to an (n-l, k-l) code, where the shortened code is taught of having l "0s" of information bits appended to k-l relevant information bits.

The encoding process for a shortened cyclic (n-l, k-l) is no different than the encoding process of a cyclic (n, k) code. The encoder state of the register that generates the parity bits is reset to all 0s. The leading 0s of a shortened code will not cause a change in the state of the encoder. Hence, the encoding process can be performed by directly feeding in the k-l relevant information bits to the encoder. The parity FEC bits shall be computed when the last of the k-l information bits is fed to the encoder.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

J.2 Shortened cyclic codes decoding (cont'd)

The decoding process for a shortened (n-l, k-l) can use the same Meggitt decoder as that of non-shortened (n, k) code, but it will be inefficient as it will require up to n clock cycles (or k clock cycles if we only are interested in decoding the information bits) to perform this operation. The inefficiency stems from the fact that one zero is assumed to precede the received word and therefore calculating the syndrome of the rotated codeword, where the relevant information starts, takes l extra syndrome computations (or n-l if the codeword is rotated counterclockwise). The additional step in computing the syndrome of the shortened codeword can be eliminated as follows:

Let p(x) be the remainder of dividing x^{n-k+l} by g(x). We then have:

$$p(x) = x^{n-k+l} + a(x)\, g(x)$$

Multiply both sides of the above equation by r(x) yields:

$$r(x)p(x) = r(x)\, x^{n-k+l} + r(x)\, a(x)\, g(x)$$

and then apply mod g(x) to both sides:

$$\begin{aligned} r(x)\, p(x) \bmod g(x) &= r(x)\, x^{n-k+l} \bmod g(x) + r(x)\, a(x)\, g(x) \bmod g(x) \\ &= r(x)\, x^{n-k+l} \bmod g(x) \\ &= r(x)\, x^{n-k+l} \bmod x^{n}\text{-}1 \bmod g(x), \text{ since } g(x) \text{ divides } x^{n}\text{-}1 \\ &= r^{n-k+l}(x) \bmod g(x) \\ &= s^{n-k+l}(x) \end{aligned}$$

Therefore, in order to compute the syndrome of $r^{n-k+l}(x)$, first calculate $p(x) = x^{n-k+l} \bmod g(x)$, then generate r(x) p(x) mod g(x). Figure J.111 shows the circuit to compute r(x) p(x) mod g(x), where

$$p(x) = x^{24} + x^{21} + x^{19} + x^{18} + x^{17} + x^{9} + x^{7} + x^{4} + x^{2} + x + 1$$

$$g(x) = x^{26} + x^{21} + x^{17} + x^{9} + x^{4} + 1$$

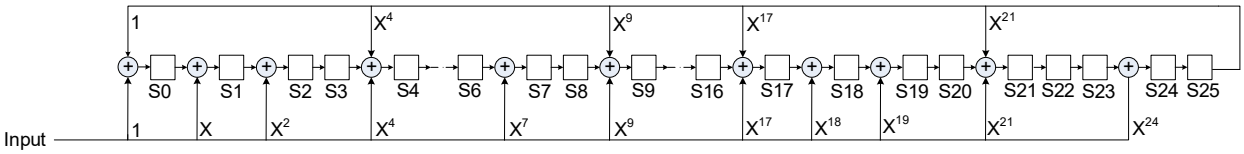


Figure J.111 — Shortened Codeword Syndrome Computation

Note that the calculated syndrome $s^{n-k+l}(x)$ not only rotates the syndrome l times to account for the shortened codeword, but it also rotates it an additional n-k times. The extra n-k additional syndrome rotations ensure that the error is trapped on the most significant bits of the syndrome register correspond to the errors on the most significant bits of the received word, hence simplifying the correction process. Figure J.112 illustrates and example where the error affects the most significant bits of the received word. The trapped error in $s^{n-k+l}(x)$ is aligned with the erroneous bits on the received word:

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

J.2 Shortened cyclic codes decoding (cont'd)

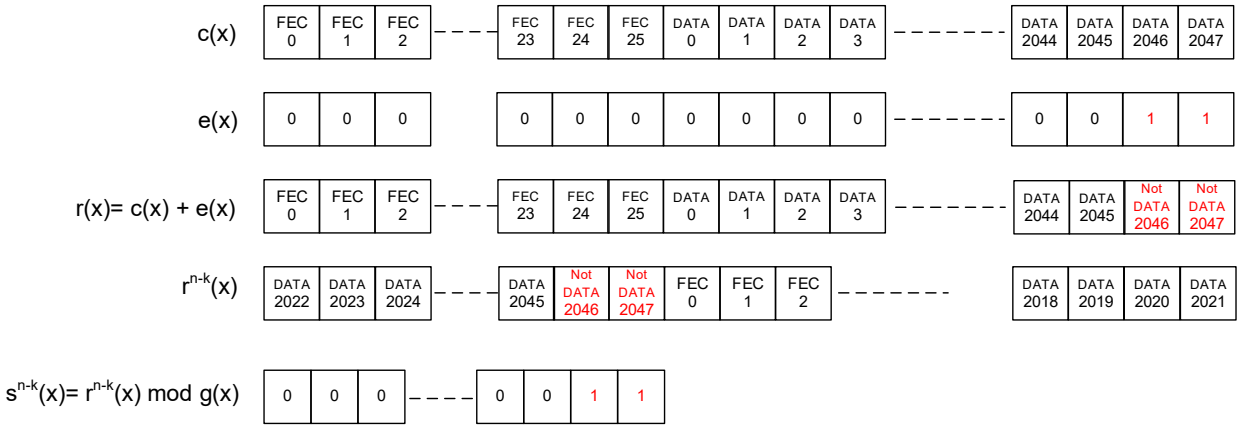


Figure J.112 — Rotating the syndrome by n-k bits aligns the error with the received word

