Details of 4-lane Interleaved 100G FEC

Shawn Nicholl, Xilinx

Ben Jones, Xilinx

IEEE P802.3ck Task Force Ad Hoc Call

April 24, 2019

Previous Work

- gustlin_3ck_01_0518.pdf, proposes re-use of the 802.3bs and 802.3cd PCS/FEC/PMA sublayers in this project
- anslow_3ck_adhoc_01_072518.pdf, initial FEC performance analysis
- gustlin_3ck_01_0718.pdf, proposes a possible RS symbol muxing scheme in the PMA sublayer, this is no longer being considered
- anslow_3ck_01_0918.pdf, updated FEC performance analysis
- gustlin_3ck_01_1118.pdf, proposes a new interleaved FEC sublayer for the most difficult channels at 100GbE
- anslow_3ck_01_1118.pdf, analyses the interleaved FEC performance
- nicholl_3cn_01b_181211.pdf, proposes a baseline for CGMII extender (not adopted)

Previous Work

- gustlin_3ck_01_0119.pdf, proposes a baseline for the new interleaved FEC sublayer
- nicholl_3ct_01a_0319.pdf, contains adopted baseline for Inverse RS-FEC
- gustlin_3ck_01_0319.pdf, contains adopted baseline for C2M and C2C-S channels

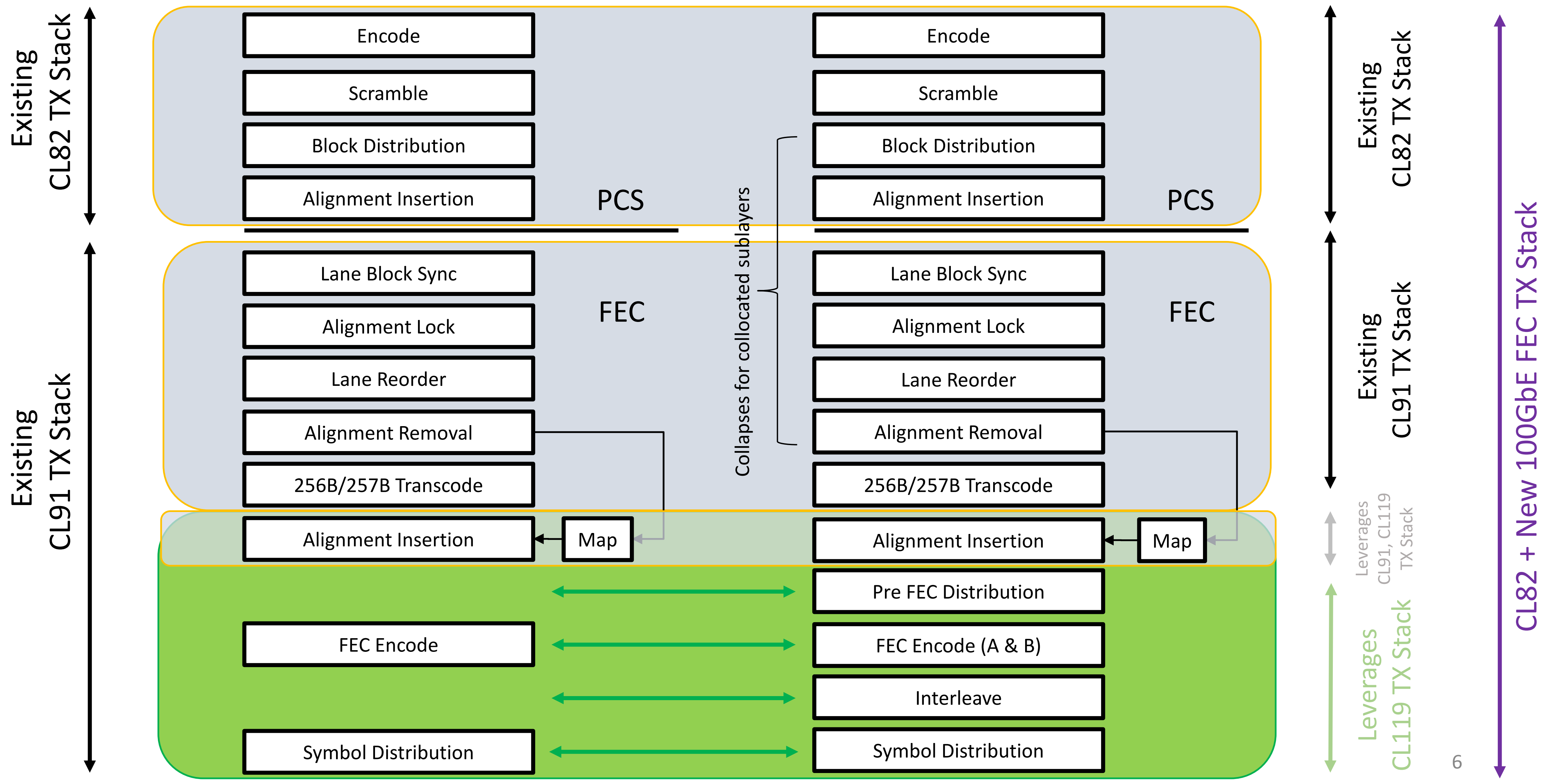
Acknowledgements

- **Thanks to Jeff Slavick and Eric Baden for their feedback on the content of this presentation**

