# 48. Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer, type 10GBASE-X

## 48.1 Overview

This clause specifies the Physical Coding Sublayer (PCS) and the Physical Medium Attachment (PMA) sublayer that are common to a family of 10 Gb/s Physical Layer implementations, collectively known as 10GBASE-X. The 10GBASE-LX4 PMD described in Clause 53 and 10GBASE-CX4 described in Clause 54 are members of the 10GBASE-X PHY family. The term 10GBASE-X is used when referring to issues common to any of the variants within this family.

The 10GBASE-X PCS and PMA sublayers are also utilized by the XGXS specified in Clause 47.

10GBASE-X PCS and PMA sublayers map the interface characteristics of the PMD sublayer (including MDI) to the services expected by the Reconciliation Sublayer (RS) and the logical and electrical characteristics of the 10 Gigabit Media Independent Interface (XGMII). Although the XGMII is optional, it is used as the basis for the definition of the 10GBASE-X PCS and PMA sublayers.

10GBASE-X assumes the use of the MDIO interface and register set for communication between PHY and Station Management (STA) entities, see Clause 45.

10GBASE-X has the following characteristics:

a) The capability of supporting 10 Gb/s operation at the XGMII and RS
b) Clock references embedded in all data and control code-groups
c) Data paths consisting of independent serial links called lanes
d) Independent four-lane-wide transmit and receive data paths
e) Simple signal mapping to the XGMII and RS
f) Full duplex operation
g) Shared technology with other 10 Gb/s interfaces
h) Shared functionality with other 10 Gb/s Ethernet blocks

### 48.1.1 Objectives

The following are the objectives of 10GBASE-X:

a) Support the IEEE 802.3 MAC
b) Provide a data rate of 10 Gb/s at the XGMII
c) Support cable plants using optical fiber compliant with second edition of ISO/IEC 11801: 1995
d) Support a BER objective of $10^{-12}$
e) Support the optional XAUI
f) Support link fault and error indications

### 48.1.2 Relationship of 10GBASE-X to other standards

Figure 48–1 depicts the relationships among the 10GBASE-X sublayers (shown shaded), the IEEE 802.3 MAC and RS, and the IEEE 802.2 LLC.

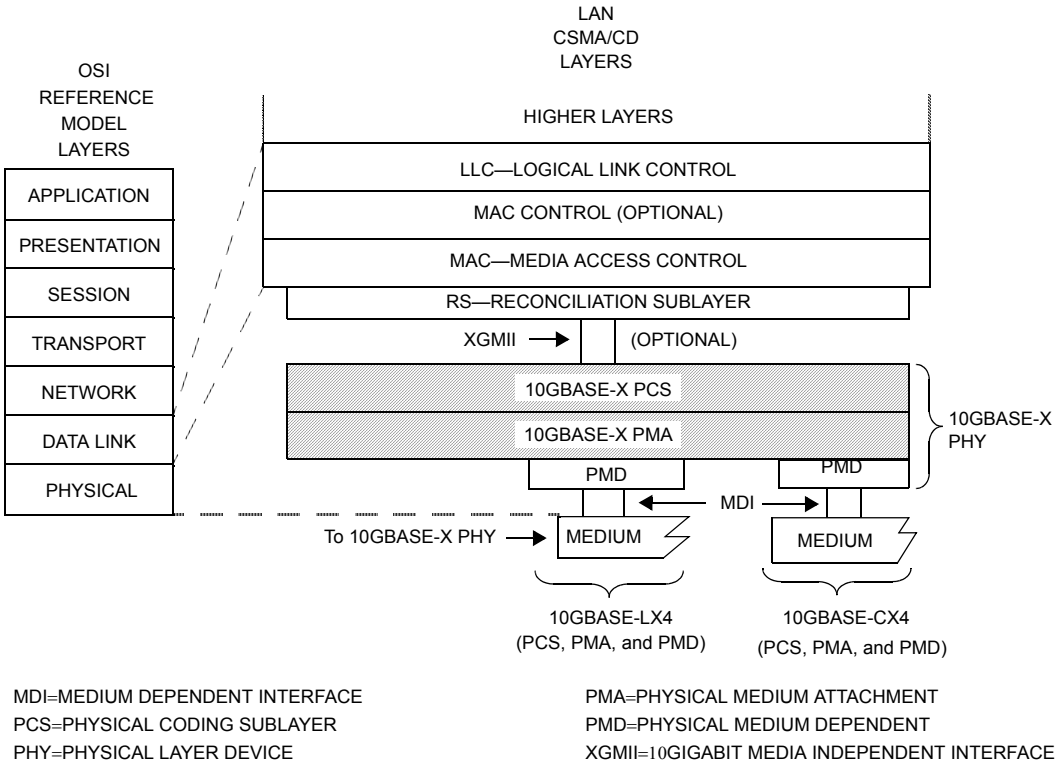



**Figure 48–1—10GBASE-X PCS and PMA relationship to the ISO/IEC Open Systems Interconnection (OSI) reference model and the IEEE CSMA/CD LAN Model**

### 48.1.3 Summary of 10GBASE-X sublayers

The following provides an overview of the 10GBASE-X sublayers.[7]

#### 48.1.3.1 Physical Coding Sublayer (PCS)

The interface between the PCS and the RS is the XGMII as specified in Clause 46. The 10GBASE-X PCS provides services to the XGMII in a manner analogous to how the 1000BASE-X PCS provides services to the 1000 Mb/s GMII.

The 10GBASE-X PCS provides all services required by the XGMII and in support of the 10GBASE-X PMA, including:

a) Encoding of 32 XGMII data bits and 4 XGMII control bits to four parallel lanes conveying 10-bit code-groups each, for communication with the underlying PMA.
b) Decoding of four PMA parallel lanes, conveying 10-bit code-groups each, to 32 XGMII data bits and 4 XGMII control bits.
c) Synchronization of code-groups on each lane to determine code-group boundaries.

[7] The 10GBASE-X PHY consists of that portion of the Physical Layer between the MDI and XGMII consisting of the PCS, PMA, and PMD sublayers. The 10GBASE-X PHY is roughly analogous to the 1000BASE-X PHY.

d) Deskew of received code-groups from all lanes to an alignment pattern.
e) Support of the MDIO interface and register set as specified in Clause 45 to report status and enable control of the PCS.
f) Conversion of XGMII Idle control characters to (from) a randomized sequence of code-groups to enable serial lane synchronization, clock rate compensation and lane-to-lane alignment.
g) Clock rate compensation protocol.
h) Link Initialization based on the transmission and reception of the Idle sequence.
i) Link status reporting for fault conditions.

#### 48.1.3.2 Physical Medium Attachment (PMA) sublayer

The PMA provides a medium-independent means for the PCS to support the use of a range of serial-bit-oriented physical media. The 10GBASE-X PMA performs the following functions:

a) Mapping of transmit and receive code-groups between the PCS and PMA via the PMA service interface.
b) Serialization (deserialization) of code-groups for transmission (reception) on the underlying serial PMD.
c) Clock recovery from the code-groups supplied by the PMD.
d) Mapping of transmit and receive bits between the PMA and PMD via the PMD service interface.
e) Direct passing of signal_detect from the PMD to the PCS through the PMA via the PMD and PMA service interfaces.

#### 48.1.3.3 Physical Medium Dependent (PMD) sublayer

10GBASE-X supports the PMD sublayer and MDI specified in Clause 53. The 10GBASE-LX4 PMD and 10GBASE-CX4 performs the following functions:

a) Transmission of quad serial bit streams on the underlying medium.
b) Reception of quad serial bit streams on the underlying medium.

### 48.1.4 Rate of operation

The 10GBASE-X PCS and PMA support the 10 Gb/s MAC data rate. The line rate of each of four PMA lanes is 3.125 GBaud ± 100 ppm.

### 48.1.5 Allocation of functions

PCS and PMA functions directly map onto the 10GBASE-X PMD, MDI and medium which attach, in turn, to another 10GBASE-X PHY. In addition, 10GBASE-X PCS and PMA functions embodied in the XGXS described in Clause 47 may be used to attach to alternate 10 Gb/s PHYs such as 10GBASE-R or 10GBASE-W.

The longer interconnect distances afforded through the specification of a self-clocked serial architecture enable significant implementation flexibility while imposing a requirement on those implementations to ensure sufficient signal fidelity over the link. The implementer of this standard is expected to meet the required specifications in this and related clauses through implementation methods not specified by this standard.

### 48.1.6 Inter-sublayer interfaces

There are a number of interfaces employed by 10GBASE-X. Some (such as the PMA service interface) use an abstract service model to define the operation of the interface. Figure 48–2 depicts the relationship and mapping of the services provided by all of the interfaces relevant to 10GBASE-X.

Multiple optional physical instantiations of the PCS service interface have been defined. One is the XGMII described in Clause 46. The other is the interface to the XGXS described in Clause 47.

Physical instantiations of the 10GBASE-X PMA and PMD service interfaces are not defined in this standard.

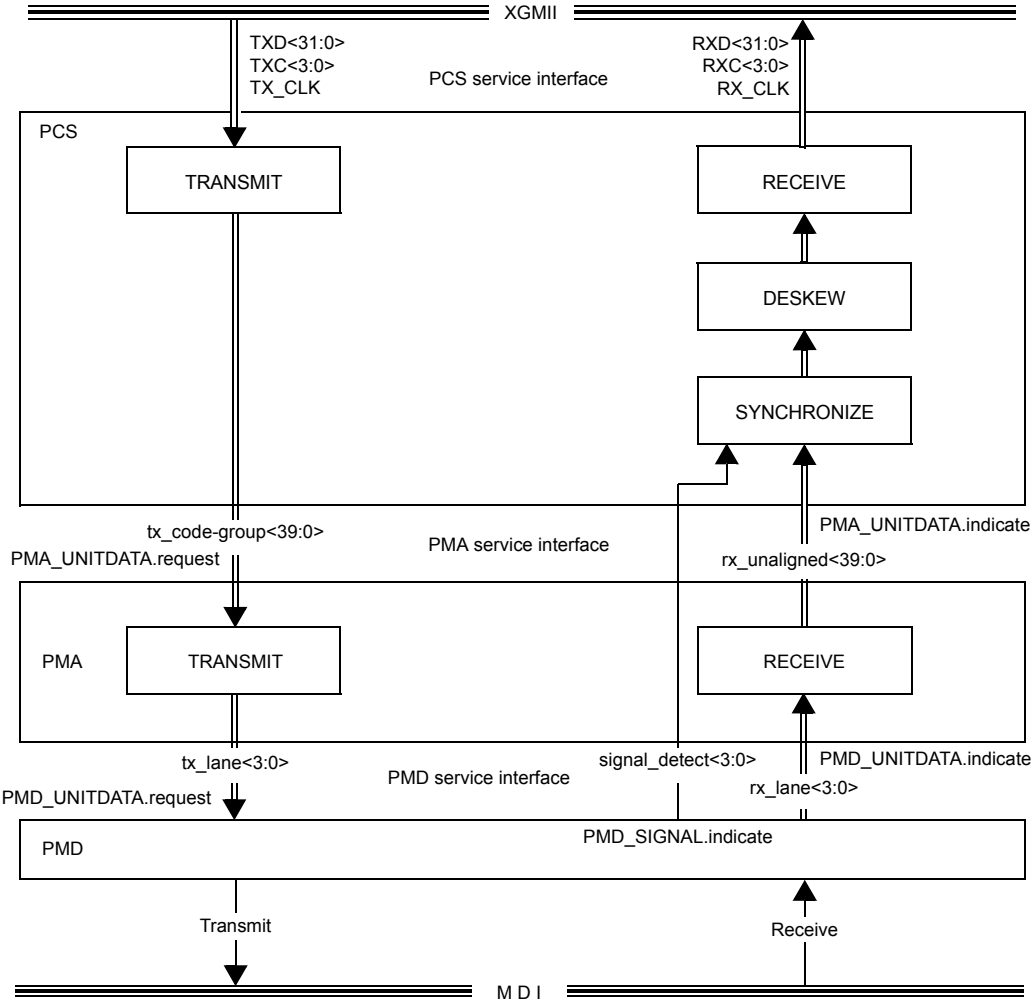



**Figure 48–2—Functional block diagram**

### 48.1.7 Functional block diagram

Figure 48–2 provides a functional block diagram of the 10GBASE-X PHY.