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex K **(informative) Clock Terminology for 64B/66B and 64B/80B link layer**

Table K.66 — Clock ratios with respect to extended multiblock clock

Clock name	Ratio (w.r.t. LEMC)	Notes
SYSREF Clock**	1 / *R*	*R* = integer
device clock*	x *D*	*D* = integer in case of subclass 1
Local Extended Multiblock Clock (LEMC)*	(1)	
frame clock	x *K*	*K* = 256·*E*/*F*
sample clock	x *K* x *S*	
conversion clock	x *K* x *S* x *C*	*C* = interpolation- or decimation-factor
multiblock clock	x *E*	
block clock	x *E* x 32	
bit clock	x *E* x 32 x *BkW*	
octet clock	x *K* x *F* or x *E* x 32 x 8	
*- Required physical signals ** - SYSREF Clock is defined in JESD204C clause 4.3.5		

weijifadian

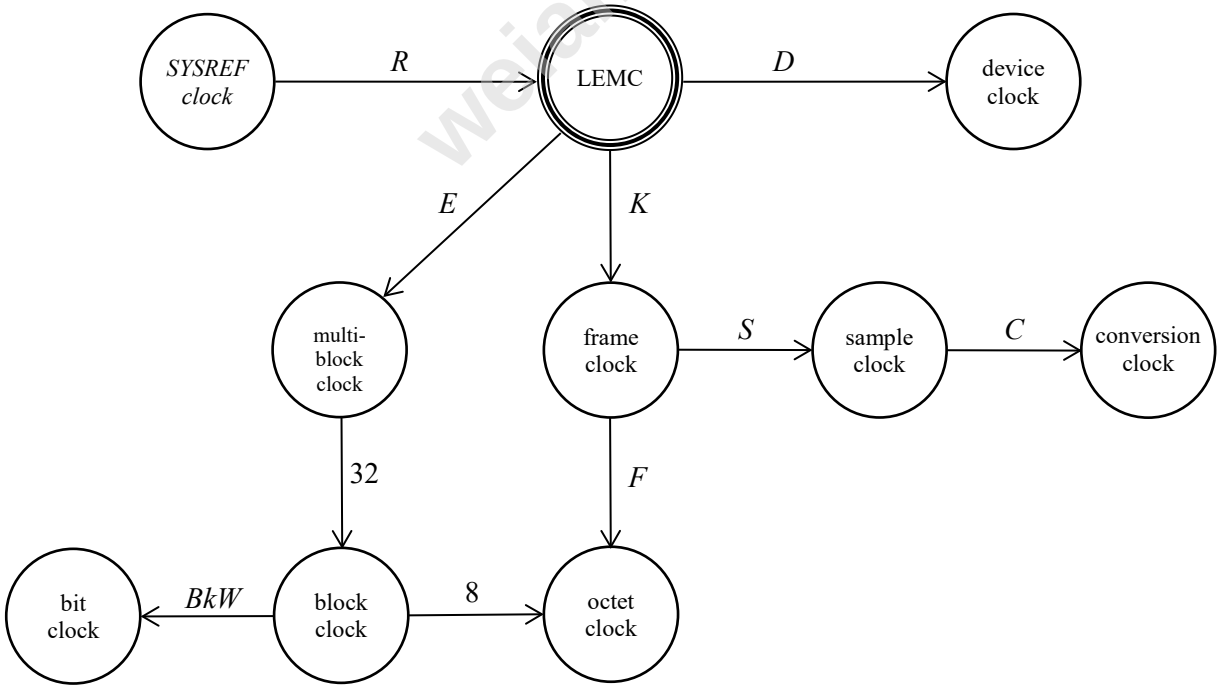


Figure K.113 — Relation diagram for JESD204 clocks

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex L **(informative) Clock Terminology for 8B/10B link layer**

Table L.67 — Clock ratios with respect to the multiframe clock

Clock name	Ratio (w.r.t. multiframeclock)	Notes
SYSREF Clock **	1 / *R*	*R* = integer
(Local) MultiFrame Clock (LMFC)*	(1)	
Frame clock	× *K*	ceil (17/F) ≤ K ≤ min(256, floor(1024/F))
Character clock	× *K* × *F*	*F*
Bit clock	× *K* × *F* × 10	8B/10B encoding
Sample clock	× *K* × *S*	*S*
Conversion clock	× *K* × *S* × *C*	*C* = interpolation- or decimation-factor
Device clock *	× *D*	*D* = integer in case of subclass 1, 2
*- Required physical signals (LMFC not required in subclass 0). ** - SYSREF Clock is defined in 4.3.5.		

weijiafadian

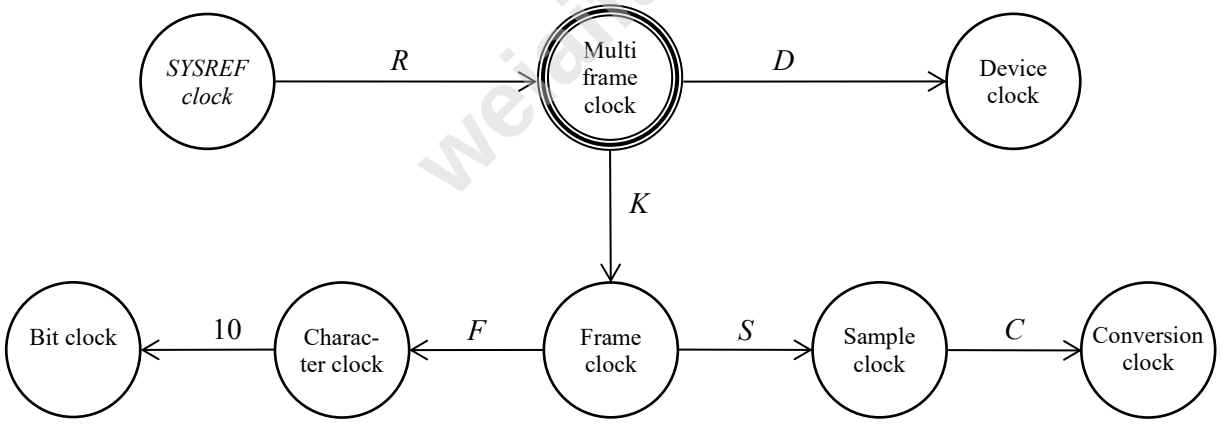


Figure L.114 — Relation diagram for JESD204 clocks

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex M (informative) Backward compatibility of 8B/10B link layer with previous versions of JESD204