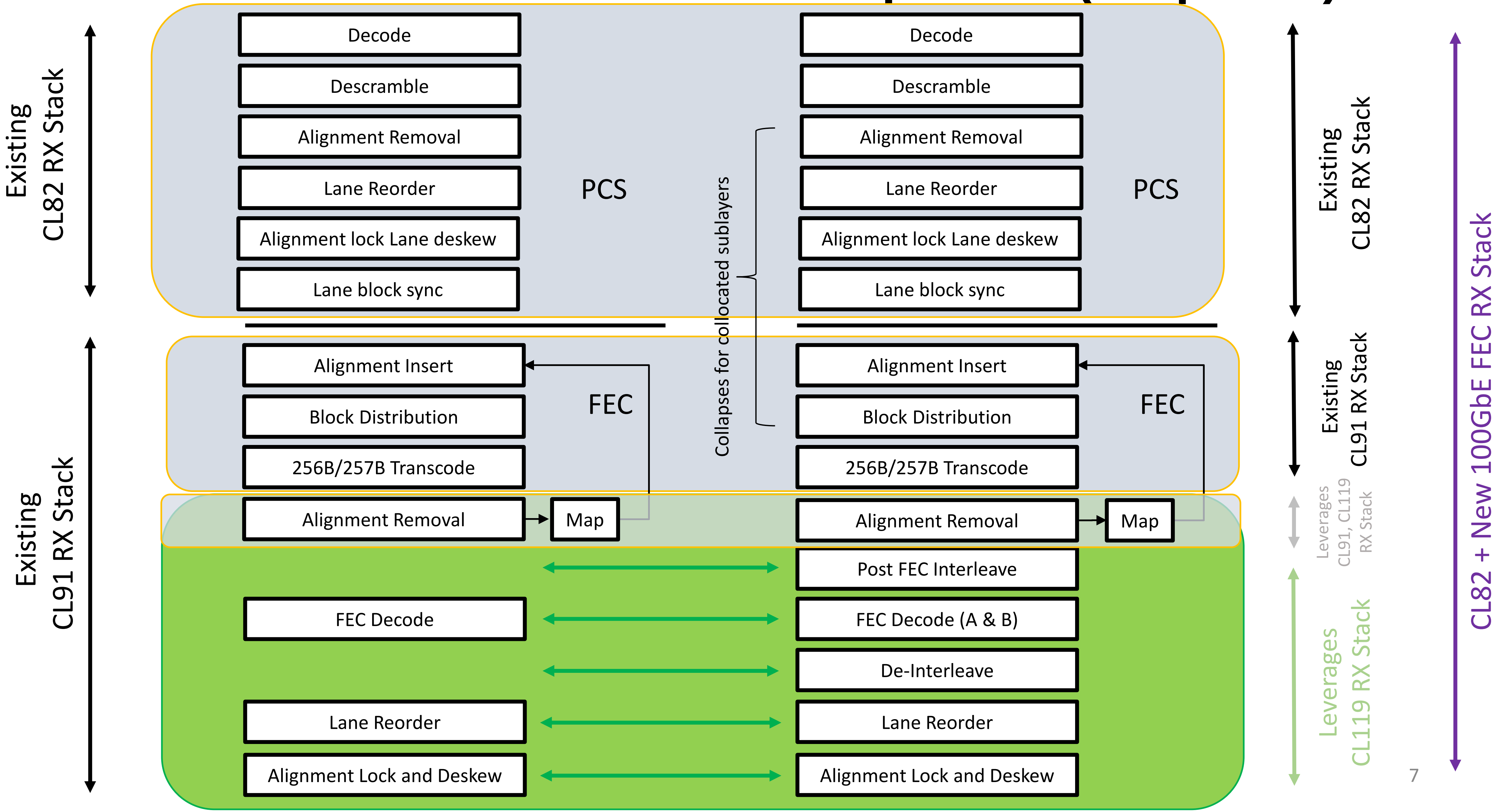
Overview

- **This presentation works through details associated with a proposed 4-lane interleaved 100G FEC for P802.3ck**
- **Use of 4 FEC lanes differs from previous 2-lane interleaved FEC proposal**
- **During offline consensus building, folks expressed a stronger appetite for 4 lanes**
    - Greater commonality with CL91
        - Alignment marker method same as 802.3cd-2018 CL91
    - Consistent with the capacity/lanes ratio already found in FEC (CL91), PCS (CL119)
        - 100G: 4 FEC lanes → 25Gbps / lane
        - 200G: 8 PCS lanes → 25Gbps / lane
        - 400G: 16 PCS lanes → 25Gbps / lane
    - Possibly allows for reduced development effort due to re-use from existing implementations
    - Allows system integrators to use 25G-based devices (eg. FPGA’s) to support new interleaved FEC with external bit muxes/gearboxes
    - Expect FLR performance similar to 400G (CL119)
- **The resultant 4-lane interleaved FEC heavily leverages existing CL91 and CL119**

Interleaved 100G FEC Tx Stack Comparison (Proposed)



Interleaved 100G FEC Rx Stack Comparison (Proposed)



Existing CL82 RX Stack

Existing CL91 RX Stack

Decode

Descramble

Alignment Removal

Lane Reorder

Alignment lock Lane deskew

Lane block sync

PCS

Alignment Insert

Block Distribution

256B/257B Transcode

FEC

Alignment Removal

Map

FEC Decode

Lane Reorder

Alignment Lock and Deskew

Collapses for collocated sublayers

Decode

Descramble

Alignment Removal

Lane Reorder

Alignment lock Lane deskew

Lane block sync

PCS

Alignment Insert

Block Distribution

256B/257B Transcode

FEC

Alignment Removal

Map

Post FEC Interleave

FEC Decode (A & B)