### 48.1.8 Special symbols

/x/ The code-group x is represented by preceding and following slash characters.

||y|| Four code-groups, one each in lanes 0 through 3 inclusive, synchronous to each other and arranged in a column identified by the value y, is represented by preceding and following double bar characters.

## 48.2 Physical Coding Sublayer (PCS)

### 48.2.1 PCS service interface (XGMII)

The PCS service interface allows the 10GBASE-X PCS to transfer information to and from the PCS client. The PCS client is the RS defined in Clause 46, or the XGXS defined in Clause 47. An instantiation of the PCS service interface is the XGMII defined in Clause 46.

In the transmit direction, the 10GBASE-X PCS accepts packets from the PCS client on the XGMII. Due to the continuously signaled nature of the underlying PMA, and the encoding performed by the PCS, the 10GBASE-X PCS maps XGMII data and control characters into a code-group stream. In the receive direction, the PCS decodes the code-group stream received from the PMA, maps the code-groups to XGMII data and control characters and forwards the character stream to the XGMII to the PCS client for further processing.

### 48.2.2 Functions within the PCS

The PCS includes the Transmit, Receive, Synchronization, and Deskew processes for 10GBASE-X. The PCS shields the RS (and MAC) from the specific nature of the underlying channel.

When communicating with the XGMII, the PCS uses, in each direction, 32 data signals (TXD <31:0> and RXD <31:0>), four control signals (TXC <3:0> and RXC <3:0>), and a clock (TX_CLK and RX_CLK).

When communicating with the PMA, the PCS uses the data signals tx_code-group <39:0> in the transmit direction and rx_unaligned <39:0> in the receive direction. Each set of data signals conveys four lanes of 10-bit code-groups. At the PMA service interface, code-group alignment, lane-to-lane deskew, and provision for PHY clock rate compensation are made possible by embedding special non-data code-groups in the idle stream. The PCS provides the functions necessary to map packets between the XGMII format and the PMA service interface format.

The tx_code-group and rx_unaligned signals are organized into four lanes in a manner similar to that of the XGMII. On transmit, the first PCS code-group is aligned to lane 0, the second to lane 1, the third to lane 2, the fourth to lane 3, then repeating with the fifth to lane 0, etc. This lane-oriented organization extends through the PMA to the PMD service interface. (See Table 48–1.)

**Table 48–1—Transmit and receive lane associations**

| Lane | XGMII TXD RXD | XGMII TXC RXC | PMA tx_code-group rx_unaligned | PMD tx_lane rx_lane |
|---|---|---|---|---|
| 0 | <7:0> | <0> | <9:0> | <0> |
| 1 | <15:8> | <1> | <19:10> | <1> |
| 2 | <23:16> | <2> | <29:20> | <2> |
| 3 | <31:24> | <3> | <39:30> | <3> |

The PCS Transmit process continuously generates code-groups based upon the TXD <31:0> and TXC <3:0> signals on the XGMII, sending them to the PMA service interface via the PMA_UNITDATA.request primitive.

The PCS Synchronization process continuously accepts unaligned and unsynchronized code-groups via the PMA_UNITDATA.indicate primitive, obtains 10-bit code-group synchronization, and conveys synchronized 10-bit code-groups to the PCS Deskew process via the SYNC_UNITDATA.indicate message. The PCS Synchronization process sets the lane_sync_status <3:0> flags to indicate whether the PMA is functioning dependably (as well as can be determined without exhaustive error-rate analysis).

The PCS Deskew process continuously accepts synchronized code-groups via the SYNC_UNITDATA.indicate message, aligns the code-groups to remove skew between the lanes that has been introduced by the link, and conveys aligned and synchronized code-groups to the PCS Receive process via the ALIGN_UNITDATA.indicate message. The PCS Deskew process asserts the align_status flag to indicate that the PCS has successfully deskewed and aligned code-groups on all PCS lanes. The PCS Deskew process attempts deskew and alignment whenever the align_status flag is de-asserted. The PCS Deskew process is otherwise idle.

The PCS Receive process continuously accepts code-groups from the PMA service interface via the ALIGN_UNITDATA.indicate message. The PCS Receive process monitors these code-groups and generates RXD and RXC on the XGMII. All code-groups received that represent idle are replaced with Idle characters prior to forwarding to the XGMII.

The PCS Transmit and Receive processes provide support for Link status reporting, which supports the transmit fault and receive fault conditions.

All PCS processes are described in detail in the state diagrams in 48.2.6.2.

### 48.2.3 Use of code-groups

The transmission code used by the PCS, referred to as 8B/10B, is identical to that specified in Clause 36. The PCS maps XGMII characters into 10-bit code-groups, and vice versa, using the 8B/10B block coding scheme. Implicit in the definition of a code-group is an establishment of code-group boundaries by a PCS Synchronization process. The 8B/10B transmission code as well as the rules by which the PCS ENCODE and DECODE functions generate, manipulate, and interpret code-groups are specified in 36.2.4. A 10GBASE-X PCS shall meet the requirements specified in 36.2.4.1 through 36.2.4.6, 36.2.4.8, and 36.2.4.9. PCS lanes are independent of one another. All code-group rules specified in 36.2.4 are applicable to each lane. The mapping of XGMII characters to PCS code-groups is specified in Table 48–2. The mapping of PCS code-groups to XGMII characters is specified in Table 48–3.

Figure 48–3 illustrates the mapping of an example XGMII character stream into a PCS code-group stream.

The relationship of code-group bit positions to XGMII, PCS and PMA constructs and PMD bit transmission order, exemplified for lane 0, is illustrated in Figure 48–4.

XGMII

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T/RXD<7:0> | I | I | S | Dp | D | D | D | --- | D | D | D | D | I | I | I | I | I | I |
| T/RXD<15:8> | I | I | Dp | Dp | D | D | D | --- | D | D | D | T | I | I | I | I | I | I |
| T/RXD<23:16> | I | I | Dp | Dp | D | D | D | --- | D | D | D | I | I | I | I | I | I | I |
| T/RXD<31:24> | I | I | Dp | Ds | D | D | D | --- | D | D | D | I | I | I | I | I | I | I |

PCS

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LANE 0 | K | R | S | Dp | D | D | D | --- | D | D | D | D | A | R | R | K | K | R |
| LANE 1 | K | R | Dp | Dp | D | D | D | --- | D | D | D | T | A | R | R | K | K | R |
| LANE 2 | K | R | Dp | Dp | D | D | D | --- | D | D | D | K | A | R | R | K | K | R |
| LANE 3 | K | R | Dp | Ds | D | D | D | --- | D | D | D | K | A | R | R | K | K | R |

Legend:
Dp represents a data character containing the preamble pattern
Ds represents a data character containing the SFD pattern

**Figure 48–3—XGMII character stream to PCS code-group stream mapping example**

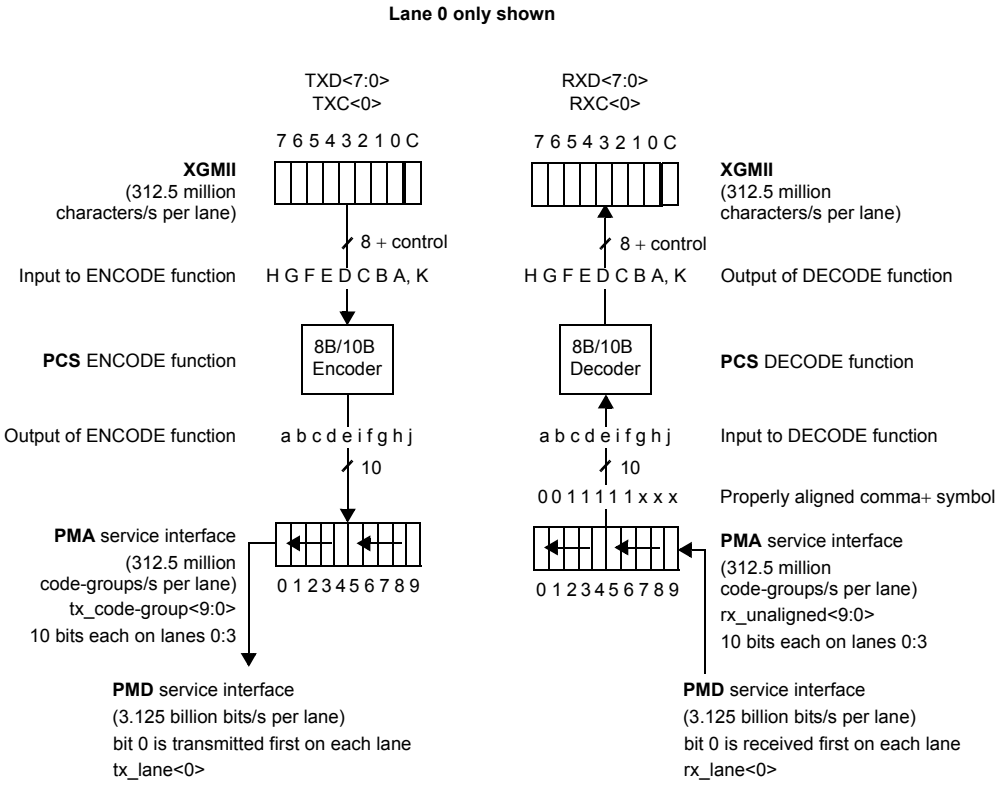



**Figure 48–4—PCS reference diagram**

### 48.2.4 Ordered_sets and special code-groups

10GBASE-X PCS ordered_sets consist of combinations of special and data code-groups (defined as a column of code-groups). All ordered_sets are four code-groups in length and begin in lane 0. In addition to ordered_sets, the PCS defines several special code-groups for control purposes. Ordered_sets and special code-groups provide the following capabilities:

a) PCS Synchronization process ability to obtain bit and code-group synchronization
b) Packet delineation
c) Synchronization between the transmitter and receiver circuits at opposite ends of a link
d) Deskew of received code-groups from all serial lanes to an alignment pattern
e) Clock rate compensation protocol
f) Link status reporting protocol
g) Column fill
h) Error indication

Table 48–4 lists the defined ordered_sets and special code-groups.

