Proposal of a Physical Coding Sublayer for GEPOF technical feasibility

Rubén Pérez-Aranda
rubenpda@kdpof.com

Agenda

- Objectives

- Physical Coding Sublayer proposal

  - General architecture

  - GMII data encapsulation

  - GMII data decapsulation

  - GMII encapsulation capacity

  - GMII encapsulation rate-matching

  - Binary and symbol scramblers

  - Physical Header Data description

  - Physical Header encoding

Motivation

- In [6] was presented a transmission structure proposal for GEPOF PHY able to fullfil with the special requirements of automotive applications like:

  - Clock frequency deviation: +/- 200 ppm (aging and temperature)

  - Max. wake time (from power off to Gigabit link ready): 100 ms

  - Clock frequency drift of 5ppm/°C

  - Fast temperature drift of ~4°C/s

  - Continuous tracking of non-static communication channel caused by vibrations and temperature

- An identified gap for technical feasibility is how to map the GMII Ethernet frames into the proposed transmission structure considering requirements as:

  - Low latency

  - Low and bounded jitter

  - Error handling with Low Mean Time to False Packet Acceptance (MTTFPA)

  - Rate-matching

Objectives

- The main objective of this presentation is to propose a Physical Coding Sublayer (PCS) able to fit with the requirements described in [1], [2] and [6] and compatible with the transmission structure proposed in [6], for the technical feasibility assessment

- The proposal provides a method for mapping the Ethernet frames entering the GMII interface into FEC code-words to be transmitted along the communication channel

- An encapsulation method of Ethernet frames that is asynchronous to FEC alignement is proposed, therefore providing a high freedom for FEC selection among all the proposals described in [5]
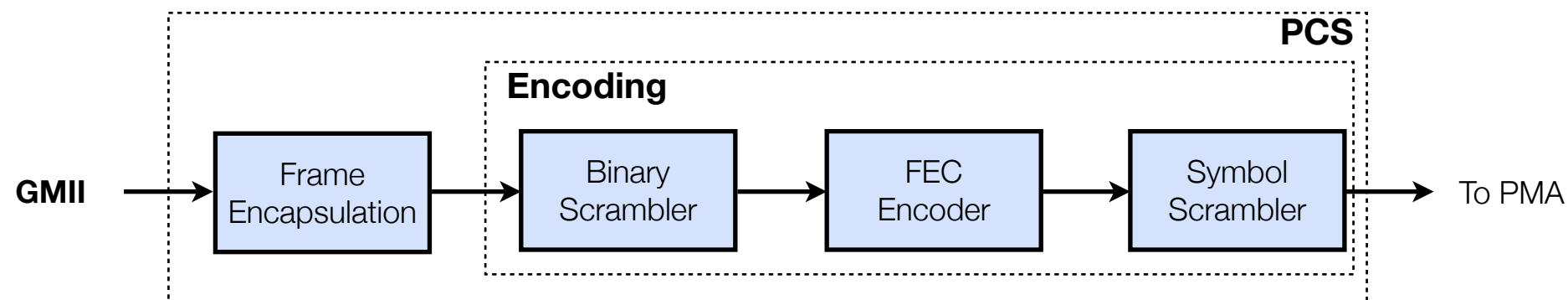
Physical Coding Sublayer proposal

General architecture

PCS: Physical Coding Sublayer

PMA: Physical Medium Attachment

PMD: Physical Medium Dependent

F_S: symbol rate
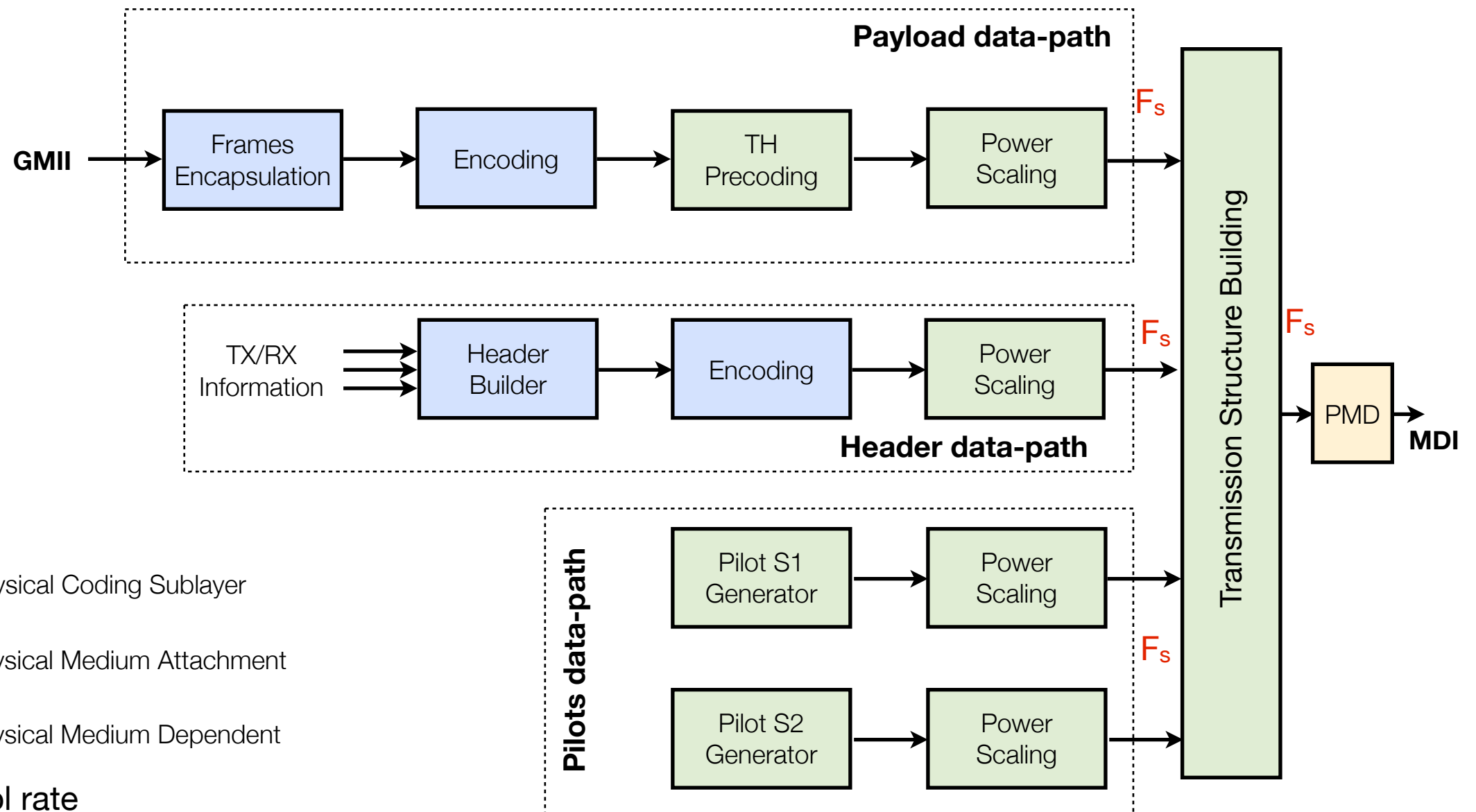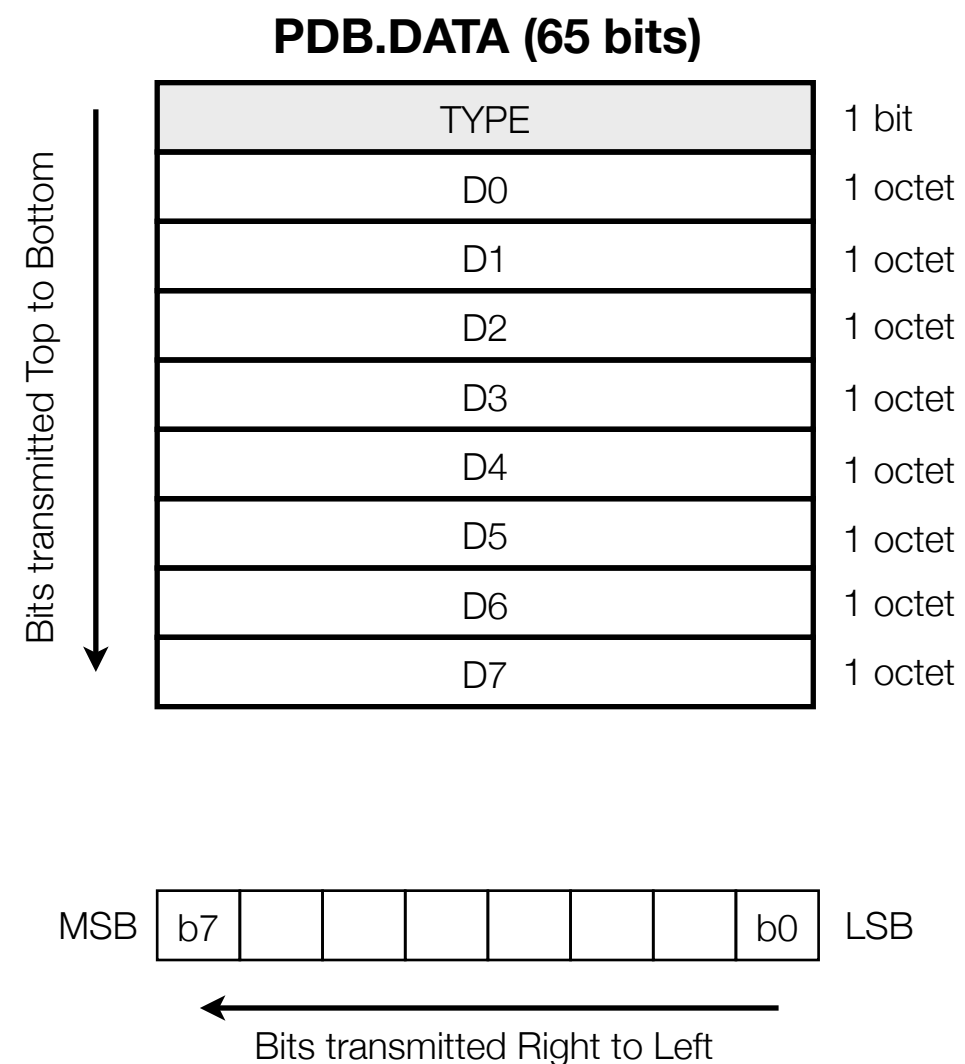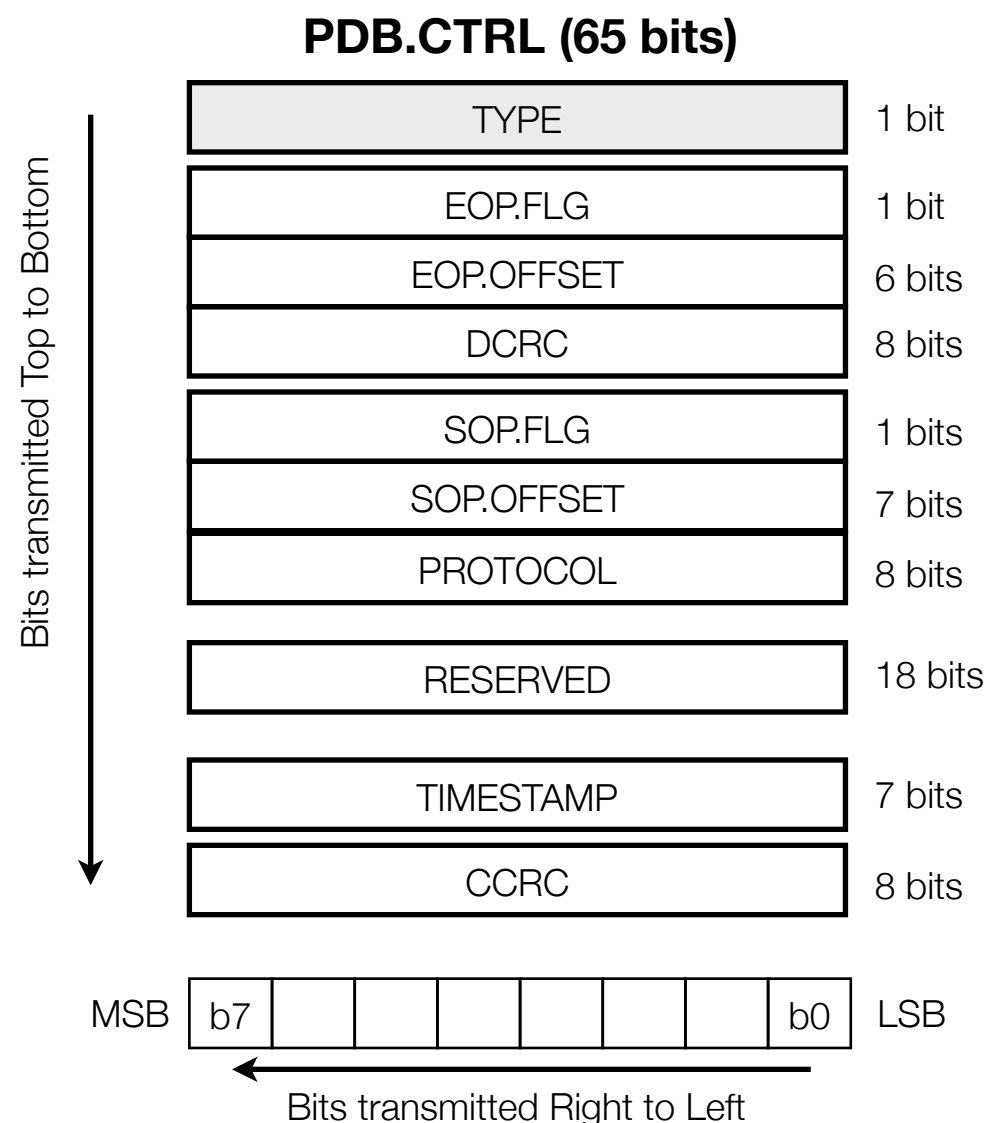
PCS - Payload data-path

