XX.1 Forward Error Correction for 10GBASE-KR PHY

XX.1.1 Overview

This subclause specifies an optional Forward Error Correction (FEC) mechanism for the 10GBASE-KR PHY. The FEC provides coding gain to increase the link budget and BER performance on a broader set of back plane channels as defined in clause 69. It provides additional margin to account for variations in manufacturing and evironmental conditions.

XX.1.2 Objectives

The following are the objectives for the FEC:

1. To support Forward Error Correction mechanism for 10GBASE-KR PHYs.
2. To support the full duplex mode of operation of the Ethernet MAC.
3. To support the PCS, PMA and PMD sublayers defined for 10GBASE-KR.
4. To provide a 10.3125 Gb/s effective data rate at the service interface presented by the PMA sublayer.
5. Support operations over links consistent with differential, controlled impedance traces on a printed circuit board with 2 connectors and total length up to at least 1m meeting the requirements of 69.3.
6. Support a BER objective of 10^{-12} or better.

XX.1.2 Functional Block Diagram

Figure XX-1 shows the functional block diagram of FEC for 10GBASE-KR PHY and the relationship between the PCS and PMA sublayers.

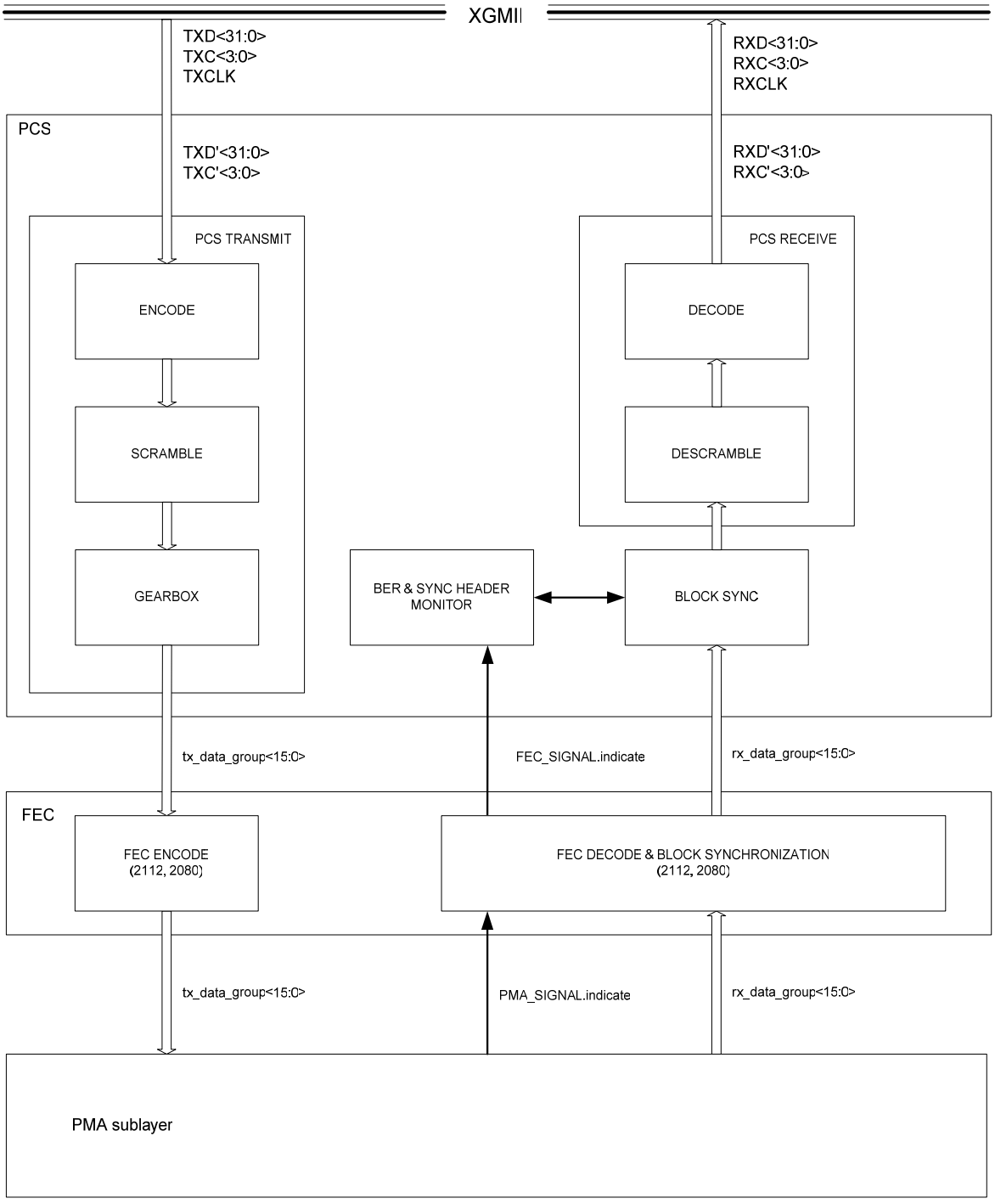

Figure XX-1 FEC Functional block diagram

XX.1.3 FEC Service Interface

The FEC Service Interface is provided to allow the 10GBASE-KR PCS to transfer information to and from the FEC. These services are defined in an abstract manner and do not imply any particular implementation. The FEC Service Interface supports exchange of data-units between PCS entities on either side of a 10GBASE-KR link using request and indication primitives. Data-units are mapped into FEC frames by the FEC and passed to the PMA, and vice versa.