**Table 48–2—XGMII character to PCS code-group mapping**

| XGMII TXC | XGMII TXD | PCS code-group | Description |
|---|---|---|---|
| 0 | 00 through FF | Dxx.y | Normal data transmission |
| 1 | 07 | K28.0 or K28.3 or K28.5 | Idle in ||I|| |
| 1 | 07 | K28.5 | Idle in ||T|| |
| 1 | 9C | K28.4 | Sequence |
| 1 | FB | K27.7 | Start |
| 1 | FD | K29.7 | Terminate |
| 1 | FE | K30.7 | Error |
| 1 | Other value in Table 36-2 | See Table 36-2 | Reserved XGMII character |
| 1 | Any other value | K30.7 | Invalid XGMII character |
| NOTE—Values in TXD column are in hexadecimal. | | | |

#### 48.2.4.1 Data (/D/)

A data code-group, when not used to distinguish or convey information for a defined ordered_set, conveys one octet of data between the XGMII and the PCS. Any data code-group can be followed by any other data code-group. Data code-groups are encoded and decoded but not interpreted by the PCS.

#### 48.2.4.2 Idle (||I||)

Idle ordered_sets (||I||) are transmitted in full columns continuously and repetitively whenever the XGMII is idle (TXD <31:0>=0x07070707 and TXC <3:0>=0xF). ||I|| provides a continuous fill pattern to establish and maintain lane synchronization, perform lane-to-lane deskew and perform PHY clock rate compensation. ||I|| is emitted from, and interpreted by, the PCS.

**Table 48–3—PCS code-group to XGMII character mapping**

| XGMII RXC | XGMII RXD | PCS code-group | Description |
|---|---|---|---|
| 0 | 00 through FF | Dxx.y | Normal data reception |
| 1 | 07 | K28.0 or K28.3 or K28.5 | Idle in ||I|| |
| 1 | 07 | K28.5 | Idle in ||T|| |
| 1 | 9C | K28.4 | Sequence |
| 1 | FB | K27.7 | Start |
| 1 | FD | K29.7 | Terminate |
| 1 | FE | K30.7 | Error |
| 1 | FE | Invalid code-group | Received code-group |
| 1 | See Table 36-2 | Other valid code-group | Received reserved code-group |
| NOTE—Values in RXD column are in hexadecimal. | | | |

A sequence of ||I|| ordered_sets consists of one or more consecutively transmitted ||K||, ||R|| or ||A|| ordered_sets, as defined in Table 48–4. Rules for ||I|| ordered_set sequencing shall be as follows:

a) ||I|| sequencing starts with the first column following a ||T||.
b) The first ||I|| following ||T|| alternates between ||A|| or ||K|| except if an ||A|| is to be sent and less than r [see item d)] columns have been sent since the last ||A||, a ||K|| is sent instead.
c) ||R|| is chosen as the second ||I|| following ||T||.
d) Each ||A|| is sent after r non-||A|| columns where r is a uniform randomly distributed number between 16 and 31, inclusive. The corresponding minimum spacing of 16 non-||A|| columns between two ||A|| columns provides a theoretical 85-bit deskew capability.
e) When not sending an ||A||, either ||K|| or ||R|| is sent with a random uniform distribution between the two.
f) Whenever sync_status=OK, all ||I|| received during idle are translated to XGMII Idle control characters for transmission over the XGMII. All other !||I|| received during idle are mapped directly to XGMII data or control characters on a lane by lane basis.

The purpose of randomizing the ||I|| sequence is to reduce 10GBASE-X electromagnetic interference (EMI) during idle. The randomized ||I|| sequence produces no discrete spectrum. Both ||A|| spacing as well as ||K||, ||R||, or ||A|| selection shall be based on the generation of a random integer r generated by a PRBS based on one of the 7th order polynomials listed in Figure 48–5. ||A|| spacing is set to the next generated value of r. The rate of generation of r is once per column, 312.5 MHz ± 100 ppm. Once the ||A|| spacing count goes to zero (A_CNT=0), ||A|| is selected for transmission at the next opportunity during the Idle sequence. ||K|| and ||R|| selection follows the value of code_sel, which is continuously set according to the even or odd value of r. The method of generating the random integer r is left to the implementer. PCS Idle randomizer logic is illustrated in Figure 48–5

**Table 48–4—Defined ordered_sets and special code-groups**

| Code | Ordered_Set | Number of code-groups | Encoding |
|---|---|---|---|
| **\|\|I\|\|** | **Idle** | | **Substitute for XGMII Idle** |
| \|\|K\|\| | Sync column | 4 | /K28.5/K28.5/K28.5/K28.5/ |
| \|\|R\|\| | Skip column | 4 | /K28.0/K28.0/K28.0/K28.0/ |
| \|\|A\|\| | Align column | 4 | /K28.3/K28.3/K28.3/K28.3/ |
| | **Encapsulation** | | |
| \|\|S\|\| | Start column | 4 | /K27.7/Dx.y/Dx.y/Dx.y/[a] |
| \|\|T\|\| | Terminate column | 4 | Terminate code-group in any lane |
| $\|\|T_0\|\|$ | Terminate in Lane 0 | 4 | /K29.7/K28.5/K28.5/K28.5/ |
| $\|\|T_1\|\|$ | Terminate in Lane 1 | 4 | /Dx.y/K29.7/K28.5/K28.5/[a] |
| $\|\|T_2\|\|$ | Terminate in Lane 2 | 4 | /Dx.y/Dx.y/K29.7/K28.5/[a] |
| $\|\|T_3\|\|$ | Terminate in Lane 3 | 4 | /Dx.y/Dx.y/Dx.y/K29.7/[a] |
| | **Control** | | |
| /E/ | Error code-group | 1 | /K30.7/ |
| | **Link Status** | | |
| \|\|Q\|\| | Sequence ordered_set | 4 | /K28.4/Dx.y/Dx.y/Dx.y/[a] |
| \|\|LF\|\| | Local Fault signal | 4 | /K28.4/D0.0/D0.0/D1.0/ |
| \|\|RF\|\| | Remote Fault signal | 4 | /K28.4/D0.0/D0.0/D2.0/ |
| \|\|Qrsvd\|\| | Reserved | 4 | !\|\|LF\|\| and !\|\|RF\|\| |
| | **Reserved** | | |
| \|\|Fsig\|\| | Signal ordered_set | 4 | /K28.2/Dx.y/Dx.y/Dx.y/[a,b] |

[a]/Dx.y/ indicates any data code-group.
[b]Reserved for INCITS T11.

.

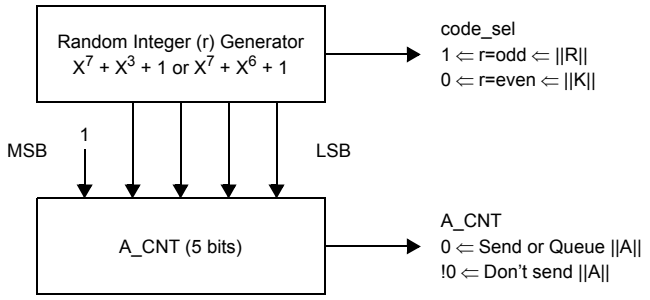



**Figure 48–5—PCS idle randomizer**

#### 48.2.4.2.1 Sync ||K||

Code-group synchronization is the process by which the receiver detects code-group boundaries in the incoming bit stream of each lane. The detection of the comma pattern in the incoming bit stream identifies a code-group boundary. The proper alignment of a comma used for code-group synchronization is depicted in Figure 48–4. The Sync or ||K|| ordered_set included in the PCS Idle sequence guarantees a sufficient frequency of commas in each lane. The comma pattern is defined in 36.2.4.9. The Sync ordered_set is defined in Table 48–4. /K/ code-groups are interpreted per lane by the PCS Synchronization process which is described in 48.2.6.2.2. Detection of both comma+ and comma- variants of /K/ shall be required.

#### 48.2.4.2.2 Align ||A||

Skew is introduced between lanes by both active and passive elements of a 10GBASE-X link. The PCS deskew function compensates for all lane-to-lane skew observed at the receiver. The Align or ||A|| ordered_set consists of a unique special code-group, also known as Align or /A/ in each lane. /A/ is not used in any other ordered_set. The definition of a 10GBASE-X ordered_set guarantees that /A/ code-groups are simultaneously initiated on all lanes at the transmitter, resulting in minimal lane-to-lane skew at the transmitter. The Align ordered_set is defined in Table 48–4. Allowable skew for all link elements shall be as specified in Table 48–5.

**Table 48–5—Skew budget**

| Skew Source | # | Skew | Total Skew |
|---|---|---|---|
| PMA Tx | 1 | 1 UI[a] | 1 UI |
| PCB | 2 | 1 UI | 2 UI |
| Medium | 1 | <18 UI | <18 UI |
| PMA Rx[b] | 1 | 20 UI | 20 UI |
| Total | | | <41 UI |

[a]UI represents unit interval. For 10GBASE-X, 1 UI = 320 ps.
[b]Includes deserialization function, physical deserializer skew and clock boundary transition.

#### 48.2.4.2.3 Skip ||R||

The 10GBASE-X PHY allows for multiple clock domains along a single link. The Skip or ||R|| ordered_set is included in the PCS Idle sequence to allow for clock rate compensation for the case of multiple clock domains. Clock rate compensation may be performed via insertion or removal of either Idle characters in the unencoded data stream or ||R|| in the encoded Idle stream. Any ||R|| may be removed. ||R|| may be inserted anywhere in the Idle stream with the exception of the first column following ||T||.

When clock compensation is done in the unencoded data stream, rules for idle insertion and deletion shall be as follows:

a) Idle insertion or deletion occurs in groups of four Idle characters.
b) Idle characters are added following idle or ordered_sets.
c) Idle characters are not added while data is being received.
d) When deleting idles, the minimum IPG of five characters is maintained.

e) Sequence ordered_sets are deleted to adapt between clock rates.
f) Sequence ordered_set deletion occurs only when two consecutive sequence ordered_sets have been received and deletes only one of the two.
g) Only idles are inserted for clock compensation.

The disparity of the /R/ code-group is neutral, allowing its removal or insertion without affecting the current running disparity of the serial stream. The correct current running disparity version of /R/ must be inserted in each lane during Skip insertion (see Table 36-2). The Skip ordered_set is defined in Table 48–4.

#### 48.2.4.3 Encapsulation

The Start and Terminate ordered_sets correspond to columns containing the XGMII Start and Terminate control characters, respectively.