- A frame encapsulation method is going to be defined to encapsulate the Ethernet frames into FEC codewords in an asynchronous way

- The binary stream from encapsulation is scrambled to get uncorrelated data entering the FEC independent of the correlation patterns of user information carried in Ethernet frames, therefore the power spectral density of the symbols stream can be considered white

- It is advantageous to use MLCC encoder for FEC as it is described in [5] to get the highest coding gain, hence highest link budget

- The encoded modulated M-PAM symbols outgoing the FEC encoder are scrambled to provide that every point belonging to M-PAM constellation is equally affected by the non-linearity of communication channel

PCS - Payload data-path: GMII Encapsulation

- Ethernet frames from GMII TX are segmented and encapsulated into blocks of 8 octets (64 bits)

- One control bit is prepended to each 64 bits unit, composing the basic transmission unit that we are going to call Physical Data Block (PDB), of 65 bits each one

- Type bit (prefix) is used to indicate if the PDB contains user data (PDB.DATA) from GMII or control information of the encapsulation protocol (PDB.CTRL)

PDB.CTRL (65 bits)

TYPE	1 bit
EOP.FLG	1 bit
EOP.OFFSET	6 bits
DCRC	8 bits
SOP.FLG	1 bits
SOP.OFFSET	7 bits
PROTOCOL	8 bits
RESERVED	18 bits
TIMESTAMP	7 bits
CCRC	8 bits

Bits transmitted Top to Bottom

MSB | b7 | | | | | | | b0 | LSB

Bits transmitted Right to Left

PDB.DATA (65 bits)

TYPE	1 bit
D0	1 octet
D1	1 octet
D2	1 octet
D3	1 octet
D4	1 octet
D5	1 octet
D6	1 octet
D7	1 octet

Bits transmitted Top to Bottom

MSB | b7 | | | | | | | b0 | LSB

Bits transmitted Right to Left

PCS - Payload data-path: GMII Encapsulation

- PDB.CTRL description

Symbol	Description	# of bits	Valid values
PDB.TYPE	Indicates the type of PDB (for control or user data)	1	1: The current PDB is PDB.CTRL
PDB.CTRL.EOP.FLG	Indicates packet finishes in the next PDB.DATA block. Offset where the packet ends is indicated by PDB.CTRL.EOP.OFFSET.	1	0: No end of packet in the next PDB.DATA 1: End of packet in the next PDB.DATA
PDB.CTRL.EOP.OFFSET	When PDB.CTRL.EOP.FLG is 1, this field indicates offset in # of bits to the last bit of the packet in the next PDB.DATA. PDB.TYPE is not counted.	6	0x00 .. 0x3F
PDB.CTRL.DCRC	When PDB.CTRL.EOP.FLG is 1, this field indicates the CRC8 of data (contained in PDB.DATA units) corresponding to the packet referred by PDB.CTRL.EOP.FLG and PDB.CTRL.EOP.OFFSET.	8	0x00 .. 0xFF
PDB.CTRL.SOP.FLG	Indicates a packet starts in the next PDB.DATA block. Offset where the packet starts is indicated by PDB.CTRL.SOP.OFFSET.	1	0: No start of packet in the next PDB.DATA 1: Start of packet in the next PDB.DATA
PDB.CTRL.SOP.OFFSET	When PDB.CTRL.SOP.FLG is 1, this field indicates offset in # of bits to the first bit of the packet in the next PDB.DATA. PDB.TYPE is not counted. The value 0x40 indicates a packet starts two PDB.DATA packets after the current PDB.CTRL. This allows for a more efficient back-to-back packet encapsulation, when PDB.CTRL.EOP.OFFSET of the previous packet takes value 0x3F.	7	0x00 .. 0x40