The following primitives are defined within the FEC Service Interface:
FEC_UNITDATA.request(tx_data-group<15:0>)
FEC_UNITDATA.indication(rx_data-group<15:0>)
FEC_SIGNAL.indication(SIGNAL_OK)

XX.1.3.1 FEC_UNITDATA.request

This primitive defines the transfer of data in the form of constant-width data-units from the PCS to the FEC. The data supplied via FEC_UNITDATA.request is mapped by the FEC Transmit process into the payload capacity of the outgoing FEC frame stream.

XX.1.3.2 Semantics of the service primitive

FEC_UNITDATA.request(tx_data-group<15:0>)

The data conveyed by FEC_UNITDATA.request is a 16-bit vector representing a single data-unit that has been prepared for transmission by the 10GBASE-KR PCS Transmit process.

XX.1.3.3 When generated

The 10GBASE-KR PCS sends tx_data-group<15:0> to the FEC at a nominal rate of 644.53125MHz, corresponding to the 10GBASE-KR signaling speed of 10.3125Gbd.

XX.1.3.4 Effect of receipt

Upon receipt of this primitive, the FEC Transmit process maps the data conveyed by the tx_data unit<15:0> parameter into the payload of the transmitted FEC frame block stream, adds FEC overhead as required, scrambles the data, and transfers the result to the PMA via the PMA_UNITDATA.request primitives.

XX.1.4 FEC_UNITDATA.indication

This primitive defines the transfer of received data in the form of constant-width data-units from the FEC to the PCS. FEC_UNITDATA.indication is generated by the FEC Receive process in response to FEC block data received from the PMA.

XX.1.4.1 Semantics of the service primitive

FEC_UNITDATA.indication(rx_data-group<15:0>)

The rx_data-group<15:0> parameter is a 16-bit vector that represents the data-unit transferred by the FEC to the 10GBASE-KR PCS.

XX.1.4.2 When generated

The FEC sends one rx_data-group<15:0> to the 10GBASE-KR PCS whenever it has delineated exactly 16 bits of valid payload information from the incoming FEC data stream received from the PMA sublayer. The nominal rate of generation of the FEC_UNITDATA.indication primitive is 644.53125 Mtransfers/s.

XX.1.4.3 Effect of receipt

The effect of receipt of this primitive by the FEC client is unspecified by the FEC sublayer.

XX.1.5 FEC_SIGNAL.indication

This primitive is sent by the FEC to the PCS to indicate the status of the Receive process. FEC_SIGNAL.indication is generated by the FEC Receive process in order to propagate the detection of severe error conditions (e.g., no valid signal being received from the PMA sublayer) to the PCS.

XX.1.5.1 Semantics of the service primitive

FEC_SIGNAL.indication(SIGNAL_OK)

The SIGNAL_OK parameter can take one of two values: OK or FAIL. A value of OK denotes that the FEC Receive process is successfully delineating valid payload information from the incoming data stream received from the PMA sublayer, and this payload information is being presented to the PCS via the FEC_UNITDATA.indication primitive. A value of FAIL denotes that errors have been detected by the Receive process that prevent valid data from being presented to the PCS, in this case the FEC_UNITDATA.indication primitive and its associated rx_data-group<15:0> parameter are meaningless.

XX.1.5.2 When generated

The FEC generates the FEC_SIGNAL.indication primitive to the 10GBASE-KR PCS whenever there is a change in the value of the SIGNAL_OK parameter.

XX.1.5.3 Effect of receipt

The effect of receipt of this primitive by the FEC client is unspecified by the FEC sublayer.

XX.2.1 FEC Principle of operation

On transmission, the FEC sublayer receives data from the PCS, transcodes 64b66b words, performs the FEC coding / framing, scrambles and sends the data to the PMA. On reception, the FEC sublayer receives data from the PMA, performs descrambling, achieves FEC framing synchronization, decodes the FEC code, correcting data where necessary and possible, recodes 64b/66b words and sends the data to the PCS.

XX.2.1.1 FEC Code

The FEC code used is a shortened cyclic code (2112, 2080) for error checking and forward error correction. The FEC block length is 2112 bits. The code encodes 2080 bits of payload (or information symbols) and adds 32 bits of overhead (or parity symbols). The code is systematic – meaning that the information symbols are not disturbed in anyway in the encoder and the parity symbols are added separately to the end of each block.

The (2112,2080) code is constructed by shortening the cyclic code (42987, 42955). This Firecode is guranteed to correct an error burst of t=11bits per block. It is a systematic code well suited for correction of the burst errors typical in a backplane channel (Clause 69.3) resulting from error propogation in the receive equalizer.

XX.2.1.2 FEC frame Format

The frame format for the FEC block is shown in table XX-1. The length of the FEC block is 2112 bits. Each FEC block contains 32 rows of 65 bits each; 64 bits of payload and 1 bit transcoding overhead (T bits) At the end of each frame block there is 32-bit overhead or parity check bits. Transmission is from left to right within each row and from top to bottom between rows. The payload bits carry the information symbols from the PCS layer.