JESD204A and JESD204B impose an upper limit of 32 to *K*, the number of frames per multiframe. JESD204C increases this limit to 256, to have more freedom in the creation of multiframes with a duration longer than the maximum specified skew. *K*-values above 32 should not be used in the JESD204A or JESD204B compatible mode.

JESD204C introduces the possibility to use a soft SYNC interface, i.e., communicating synchronization requests and error reports over the control interface. JESD204 (initial version) through JESD204B support only the hard-wired SYNC interface. Therefore, a JESD204C device that supports the 8b/10B link layer must have a hard-wired SYNC interface for interoperability with a JESD204 through JESD204B device.

The timing specifications for the hard-wired SYNC interface are driven by the capabilities (1) to guarantee detection of an error report over this interface by the TX and (2) to distinguish an error report from a synchronization request. The timing specifications differ between the different versions of this standard, because the assumptions on the clocking scheme have changed.

JESD204 (initial version) and JESD204A assumed an external frame clock. The same frame clock was supplied to all devices.

In JESD204B, the external frame clock was replaced by an external device clock. Different devices could use different device clocks, but all device clocks should be generated from a common source clock and be phase-locked to each other. The frame clock is an internal (local) signal in JESD204B.

In the JESD204C 8B/10B link layer there is no change to the device clocks. However, the existence of a local frame clock signal is no longer assumed.

The external frame clock in JESD204 and JESD204A enabled the TX to reliably detect the position of each edge of the SYNC~ signal that was generated by the RX. Therefore, the duration of an error report was defined as one frame. Activation of the SYNC~ signal for the duration of two frames and more was interpreted as a synchronization request. In JESD204B, the change to internal frame clocks introduced uncertainty in the detection of the edges of the SYNC~ signal. Therefore, the duration of the error report as generated by the JESD204B RX was extended to two frames. The JESD204B TX interprets SYNC~ activation as a synchronization request if the detected duration is at least four frames. To ensure this detected duration, the JESD204B RX must activate the SYNC~ signal for the duration of at least five frames. Nine octets were added to these five frames to guarantee that the TX would transmit enough /K/ characters for reliable link reinitialization over the data interface.

In the JESD204C 8B/10B link layer, the timings of generation and detection of the SYNC~ signal are no longer specified relative to the frame clock. In most implementations, these timings will be relative to the link layer clock. However, JESD204C allows these timings to be relative to another internal or external clock, if this clock meets certain conditions (see 4.3.7). Hence, JESD204C does not preclude designing devices that are backward compatible with a previous version of this standard. However, full backward compatibility with JESD204A and the initial version of JESD204 will require the use of an external frame clock.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex M (informative) Backward compatibility of 8B/10B link layer with previous versions of JESD204 (cont'd)

If error reporting over the SYNC interface is disabled in the RX, the ability of the TX to detect error reports and to distinguish them from synchronization requests is not relevant. Therefore, in those applications where error reporting over the SYNC~ interface is not essential, the 8B/10B link layers of different versions of this standard can be made to interoperate by disabling the error reporting over the SYNC interface in the RX.

Table M.1 outlines the various options for SYNC~ error reporting when interfacing various classes of JESD204 devices.

Table M.1 — SYNC~ conditions for interoperability

TX device	RX device	Conditions for the JESD204 device
JESD204B	JESD204C	For guaranteed detection of error reports and no false detection of synchronization requests, the RX must use error reports of two frame durations (as per JESD204B). The RX may use error reports shorter than two frames if their guaranteed detection in the TX is not essential. If these conditions cannot be met, the RX must be configured not to issue error reports over the SYNC interface.
JESD204 or JESD204A	JESD204C	For guaranteed detection of error reports and no false detection of synchronization requests, an external frame clock is necessary in both devices and the RX must use error reports of one frame duration (as per JESD204A). If no external frame clock can be used, the error report must last shorter than one frame and there is a possibility that error reports are not (or not all) detected in the TX. If these conditions cannot be met, the RX must be configured not to issue error reports over the SYNC interface.
JESD204C	JESD204B	The period of the SYNC~ detection clock must be one frame duration or a unit fraction of the frame duration. In case of long frame lengths, the decision between an error report and a synchronization request in the TX may require longer detection thresholds than specified in 8.8.2. If these conditions cannot be met, the RX must be configured not to issue error reports over the SYNC interface.
JESD204C	JESD204 or JESD204A	The same conditions as for interoperability with a JESD204B RX device above and in addition the following: To ensure detection of error reports, an external frame clock is necessary in both devices.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex N (informative) Device level implementation for 8B/10B link layer

N.1 TX link layer

Figure Q.1 outlines a possible implementation of the TX link layer of this part of the JESD204 standard. The frame data from the transport layer is first converted to an octet stream. Optionally, the data can be scrambled. In Figure N.1, the scrambler operates on octets, but the scrambler could also process larger groups of bits simultaneously. After optional scrambling, alignment characters are substituted for the purpose of monitoring frame and lane alignment in the receiver. The alignment characters replace the original data symbols in such a way that the receiver can still reconstruct the original data symbols. At link initialization, an initial lane alignment sequence is transmitted. All data is encoded as 8B/10B symbols and subsequently passed to the physical layer for serialization.