PCS - Payload data-path: GMII Encapsulation

- PDB.CTRL description

Symbol	Description	# of bits	Valid values
PDB.CTRL.PROTOCOL	When PDB.CTRL.SOP.FLG is 1, this field indicates the encapsulated protocol identifier of the packet referred by PDB.CTRL.SOP.FLG and PDB.CTRL.SOP.OFFSET.	8	0x0,0x02 .. 0xFF (reserved for future extensions) 0x01 (Ethernet)
	When both PDB.CTRL.SOP.FLG and PDB.CTRL.EOP.FLG are 0 this field identifies the type of special PDB.CTRL packet.	8	0x00 (reserved for IDLE PDBs) 0xFF (reserved for PAD PDBs) 0x01..0xFE (reserved for future extensions).
		18	0: (reserved)
PDB.CTRL.TIMESTAMP	When PDB.CTRL.SOP.FLG is 1: Time-stamp obtained from a free running counter at F_s frequency that indicates when the ethernet frame arrives to data encapsulator, used to minimize the end-to-end (GMII TX to GMII RX) jitter	7	0 .. 127 Only valid if PDB.CTRL.SOP.FLG = 1
CCRC	CRC8 of the current PDB.CTRL, including all bits (from PDB.TYPE to PDB.CTRL.TIMESTAMP) except the CCRC itself	8	0x00 .. 0xFF
Total (bits)		**65**	

PCS - Payload data-path: GMII Encapsulation

- PDB.IDLE description

- Special case of PDB.CTRL

- PDB.IDLE must be inserted if no data are available from GMII TX

Bits transmitted Top to Bottom

Symbol	Description	# of bits	Valid values
PDB.TYPE	Indicates the type of PDB (for control or user data)	1	1: The current PDB is PDB.CTRL
PDB.CTRL.EOP.FLG		1	0: No end of packet in next PDB.DATA
PDB.CTRL.EOP.OFFSET		6	0x00
PDB.CTRL.DCRC		8	0x00
PDB.CTRL.SOP.FLG		1	0: No start of packet in next PDB.DATA
PDB.CTRL.SOP.OFFSET		7	0x00
PDB.CTRL.PROTOCOL		8	**0x00**
		18	0x00
PDB.CTRL.TIMESTAMP		7	0x00
CCRC	CRC8 of the current PDB.CTRL, including all bits (also PDB.TYPE) except the CCRC itself	8	0x87
Total (bits)		**65**	

PCS - Payload data-path: GMII Encapsulation

- PDB.PAD description

- Special case of PDB.CTRL for padding between PDB.DATA blocks

- This is necessary to be inserted by the encapsulator for rate-matching only in case of FEC Data-rate > GMII rate, between two PDB.DATA blocks of a packet, allowing minimum buffer

Bits transmitted Top to Bottom

Symbol	Description	# of bits	Valid values
PDB.TYPE	Indicates the type of PDB (for control or user data)	1	1: The current PDB is PDB.CTRL
PDB.CTRL.EOP.FLG		1	0: No end of packet in next PDB.DATA
PDB.CTRL.EOP.OFFSET		6	0x00
PDB.CTRL.DCRC		8	0x00
PDB.CTRL.SOP.FLG		1	0: No start of packet in next PDB.DATA
PDB.CTRL.SOP.OFFSET		7	0x00
PDB.CTRL.PROTOCOL		8	**0xFF**
		18	0x00
PDB.CTRL.TIMESTAMP		7	0x00
CCRC	CRC8 of the current PDB.CTRL, including all bits (also PDB.TYPE) except the CCRC itself	8	0x90
Total (bits)		**65**	

PCS - Payload data-path: GMII Encapsulation

- PDB.DATA description

- A packet may start and/or end in any bit of PDB.DATA.Dx

- Length of packets does not require to be multiple of 8 bits; packets are bit aligned

- In case of the packets are not transmitted back to back, padding bits may be added to complete the PDB.DATA.

Symbol	Description	# of bits	Valid values
PDB.TYPE	Indicates the type of PDB (for control or user data)	1	0: The current PDB is PDB.DATA
PDB.DATA.D0	First data octet of PDB.DATA	8	0x00 .. 0xFF
PDB.DATA.D1	Second data octet of PDB.DATA	8	0x00 .. 0xFF
PDB.DATA.D2	Third data octet of PDB.DATA	8	0x00 .. 0xFF
PDB.DATA.D3	Fourth data octet of PDB.DATA	8	0x00 .. 0xFF
PDB.DATA.D4	Fifth data octet of PDB.DATA	8	0x00 .. 0xFF
PDB.DATA.D5	Sixth data octet of PDB.DATA	8	0x00 .. 0xFF
PDB.DATA.D6	Seventh data octet of PDB.DATA	8	0x00 .. 0xFF
PDB.DATA.D7	Eighth data octet of PDB.DATA	8	0x00 .. 0xFF
Total (bits)		**65**	

Bits transmitted Top to Bottom

PCS - Payload data-path: GMII Encapsulation

Encapsulation of Ethernet frames from GMII
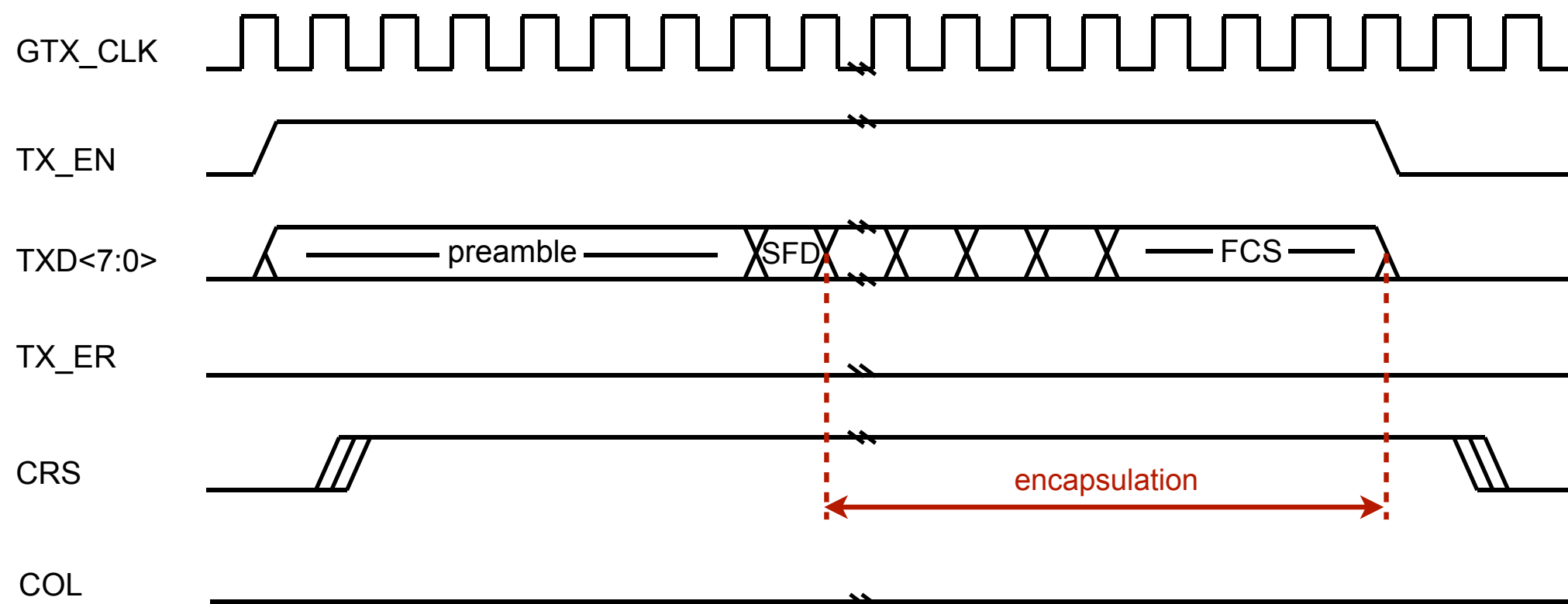
PCS - Payload data-path: GMII Encapsulation

- PHD.TX.NEXT.PDB.OFFSET

    - In general, the first PDB of a transmit block is not aligned with the beginning of the Payload section (first FEC code-word).