##### 48.2.4.3.1 Start ||S||

The Start or ||S|| ordered_set directly maps to the XGMII Start control character in lane 0 followed by any three data characters in XGMII lanes 1 through 3. Normally, the three data characters will be the preamble pattern, but the PCS neither checks nor alters their contents. ||S|| indicates to the PCS that a packet has been initiated. The Start ordered_set is defined in Table 48–4.

##### 48.2.4.3.2 Terminate ||T||

The Terminate or ||T|| ordered_set directly maps to the XGMII Terminate control character located in any lane, preceded by data characters if Terminate is not in lane 0, and followed by Idle characters if Terminate is not in lane 3. ||T|| indicates to the PCS that a packet has been terminated. All XGMII control characters following ||T|| are translated by the PCS until the recognition of the next XGMII Start control character. The PCS considers the MAC interpacket gap (IPG) to have begun with the reception of ||T||. The Terminate ordered_set is defined in Table 48–4.

Unrecognized running disparity errors which propagate to any Idle code-groups in ||T|| or to the column following ||T|| are indicated as /E/ in the preceding column in the same lane in which the errors were recognized. All ||I|| ordered_sets are selected to ensure that propagated code violations are recognized and not propagated further.

The cvtx_terminate function is used to convert all XGMII Idle control characters in the same column as the XGMII Terminate control character to /K/ code-groups. The cvrx_terminate function is used to convert all /K/ code-groups in ||T|| to XGMII Idle control characters.

#### 48.2.4.4 Error /E/

The Error code-group is directly mapped to the XGMII Error control character. /E/ may also be generated by the PCS client to indicate a transmission error to its peer entity or deliberately corrupt the contents of the frame in such a manner that a receiver will detect the corruption with the highest degree of probability. Error is signaled per lane since code-violations are detected on a per lane basis. The Error code-group is defined in Table 48–4.

The presence of /E/ or any invalid code-group on the medium denotes an error condition. 10GBASE-X elements that detect code-group violations shall replace the invalid code-group with /E/ prior to retransmission.

#### 48.2.4.5 Link status

Link status reporting uses the Sequence ordered-set to transport the transmit fault and receive fault link status conditions.

##### 48.2.4.5.1 Sequence ||Q||

The Sequence or ||Q|| ordered_set directly maps to the XGMII Sequence control character on lane 0 followed by three data characters in XGMII lanes 1 through 3. ||Q|| indicates to the PCS that a link status message has been initiated. The PCS Receive process may also initiate Sequence ordered-sets upon detection of a link status condition. Sequence ordered-sets are always sent over the PMA service interface in the column that follows an ||A|| ordered-set. The Sequence ordered-sets do not otherwise interfere with the randomized ||I|| sequence. Sequence ordered-sets corresponding to Local Fault signal and Remote Fault signal are specified in Table 48–4.

### 48.2.5 Management function requirements

The 10GBASE-X PCS supports a set of required and optional management objects to permit it to be controlled by the Station Management entity (STA). Access to management objects within the 10GBASE-X sublayer is accomplished by means of a set of registers within the MDIO register space as defined in 45.2.4 and 45.2.5. The details of the register bit allocations and general usage are given in Clause 45. Table 48–6, Table 48–8, and Table 48–8 describe how the PCS state diagram variables map to management register bits. If an MDIO interface is provided (see Clause 45), they are accessed via that interface. If not, it is recommended that an equivalent access be provided.

**Table 48–6—State diagram variable to management register mapping for PCS**

| State diagram variable | Management Register Bit |
|---|---|
| reset | 3.0.15 Reset |
| sync_status | 3.24.12 10GBASE-X lane alignment status |
| lane_sync_status<3> | 3.24.3 Lane 3 sync |
| lane_sync_status<2> | 3.24.2 Lane 2 sync |
| lane_sync_status<1> | 3.24.1 Lane 1 sync |
| lane_sync_status<0> | 3.24.0 Lane 0 sync |

**Table 48–7—State diagram variable to management register mapping for PHY XS**

| State diagram variable | Management Register Bit |
|---|---|
| reset | 4.0.15 Reset |
| align_status | 4.8.10 Receive local fault |
| sync_status | 4.24.12 PHY XGXS lane alignment status |
| lane_sync_status<3> | 4.24.3 Lane 3 sync |
| lane_sync_status<2> | 4.24.2 Lane 2 sync |
| lane_sync_status<1> | 4.24.1 Lane 1 sync |
| lane_sync_status<0> | 4.24.0 Lane 0 sync |

**Table 48–8—State diagram variable to management register mapping for DTE XS**

| State diagram variable | Management Register Bit |
|---|---|
| reset | 5.0.15 Reset |
| align_status | 5.8.10 Receive local fault |
| sync_status | 5.24.12 PHY XGXS lane alignment status |
| lane_sync_status<3> | 5.24.3 Lane 3 sync |
| lane_sync_status<2> | 5.24.2 Lane 2 sync |
| lane_sync_status<1> | 5.24.1 Lane 1 sync |
| lane_sync_status<0> | 5.24.0 Lane 0 sync |

### 48.2.6 Detailed functions and state diagrams

The body of this clause is comprised of state diagrams, including the associated definitions of variables, constants, and functions. Should there be a discrepancy between a state diagram and descriptive text, the state diagram prevails.

The notation used in the state diagrams in this clause follows the conventions in 21.5. State diagram variables follow the conventions of 21.5.2 except when the variable has a default value. Variables in a state diagram with default values evaluate to the variable default in each state where the variable value is not explicitly set.

Timeless states are employed as an editorial convenience to facilitate the distribution of transition conditions from prior states. No actions are taken within these states. Exit conditions are evaluated for timeless states. There is one timeless state. It is PCS Receive state RECEIVE.

#### 48.2.6.1 State variables

##### 48.2.6.1.1 Notation conventions

/x/
: Denotes the constant code-group specified in 48.2.6.1.2 (valid code-groups must follow the rules of running disparity as per 36.2.4.5 and 36.2.4.6).

[/x/]
: Denotes the latched received value of the constant code-group (/x/) specified in 48.2.6.1.2 and conveyed by the SYNC_UNITDATA.indicate message described in 48.2.6.1.6.

||y||
: Denotes the column of constant code-groups in lanes 0 through 3, inclusively, specified in 48.2.6.1.2 (valid code-groups must follow the rules of running disparity as per 36.2.4.5 and 36.2.4.6).

[||y||]
: Denotes the latched received value of the column of constant code-groups in lanes 0 through 3, inclusively (||y||), specified in 48.2.6.1.2 and conveyed by the SYNC_UNITDATA.indicate message described in 48.2.6.1.6 or the ALIGN_UNITDATA.indicate message described in 48.2.6.1.6.

#### 48.2.6.1.2 Constants

/A/
: The /K28.3/ code-group used in the Idle Align function specified in 48.2.4.2.2.

||A||
: The column of four identical Idle Align code-groups corresponding to the Idle Align function specified in 48.2.4.2.2.

/COMMA/
: The set of special code-groups which include a comma as specified in 36.2.4.9 and listed in Table 36-2.

/D/
: The set of 256 code-groups corresponding to valid data, as specified in 48.2.4.1 and listed in Table 36-1.

||D||
: The column of four Data code-groups present during packet reception.

/Dx.y/
: One of the set of 256 code-groups corresponding to valid data, as specified in 48.2.4.1 and listed in Table 36-1.

/E/
: The /K30.7/ code-group corresponding to the Error function specified in 48.2.4.4.

||I||
: The column of four identical Idle code-groups corresponding to the Idle function specified in 48.2.4.2. Also used to represent the corresponding XGMII control characters.

||IDLE||
: Alias for ||I||.

/INVALID/
: The set of invalid data or special code-groups, as specified in 36.2.4.6.

/K/
: The /K28.5/ code-group used in the Idle Sync function specified in 48.2.4.2.1. Also used in the Terminate function specified in 48.2.4.3.2.

||K||
: The column of four identical Idle Sync code-groups corresponding to the Idle Sync function specified in 48.2.4.2.1.

/Kx.y/
: One of the set of 12 code-groups corresponding to valid special code-groups, as specified in Table 36-2.

LFAULT
: A vector of bits RXD<31:0> and RXC<3:0> containing a Local Fault sequence ordered_set. The Local Fault sequence ordered_set is defined in 46.3.4.

||Q||
: The column of four code-groups corresponding to the Sequence function specified in 48.2.4.5.1. Also used to represent the corresponding set of XGMII control and data characters.

/R/
: The /K28.0/ code-group used for the Idle Skip function specified in 48.2.4.2.3.

||R||

The column of four identical Idle Skip code-groups corresponding to the Idle Skip function specified in 48.2.4.2.3.

||S||

The column of code-groups including the Start code-group as specified in 48.2.4.3.1. Also used to represent the corresponding set of XGMII control and data characters.

||T||

The column of code-groups including the Terminate code-group as specified in 48.2.4.3.2. Also used to represent the corresponding XGMII control and data characters.

#### 48.2.6.1.3 Variables

align_column <39:0>

A vector of bits represented by the most recently received column of aligned 10-bit code-groups on all four lanes from the PCS Deskew process. For lane 0, the element align_column <0> corresponds to the least recently received (oldest) rx_bit of the code-group; align_column <9> corresponds to the most recently received (newest) rx_bit of the code-group. The same bit aging is applicable to the bits in align_column <39:10> with respect to rx_lane <3:1>. Code-group to lane assignment is specified in Table 48–1.

align_status

A parameter set by the PCS Deskew process to reflect the status of the lane-to-lane code-group alignment.

Values: FAIL; The deskew process is not complete.
OK; All lanes are synchronized and aligned.

code_sel

A boolean derived from a uniformly distributed random integer r generated by a PRBS based on a 7th order polynomial.

Values: 0; LSB of random number is zero.
1; LSB of random number is one.

deskew_error

A boolean used by the PCS Deskew process to indicate that a lane-to-lane alignment error has been detected.

Values: FALSE; /A/ not recognized in any lane or recognized in all lanes simultaneously.
TRUE; /A/ recognized in fewer than all lanes.

enable_cgalign

A boolean that indicates the enabling and disabling of code-group comma alignment. The code-group boundary may be changed whenever code-group comma alignment is enabled. This process is known as code-group alignment.

Values: FALSE; Code-group alignment is disabled.
TRUE; Code-group alignment is enabled.

enable_deskew

A boolean that indicates the enabling and disabling of the deskew process. Code-groups may be discarded whenever deskew is enabled. This process is known as code-group slipping.

Values: FALSE; Deskew is disabled.
TRUE; Deskew is enabled.

IDLE

A vector of bits RXD<31:0> and RXC<3:0> containing Idle. Idle is defined in Table 46-4.

lane_sync_status <3:0>

A parameter set by the PCS Synchronization process to reflect the status of the link for each lane as viewed by the receiver. lane_sync_status <n> represents lane_sync_status on lane n where n=0:3.