De-Interleave

Lane Reorder

Alignment Lock and Deskew

Existing CL82 RX Stack

Existing CL91 RX Stack

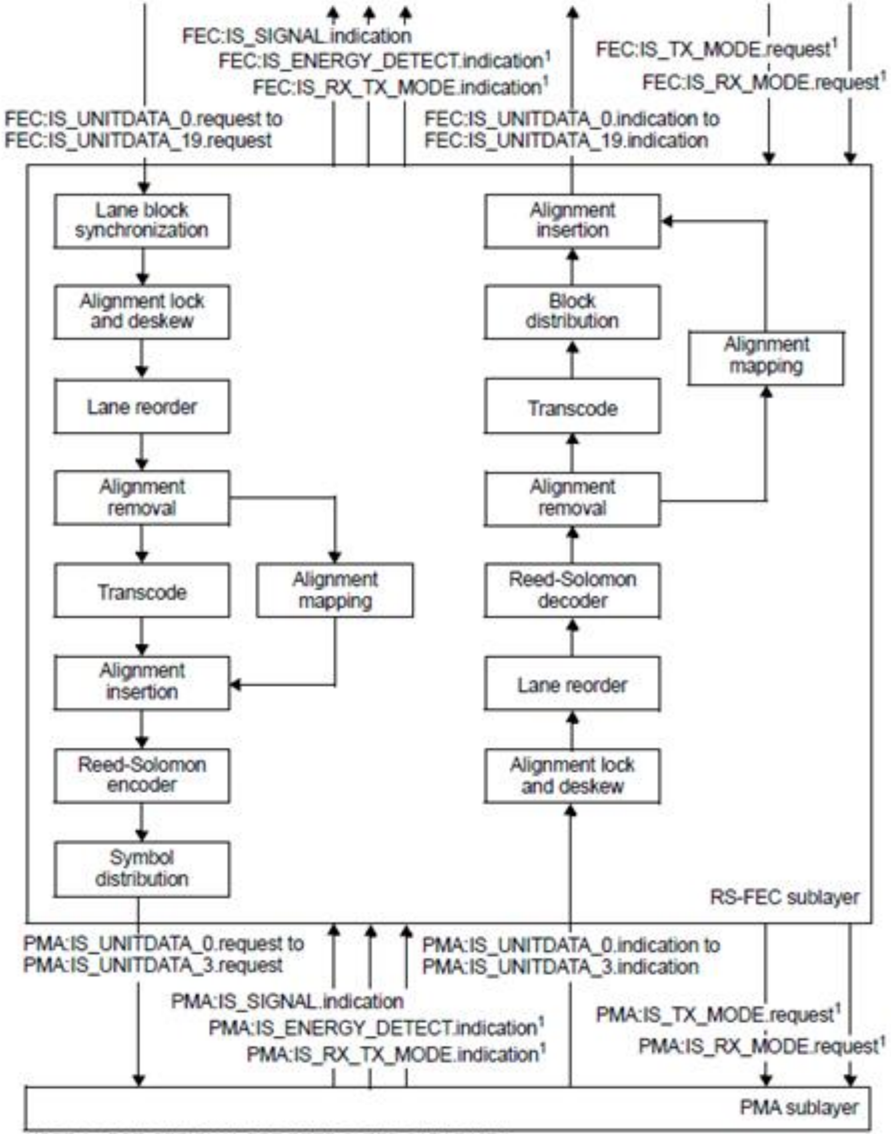
Leverages CL91, CL119 RX Stack

Leverages CL119 RX Stack

CL82 + New 100GbE FEC RX Stack

Current 802.3-2018 Clause 91

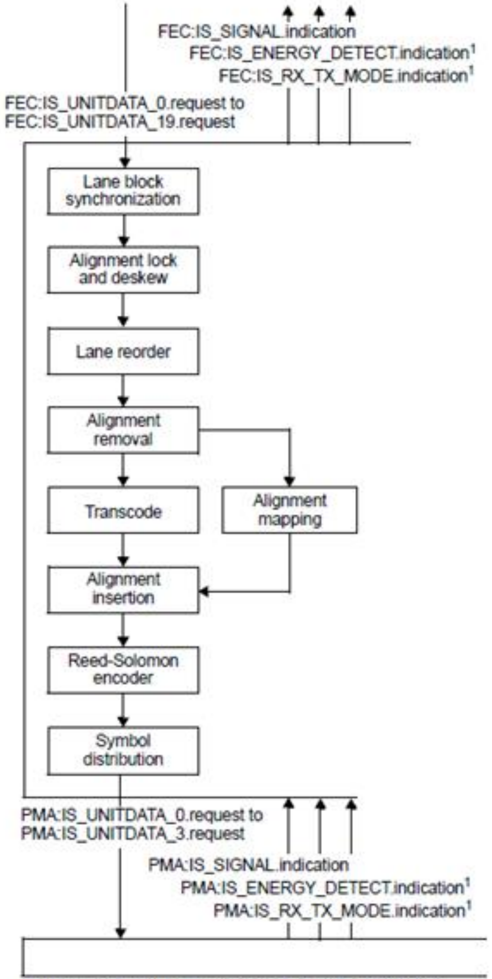
- **This is the existing Clause 91 RS-FEC**


FEC:IS_SIGNAL.indication
FEC:IS_ENERGY_DETECT.indication[1]
FEC:IS_RX_TX_MODE.indication[1]
FEC:IS_TX_MODE.request[1]
FEC:IS_RX_MODE.request[1]
FEC:IS_UNITDATA_0.request to
FEC:IS_UNITDATA_19.request
FEC:IS_UNITDATA_0.indication to
FEC:IS_UNITDATA_19.indication
Lane block synchronization
Alignment lock and deskew
Lane reorder
Alignment removal
Transcode
Alignment mapping
Alignment insertion
Reed-Solomon encoder
Symbol distribution
Alignment insertion
Block distribution
Alignment mapping
Transcode
Alignment removal
Reed-Solomon decoder
Lane reorder
Alignment lock and deskew
RS-FEC sublayer
PMA:IS_UNITDATA_0.request to
PMA:IS_UNITDATA_3.request
PMA:IS_UNITDATA_0.indication to
PMA:IS_UNITDATA_3.indication
PMA:IS_SIGNAL.indication
PMA:IS_ENERGY_DETECT.indication[1]
PMA:IS_RX_TX_MODE.indication[1]
PMA:IS_TX_MODE.request[1]
PMA:IS_RX_MODE.request[1]
PMA sublayer

NOTE 1—FOR OPTIONAL EEE DEEP SLEEP CAPABILITY

Figure 91–2—Functional block diagram

Current 802.3-2018 CL91, CL119 Tx RS-FEC(s)


FEC:IS_SIGNAL.indication
FEC:IS_ENERGY_DETECT.indication[1]
FEC:IS_RX_TX_MODE.indication[1]
FEC:IS_UNITDATA_0.request to
FEC:IS_UNITDATA_19.request
Lane block synchronization
Alignment lock and deskew
Lane reorder
Alignment removal
Transcode
Alignment mapping
Alignment insertion
Reed-Solomon encoder
Symbol distribution
PMA:IS_UNITDATA_0.request to
PMA:IS_UNITDATA_3.request
PMA:IS_SIGNAL.indication
PMA:IS_ENERGY_DETECT.indication[1]
PMA:IS_RX_TX_MODE.indication[1]