    - For fast synchronization and checking, the offset of the first PDB in Payload of the next transmit block is indicated by the Physical Header Data (PHD)

    - **PHD.TX.NEXT.PDB.OFFSET[k+1] = (PHD.TX.NEXT.PDB.OFFSET[k] + Block Length) mod 65** (see PHD section)

    - Block Length (information bits)= $28 \cdot 4 \cdot (1664 + 1994 + 3024)$ - calculated for MLCC proposed in [5]
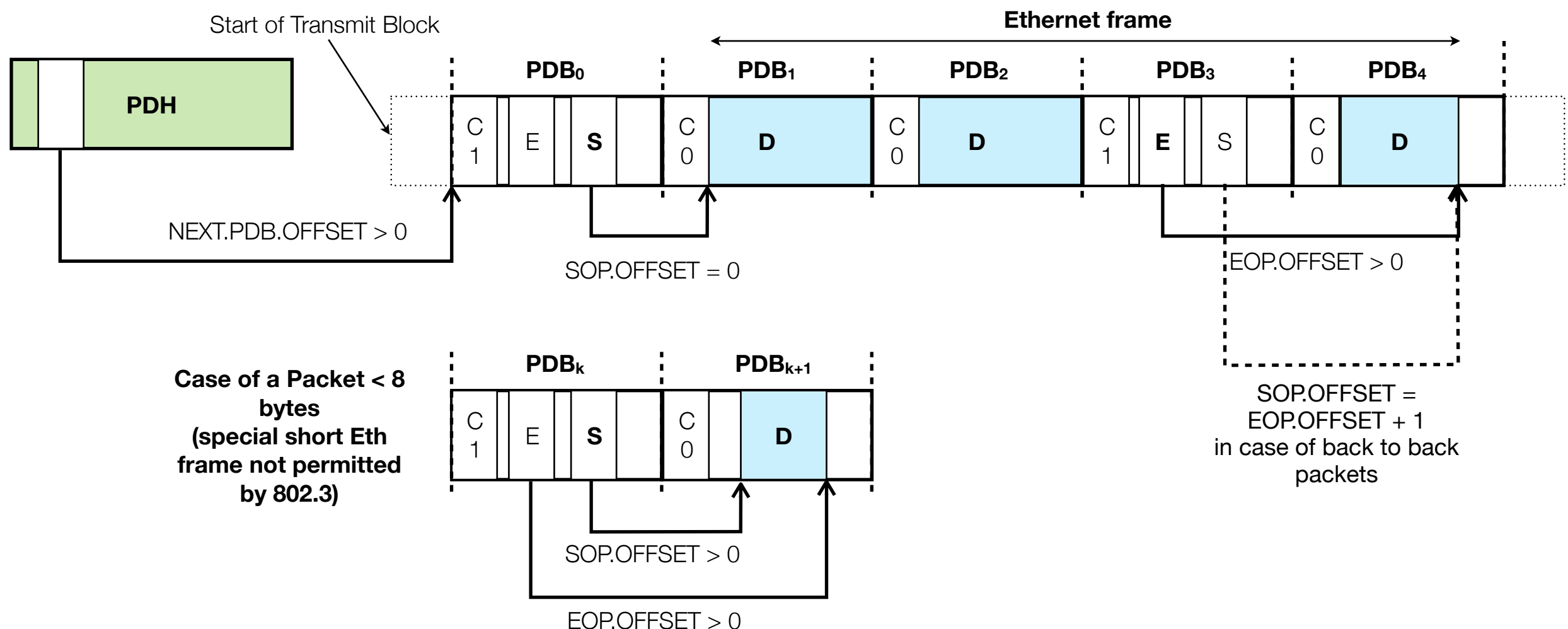
PCS - Payload data-path: GMII Encapsulation

- Only Full-Duplex of Clause 35 (IEEE 802.3) is supported (green fields)

**Table 35–1—Permissible encodings of TXD<7:0>, TX_EN, and TX_ER**

TX_EN	TX_ER	TXD<7:0>	Description	PLS_DATA.request parameter
0	0	00 through FF	Normal inter-frame	TRANSMIT_COMPLETE
0	1	00 through 0E	Reserved	—
0	1	0F	Carrier Extend	EXTEND (eight bits)
0	1	10 through 1E	Reserved	—
0	1	1F	Carrier Extend Error	EXTEND_ERROR (eight bits)
0	1	20 through FF	Reserved	—
1	0	00 through FF	Normal data transmission	ZERO, ONE (eight bits)
1	1	00 through FF	Transmit error propagation	No applicable parameter
NOTE—Values in TXD<7:0> column are in hexadecimal.				

By deliberately DCRC corruption when data is encapsulated

GTX_CLK

TX_EN

PCS - Payload data-path: GMII Encapsulation

Figure 35–3—Basic frame transmission

GTX_CLK

TX_EN

TXD<7:0>   preamble   FCS
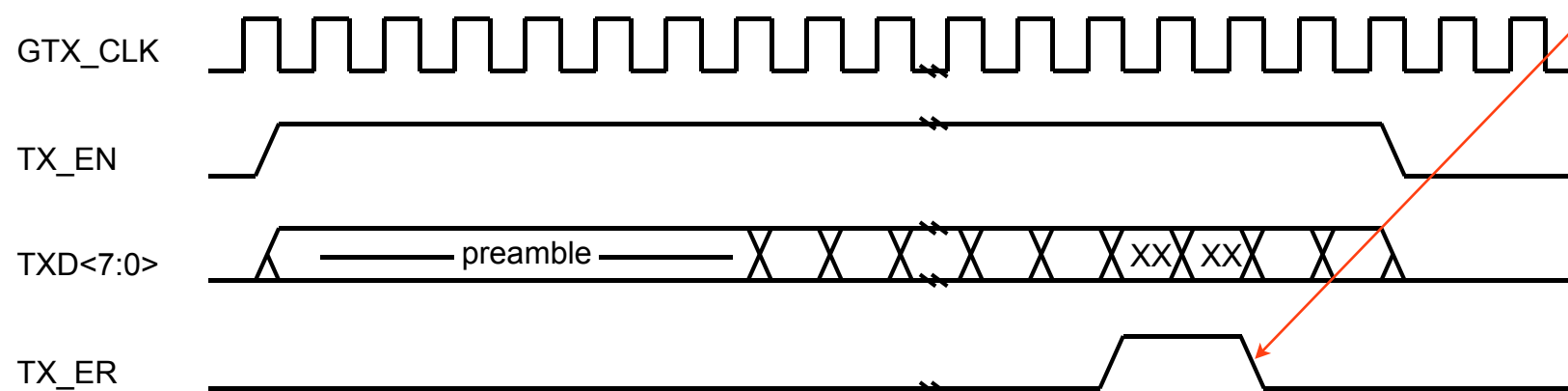
TX_ER
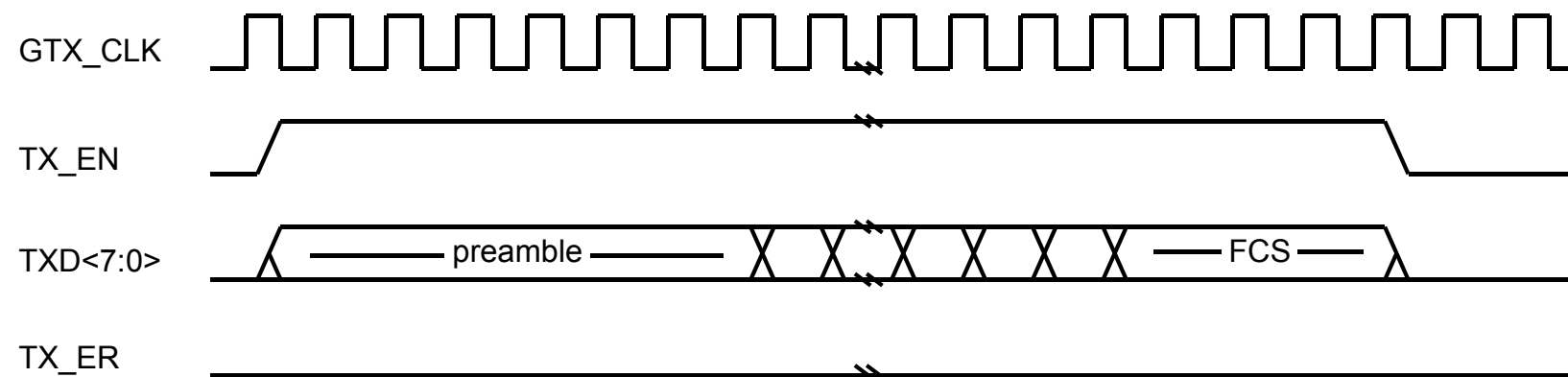
By deliberately data or DCRC corruption when data is encapsulated

Figure 35–4—Propagating an error within a frame

GTX_CLK

TX_EN

TXD<7:0>   preamble   XX XX

TX_ER

PCS - Payload data-path: GMII Decapsulation

- Special error management in decapsulation:

  1. PDB.DATA is detected as PDB.CTRL

     1. CCRC shall be erroneous

     2. The current packet is discarded (GMII RX_ER)

     3. The RX waits for a new (PDB.CTRL != PDB.IDLE) & (PDB.CTRL != PDB.PAD) announcing end of packet and possible new data packet

  2. PDB.CTRL is detected as PDB.DATA

     1. Continue PDBs reception until the next PDB.CTRL != PDB.PAD

     2. If (PDB.CTRL == PDB.IDLE) or CCRC is erroneous

        1. The current packet is discarded

        2. The Rx waits for a new (PDB.CTRL != PDB.IDLE) & (PDB.CTRL != PDB.PAD) announcing a new data packet.

     3. If (PDB.CTRL != PDB.IDLE) and CCRC is correct

        1. DCRC shall be erroneous, since a PDB.CTRL was detected as data and included in DCRC checking, therefore it is signalled as error in GMII

        2. The Rx proceeds with the new packet signaled in the PDB.CTRL or waits for a new (PDB.CTRL != PDB.IDLE) & (PDB.CTRL != PDB.PAD) announcing a new data packet.