Table XX–1—FEC frame format

T_0	64 Bit Payload Word 0	T_1	64 Bit Payload Word 1	T_2	64 Bit Payload Word 2	T_3	64 Bit Payload Word 3
T_4	64 Bit Payload Word 4	T_5	64 Bit Payload Word 5	T_6	64 Bit Payload Word 6	T_7	64 Bit Payload Word 7
T_8	64 Bit Payload Word 8	T_9	64 Bit Payload Word 9	T_{10}	64 Bit Payload Word 10	T_{11}	64 Bit Payload Word 11
T_{12}	64 Bit Payload Word 12	T_{13}	64 Bit Payload Word 13	T_{14}	64 Bit Payload Word 14	T_{15}	64 Bit Payload Word 15
T_{16}	64 Bit Payload Word 16	T_{17}	64 Bit Payload Word 17	T_{18}	64 Bit Payload Word 18	T_{19}	64 Bit Payload Word 19
T_{20}	64 Bit Payload Word 20	T_{21}	64 Bit Payload Word 21	T_{22}	64 Bit Payload Word 22	T_{23}	64 Bit Payload Word 23
T_{24}	64 Bit Payload Word 24	T_{25}	64 Bit Payload Word 25	T_{26}	64 Bit Payload Word 26	T_{27}	64 Bit Payload Word 27
T_{28}	64 Bit Payload Word 28	T_{29}	64 Bit Payload Word 29	T_{30}	64 Bit Payload Word 30	T_{31}	64 Bit Payload Word 31
32 parity bits							

Total Block length = (32 x 65) + 32 = 2112 bits

XX.2.1.3 Placing 64b/66b PCS blocks into FEC frame

The FEC module does not decrease the symbol rate from the PCS layer, instead uses 32 compressed sync bits from the 64b/66b encoded data of the PCS sublayer to add the parity check bits.

The 10GBASE-KR 64b/66b PCS maps 64 bits of scrambled payload and 2 bits of unscrambled sync header into 66-bit encoded blocks. The 2-bit sync header allows establishment of 64b/66b block boundaries by the PCS sync process. The sync header is 01 for data blocks and 10 for control blocks, the sync header is the only position in the PCS block that always contain a transition and this feature of the code is used to establish 64b/66b block boundaries.

The FEC module compresses the 2 bits of the sync header to 1 transcode bit. The transcode bit carries the state of 10GBASE-KR sync bits for the associated payload. This is achieved by eliminating the 1st bit in 64b/66b block, which is also the first sync bit, and preserving the 2nd bit. The value of the 2nd bit defines the value of the removed 1st bit uniquely, since it is always an inversion of the 1st bit. The transcode bits are further scrambled (as explained below in a subsequent section) to ensure DC balance.

The 32 sequential 64b66b blocks are transcoded in this fashion, and then 32 bits of FEC parity are computed for them. The 32 transcoded words and the 32 FEC parity bits constitute an FEC block.

The error detection property of the FEC cyclic code is used to establish block syncronization at FEC block boundaries at the receiver. If decoding passes successfuly, the FEC decoder produces 32 65 bit words, the first decoded bit of each word being the transcode bit. Then the 1st sync bit in 64b/66b code is constructed by the inversion of the transcode bit, and the value of the 2nd sync bit is equal to the transcode bit.

The 16-bit data transmitted from the PCS gearbox function is encoded by the FEC encoder and sent to the PMA sublayer, similarly the 16-bit data received from the PMA sublayer is decoded by the FEC decoder. The resulting 64b/66b blocks are sent to the PCS sublayer.

XX 2.1.4 Functions within FEC Module

The FEC module comprises four functional blocks; FEC Encoder, Reverse Gearbox function, FEC decoder and FEC block synchronization.

The transmit data-units are sent from the FEC encoder to the PMA service interface via the PMA_UNITDATA.request primitive. When the transmit channel is in test-pattern mode, a test pattern is packed into the transmit data-units that are sent to the PMA service interface via the PMA_UNITDATA.request primitive.

XX.2.1.4.1 Reverse Gearbox function

The reverse gearbox function adapts between the 66-bit width of the 64b/66b blocks and the 16-bit width of the PCS interface. It receives the 16-bit stream from the PCS interface and converts them back to 66-bit encoded blocks for the FEC Encoder to process. The reverse gerabox function operates in the same manner as the block sync function defined in 49.2.9.

When the transmit channel is operating in normal mode, the reverse gearbox function receives data via 16-bit FEC_UNITDATA.request primitive. It will form a bit stream from the primitives by concatenating requests with the bits of each primitive in order to form tx_data-group<0> to tx_data-group<15> (see Figure 49–6). It obtains lock to the 66-bit blocks in the bit stream using the sync headers and outputs 66-bit blocks. Lock is obtained as specified in the block lock state machine shown in Figure 49–12.

The reverse gearbox functionality is necessary only when the optional PMA compatibility interface named XSBI is implemented between the PCS and FEC functions, since that interface passes data via a 16-bit wide path. When the XSBI is not implemented the internal data-path width between the PCS and FEC is an implementation choice. Depending on the path width, the reverse gearbox function may not be necessary.