NOTE 1—FOR OPTIONAL EEE DEEP SLEEP CAPABILITY

Figure 91–2—Functional block diagram

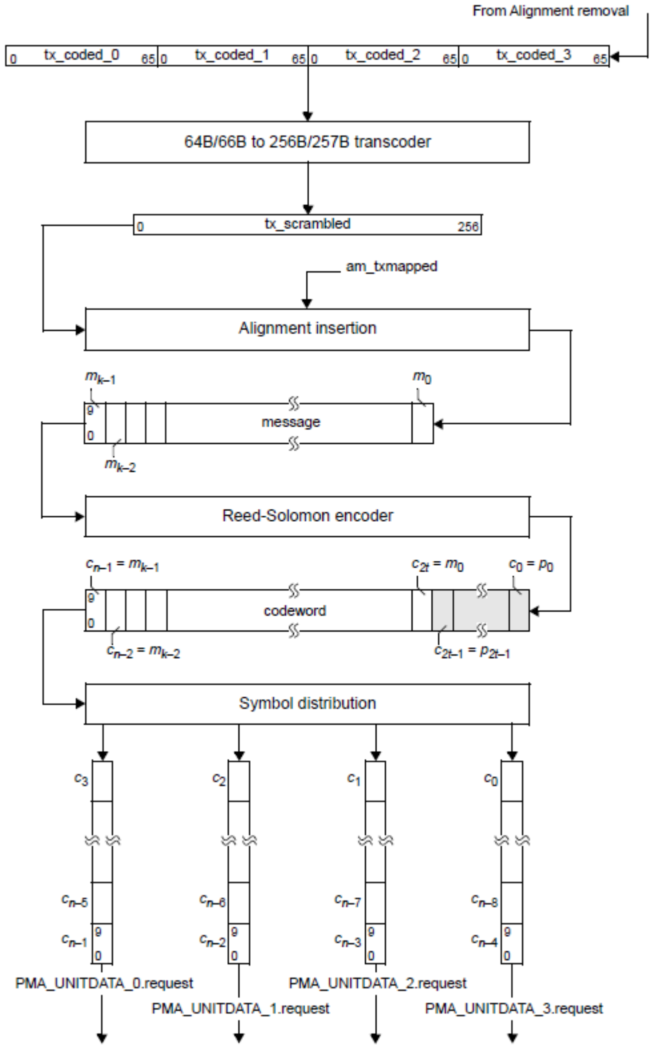


Figure 91–6—Transmit bit ordering

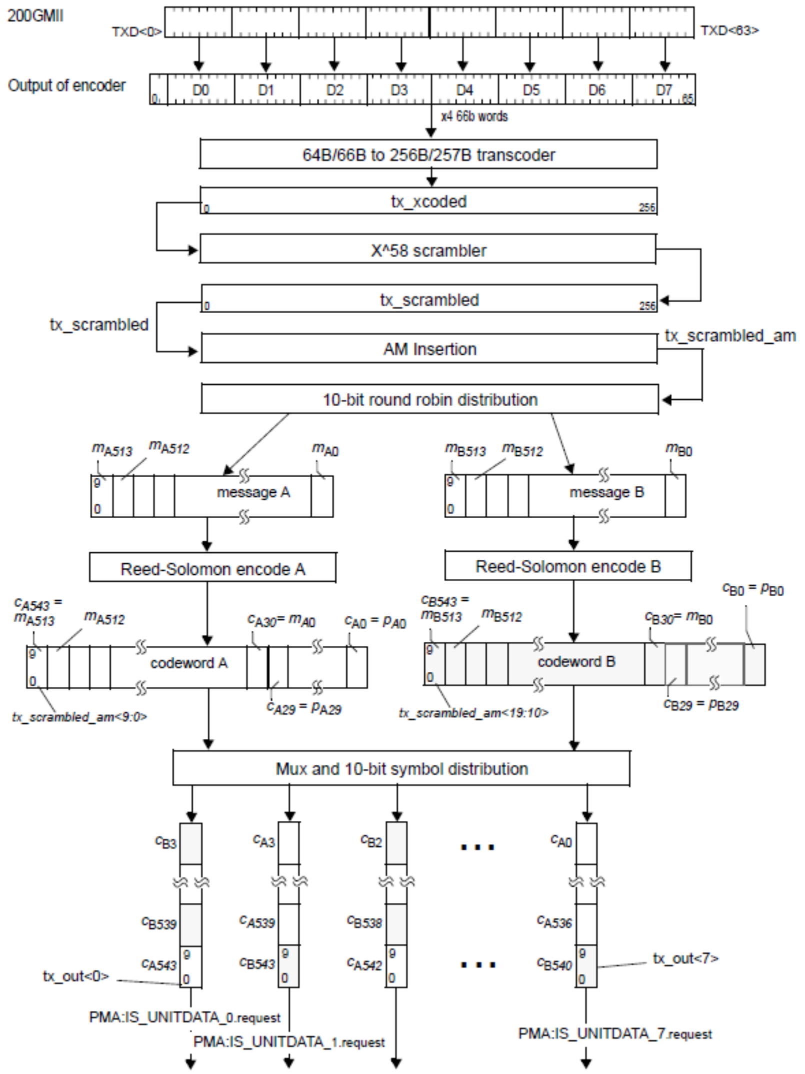


Figure 119–10—200GBASE-R Transmit bit ordering and distribution

Current 802.3-2018 CL91, CL119 Rx RS-FEC(s)

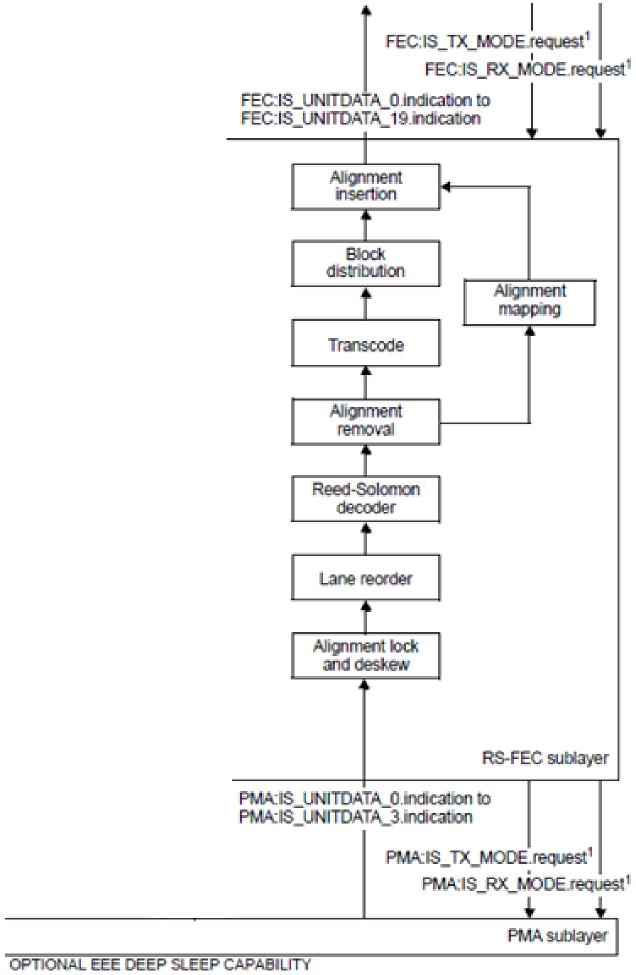

FEC:IS_TX_MODE.request[1]
FEC:IS_RX_MODE.request[1]
FEC:IS_UNITDATA_0.indication to
FEC:IS_UNITDATA_19.indication
Alignment insertion
Block distribution
Alignment mapping
Transcode
Alignment removal
Reed-Solomon decoder
Lane reorder
Alignment lock and deskew
RS-FEC sublayer
PMA:IS_UNITDATA_0.indication to
PMA:IS_UNITDATA_3.indication
PMA:IS_TX_MODE.request[1]
PMA:IS_RX_MODE.request[1]
PMA sublayer
OPTIONAL EEE DEEP SLEEP CAPABILITY