Each TX link layer monitors synchronization requests and optionally error reports from the RX device on the SYNC interface, which is shared between all link layers of the link. In Figure N.1 the hard-wired variant of the SYNC interface is shown. If the SYNC~ signal is active for at least a programmable duration (see 8.8.2), all transmitters in the link enter synchronization mode until the SYNC~ signal is deactivated by the RX device. On a multipoint link, it is possible, but not required, to share the SYNC interface between all links of the multipoint link.

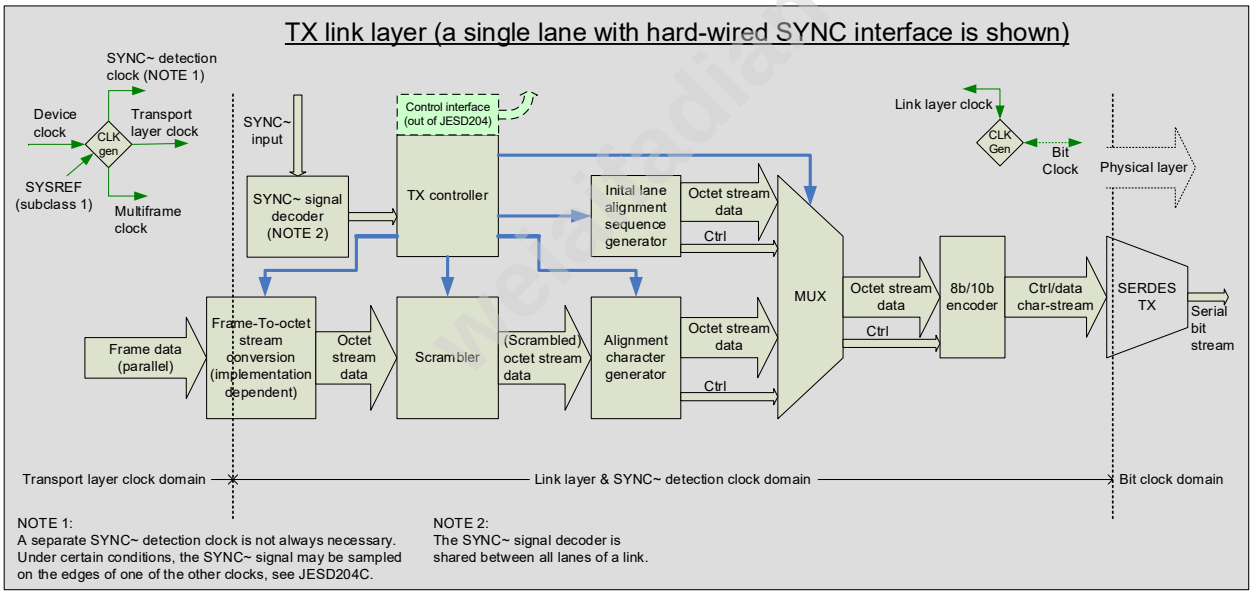


Figure N.1 — TX link layer (not supporting early synchronization option in scrambling)

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

N.2 RX Link Layer

Figure N.2 outlines a possible implementation of the RX link layer of this part of the JESD204 standard. Parallel symbols from the physical layer are decoded in an 8B/10B decoder as described in [2]. If decoding errors are detected, they are reported to the RX controller, which then generates a message over the SYNC interface to flag the transmitter when an error occurs or reinitialization is needed. The SYNC interface is shared between all link layers of the link. On a multipoint link, it is possible, but not required, to share the SYNC interface between all links of the multipoint link. In Figure N.2 the hard-wired variant of the SYNC interface is shown.

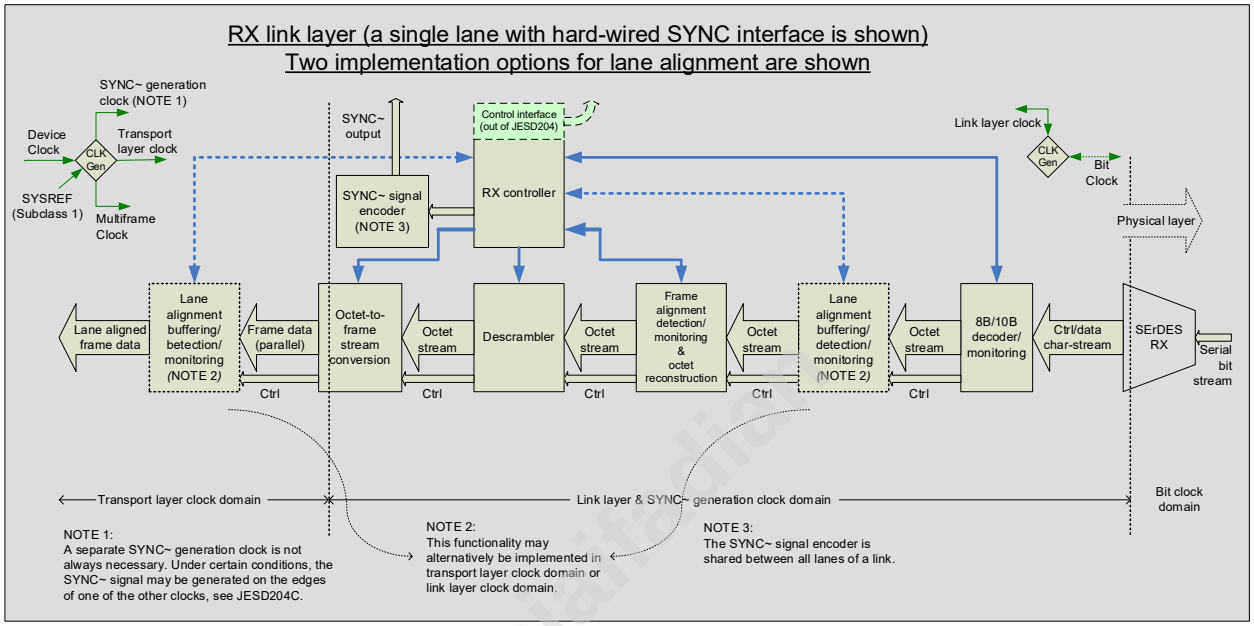


Figure N.2 — RX Link Layer

After 8B/10B decoding, any substitution characters are restored in the octet stream. If the data was scrambled at the transmitter, it is also passed through the receiver's optional descrambler. After de-scrambling, the data is framed into its original format as generated in the TX transport layer. In applications with multiple lanes or multiple devices, a FIFO and appropriate control mechanism may be required in the final assembly stages to align the frames across multiple lanes.