Values: FAIL; The receiver is not synchronized to the code-group boundary.
OK; The receiver is synchronized to the code-group boundary.

next_ifg

Controls the ||IDLE|| pattern immediately following the next frame. It is used to ensure an equal and deterministic presence of both ||A|| and ||K||

Values: A; The first ||IDLE|| following the end of the next frame will be ||A||.
K; The first ||IDLE|| following the end of the next frame will be ||K||.

reset

Condition that is true until such time as the power supply for the device that contains the PCS has reached the operating region. The condition is also true when the device has low-power mode set via Control register bit 4.0.11 or 5.0.11. The condition is also true when a reset request is detected via Control register bit 4.0.15 or 5.0.15.

Values: FALSE; The device is completely powered and has not been reset (default).
TRUE; The device has not been completely powered or has been reset.

NOTE—Reset evaluates to its default value in each state where it is not explicitly set.

RX

Alias for RXD <31:0> and RXC <3:0> representing the XGMII Receive Data and Control signals.

RXC <3:0>

Receive Control signals of the XGMII as specified in Clause 46. Set by the PCS Receive process.

RXD <31:0>

Receive Data signals of the XGMII as specified in Clause 46. Set by the PCS Receive process.

rx_lane <3:0>

A vector of bits representing the serial lanes used to convey data from the PMD to the PMA via the PMD_UNITDATA.indicate service primitive as specified in Clause 53.

Bit values: ZERO; Data bit is a logical zero.
ONE; Data bit is a logical one.

rx_unaligned <39:0>

A vector of bits represented by the most recently received column of unaligned 10-bit code-groups on all four lanes from the PMA. For lane 0, the element rx_unaligned <0> is the least recently received (oldest) rx_bit from the PMD; rx_unaligned <9> is the most recently received (newest) rx_bit from the PMD. The same bit aging is applicable to the bits in rx_unaligned <39:10> with respect to rx_lane <3:1>. Code-group to lane assignment is specified in Table 48–1.

signal_detect <3:0>

A parameter set continuously and directly from the PMD_SIGNAL.indicate(signal_detect <3:0>) primitive to reflect the status of the incoming link signal for each lane. Used by the PCS Synchronization process to validate the data received on rx_unaligned <39:0>. Signal_detect <n> represents signal_detect on lane n where n=0:3.

Values: FAIL; A signal is not present on the lane.
OK; A signal is present on the lane.

sync_code-group <39:0>

A vector of bits represented by the most recently received column of unaligned 10-bit code-groups on all four lanes from the PCS Synchronization process. For lane 0, the element sync_code-group <0> corresponds to the least recently received (oldest) rx_bit; sync_code-group <9> corresponds to the most recently received (newest) rx_bit. The same bit aging is applicable to the bits in sync_code-group<39:10> with respect to rx_lane <3:1>. Code-group to lane assignment is specified in Table 48–1.

sync_status

A boolean that represents the following behavior: For all n in lane_sync_status<n>.

Values: FAIL; At least one lane is not in sync.
OK; All lanes are in sync.

TQMSG

A vector of bits representing the last link status message received over the XGMII once the link status message is recognized. Used by the PCS Transmit process to load tx_code-group <39:0>.

tx_code-group <39:0>

A vector of bits representing a column of four aligned 10-bit code-groups which has been prepared for transmission by the PCS Transmit process. This vector is conveyed to the PMA as the parameter of a PMA_UNITDATA.request(tx_lane <3:0>) service primitive. For lane 0, the element tx_code-group <0> is the first tx_bit transmitted; tx_code-group <9> is the last tx_bit transmitted. The same bit aging is applicable to the bits in tx_code-group <39:10> with respect to tx_lane <3:1>. Code-group to lane assignment is specified in Table 48–1.

TX

Alias for either TXD <31:0> and TXC <3:0> representing the XGMII Transmit Data and Control signals, or the Local Fault ordered_set as defined in 46.3.4 when a fault condition is detected on the transmit path.

TXC <3:0>

Transmit Control signals of the XGMII as specified in Clause 46. Interpreted by the PCS Transmit process.

TXD <31:0>

Transmit Data signals of the XGMII as specified in Clause 46. Interpreted by the PCS Transmit process.

tx_lane <3:0>

A vector of bits representing the serial lanes used to convey data from the PMA to the PMD via the PMD_UNITDATA.request service primitive as specified in Clause 53.

Bit values: ZERO; Data bit is a logical zero.
ONE; Data bit is a logical one.

#### 48.2.6.1.4 Functions

check_end

Prescient Terminate function used by the PCS Receive process to set the RXD<31:0> and RXC<3:0> signals to indicate Error if a running disparity error was propagated to any Idle code-groups in ||T||, or to the column following ||T||. The XGMII Error control character is returned in all lanes less than n in ||T||, where n identifies the specific Terminate ordered-set $||T_n||$, for which a running disparity error or any code-groups other than /A/ or /K/ are recognized in the column following ||T||. The XGMII Error control character is also returned in all lanes greater than n in the column prior to ||T||, where n identifies the specific Terminate ordered-set $||T_n||$, for which a running disparity error or any code group other than /K/ is recognized in the corresponding lane of ||T||. For all other lanes the value set previously is retained.

cvrx_terminate

Conversion function used by the PCS Receive process when Terminate is indicated to convert all /K/ code-groups to Idle control characters signaled via RX. Conversion is performed for all lanes.

cvtx_terminate

Conversion function used by the PCS Transmit process when Terminate is indicated to convert all Idle control characters signaled via TX to /K/ code-groups. Conversion is performed for all lanes.

DECODE ([||y||])

Consists of four independent synchronous processes, one each per lane. In the PCS Receive process, this function takes as its arguments the latched value of align_column ([||y||]) and the current running disparity, and returns XGMII RX as specified in 48.2.3 and 48.2.4. When decoding ||T||, the returned XGMII RX value is further modified by the cvrx_terminate and check_end functions, the result of the check_end function takes priority over the result of the cvrx_terminate function. DECODE also updates the current running disparity per the running disparity rules outlined in 36.2.4.4.

ENCODE(TX)

Consists of four independent synchronous processes, one each per lane. In the PCS Transmit process, this function takes as its argument the XGMII TX signals and the current running disparity for each lane, and returns four corresponding 10-bit code-groups as specified in 48.2.3 and 48.2.4. When encoding ||T||, the XGMII TX values are modified by the result of the cvtx_terminate function. ENCODE also updates the current running disparity per Tables 36-1 or 36-2.

Q_det

Function to determine the need to transmit sequence ordered_sets. If TX=||Q|| then Q_det is set to true and TQMSG is set to the result of ENCODE(TX). Q_det remains true until set to false by the PCS transmit source state diagram. In the event that this function and the state diagram both attempt to modify Q_det, the setting of Q_det by this function to true will take priority.

signal_detectCHANGE <3:0>

In the PCS Synchronization process, this function monitors the signal_detect parameter on a per lane basis for a state change. The function is set upon state change detection, which is required to detect signal_detect changes which occur asynchronously to PUDI. signal_detectCHANGE <n> represents signal_detectCHANGE on lane n where n=0:3.

Values: TRUE; The output of this function changes to true when the function detects a change in signal_detect and stays true until the false condition is satisfied.
FALSE; The output of this function changes to false when the LOSS_OF_SYNC state of the PCS synchronization state diagram is entered.

#### 48.2.6.1.5 Counters

A_CNT
: A 5-bit down counter used to control Align code-group spacing. Loaded with a value between 16 and 31, inclusive, by a uniformly distributed random integer r generated by a PRBS based on a 7th order polynomial as described in 48.2.4.2. A_CNT is decremented once per PUDR. The count remains at zero until ||A|| is transmitted, at which time a new value is loaded.

good_cgs
: A 2-bit consecutive valid code-groups received counter.

#### 48.2.6.1.6 Messages

AUDI([||y||])
: Alias for ALIGN_UNITDATA.indicate(parameters).

ALIGN_UNITDATA.indicate([align_column <39:0>])
: A signal sent by the PCS Deskew process to the PCS Receive process conveying the latched value of the indicated column of code-groups over each lane ([||y||]) (see 48.2.6.2.3).

PMA_UNITDATA.indicate(rx_unaligned <39:0>)
: A signal sent by the PMA Receive process to the PCS Synchronization process conveying the next code-group set received over each lane of the medium (see 48.3.2.2).

PMA_UNITDATA.request(tx_code-group <39:0>)
: A signal sent by the PCS Transmit process conveying the next code-group set for all lanes ready for transmission over the medium (see 48.3.2.1).

PMD_SIGNAL.indicate(signal_detect <3:0>)
: Indicates the status of the incoming link signal. A signal mapped to the PMD_SIGNAL.indicate(SIGNAL_DETECT) service primitive specified in Clause 53. signal_detect <n> is set to the same value for all lanes n where n=0:3.

  Values: FAIL; A signal is not present on the lane.
  OK; A signal is present on the lane.

PUDI
: Alias for PMA_UNITDATA.indicate(parameters).

PUDR
: Alias for PMA_UNITDATA.request(parameters).

SUDI
: Alias for SYNC_UNITDATA.indicate(parameters).

SYNC_UNITDATA.indicate(sync_code-group <39:0>)
: A signal sent by the PCS Synchronization process to the PCS Deskew process conveying code-groups over each lane (see 48.2.6.2.2).

### 48.2.6.2 State diagrams

#### 48.2.6.2.1 Transmit

The PCS shall implement its Transmit process as depicted in Figure 48–6, including compliance with the associated state variables as specified in 48.2.6.1. This state machine makes exactly one transition for each transmitted ordered_set that is processed.

The Transmit Source process determines whether XGMII data and control information should be passed through to the PMA for serialization or converted before passing to the PMA. In all cases, XGMII data and control information is encoded before it is passed to the PMA. Data and control information may be sourced either directly from the XGMII or generated by the PCS depending on whether packet or Idle is being sourced by the XGMII. The recognition of the XGMII Start, Terminate, Idle and Sequence control characters is used to determine whether packet, Idle, or link status is being sourced by the XGMII.

The detection of a link status condition including the receipt of link status messages over the XGMII causes the PCS Transmit process to generate link status messages interspersed in an Idle sequence.

The PCS Transmit process continuously sources tx_code-group<39:0> to the PMA. The Transmit process determines the proper code-group to source on each lane based on running disparity requirements.

#### 48.2.6.2.2 Synchronization

The PCS shall implement four Synchronization processes as depicted in Figure 48–7 including compliance with the associated state variables as specified in 48.2.6.1. The Synchronization process is responsible for determining whether the underlying receive channel is ready for operation. Failure of the underlying channel typically causes the PMA client to suspend normal actions. A Synchronization process operates independently on each lane, and synchronization is complete only when synchronization is acquired on all lanes. The synchronization process described in the following paragraphs applies to each lane.

The PCS Synchronization process continuously accepts code-groups via the PMA_UNITDATA.indicate primitive and conveys received code-groups to the PCS Deskew process via the SYNC_UNITDATA.indicate message.