Figure 91–2—Functional block diagram

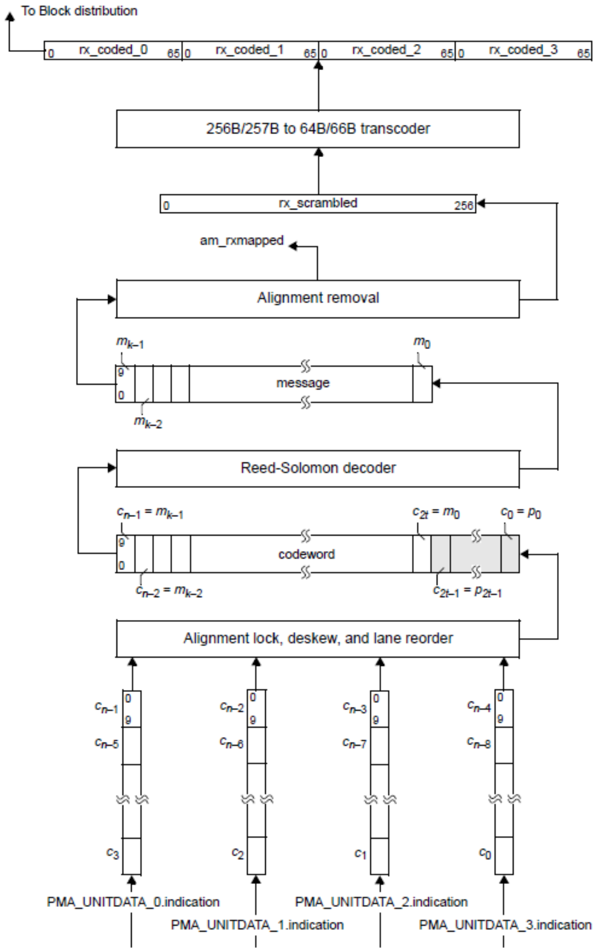
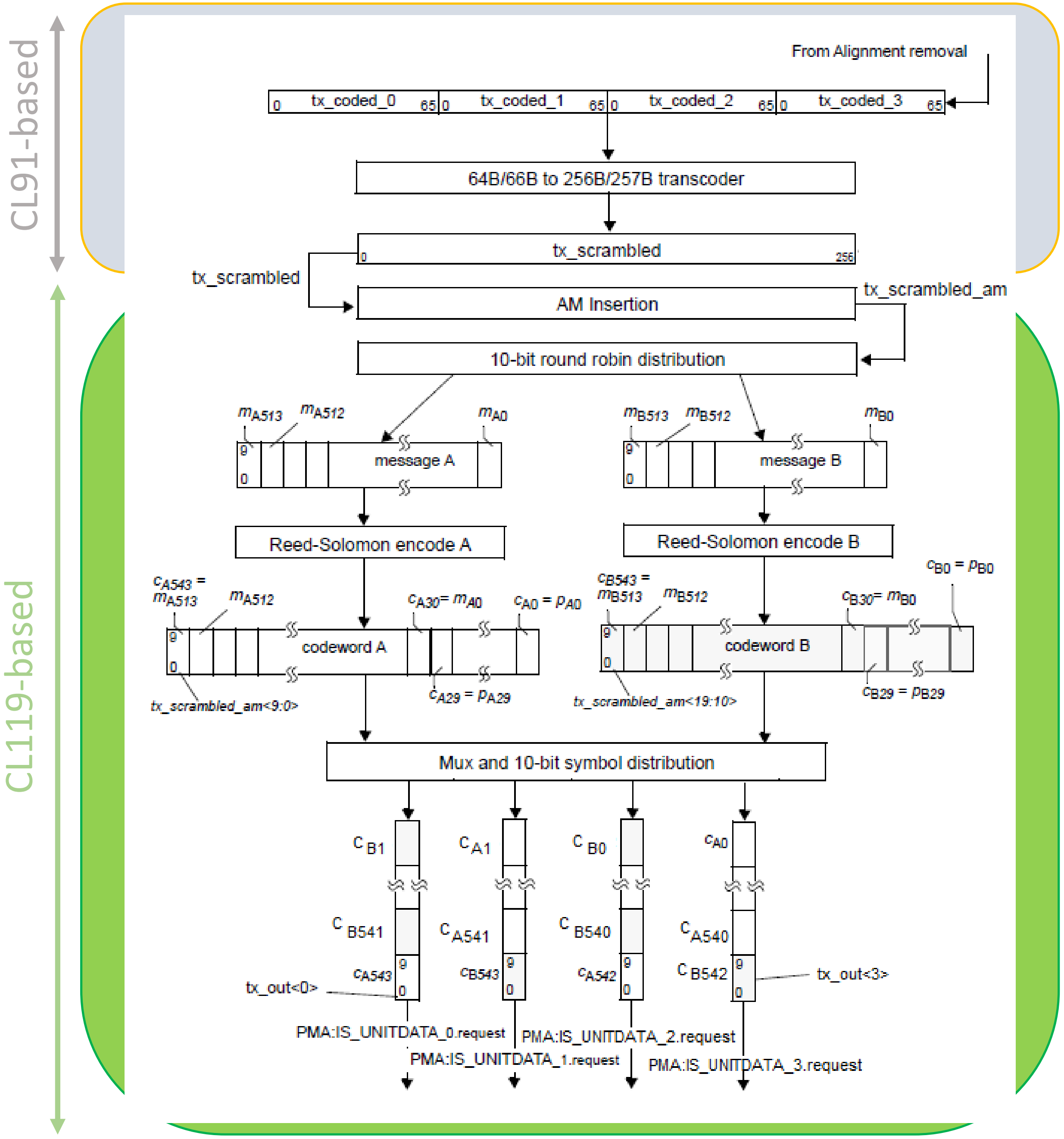