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex O (informative) Parallel scrambler and descrambler implementations

The scrambler and descrambler are defined in their serial implementation form. However, in actual hardware it may be advantageous to use an equivalent parallel implementation. Parallel implementations can be synthesized from the parallel update equations in the scrambler definitions**,** Figure 72 and Figure 73. In the 8-bit implementations, the synthesis makes use of the fact that scrambled bit S_i is followed exactly one parallel clock cycle later by scrambled bit S_{i+8}. In this way, the parallel update equations of Figure 72 are transformed into the 8-bits scrambler of Figure O.115 and the 8-bits descrambler of Figure O.116. Similarly, the parallel update equations of the alternative scrambler of Figure 73 are transformed into the alternative 8-bits scrambler of Figure O.117. In the 16-bit implementations, the synthesis is based on the fact that scrambled bit S_i is followed exactly one parallel clock cycle later by scrambled bit S_{i+16}. This way the 16-bit scrambler of Figure O.118 and the 16-bits descrambler of Figure O.119 are derived. Note that the 16-bit implementations have the disadvantage of double propagation time for the two least significant bits. In all these figures, the logic symbols are according to IEEE Std. 91a-1991 (Ref. 7).

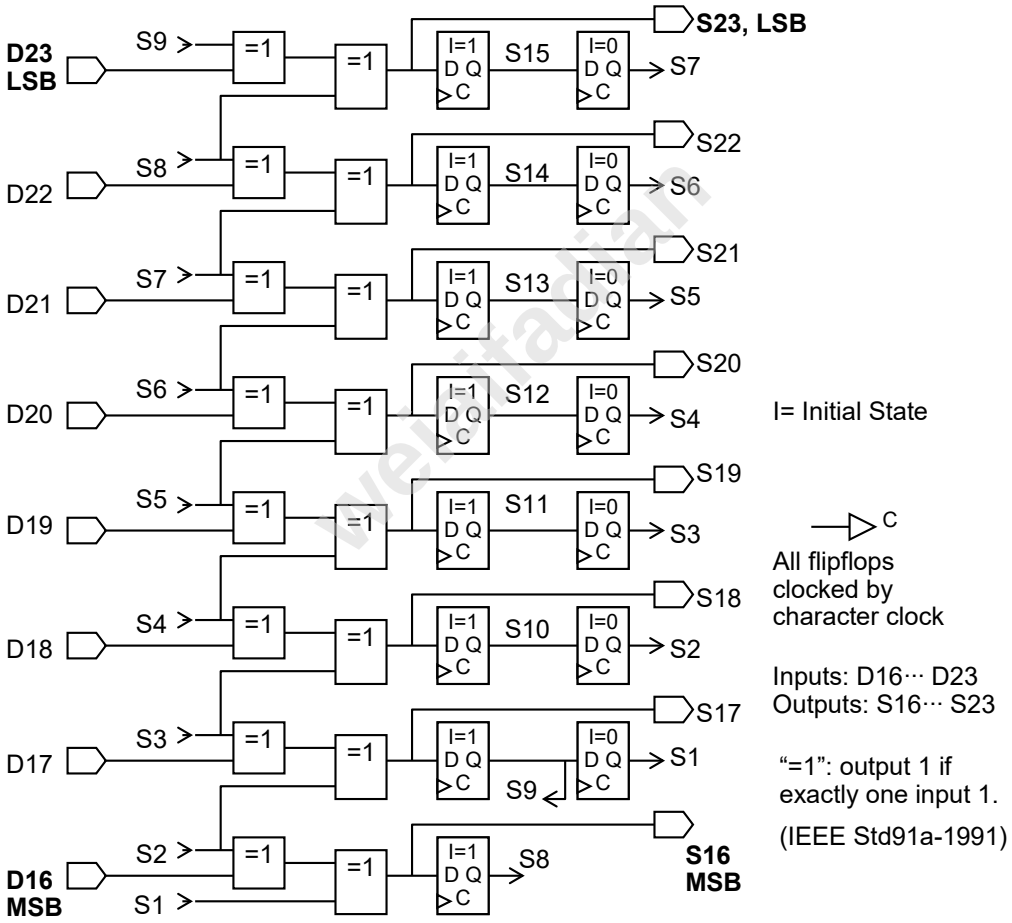


Figure O.115 — 8-bit parallel implementation of self-synchronous scrambler based on $1+x^{14}+x^{15}$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex O (informative) Parallel scrambler and descrambler implementations (cont'd)