PCS - Payload data-path: GMII Decapsulation

- Only Full-Duplex of Clause 35 (IEEE 802.3) is supported (green fields)

**Table 35–2—Permissible encoding of RXD<7:0>, RX_ER, and RX_DV**

RX_DV	RX_ER	RXD<7:0>	Description	PLS_DATA.indication parameter
0	0	00 through FF	Normal inter-frame	No applicable parameter
0	1	00	Normal inter-frame	No applicable parameter
0	1	01 through 0D	Reserved	—
0	1	0E	False Carrier indication	No applicable parameter
0	1	0F	Carrier Extend	EXTEND (eight bits)
0	1	10 through 1E	Reserved	—
0	1	1F	Carrier Extend Error	ZERO, ONE (eight bits)
0	1	20 through FF	Reserved	—
1	0	00 through FF	Normal data reception	ZERO, ONE (eight bits)
1	1	00 through FF	Data reception error	ZERO, ONE (eight bits)
NOTE—Values in RXD<7:0> column are in hexadecimal.				

Asserted by Phy when DCRC is not valid or packet is discarded by de-encapsulator
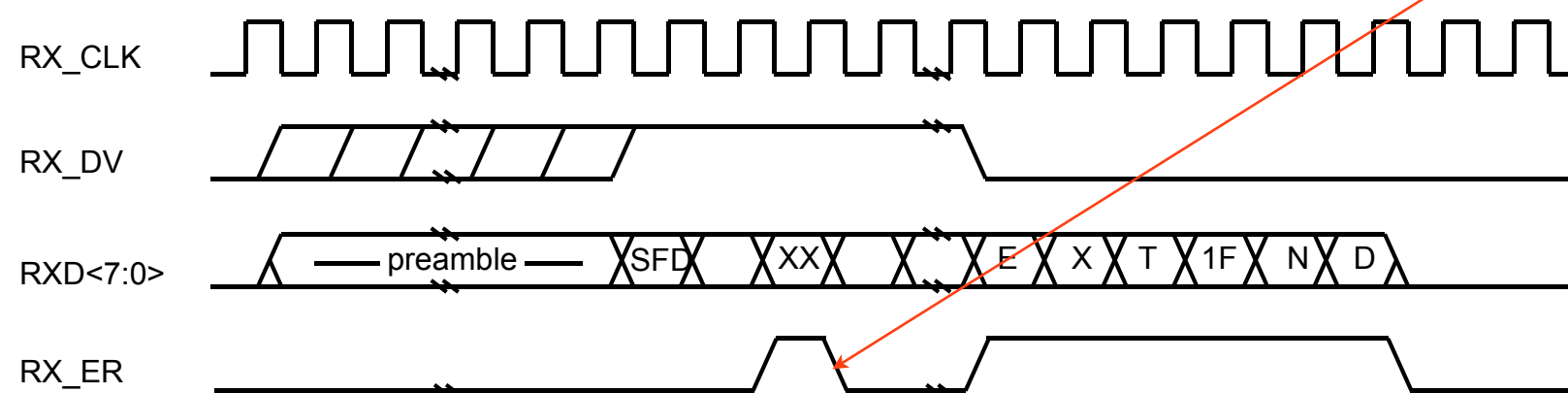
PCS - Payload data-path: GMII Decapsulation

Figure 35–8—Basic frame reception

RX_CLK

RX_DV

RXD<7:0>   preamble   SFD   FCS

RX_ER
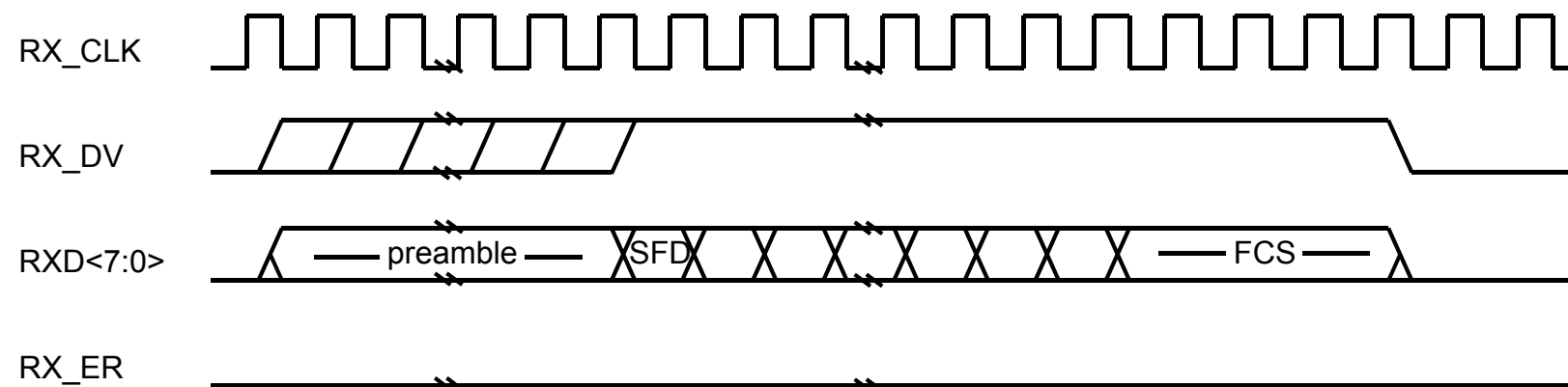
Asserted by Phy when DCRC is not valid or packet is discarded by de-encapsulator

Figure 35–11—Two examples of reception with error

RX_CLK

RX_DV

RXD<7:0>   preamble   SFD   XX   E X T 1F N D

RX_ER

PCS - Payload data-path: capacity

- For each Ethernet frame received from GMII TX:

  - 7 octets of Preamble + 1 octet of SFD,

  - InterPacketGap = 12 octets, with possible shrinkage in practical implementations of Ethernet bridges to accomodate clock deviations

- Case of back-to-back frames with no IPG nor Preamble shrinkage:

  - 2 PDB.CTRL per frame are used to indicate start and stop of frame independently

  - 2 PDB.CTRL = 16 octets, which is less than 20 octets of Preamble + IPG

- Case of back-to-back frames with IPG or Preamble shrinkage:

  - The encapsulation system supports to indicate in a single PDB.CTRL the stop of previous frame and start of next frame, without padding between them in PDB.DATA

  - 1 PDB.CTRL per frames, so 8 octets is always to be less than Preamble + IPG

- Therefore, data encapsulation is able to transport all the required bit-rate from GMII

- Required output bit-rate of data-encapsulation / input FEC bit-rate:

  - $65/64 \cdot 1000 = 1015.625$ Mbps

PCS - Payload data-path: rate-matching

1.PHY rate > GMII rate:

- This case is only produced if FEC provides an average bit-rate higher than $65/64 \cdot 1000$ Mbps, due to design limitations in, e.g. binary codes selection

- Solved by PDB.PAD blocks insertion between PDB.DATA blocks

- Extra latency (buffer) is required in this case to avoid FIFO underflow in transmitting decapsulated data to GMII RX

- Buffer will finally depend on the largest supported jumbo frame

2.PHY rate == G/MII rate (preferred one):

- Minimum buffers at TX and RX for encapsulation/decapsulation

- This case is only possible if FEC rate is designed accordingly to data encapsulation to provide just the same average bit-rate

- That is the case of MLCC encoder proposed in [5] combined with transmission structure proposed in [6]. See last slide of [6]

PCS - Payload data-path: rate-matching

- Several jitter sources are identified, which have to be resolved:

  - As it was described in [6], the transmission structure may be composed by different parts (pilots, header, and payload), being the structure periodic in time

  - When a packet is received from the GMII TX and it is encapsulated, it could be delayed due to the transmission of pilot or headers parts, introducing a deterministic jitter by the nature of the transmission structure