XX.2.1.4.2 FEC Encoder

The FEC encoder connects to the PCS Gearbox function using the 16-bit tx_data_goup. The FEC encoder takes 32 x 64b/66b blocks from PCS and encodes it into a single FEC block of 2112 bits. The FEC Encoder compresses the two sync bits to one transcode bits as expained in the previous section. The resulting 32 x 65b = 2080 bits with the frame format as shown in Table XX-1 are fed to the (2112,2080) encoder, which produces 32 parity-check bits. The parity check bits are appended to the end of the FEC frame. The FEC frame is scrambled using the PN-2112 pseudo-noise sequence as described in XX.2.1.4.4.1. and sent to the PMA interface.

XX.2.1.4.3 FEC transmission bit ordering

The format of the FEC block and the transmit bit ordering is shown in Figure XX-2 below.

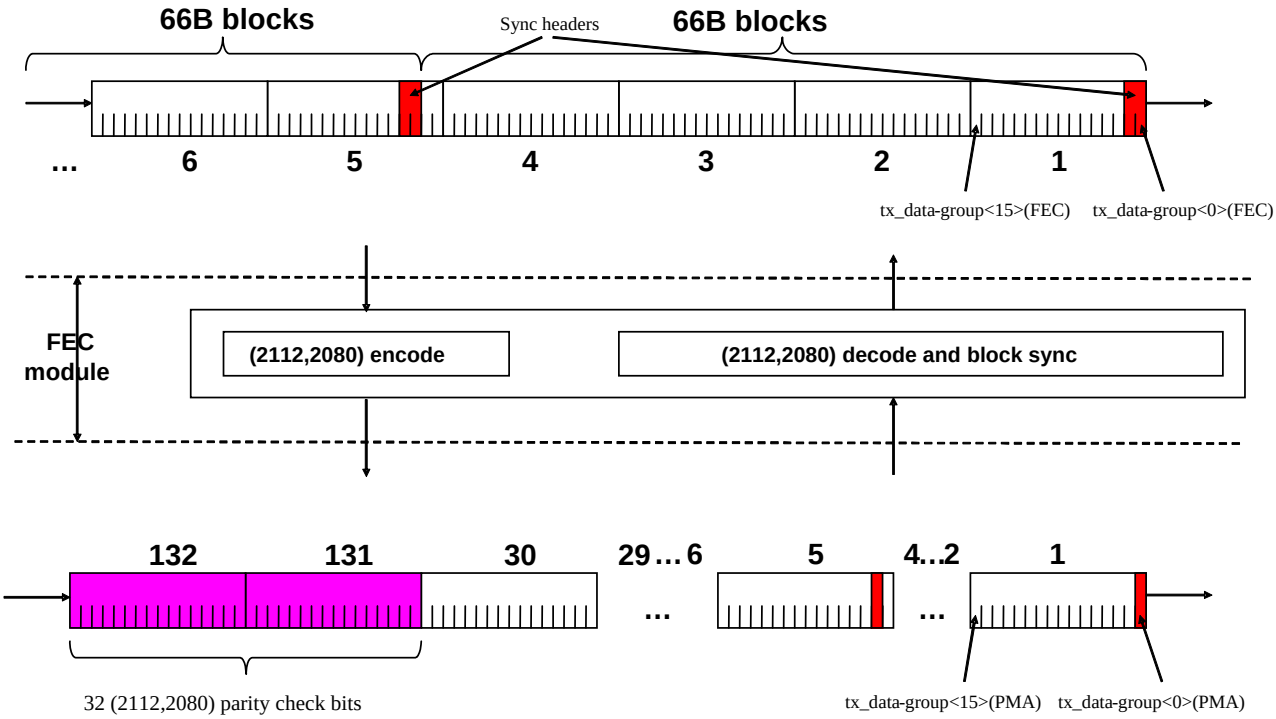

Figure XX–2—Transmission bit ordering

XX.2.1.4.4 FEC (2112,2080) Encoder

The block diagram of the FEC Encoder is illustrated in figure XX-3. The 32 x 65 bit payload blocks are encoded by the (2112,2080) code. This code is a shortened cyclic code that can be encoded by generator polynomial g(x). The resulting payload block including the T bits is scrambled using the PN-2112 pseudo-noise sequence as described in XX.2.1.4.4.1.

The generator polynomial g(x) for the (2112, 2080) parity-check bits is defined as given below.

$$g(x) = x^{32} + x^{23} + x^{21} + x^{11} + x^2 + 1,$$

then, if the polynomial representation of information bits is m(x), the codeword c(x) can be calculated in systematic form as

$$r(x) = x^{32} m(x) \mod g(x),$$
$$c(x) = r(x) + x^{32} m(x).$$

(Multiplication on x^{32} is performed using shifts). Systematic form of the codeword means that first 2080 bits of the codeword are information bits that can be extracted directly.