Figure 91–7—Receive bit ordering

No
Diagram
In CL119
For Receive
Bit Ordering

P802.3ck Tx Interleaved FEC

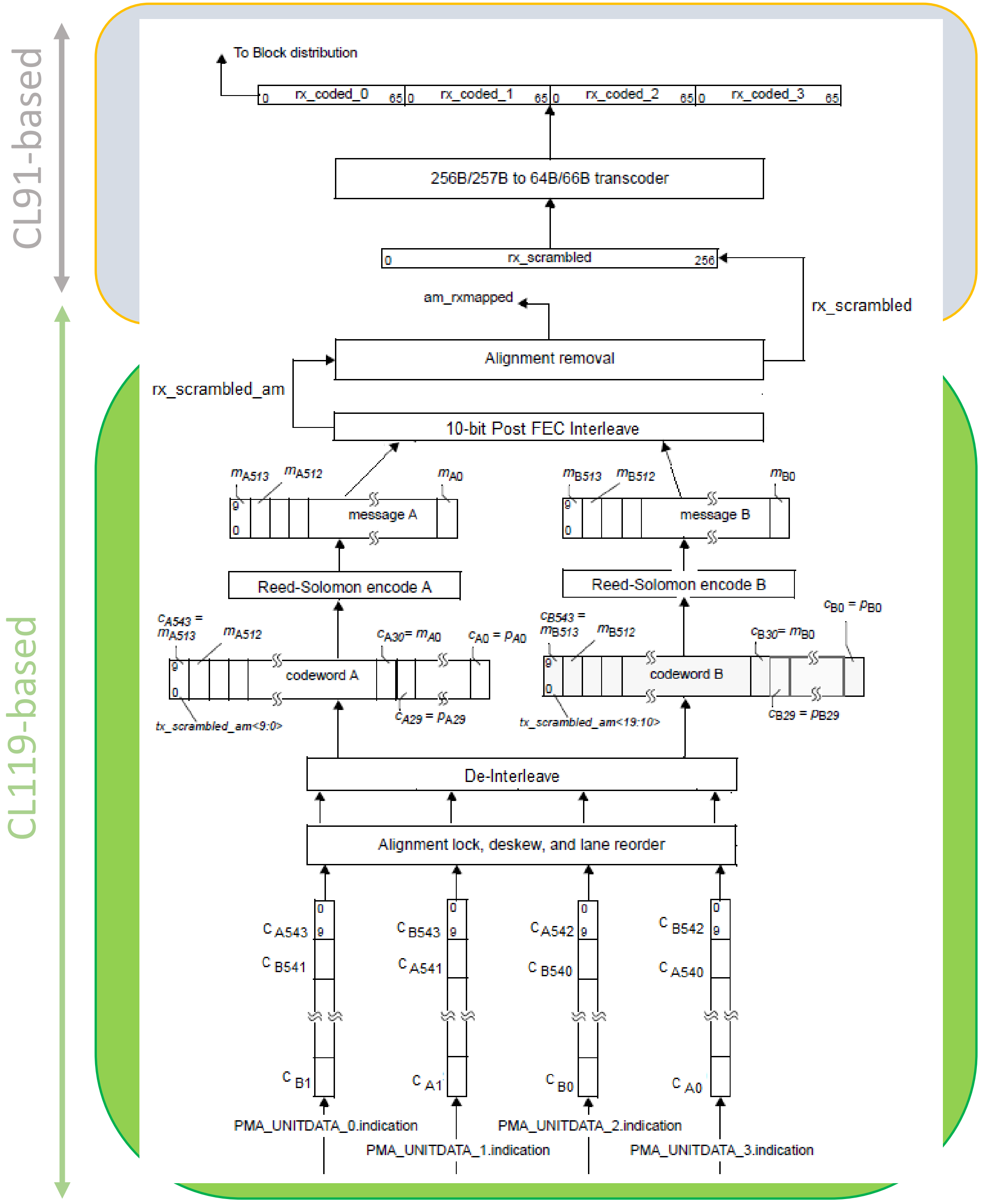
- **Transmit Ordering for 4 FEC lanes**
  - The ordering is consistent with CL119
- **Symbol distribution**
  - 10-bit symbols are interleaved onto FEC lanes in the same manner as CL119
    - “A” character below represents a 10-bit symbol from codeword A
    - “B” character below represents a 10-bit symbol from codeword B
  - ABABBABAABABBABA…ABABBABA
  - Diagram Note:
    - The bottom row corresponds to the yellow highlighted symbols above
    - The second-from-bottom row corresponds to the green-highlighted symbols above
    - The third-from-bottom row represents many symbols and corresponds to the non-highlighted symbols above



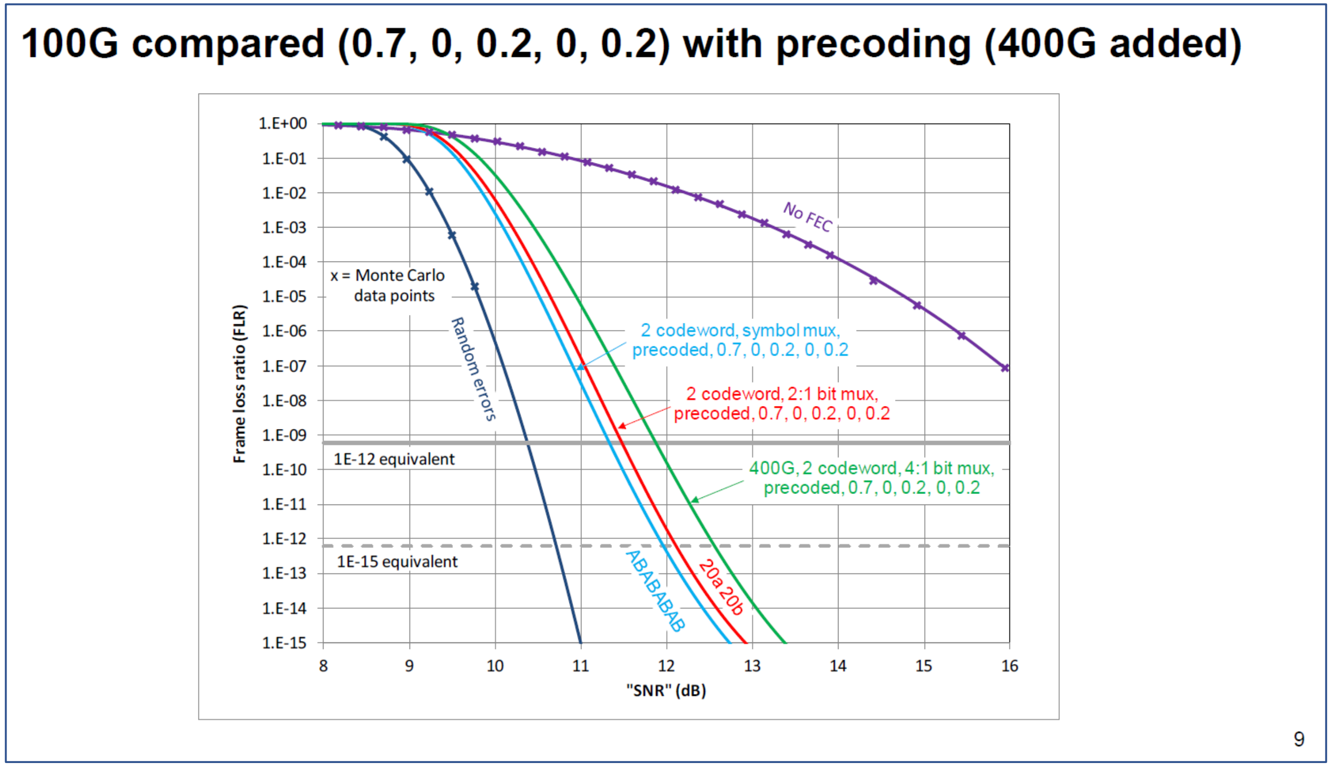
CL91-based
CL119-based
From Alignment removal
0 tx_coded_0 65
0 tx_coded_1 65
0 tx_coded_2 65
0 tx_coded_3 65
64B/66B to 256B/257B transcoder
0 tx_scrambled 256
tx_scrambled
AM Insertion
tx_scrambled_am
10-bit round robin distribution
m_{A513} m_{A512} m_{A0}
9 0 message A
m_{B513} m_{B512} m_{B0}
9 0 message B
Reed-Solomon encode A
Reed-Solomon encode B
$c_{A543} = m_{A513}$ m_{A512} $c_{A30} = m_{A0}$ $c_{A0} = p_{A0}$
9 0 codeword A
$c_{A29} = p_{A29}$
tx_scrambled_am<9:0>
$c_{B543} = m_{B513}$ m_{B512} $c_{B30} = m_{B0}$ $c_{B0} = p_{B0}$
9 0 codeword B
$c_{B29} = p_{B29}$
tx_scrambled_am<19:10>
Mux and 10-bit symbol distribution
c_{B1} c_{B541} c_{A543} 9 0
c_{A1} c_{A541} c_{B543} 9 0
c_{B0} c_{B540} c_{A542} 9 0
c_{A0} c_{A540} c_{B542} 9 0
tx_out<0>
tx_out<3>
PMA:IS_UNITDATA_0.request
PMA:IS_UNITDATA_1.request
PMA:IS_UNITDATA_2.request
PMA:IS_UNITDATA_3.request