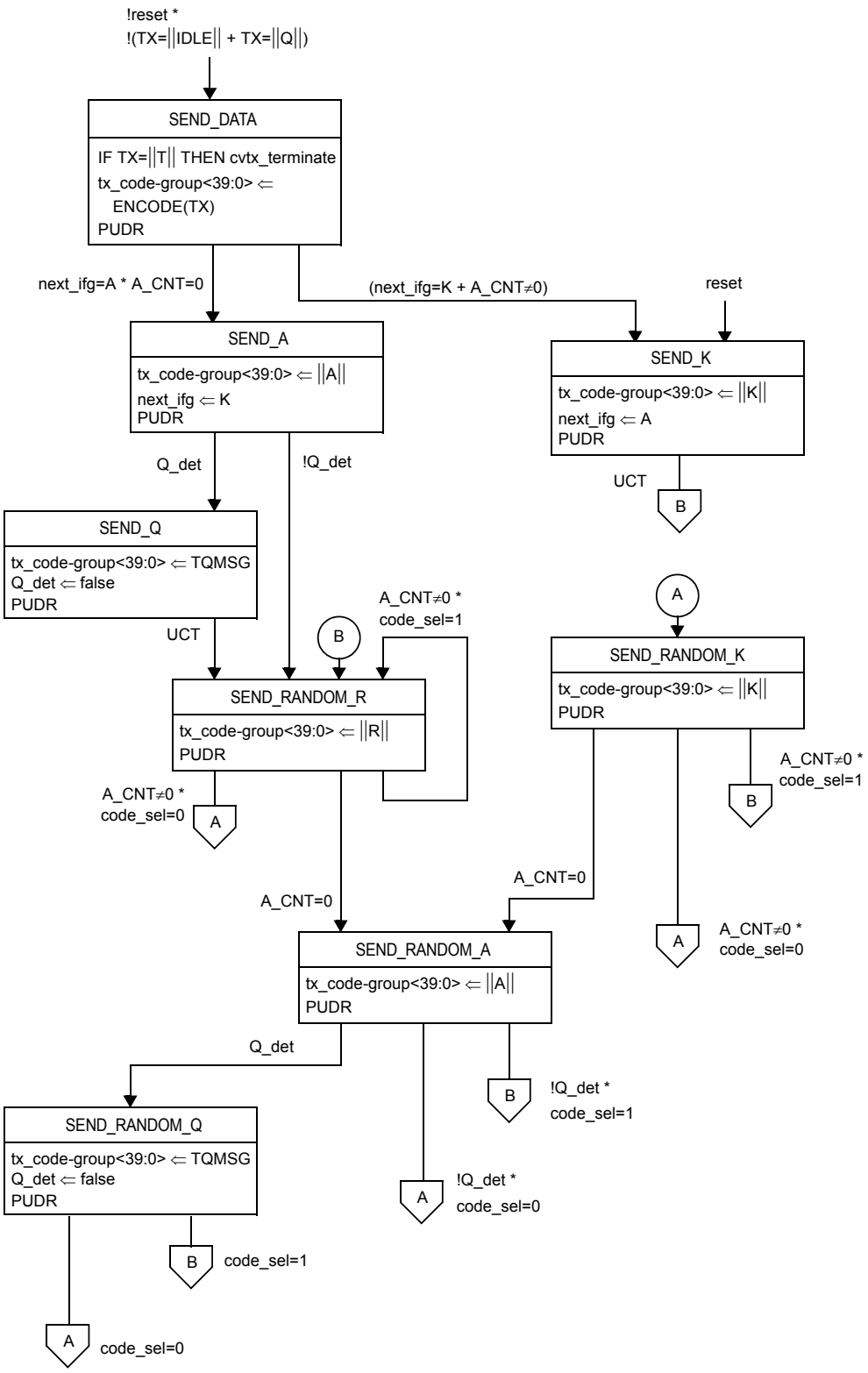
When in the LOSS_OF_SYNC state, the PCS may attempt to realign its current code-group boundary to one which coincides with the code-group boundary defined by a comma (see 36.2.4.9). This process is referred to in this document as code-group alignment.

Once synchronization is acquired, the Synchronization process tests received code-groups in sets of four code-groups and employs multiple sub-states, effecting hysteresis, to move between the SYNC_ACQUIRED_1 and LOSS_OF_SYNC states. The Synchronization process sets the lane_sync_status <3:0> flags to indicate whether the PMA is functioning dependably (as well as can be determined without exhaustive error-rate analysis). Whenever any PMA lane is not operating dependably, as indicated by the setting of lane_sync_status <3:0>, the align_status flag is set to FAIL.

#### 48.2.6.2.3 Deskew

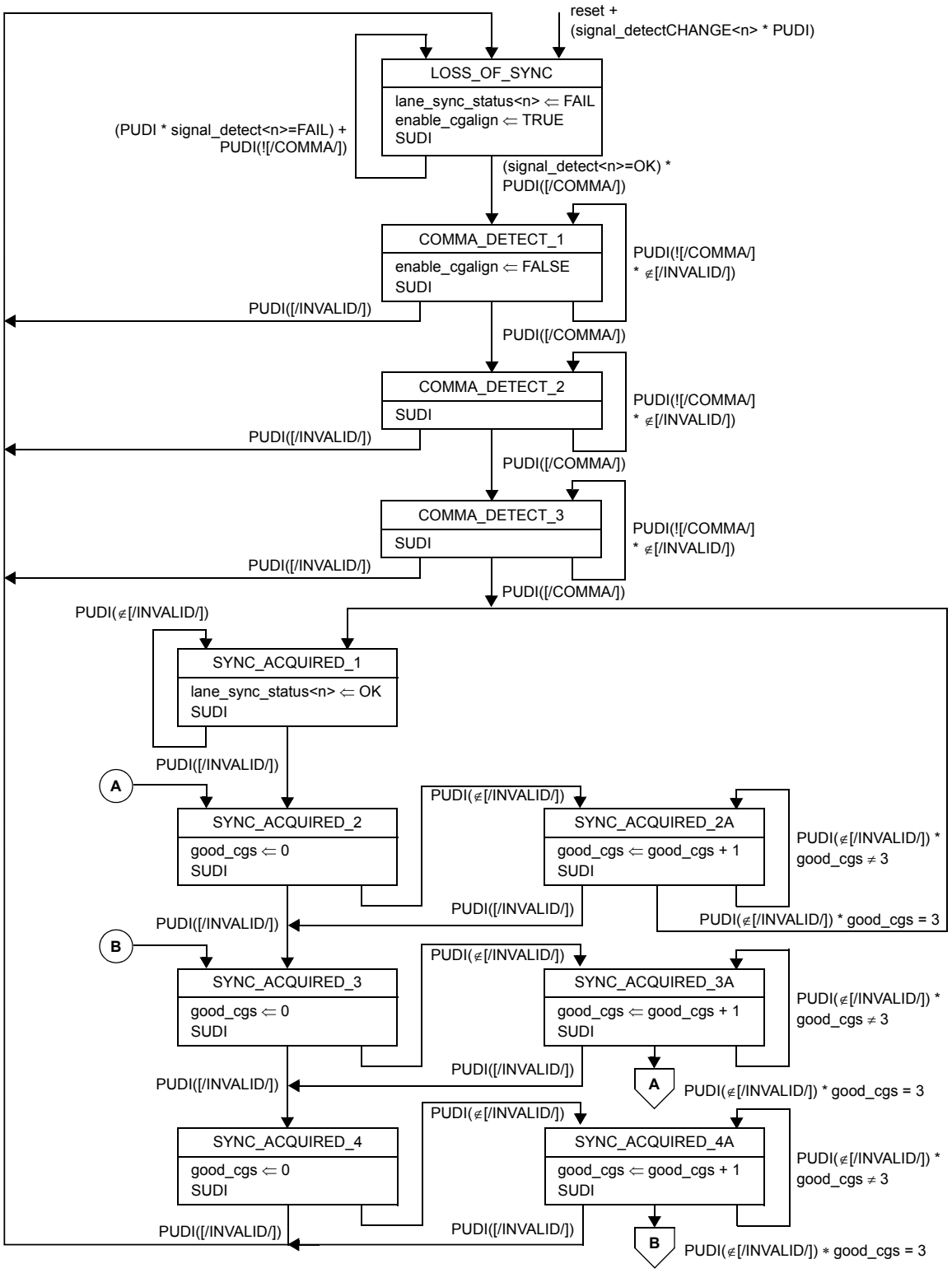
The PCS shall implement the Deskew process as depicted in Figure 48–8 including compliance with the associated state variables as specified in 48.2.6.1. The Deskew process is responsible for determining whether the underlying receive channel is capable of presenting coherent data to the XGMII. The Deskew process asserts the align_status flag to indicate that the PCS has successfully deskewed and aligned code-groups on all lanes. The Deskew process attempts deskew and alignment whenever the align_status flag is de-asserted. The Deskew process is otherwise idle. Whenever the align_status flag is set to FAIL the condition is indicated as a link_status=FAIL condition in the status register bit 4.1.2 or 5.1.2.

Once alignment is acquired, the Deskew process tests received columns and employs multiple sub-states, effecting hysteresis, to move between the ALIGN_ACQUIRED_1 and LOSS_OF_ALIGNMENT states. These states monitor the link for continued alignment, tolerate alignment inconsistencies due to a reasonably low BER, and restart the Deskew process if alignment can not be reliably maintained.




NOTE—The state machine makes exactly one transition for each transmit code-group processed.

**Figure 48–6—PCS transmit source state diagram**




NOTE— lane_sync_status<n>, signal_detect<n> and signal_detectCHANGE<n>, refer to the number of the received lane n where n=0:3