  - On the other hand, the regular structure of data encapsulator is not a priori aligned to the Ethernet frames received from GMII TX, therefore there exist an additional jitter due to the random latency between SFD reception and the time the beginning of frame is encapsulated and ready to be transmitted to FEC

- The proposed PCS is able to solve jitter by means of latency control:

  - Flow control implemented in FEC encoder, in such a way the bit-rate demanded in the encoder input is constant and equal to the average one
    - According to [6], target constant bit-rate should be: $((1664+1994)/2016+1.5) \cdot 312.5 \cdot 8064/(8064 + 160) = 1015.625$ Mbps

  - Residual jitter caused by rate wander due to imperfect flow control at FEC encoder input and jitter from encapsulation is solved by the use of PDB.CTRL.TIMESTAMP field in decapsulation (PCS receive function)
    - Receiver also implements a free running counter with the recovered symbol frequency, therefore it knows, by comparison, how much an Ethernet frame has to be delayed before transmit it to GMII RX for constant latency

- Estimated achievable jitter with latency control: ≤ 32ns (4 GMII clock cycles)

PCS - Payload data-path: DCRC

All the adders are mod-2 (XOR)
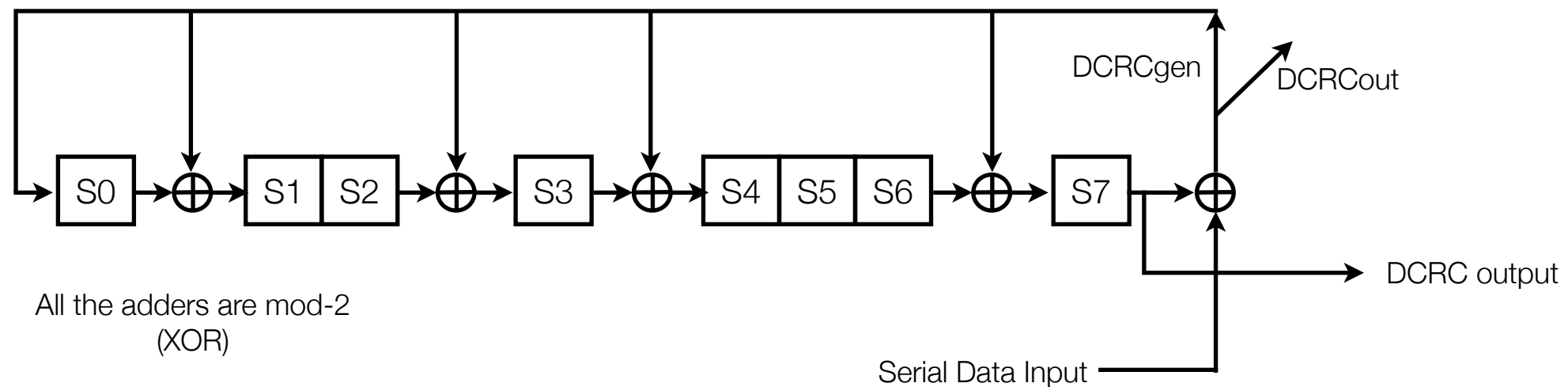
- Generator Polynomial: $1 + x + x^3 + x^4 + x^7 + x^8$

- The 8 delay elements S0, ... S7, shall be initialized to 0.

- All bits of data packet are used to compute the DCRC with switch connected (DCRCgen setting).

- After all the bits have been serially processed, the switch is disconnected (DCRCout setting) and the 8 stored values (S0...S7) are the DCRC.

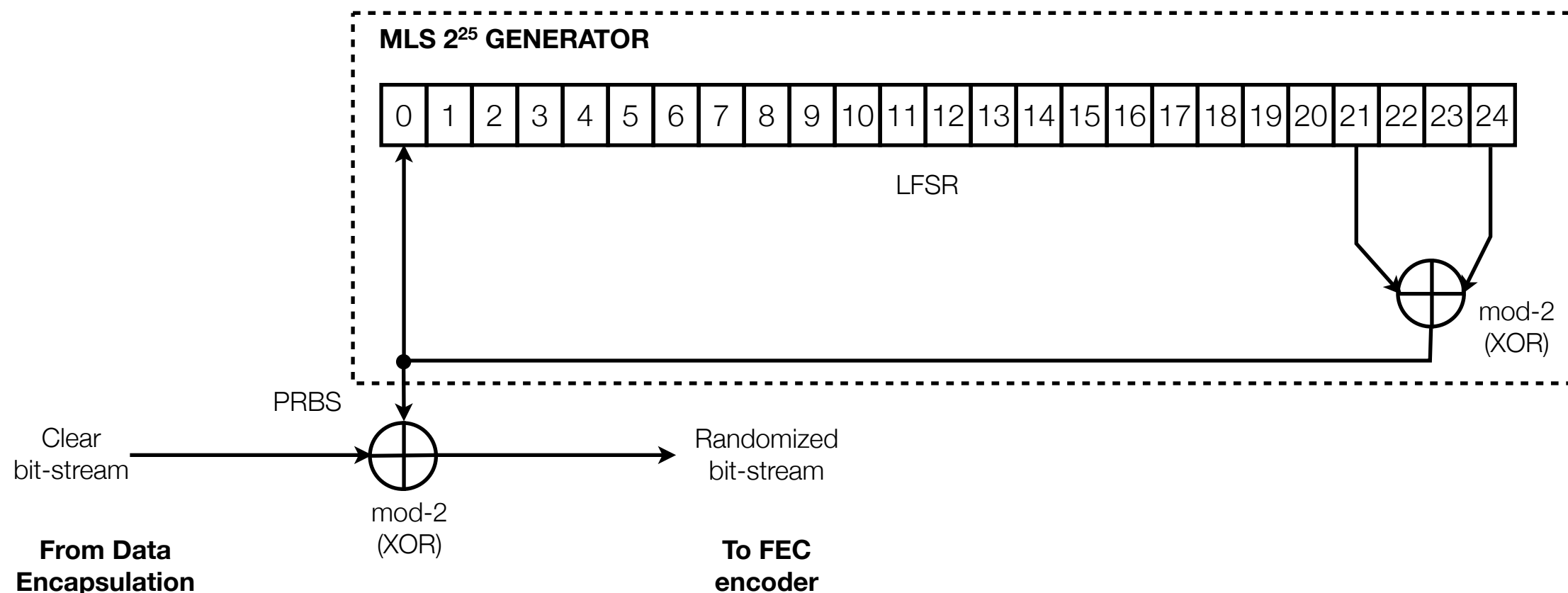- DCRC is transmitted in order from S7 to S0.
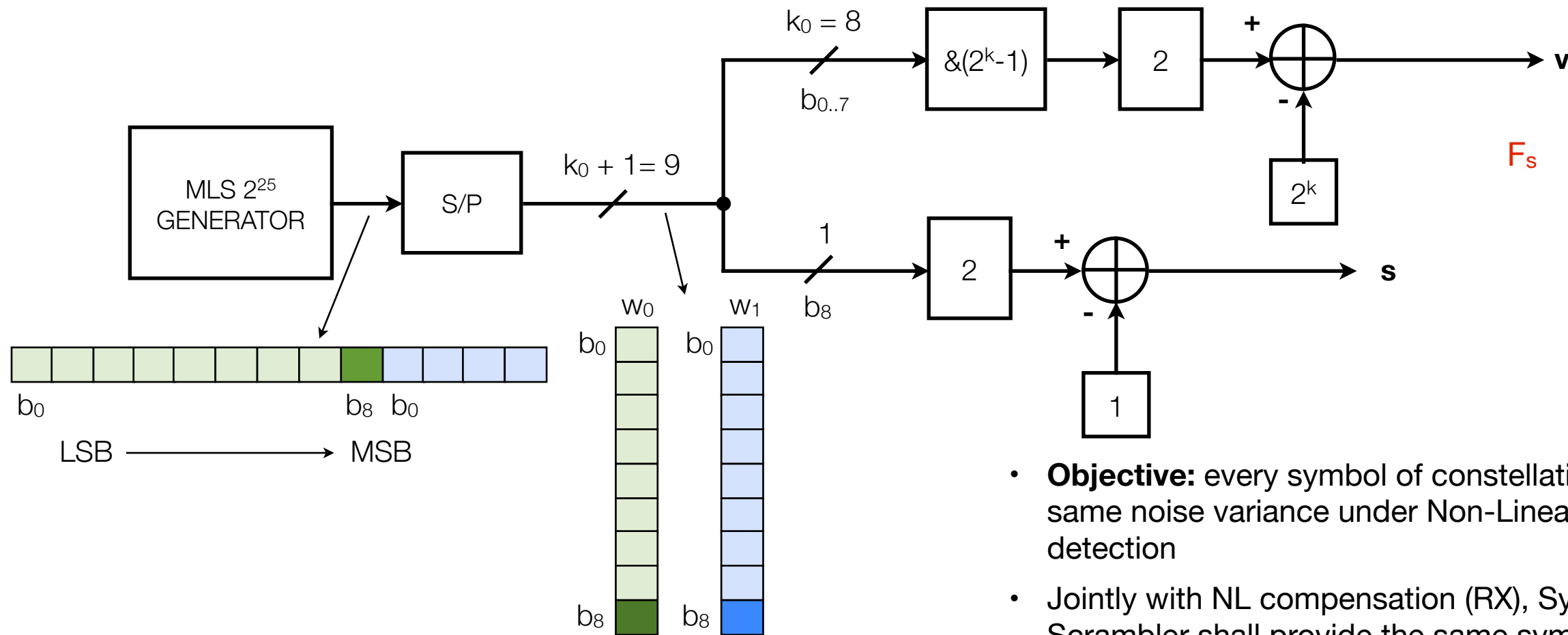
PCS - Payload data-path: CCRC
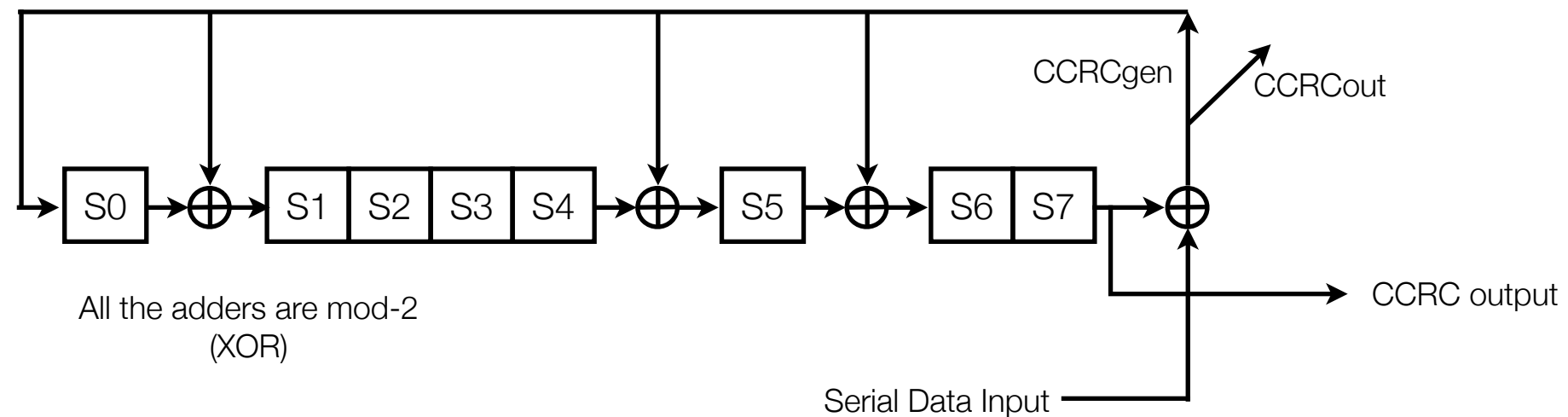
- Generator Polynomial: $1 + x + x^5 + x^6 + x^8$