P802.3ck Rx Interleaved FEC

- **Receive Ordering for 4 FEC lanes**
  - The ordering is consistent with CL119
- **Symbol distribution**
  - 10-bit symbols are de-interleaved from FEC lanes in the same manner as CL119
    - “A” character below represents a 10-bit symbol from codeword A
    - “B” character below represents a 10-bit symbol from codeword B
  - ABABBABAABABBABA…ABABBABA
  - Diagram Note:
    - The row below “Alignment lock, deskew, and lane reorder” block corresponds to the yellow highlighted symbols above
    - The second-row below “Alignment lock, deskew, and lane reorder” corresponds to the green-highlighted symbols above
    - The third-row below represents many symbols and corresponds to the non-highlighted symbols above



Performance Considerations



100G compared (0.7, 0, 0.2, 0, 0.2) with precoding (400G added)

9

- **The 4:1 bit mux results in slightly degraded performance compared to the 2:1 bit mux**
  - See anslow_3ck_01_1118.pdf slide 9
    - 4:1 bit mux shown by the green curve
    - 2:1 bit mux shown by the red curve
- **It is expected that the performance would be no worse than 400G**
  - In other words, regardless of the limits we put on the DFE, we will get the performance of 400G

Summary

- **This presentation works through the details of a 4-lane interleaved 100G FEC**
- **Slides that follow contain technical details of the proposed FEC**
  - Draft text is proposed for a new Clause 300

Technical Details Beyond Here!

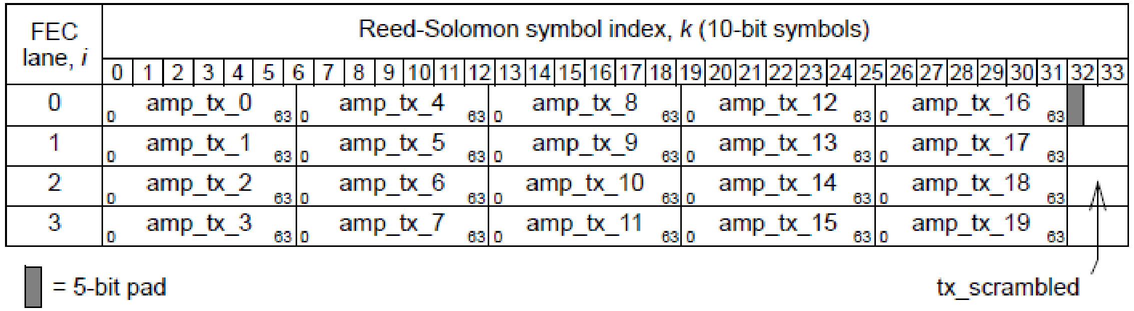
P802.3ck – Interleaved FEC Tx – CL91-based functions

- **Assume new Clause 300 for Interleaved FEC**
  - 300.5.2 Transmit function
- **Sections that could be directly used from Clause 91 are following:**
  - 300.5.2.1 Lane block synchronization
    - Same as 91.5.2.1 Lane block synchronization
  - 300.5.2.2 Alignment lock and deskew
    - Same as 91.5.2.2 Alignment lock and deskew
  - 300.5.2.3 Lane reorder
    - Same as 91.5.2.3 Lane reorder
  - 300.5.2.4 Alignment marker removal
    - Same as 91.5.2.4 Alignment marker removal
  - 300.5.2.5 64B/66B to 256B/257B transcoder
    - Same as 91.5.2.5 64B/66B to 256B/257B transcoder

P802.3ck – Interleaved FEC Tx – Changes from CL91

- **Sections that would be different from Clause 91 are following:**
    - 300.5.2.6 Alignment marker mapping and insertion
        - Based on 91.5.2.6 Alignment marker mapping and insertion (as amended by 802.3cd-2018)
            - amp_tx_x creation the same as CL91 with four_lane_pmd = false [no EEE deep sleep support]
            - amp_tx_0=am0, amp_tx_1=am0, amp_tx_2=am0, amp_tx_3=am0, other AM's are unchanged
        - Remainder of mapping process to form 10280-bit block based on 119.2.4.4 "Alignment marker mapping and insertion" to account for checkboard patterning
    - 300.5.2.7 Pre-FEC distribution
        - Same as 119.2.4.5 Pre-FEC distribution
    - 300.5.2.8 Reed-Solomon encoder
        - Same as 119.2.4.6 Reed-Solomon encoder
    - 300.5.2.9 Symbol distribution
        - Based on 119.2.4.7 Symbol distribution
        - Distribute to four FEC lanes instead of 16 (or 8)