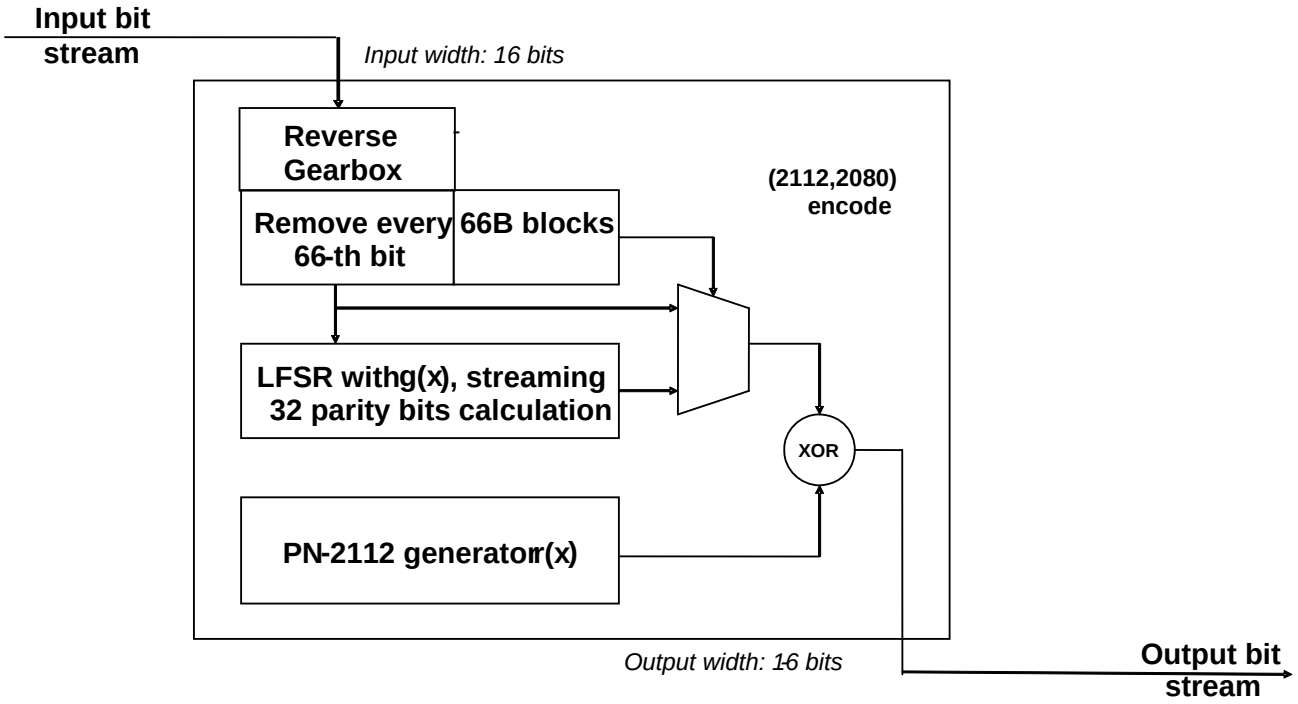

Figure XX–3—(2112,2080) encoding

XX.2.1.4.4.1 PN-2112 pseudo-noise sequence generator

PN-2112 is a pseudo-noise sequence of length 2112 generated by the polynomial r(x), which is equal to the scrambler polynomial defined in clause 49.2.6 with initial state $S_{57}=1$, $S_{i-1}=S_i$ XOR 1 or simply the binary sequence of 101010…. Before each FEC Block processing (encoding or decoding) the PN-2112 generator is initialized with this state of 101010….. Scrambling with the PN-2112 sequence at the FEC codeword boundary is necessary for establishing FEC block syncronization (to ensure that any shifted input bit sequence is not equal to another FEC codeword) and to ensure DC balance.

$$r(x) = 1 + x^{-39} + x^{-58} \qquad (49\text{-}1)$$

XX.2.1.4.5 FEC Decoder

The FEC decoder establishes FEC block synchronization based on repeated decoding of the received sequence. Decoding and error correction is performed after FEC synchronization is achieved. There is a configuration option for the FEC decoder to indicate any decoding errors to the upper layer.

The FEC decoder recovers and extracts the information bits using the parity-check data. In case of successful decoding the decoder restores the sync bits in each of the 64b/66b blocks sent to the PCS function. When the decoder is configured to indicate decoding error, the decoder indicates error to the PCS by means of setting both sync bits in each of the 64B/66B block to the same value (11), thus forcing the PCS sublayer to consider this block as invalid.

When the receive channel is in normal mode of operation the FEC Synchronization process continuously monitors PMA_SIGNAL.indicate(SIGNAL_OK). When SIGNAL_OK indicates OK the FEC Synchronization process accepts data-units via the PMA_UNITDATA.indicate primitive. It attains block sychronization based on the decoding of FEC blocks and conveys received 64b/66b blocks to the PCS Receive process. The FEC Synchronization process sets the sync_status flag to the PCS function to indicate whether the FEC has obtained synchronization.

XX.2.1.4.5.1 FEC (2112,2080) decoding

The FEC decoding function block diagram is shown in Figure XX-4. The decoder processes the 16-bit rx_data_group stream received from the PMA sublayer and descrambles the data using the PN-2112 pseudo-noise sequence as described in XX.2.1.4.4.1.