- The 8 delay elements S0, ... S7, shall be initialized to 0.

- All bits of PDB.CTRL, from PDB.TYPE to PDB.CTRL.TIMESTAMP, except PDB.CTRL.CCRC itself (= 57 bits) are used to compute the CCRC with switch connected (CCRCgen setting).

- After all the bits have been serially processed, the switch is disconnected (CCRCout setting) and the 8 stored values (S0...S7) are the CCRC.

- CCRC is transmitted in order from S7 to S0.

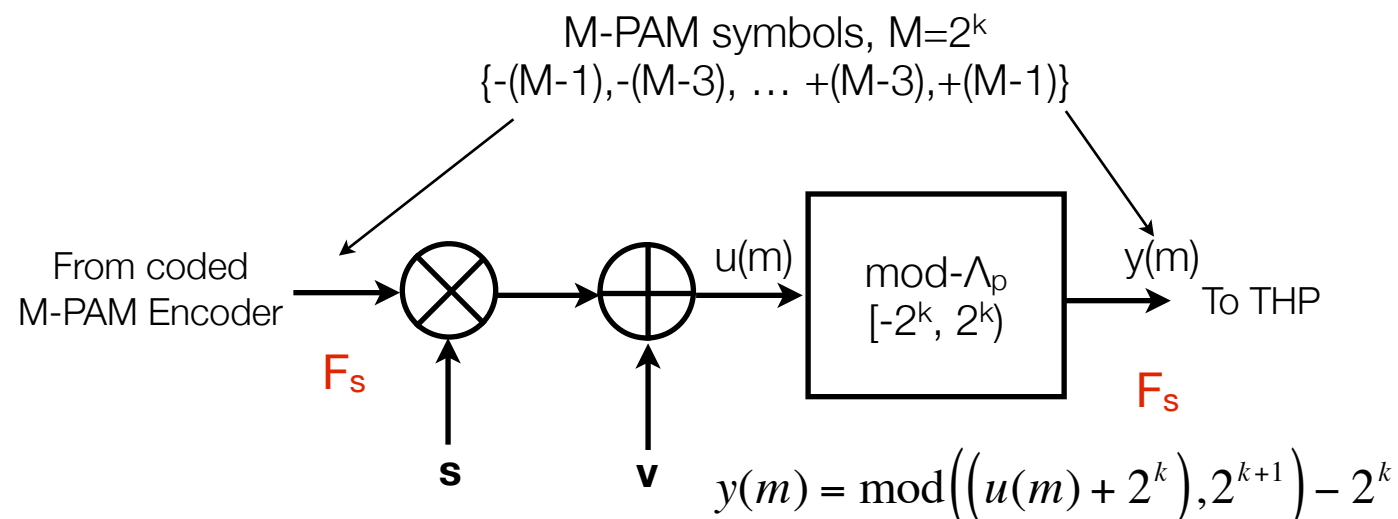# PCS - Payload data-path: binary scrambler

- Binary Scrambler:

  - LFSR polynomial: $1 + x^{22} + x^{25}$ (MLS generator)

  - Scrambler is initialized to a known state at the beginning of transmit block, therefore no LFSR synchronization is needed ➤ fast link startup