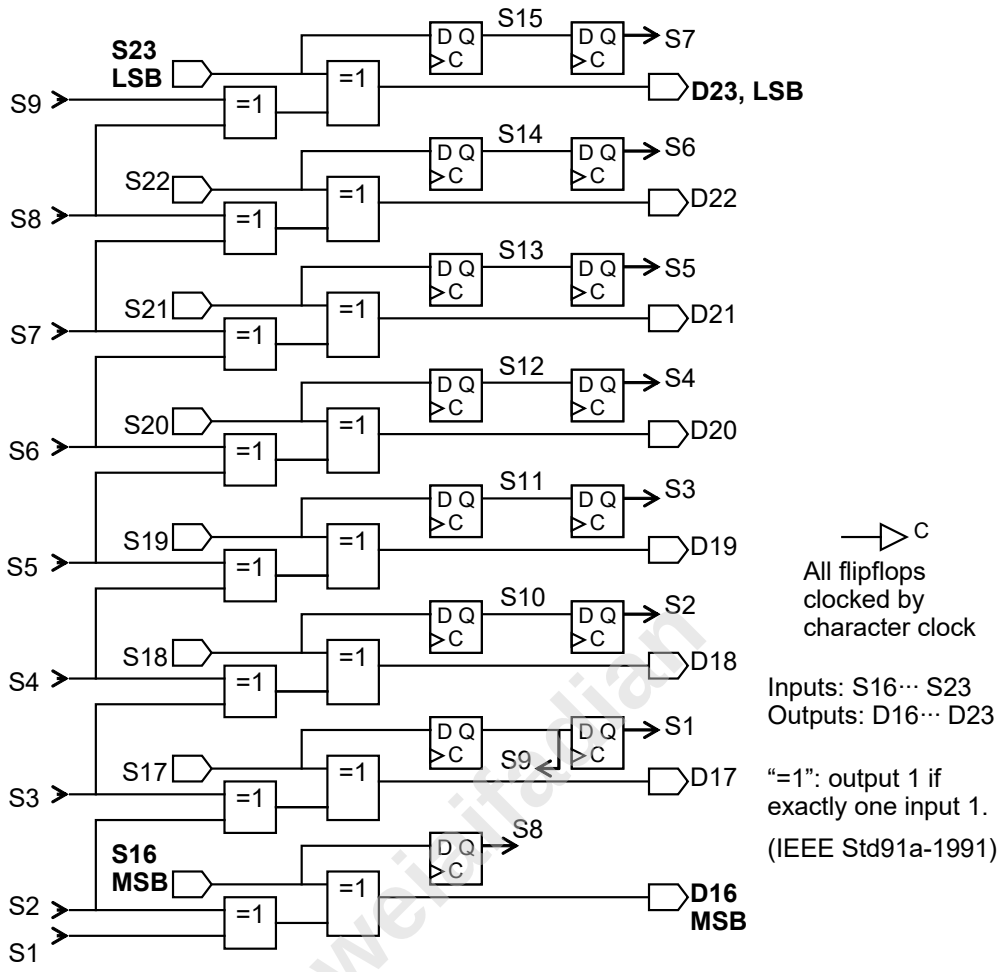


Figure O.116 — 8-bit parallel implementation of self-synchronous descrambler based on $1+x^{14}+x^{15}$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex O (informative) Parallel scrambler and descrambler implementations (cont'd)

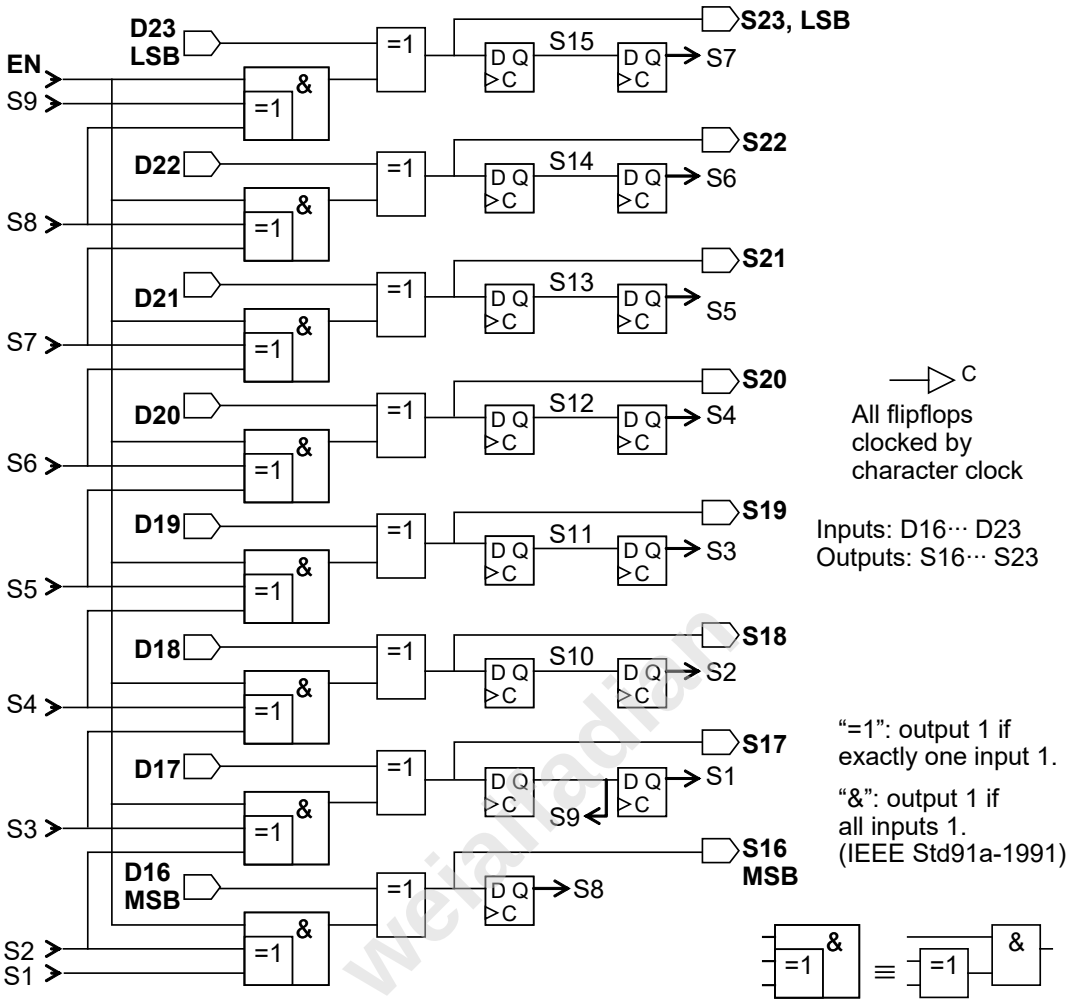


Figure O.117 — 8-bit parallel implementation of alternative early-synchronization scrambler