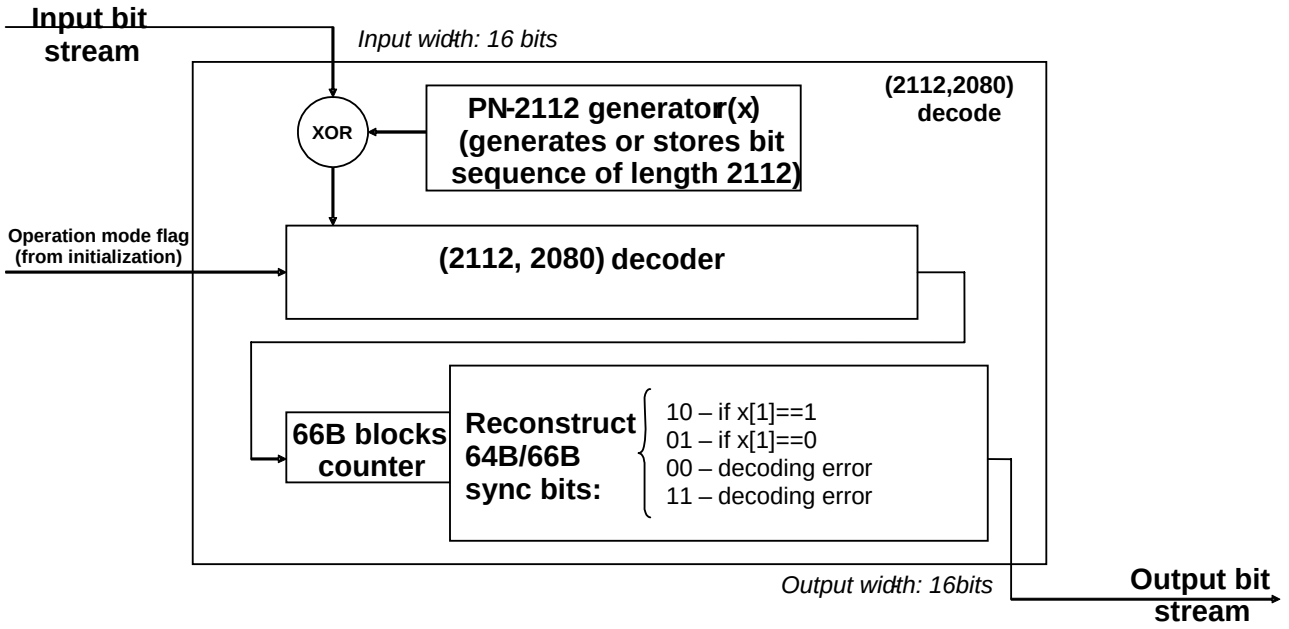

Figure XX-4— FEC (2112,2080) decoding

The synchronization of the 2112 bit FEC frame is established using FEC decoding as described in the flow diagram XX-7. Each of the 32 65-bit data words is extracted from the recovered FEC block and the 2 bit sync is reconstructed for the 64b/66b codes from the transcode bit as shown in Figures XX-5 and XX-6. The FEC decoder provides a configuration option that indicates decoding errors in the reconstructed sync bits:

- reconstruct sync bits to be one of the possible combinations: “01” or “10”, if decoding successful.
- reconstruct sync bits to additional codes “00” and “11”, if decoding error occurred.

This information corresponds to one complete (2112,2080) FEC frame that is equal to 32 64b/66b code blocks.