  - Defined reset state

$k_0 = 8$

$b_{0..7}$

$\&(2^k-1)$

2

$+$ / $-$

v

F_s

2^k

MLS 2^{25} GENERATOR

S/P

$k_0 + 1 = 9$

$k_0 = 8$

1

b_8

2

$+$ / $-$

s

1

w_0 w_1

b_0 b_0

b_8 b_8

b_0

b_8 b_0

LSB ⟶ MSB

M-PAM symbols, $M = 2^k$
$\{-(M-1), -(M-3), \ldots +(M-3), +(M-1)\}$

From coded M-PAM Encoder

$u(m)$

mod-Λ_p
$[-2^k, 2^k)$

$y(m)$

To THP

F_s

F_s

S

V

$$y(m) = \mathrm{mod}\left(\left(u(m) + 2^k\right), 2^{k+1}\right) - 2^k$$

- **Objective:** every symbol of constellation sees the same noise variance under Non-Linear distortion in detection
- Jointly with NL compensation (RX), Symbol Scrambler shall provide the same symbol error probability for all the constellation points
- LFSR polynomial: $\boldsymbol{1 + x^{22} + x^{25}}$ (MLS generator)
- Scrambler is initialized to a known state at the beginning of transmit block ➤ fast link startup
- Defined reset state
- Modulo operation reduces the scrambled symbols to the same Voronoi region of THP
- Symbol Scrambler is configured according to MLCC encoder: same ($M = 2^k$) PAM
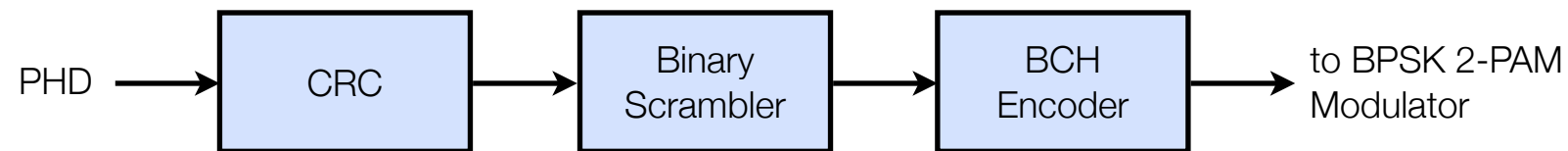
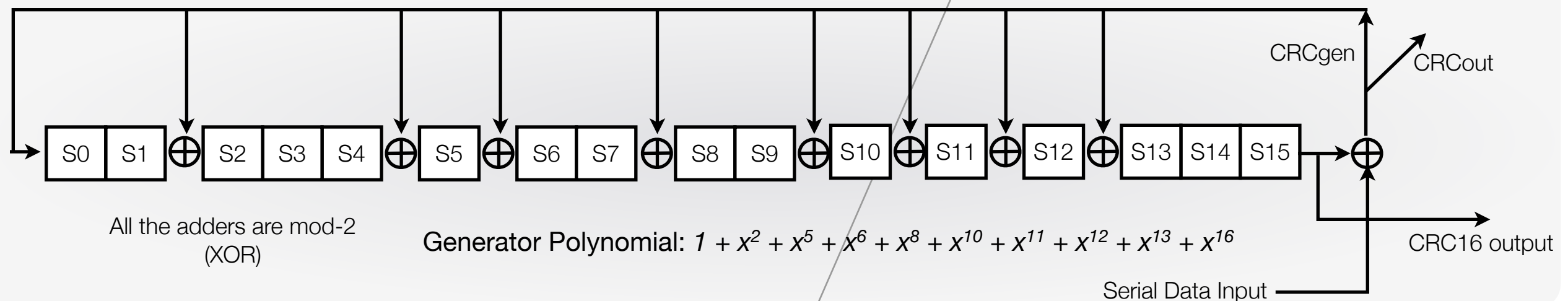| Symbol	Description	# of bits	Valid values
PHD.TX.BLOCKID	Current TX transmit block counter	8	0 .. 255
PHD.TX.CODING.ID	FEC identifier To support FEC configurations for coding-gain vs. latency tradeoff	3	0x0: MLCC 2016 symbols/CW as per [5] 0x1..0x7: Other configurations
PHD.TX.NEXT.CODING.SE	Next transmit block MLCC spectral efficiency configuration (in number of coded bits per dimension)	4	0 .. 5 (Reserved) 6 (3.5) (as per MLCC scheme in [5]) 7 .. 15: (Reserved)
PHD.TX.NEXT.THP.SETID	THP coefficients set Id that will be used in the next transmit block	2	0: the next block is not TH precoded 1 .. 3
PHD.TX.NEXT.PDB.OFFSET	Offset of the first PDB in Payload of the next transmit block	7	0x00 .. 0x40
PHD.RX.LASTBLOCKID	Last block counter received in return channel before current TX frame	8	0 .. 255
	Reserved	4	0 1 .. 15: Reserved
PHD.RX.REQ.THP.SETID	Requested THP coefficients set Id	2	0: no request for changing the THP coefficients is performed 1 .. 3
PHD.RX.REQ.THP.COEF[0..8]	Requested THP coefficients set when PHD.RX.REQ.THP.SETID is not equal to 0. 9 b(k) coefficients of 12 bits	108	Each b(k) is formatted (12, 2) Ordered from b(0) to b(8)
PHD.RX.STATUS	Indicates that local PMA receive function is able to make the reception of PAM symbols with reliability. This corresponds with the content of variable loc_rcvr_status. The PCS receive function shall use this PHD field to determine the rem_rcvr_status.	1	0: NOT_OK 1: OK

PCS - Header data-path: PHD description [6]

Symbol	Description	# of bits	Valid values
PHD.CAP.LPI	Signals the capability of the PHY to support the reception of Low Power Idles during the payload sub-blocks	3	0: LPI is not supported 1: LPI is supported 2 .. 7: reserved
	Reserved	2	0 1 .. 3: reserved
PHD.DEVID.FLG	Indicates the PHD carries device identifier information encoded in PHD.DEVID.INFO	1	0 1: reserved for future extensions
PHD.DEVID.INFO	Device identifier The identification procedure is left undefined, and this field is reserved for future extensions of this standard	48	0 $1 .. 2^{48}-1$: reserved for future extensions
	Reserved	128	0 $1 .. 2^{128}-1$: reserved for future extensions
PHD.VENDOR.FLG	Indicates the PHD carries vendor proprietary information encoded in PHD.VENDOR.INFO	1	0: no vendor information is included 1: PHD carries vendor proprietary information
PHD.VENDOR.INFO	Vendor proprietary information used for application specific implementations and extensions. In case of PHD.VENDOR.FLG = 0, this field must take the value zero for all the bits.	374	$0 .. 2^{374}-1$
PHD.CRC16	Cyclic redundancy code of 16 bits.	16	0x0000 ... 0xFFFF
Total (bits)		720	

PHD → CRC → Binary Scrambler → BCH Encoder → to BPSK 2-PAM Modulator

$1 + x^{22} + x^{25}$ (MLS)

Reset at beginning of transmit block

CRCgen   CRCout

| S0 | S1 | ⊕ | S2 | S3 | S4 | ⊕ | S5 | ⊕ | S6 | S7 | ⊕ | S8 | S9 | ⊕ | S10 | ⊕ | S11 | ⊕ | S12 | ⊕ | S13 | S14 | S15 |

All the adders are mod-2 (XOR)

Generator Polynomial: $1 + x^2 + x^5 + x^6 + x^8 + x^{10} + x^{11} + x^{12} + x^{13} + x^{16}$

CRC16 output

Serial Data Input

BCH Systematic Encoder

BCHgen   BCHCout

g(0)   g(1)   g(p-1)

| S0 | ⊕ | S1 | ------ | ⊕ | Sp-1 | ⊕ |

All the adders are mod-2 (XOR)

BCH parity D(x)

BCH information message M(x)
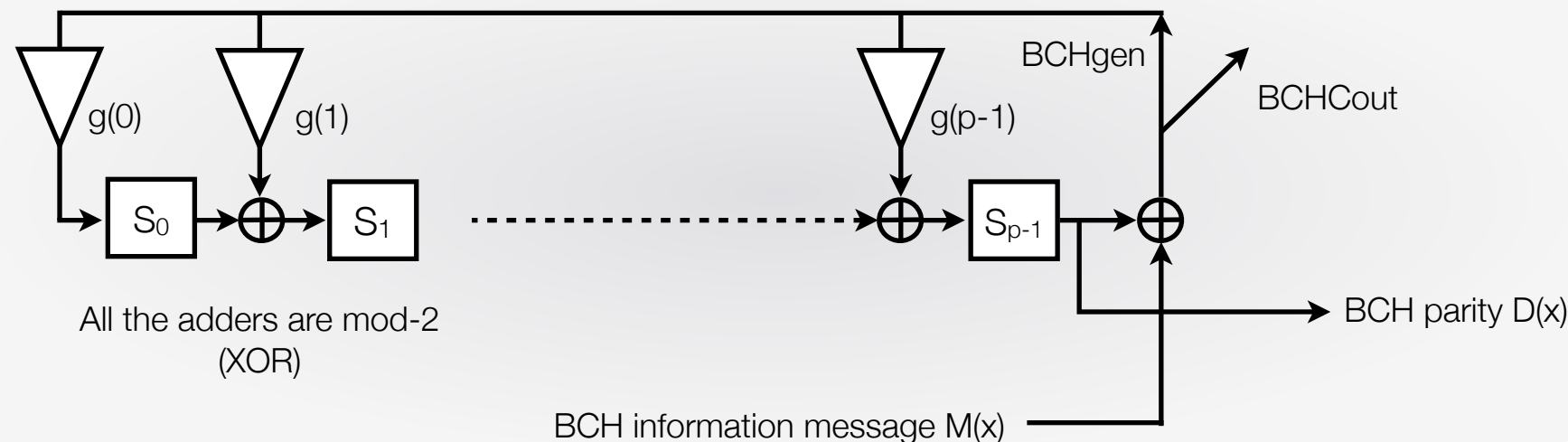
(896, 720) shortened of a GF(2^{11}) BCH (2047, 1871) t = 16

References

- [1] *Rubén Pérez-Aranda, "Shannon's capacity analysis of GEPOF for technical feasibility assessment", GEPOF SG, Interim Meeting, May 2014*

- [2] *Rubén Pérez-Aranda, "Link budget requirements for Gigabit over POF", GEPOF SG, Interim Meeting, May 2014*

- [3] *Rubén Pérez-Aranda, "Optical receiver characteristics for GEPOF technical feasibility", GEPOF SG, Interim Meeting, May 2014*

- [4] *Rubén Pérez-Aranda, "Optical transmitter characteristics for GEPOF technical feasibility", GEPOF SG, Interim Meeting, May 2014*

- [5] *Rubén Pérez-Aranda, "High spectrally efficient coded modulation schemes for GEPOF technical feasibility", GEPOF SG, Plenary Meeting, July 2014*

- [6] *Rubén Pérez-Aranda, "Proposal of a transmission structure for GEPOF technical feasibility", GEPOF SG, Plenary Meeting, July 2014*

Questions?