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex O (informative) Parallel scrambler and descrambler implementations (cont'd)

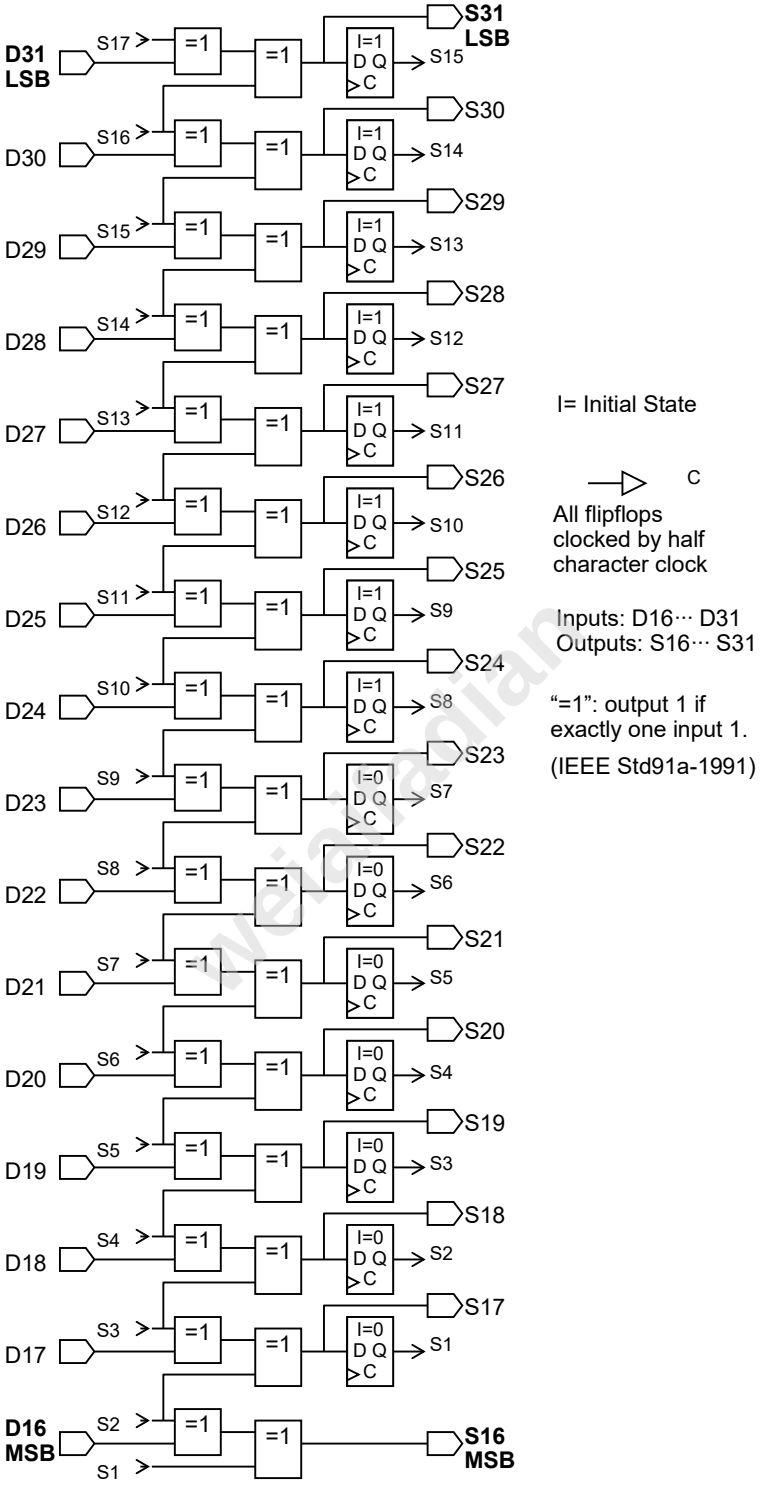


Figure O.118 — 16-bit parallel implementation of self-synchronous scrambler based on $1+x^{14}+x^{15}$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex O (informative) Parallel scrambler and descrambler implementations (cont'd)