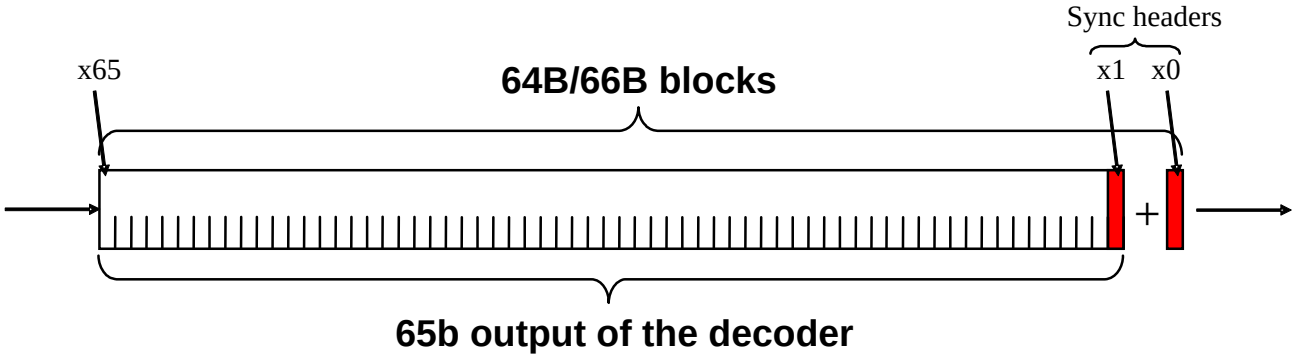

Figure XX-5—Reconstructing sync bits in 64b/66b blocks

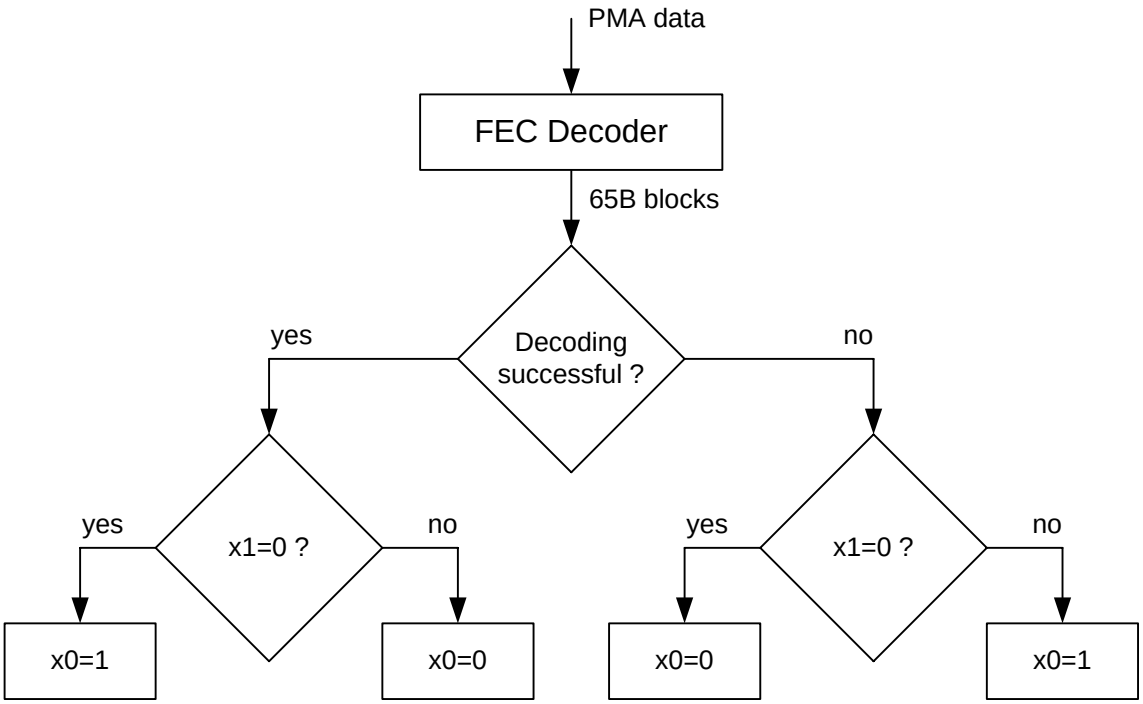

Figure XX-6—Defining the value of x0 bit

XX.2.1.4.6 FEC block synchronization

The receive synchronization of FEC blocks is illustrated by the flow diagram in figure XX-7.

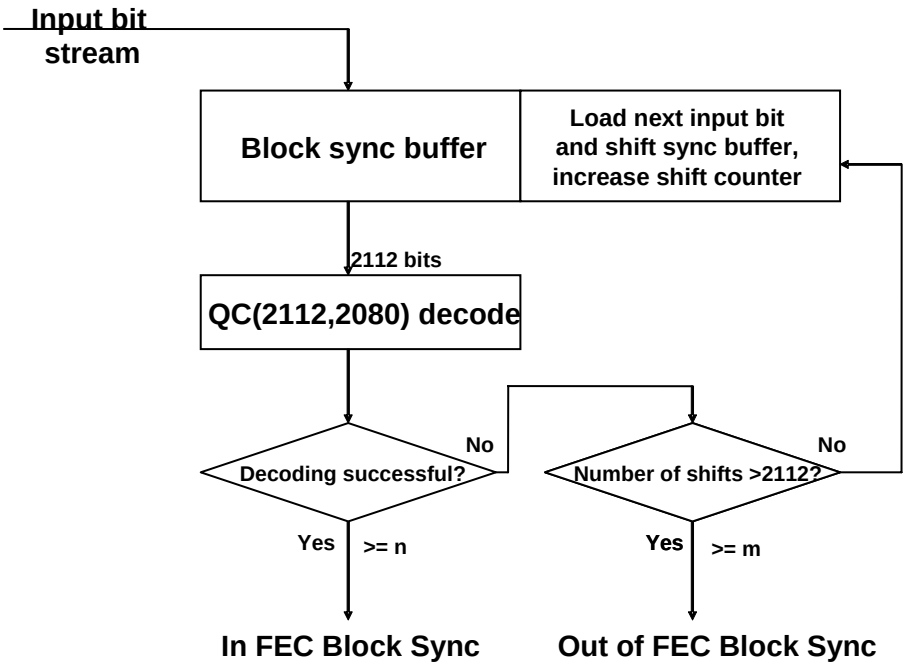

Figure XX-7 — FEC (2112,2080) block sync and decoding

Recive Frame synchronization is achived using conventional n/m serial locking techniques as described below:

- Test a candidate block start position
  - Evaluate parity for the potential block
    - If it fails shift candidate start by one bit position and try again
- Validate potential block start position has good parity for "n" consecutive blocks
  - If any of them fail shift candidate start one bit position and start again
- Block Sync is established.
- If "m" consecutive blocks are received with bad parity, drop Block Sync and restart again at 1.
- If "n" consecutive blocks are received with good parity, report Block Sync

The procedure is repeated at most 2111 times for all bits positons in the 2112 code word. The default value for variables m=8 and n=4.