**Figure 48–7—PCS synchronization state diagram**

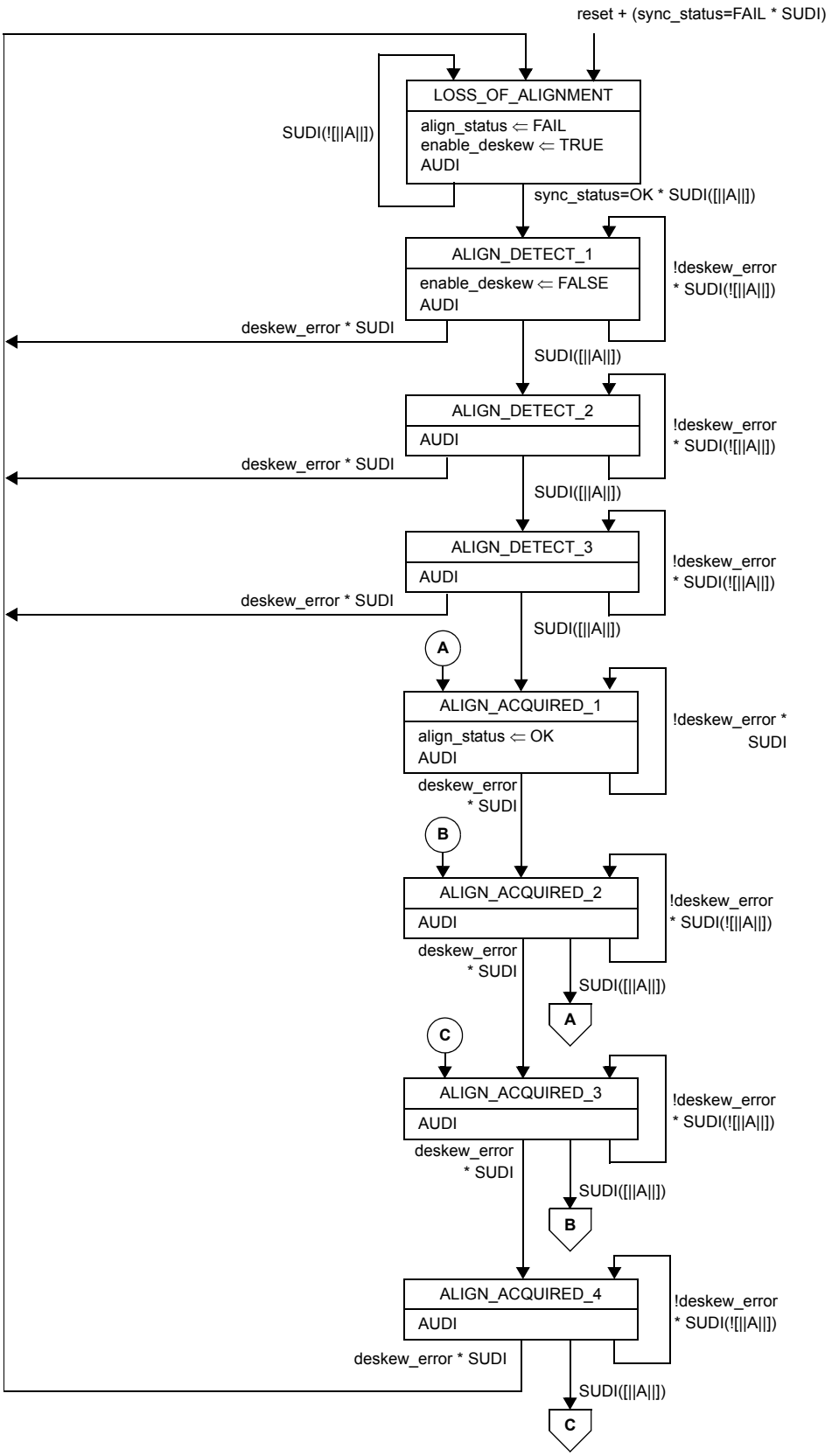



**Figure 48–8—PCS deskew state diagram**

### 48.2.6.2.4 Receive

The PCS shall implement its Receive process as depicted in Figure 48–9, including compliance with the associated state variables as specified in 48.2.6.1.

The PCS Receive process continuously performs the DECODE function on the received code-groups from the PCS Deskew Process via the ALIGNED_UNITDATA.indicate message. The PCS Receive process generates the receive clock signal of the XGMII (RX_CLK) as specified in Clause 46. State transitions in the PCS Receive state diagram that generate the data and control characters (RXD<31:0> and RXC<3:0>) on the XGMII occur synchronous to RX_CLK. The Receive process operates in the following two modes:

a) Data mode during packet reception including Start and Terminate. Additionally, Data mode is active whenever !||I|| columns are received during the Idle sequence or !||I|| or !||Q|| columns are received during the Fault sequence signifying either an error or unusual or unsupported indication. Valid code-groups received while in Data mode are mapped to corresponding XGMII data or control characters regardless of whether or not the control characters are valid XGMII control characters. Invalid or Error code-groups are mapped directly to XGMII Error control characters. All code-groups are mapped on a lane by lane basis.

b) Idle mode during idle reception excluding Start and Terminate. Idle mode is active whenever ||I|| is received during idle reception. ||I|| is translated to XGMII Idle control characters.

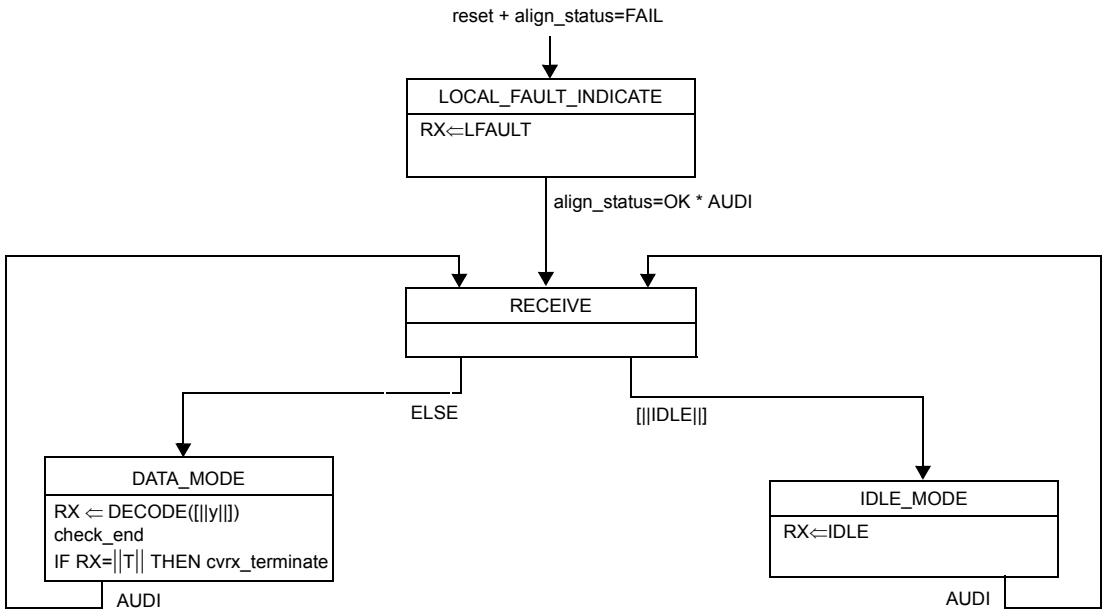



**Figure 48–9—PCS receive state diagram**

### 48.2.6.3 Initialization process

Link initialization involves the completion of the PCS Synchronization and Deskew processes and the ability to transmit and receive code-groups via the PCS Transmit and Receive processes, respectively. The status register link_status flag is set to OK whenever the align_status flag is set to OK and no errors preventing link operation are present in the PCS or PMA.

#### 48.2.6.4 Link status reporting

Link status reporting involves detection of link status conditions and the signaling of link fault status. The purpose of link status reporting is to quickly identify and convey link status conditions to the RS which can take the necessary action to activate (deactivate) the link via the setting of the RS link_fault 2-bit variable. Link status reporting and MAC packet transmission is mutually exclusive.

##### 48.2.6.4.1 Link status detection

10GBASE-X link status conditions include signal and deskew status conditions. Link status conditions include Local Fault and Remote Fault signals. A receive fault is recognized by the PCS Receive process whenever align_status=FAIL. Other fault conditions are not detected by the PCS or PMA and are detected only by the RS. A fault condition may also be recognized by any 10GBASE-X process upon detection of an error condition, which prevents continued reliable operation, but this is beyond the scope of this standard.

##### 48.2.6.4.2 Link status signaling

Link status signaling follows the detection or recognition of a link status condition and involves the generation of Sequence ordered-sets (||Q||) by the PCS Transmit process. Link status signaling involves the transmission of ||Q|| following ||A|| transmission in the Idle sequence as specified in 48.2.4.2.

##### 48.2.6.4.3 Link status messages

A 10GBASE-X link status message is a Sequence ordered-set. Ordered-sets associated with link status messages are specified in Table 48–4. Link status messages detected by the PCS Receive process are forwarded to the XGMII.

### 48.3 Physical Medium Attachment (PMA) sublayer

The PMA is specified in the form of a service interface to the PCS. These services are described in an abstract manner and do not imply any particular implementation. The PMA service interface supports the exchange of code-group information between PCS entities. The PMA converts code-groups into bits and passes these to the PMD, and vice versa.

#### 48.3.1 Functions within the PMA

The PMA comprises the PMA Transmit process and PMA Receive process. Figure 48–4 depicts the mapping of the 36-bit-wide data and control path of the XGMII to the forty-bit-wide code-groups of the PMA service interface, and on to the four lane serial PMD service interface.

NOTE—Strict adherence to manufacturer-supplied guidelines for the operation and use of PMA serializer components is required to meet the jitter specifications of Clause 47 and Clause 53. The supplied guidelines should address the quality of power supply filtering associated with the transmit clock generator, and also the purity of the reference clock fed to the transmit clock generator.

##### 48.3.1.1 PMA transmit process

The PMA Transmit process passes data unaltered (except for serializing) from the PCS directly to the PMD. Upon receipt of the PMA_UNITDATA.request primitive, the PMA shall individually serialize the four aligned 10-bit code-groups, one from each of four lanes, and transmit them to the PMD in the form of forty PMD_UNITDATA.request primitives, 10 each on four lanes.

Within each lane for each 10-bit code-group, the lowest numbered bit of the PMA_UNITDATA.indicate parameter corresponds to the first bit transmitted to the PMD and the highest numbered bit of the

PMA_UNITDATA.indicate parameter corresponds to the last bit transmitted to the PMD. There is no numerical significance ascribed to the bits within a PMA code-group; that is, the code-group is simply a bit pattern that has some predefined interpretation. PMA to PMD bit and lane association is illustrated in Table 48–1. PMA to PMD bit ordering is illustrated in Figure 48–4.

PMA_UNITDATA.request primitives shall be generated with a frequency of 312.5 MHz ± 100 ppm. An internal clock multiplier unit multiplies the frequency of the 10-bit code-group based PMA_UNITDATA.request primitives by a factor of 10 to serialize the latched data out of the PMA and into the PMD.

#### 48.3.1.2 PMA receive process

The PMA Receive process passes data unaltered (except for deserializing) from the PMD directly to the PCS. Upon receipt of forty PMD_UNITDATA.indicate primitives, 10 in succession from each of four lanes, the PMA shall assemble the bits received into a 40-bit vector representing four 10-bit unaligned code-groups and pass that vector to the PCS as the parameter of four PMA_UNITDATA.indicate primitives.

Within each lane for each 10-bit code-group, the lowest numbered bit of the PMA_UNITDATA.indicate parameter corresponds to the first bit received from the PMD and the highest numbered bit of the PMA_UNITDATA.indicate parameter corresponds to the last bit received from the PMD. There is no numerical significance ascribed to the bits within a PMA code-group; that is, the code-group is simply a bit pattern that has some predefined interpretation. Receive code-group alignment is performed by the PCS and is not applicable to PMA. PMA to PMD bit and lane association is illustrated in Table 48–1. PMA to PMD bit ordering is illustrated in Figure 48–4.

PMA_UNITDATA.indicate primitives shall be generated with a frequency of 312.5 MHz ± 100 ppm. The line rate of each of four PMD lanes is 3.125 GBaud ± 100 ppm. The serial data stream of the PMD includes an embedded clock that nominally operates at a frequency 10 times higher than that of a single lane code-group stream. The PMA Receive process shall recover a clock from a valid 8B/10B received data stream if the stream is within tolerance.