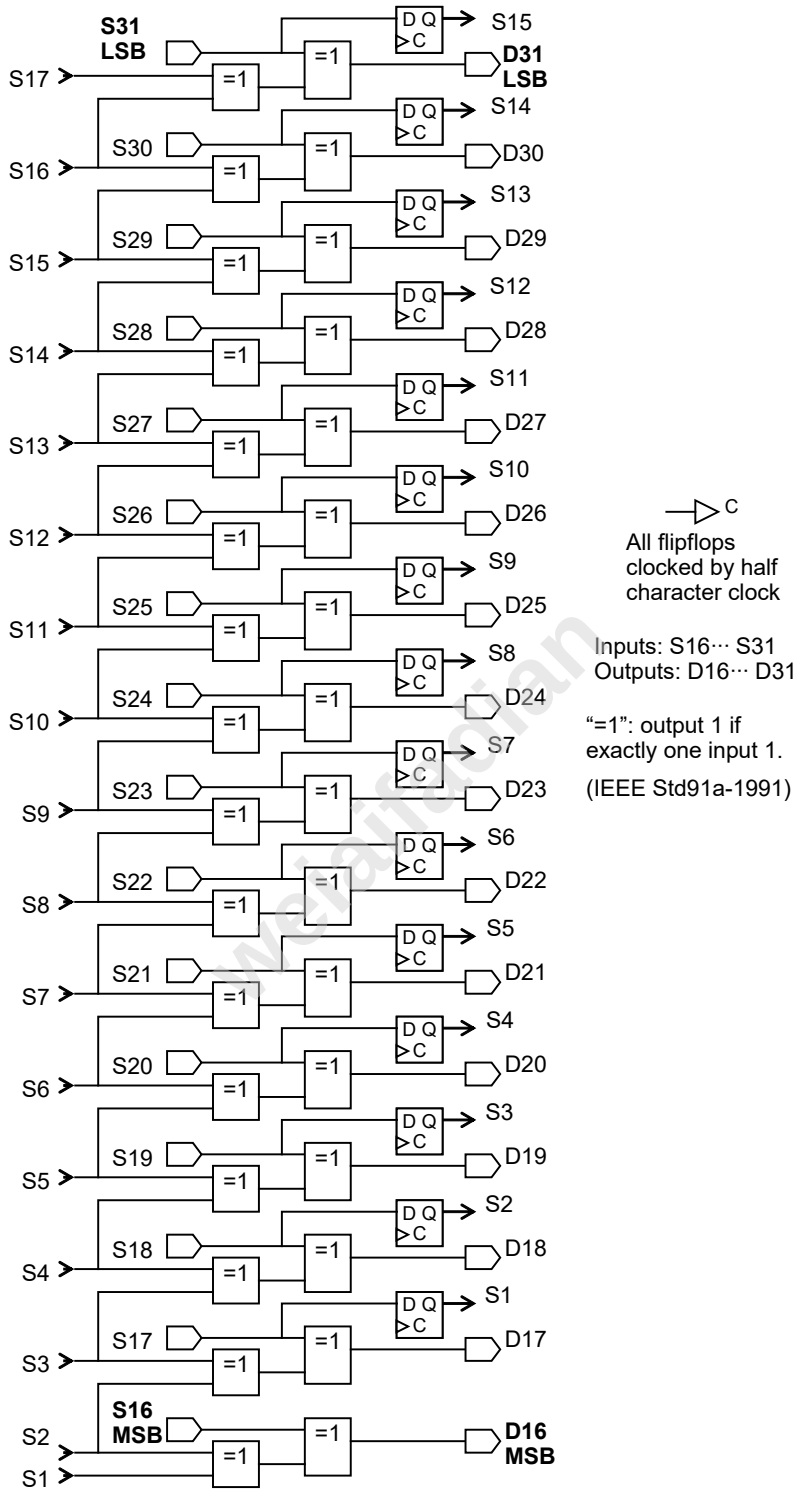


Figure O.119 — 16-bit parallel implementation of self-synchronous descrambler based on $1+x^{14}+x^{15}$

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Annex P **(informative) Bibliography**

The following standards contain provisions that, through references in the text, are informative in this standard:

1. ATIS Telecom Glossary. http://www.atis.org/glossary/

2. ANSI/IEEE Std. 91a-1991, Graphic symbols for logic functions, IEEE 1991, ANSI 1994. (Summary available at e.g., http://en.wikipedia.org/wiki/Logic_gate)

3. IEEE Standard for Ethernet, IEEE Std. 802.3™-2018 section 1, subclause 1.4, Definitions. https://standards.ieee.org/standard/802_3-2018.html.

4. IEEE Std. 802.3™-2018 (Revision of IEEE Std. 802.3™-2015) IEEE Standard for Ethernet, Section 6, June 14, 2018. https://standards.ieee.org/standard/802_3-2018.html.

5. IEEE Standard for Ethernet, IEEE Std. 802.3™-2018 section 3, clause 36, Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer, type 1000BASE-X. https://standards.ieee.org/standard/802_3-2018.html.

6. IEEE Standard for Ethernet, IEEE Std. 802.3™-2018 section 3, annex 36B, 8B/10B transmission code running disparity calculation examples. https://standards.ieee.org/standard/802_3-2018.html.

7. IEEE Standard for Ethernet, IEEE Std. 802.3™-2018 section 4, clause 48, Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer, type 10GBASE-X. https://standards.ieee.org/standard/802_3-2018.html.

8. IEEE Std. 802.3-2018®, Section Four, Clause 49, Physical Coding Sublayer (PCS) for 64B/66B, type 10GBASE-R. https://standards.ieee.org/standard/802_3-2018.html.

9. IEEE Std. 802.3-2018®, Section Six, Clause 80, Introduction to 40 Gb/s and 100 Gb/s networks. https://standards.ieee.org/standard/802_3-2018.html.

10. IBIS (I/O Buffer Information Specification) Version 6.1, September 11, 2015.

11. Lin, Shu, and Daniel J. Costello. *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1983. Print.

12. OIF-CEI-04.0, Common Electrical I/O (CEI) – Electrical and Jitter Interoperability agreements for 6G+ bps, 11G+ bps and 25G+ bps I/O, Optical Internetworking Forum, February 2017. http://www.oiforum.com/public/documents/OIF_CEI_04.0.pdf.

User Manfactian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

Standard Improvement Form JEDEC JESD204C.01

The purpose of this form is to provide the Technical Committees of JEDEC with input from the industry regarding usage of the subject standard. Individuals or companies are invited to submit comments to JEDEC. All comments will be collected and dispersed to the appropriate committee(s).

If you can provide input, please complete this form and return to:

JEDEC
Attn: Publications Department
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

Fax: 703.907.7583

1. I recommend changes to the following:

☐ Requirement, clause number _____

☐ Test method number _____ Clause number _____

The referenced clause number has proven to be:

☐ Unclear ☐ Too Rigid ☐ In Error

☐ Other _____

2. Recommendations for correction:

_____

_____

_____

_____

3. Other suggestions for document improvement:

_____

_____

_____

_____

Submitted by

Name: _____ Phone: _____

Company: _____ E-mail: _____

Address: _____

City/State/Zip: _____ Date: _____

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT

weiaifadian

Downloaded by rucheng zhao (1790402770@qq.com) on Aug 22, 2023, 6:57 pm PDT