XX.2.2 Test Pattern Generator

The test pattern generator shall be same as defined in clause 49.2.8

XX.2.3 Test Pattern Checker

The test pattern checker shall be same as defined in clause 49.2.12

XX.2.4 Enable FEC

The FEC sublayer shall have capability to enable or disable the FEC function. An MDIO interface or an equivalent management interface shall be provided to access the variable Enable_FEC (Refer to 45.2.1.75.4 register bit 1.50.3). When FEC Enable variable bit is set to a one, this enables the FEC for the 10GBASE-KR PHY. When the variable is set to zero, the FEC is disabled in the 10GBASE-KR PHY. This variable shall be set

to zero upon execution of PHY reset. When the FEC function is disabled the PHY shall have a mechanism to bypass the FEC Encode/Decode function so as not to cause additional latency associated with encoding or decoding functions.

XX.3 Error monitoring capability

The following counters apply to FEC sublayer management and error monitoring. If an MDIO interface is provided (see Clause 45), it is accessed via that interface. If not, it is recommended that an equivalent access be provided. These counters are reset to zero upon read or upon reset of the FEC sublayer. When a counter reaches all ones, it stops counting. The counters' purpose is to help monitor the quality of the link.

XX.3.1 FEC_corrected_blocks_counter

32-bit counter. FEC_corrected_blocks_counter counts once for each corrected FEC blocks processed.
This variable is provided by a management interface that may be mapped to the 45.2.1.81 register (1.157, 1.156).

XX.3.2 FEC_uncorrected_Blocks_counter

32-bit counter. FEC_uncorrected_blocks_counter counts once for each uncorrected FEC blocks processed. This variable is provided by a management interface that may be mapped to the 45.2.1.82 register (1.159, 1.158).

EDITOR's NOTE: Add following changes (FEC specific) to Auto-Neg subclauses in 73.

Insert the following subclause after 73.6.4 Technology Ability Field and renumber accordingly

73.6.6 FEC capability

FEC (F0) is encoded in bit D47 of the base Link Code Word. The default value is logic zero. When the FEC capability bit F0 is set to logic one, it indicates that the FEC function is enabled for 10GBASE-KR PHY.

Since the Local Device and the Link Partner may have enabled the FEC bits differently the priority resolution function is used to enable FEC in the respective PHYs. The FEC is enabled on the link only if both ends advertise the same status on the F0 bits.

EDITOR's NOTE: Add following changes (FEC specific) to subclauses in 45.

Insert the following rows to table 45-53 in subclause 45.2.1.75 and renumber reserved bits as shown

Bit 1.150.15:~~2~~ 5 Reserved Value always zero, writes ignored R/W
Bit 1.150.4 Enable FEC Error Indication A write of 1 to this bit configures FEC decoder to indicate Error to the upper layer R/W
Bit 1.150.3 Enable FEC A write of 1 to this bit enables FEC in the 10GBASE-KR PHY R/W
Bit 1.150.2 FEC Capable A read of 1 in this bit indicates that the 10GBASE-KR PHY is FEC capable RO

Insert the following subclauses after 45.2.1.75.2 and renumber accordingly

45.2.1.75.3 FEC Capable (1.150.2)

When read as a one, this bit indicates if the 10GBASE-KR PHY supports Forward Error Correction (FEC). When read as a zero, the 10GBASE-KR PHY does not support Forward Error Correction.

45.2.1.75.4 Enable FEC (1.150.3)

When written as a one, this bit enables the FEC for the 10GBASE-KR PHY. When written as a zero, FEC is disabled in the 10GBASE-KR PHY. This bit shall be set to zero upon execution of PHY reset.

45.2.1.75.4 Enable FEC Error Indication (1.150.4)

This bit enables the FEC decoder to indicate decoding errors to the upper layers (PCS) through the sync bits for the 10GBASE-KR PHY in the Local Device. When written as a one, this bit enables indication of decoding errors through the sync bits to the PCS layer. When written as zero the error indication function is disabled.

Insert the following subclauses after 45.2.1.80 and renumber accordingly

45.2.1.81 10GBASE-KR FEC corrected blocks counter (Register 1.157, 1.156)

The assignment of bits in the FEC corrected blocks counter register is shown in Table 45–59. See XX.3.1 for a definition of this register. These bits shall be reset to all zeroes when the register is read by the management function or upon PHY reset. These bits shall be held at all ones in the case of overflow.

Insert the following table after table 45-58 and renumber accordingly

Table 45–59—10GBASE-KR FEC corrected blocks counter register bit definitions

Editor's Note: Insert register table here

45.2.7.6 10GBASE-KR FEC uncorrected blocks counter (Register1.159, 1.158)

The assignment of bits in the FEC uncorrected blocks counter register is shown in Table 45-60. See XX.3.2 for a definition of this register. These bits shall be reset to all zeroes when the register is read by the management function or upon PHY reset. These bits shall be held at all ones in the case of overflow.

Insert the following table after table 45-59 and renumber accordingly

Table 45-50—10GBASE-KR FEC uncorrected blocks counter register bit definitions

Editor's Note: Insert register table here