### 48.3.2 Service interface

The following primitives are defined:

PMA_UNITDATA.request(tx_code-group <39:0>)

PMA_UNITDATA.indicate(rx_unaligned <39:0>)

#### 48.3.2.1 PMA_UNITDATA.request

This primitive defines the transfer of data in the form of aligned code-groups from the PCS to the PMA. PMA_UNITDATA.request is generated by the PCS Transmit process.

##### 48.3.2.1.1 Semantics of the service primitive

PMA_UNITDATA.request(tx_code-group <39:0>)

The data conveyed by PMA_UNITDATA.request is the tx_code-group <39:0> parameter defined in 48.2.6.1.3.

#### 48.3.2.1.2 When generated

The PCS continuously sends tx_code-group <39:0> to the PMA at a nominal rate of 312.5 MHz ± 100 ppm, as governed by frequency and tolerance of XGMII TX_CLK.

#### 48.3.2.1.3 Effect of receipt

Upon receipt of this primitive, the PMA generates a series of forty PMD_UNITDATA.request primitives, 10 for each of the four PMD lanes, tx_lane <3:0>, requesting the four lane serial transmission of tx_code-group <39:0> to the PMD.

### 48.3.2.2 PMA_UNITDATA.indicate

This primitive defines the transfer of data in the form of code-groups from the PMA to the PCS. PMA_UNITDATA.indicate is used by the PCS Synchronization process.

#### 48.3.2.2.1 Semantics of the service primitive

PMA_UNITDATA.indicate(rx_unaligned <39:0>)

The data conveyed by PMA_UNITDATA.indicate is the rx_unaligned <39:0> parameter defined in 48.2.6.1.3.

#### 48.3.2.2.2 When generated

The PMA continuously sends rx_unaligned <39:0> to the PCS at a nominal rate of 312.5 MHz ± 100 ppm, as governed by frequency and tolerance of PMD_UNITDATA.indicate.

#### 48.3.2.2.3 Effect of receipt

Upon receipt of this primitive, the PCS Sychronization process attempts to achieve code-group synchronization on each lane (see 48.2.6.2.2).

### 48.3.3 Loopback mode

Loopback mode shall be provided for the 10GBASE-X PMA and DTE XGXS, and optionally for the PHY XGXS, as specified in this subclause, by the transmitter and receiver of a device as a test function to the device. When Loopback mode is selected, transmission requests passed to the transmitter are shunted directly to the receiver, overriding any signal detected by the receiver on its attached link. A device is explicitly placed in Loopback mode (i.e., Loopback mode is not the normal mode of operation of a device). Loopback applies to all lanes as a group (i.e., the lane 0 transmitter is directly connected to the lane 0 receiver, the lane 1 transmitter is directly connected to the lane 1 receiver, etc.). The method of implementing Loopback mode is not defined by this standard.

Control of the Loopback function may be supported through the MDIO management interface of Clause 45 or equivalent.

NOTE—The signal path that is exercised in the Loopback mode is implementation specific, but it is recommended that this signal path encompass as much of the circuitry as is practical. The intention of providing this Loopback mode of operation is to permit diagnostic or self-test functions to test the transmit and receive data paths using actual data. Other loopback signal paths may also be enabled independently using loopback controls within other devices or sublayers.

#### 48.3.3.1 Receiver considerations

Entry into or exit from Loopback mode may result in a temporary loss of synchronization.

#### 48.3.3.2 Transmitter considerations

While in Loopback mode, the transmitter output is not defined.

### 48.3.4 Test functions

A limited set of test functions may be provided as an implementation option for testing of the transmitter function or for testing of an attached receiver.

Some test functions that are not defined by this standard may be provided by certain implementations. Compliance with the standard is not affected by the provision or exclusion of such functions by an implementation. Random jitter test patterns for 10GBASE-X are specified in Annex 48A. Test-pattern capability and selection is optional and supported via MDIO register bits defined in Clause 45. Jitter Test methodology for 10GBASE-X is specified in Annex 48B.

A typical test function is the ability to transmit invalid code-groups within an otherwise valid PHY bit stream. Certain invalid PHY bit streams may cause a receiver to lose word and/or bit synchronization. See ANSI X3.230-1994, FC-PH subclause 5.4 for a more detailed discussion of receiver and transmitter behavior under various test conditions.

## 48.4 Compatibility considerations

There is no requirement for a compliant device to implement or expose any of the interfaces specified for the PCS or PMA. Implementations of an XGMII shall comply with the requirements as specified in Clause 46.

## 48.5 Delay constraints

Predictable operation of the MAC Control PAUSE operation (Clause 31, Annex 31B) demands that there be an upper bound on the propagation delays through the network. This implies that MAC, MAC Control sublayer, and PHY implementers must conform to certain delay maxima, and that network planners and administrators conform to constraints regarding the cable topology and concatenation of devices.

The sum of transmit and receive delay contributed by the 10GBASE-X PCS shall be no more than 2048 BT.

The reference point for all MDI measurements is the 50% point of the mid-cell transition corresponding to the reference bit, as measured at the MDI.

## 48.6 Environmental specifications

All equipment subject to this clause shall conform to the requirements of 14.7 and applicable sections of ISO/IEC 11801: 1995.

## 48.7 Protocol Implementation Conformance Statement (PICS) proforma for Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer, type 10GBASE-X[8]

### 48.7.1 Introduction

The supplier of a protocol implementation that is claimed to conform to IEEE Std 802.3ae-2002, Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer, type 10GBASE-X, shall complete the following Protocol Implementation Conformance Statement (PICS) proforma. A detailed description of the symbols used in the PICS proforma, along with instructions for completing the PICS proforma, can be found in Clause 21.

### 48.7.2 Identification

#### 48.7.2.1 Implementation identification

| | |
|---|---|
| Supplier[1] | |
| Contact point for enquiries about the PICS[1] | |
| Implementation Name(s) and Version(s)[1,3] | |
| Other information necessary for full identification—e.g., name(s) and version(s) for machines and/or operating systems; System Names(s)[2] | |
| NOTES<br>1—Required for all implementations.<br>2—May be completed as appropriate in meeting the requirements for the identification.<br>3—The terms Name and Version should be interpreted appropriately to correspond with a supplier's terminology (e.g., Type, Series, Model). | |

#### 48.7.2.2 Protocol summary

| | |
|---|---|
| Identification of protocol standard | IEEE Std 802.3ae-2002, Clause 48, Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA) sublayer, type 10GBASE-X |
| Identification of amendments and corrigenda to this PICS proforma that have been completed as part of this PICS | |
| Have any Exception items been required? No [ ] Yes [ ]<br>(See Clause 21; the answer Yes means that the implementation does not conform to the standard.) | |
| Date of Statement | |

[8]*Copyright release for PICS proformas:* Users of this standard may freely reproduce the PICS proforma in this annex so that it can be used for its intended purpose and may further publish the completed PICS.

### 48.7.3 Major capabilities/options

| Item | Feature | Subclause | Value/Comment | Status | Support |
|---|---|---|---|---|---|
| MD | MDIO | 45, 48.1.3.1 | Registers and interface supported | O | Yes [ ]<br>No [ ] |
| XGXS | Support of XAUI/XGXS | 47, 48.1.5 | | O | Yes [ ]<br>No [ ] |
| XGE | XGMII compatibility interface | 46, 48.1.3.1 | Compatibility interface is supported | O | Yes [ ]<br>No [ ] |
| LX4 | Support of 10GBASE-LX4 PMD | 53, 48.1.3.3 | | O | Yes [ ]<br>No [ ] |
| CX4 | Support of 10GBASE-CX4 PMD | 54, 48.1.3.3 | | O | Yes [ ]<br>No [ ] |

### 48.7.4 PICS proforma tables for the PCS and PMA sublayer, type 10GBASE-X

#### 48.7.4.1 Compatibility considerations

| Item | Feature | Subclause | Value/Comment | Status | Support |
|---|---|---|---|---|---|
| CC1 | Jitter test patterns | 48.3.4 | As per Annex 48A | M | Yes [ ] |
| CC2 | Environmental specifications | 48.6 | | M | Yes [ ] |

#### 48.7.4.2 PCS functions

| Item | Feature | Subclause | Value/Comment | Status | Support |
|---|---|---|---|---|---|
| CG | Code-group usage | 48.2.3 | PCS support of 8B/10B code-groups | M | Yes [ ] |
| IOS | ||I|| sequencing rules | 48.2.4.2 | All rules apply | M | Yes [ ] |
| PRBS | Random integer generator | 48.2.4.2 | $X^7 + X^3 + 1$ or $X^7 + X^6 + 1$. Used for ||A|| spacing. Optional and ancillary use for testing | M | Yes [ ] |
| CMA | Comma detection | 48.2.4.2.1 | comma+ and comma- for /K/ | M | Yes [ ] |
| CKCU | Clock rate compensation in unencoded idle stream | 48.2.4.2.3 | Meets the requirements of 48.2.4.2.3 | O | Yes [ ]<br>No [ ] |
| ERR | Error indication | 48.2.4.4 | Replacement of invalid code-groups with /E/ | M | Yes [ ] |
| TSD | Transmit state diagrams | 48.2.6.2.1 | Meets the requirements of Figures 48–6 | M | Yes [ ] |

| Item | Feature | Subclause | Value/Comment | Status | Support |
|---|---|---|---|---|---|
| SSD | Synchronization state diagram | 48.2.6.2.2 | Meets the requirements of Figure 48–7 | M | Yes [ ] |
| DSD | Deskew state diagrams | 48.2.6.2.3 | Meets the requirements of Figure 48–8 | M | Yes [ ] |
| RSD | Receive state diagrams | 48.2.6.2.4 | Meets the requirements of Figure 48–9 | M | Yes [ ] |

### 48.7.4.3 PMA Functions

| Item | Feature | Subclause | Value/Comment | Status | Support |
|---|---|---|---|---|---|
| PMAT | Transmit function | 48.3.1.1 | PMA_UNITDATA.request | M | Yes [ ] |
| TXRT | Transmit rate | 48.3.1.1 | 3.125 GBaud ± 100 ppm | M | Yes [ ] |
| PMAR | Receive function | 48.3.1.2 | PMA_UNITDATA.indicate | M | Yes [ ] |
| RXRT | Receive rate | 48.3.1.2 | 3.125 GBaud ± 100 ppm | M | Yes [ ] |
| CDR | Clock and data recovery | 48.3.1.2 | Required if line rate is within 3.125 GBaud ± 100 ppm | M | Yes [ ] |

### 48.7.4.4 Interface functions

| Item | Feature | Subclause | Value/Comment | Status | Support |
|---|---|---|---|---|---|
| SKEW | Allowable lane skew | 48.2.4.2.2 | Table 48–5 | M | Yes [ ] |
| LBXPMA | 10GBASE-X PMA Loopback mode | 48.3.3 | | M | Yes [ ] |
| LBDTE | DTE XGXS Loopback mode | 48.3.3 | | M | Yes [ ] |
| LBPHY | PHY XGXS Loopback mode | 48.3.3 | | O | Yes [ ] |
| DLY | Delay constraints | 48.5 | 2048 BT | M | Yes [ ] |