Interleaved FEC Tx – AM Values



FEC lane, *i*	Reed-Solomon symbol index, *k* (10-bit symbols) 0–6	6–12	13–19	19–25	26–31	32–33
0	amp_tx_0	amp_tx_4	amp_tx_8	amp_tx_12	amp_tx_16	▮
1	amp_tx_1	amp_tx_5	amp_tx_9	amp_tx_13	amp_tx_17	
2	amp_tx_2	amp_tx_6	amp_tx_10	amp_tx_14	amp_tx_18	
3	amp_tx_3	amp_tx_7	amp_tx_11	amp_tx_15	amp_tx_19	

Figure 91–4—Alignment marker mapping to FEC lanes

- **Use 802.3cd-2018 CL91**
  - AM's for 0 to 3 are identical
  - Other are unique
    - i.e. 16 to 19 not made identical

Heavily Leverages 802.3cd CL91

For x=0 to 19, amp_tx_x<63:0> is constructed as follows:

a) Set y = 0 when x <= 3, otherwise set y = x.

b) amp_tx_x<23:0> is set to M0, M1, and M2 as shown in Figure 82-9 (bits 25 to 2) using the values in Table 82-2 for PCS lane number y.

c) amp_tx_x<31:24> = am_tx_x<33:26>

d) amp_tx_x<55:32> is set to M4, M5, and M6 as shown in Figure 82-9 (bits 57 to 34) using the values in Table 82-2 for PCS lane number y.

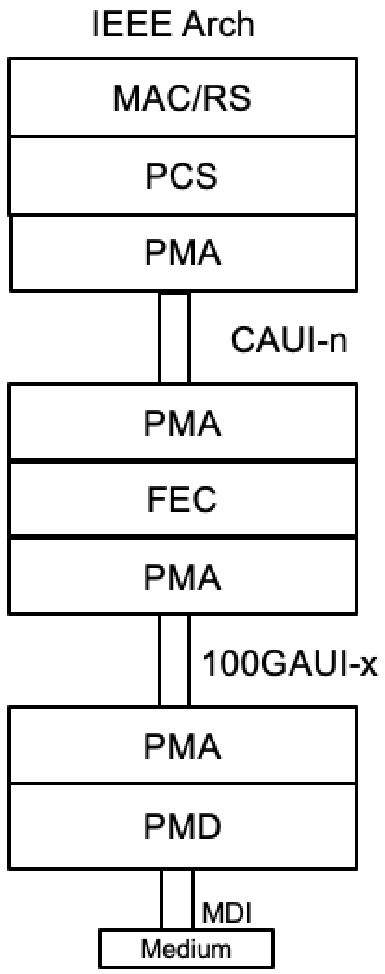
e) amp_tx_x<63:56> = am_tx_x<65:58>

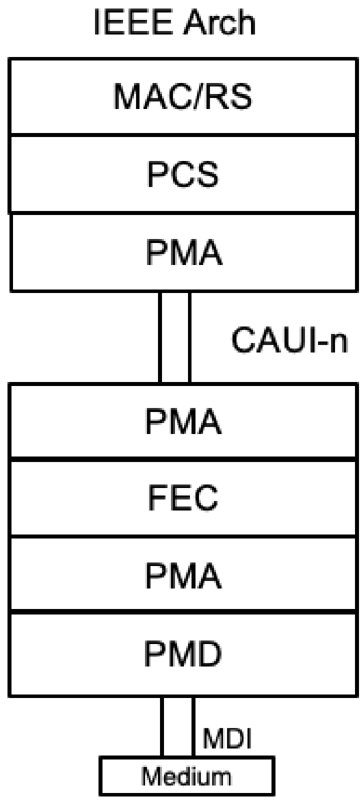
This process replaces the fixed bytes of the alignment markers received, possibly with errors, with the values from Table 82-2. In addition it substitutes the fixed bytes of the alignment markers corresponding to PCS lanes 1, 2, and 3 with the fixed bytes for the alignment marker corresponding to PCS lane 0. The variable bytes BIP are unchanged. This process simplifies receiver synchronization since the receiver only needs to search for the fixed bytes corresponding to PCS lane 0 on each FEC lane.

Interleaved FEC Tx – BIP Preservation

- **Preserve the alignment marker BIP fields as they pass through the FEC sublayer**
- **This provides support for an architecture that includes a remote FEC engine**
  - The CAUI-n that is unprotected by FEC will still contain useful BIP information

Heavily Leverages Clause 91





Interleaved FEC Tx – AM Values (am_txpayloads)

- **Formation of am_txpayloads**

Construct a matrix of 4 rows and 320 columns, am_txpayloads, as shown in Figure 300-4. Given i=0 to 3, j=0 to 4, and x=i+4j, the matrix is derived per the following expression:

am_txpayloads<i,(64j+63):64j> = amp_tx_x<63:0>

Identical
to
802.3cd
CL91

```
am_txpayloads<0,63:0>    = amp_tx_0<63:0>
am_txpayloads<1,63:0>    = amp_tx_1<63:0>
am_txpayloads<2,63:0>    = amp_tx_2<63:0>
am_txpayloads<3,63:0>    = amp_tx_3<63:0>
am_txpayloads<0,127:64>  = amp_tx_4<63:0>
am_txpayloads<1,127:64>  = amp_tx_5<63:0>
am_txpayloads<2,127:64>  = amp_tx_6<63:0>
am_txpayloads<3,127:64>  = amp_tx_7<63:0>
am_txpayloads<0,191:128> = amp_tx_8<63:0>
am_txpayloads<1,191:128> = amp_tx_9<63:0>
am_txpayloads<2,191:128> = amp_tx_10<63:0>
am_txpayloads<3,191:128> = amp_tx_11<63:0>
am_txpayloads<0,255:192> = amp_tx_12<63:0>
am_txpayloads<1,255:192> = amp_tx_13<63:0>
am_txpayloads<2,255:192> = amp_tx_14<63:0>
am_txpayloads<3,255:192> = amp_tx_15<63:0>
am_txpayloads<0,319:256> = amp_tx_16<63:0>
am_txpayloads<1,319:256> = amp_tx_17<63:0>
am_txpayloads<2,319:256> = amp_tx_18<63:0>
am_txpayloads<3,319:256> = amp_tx_19<63:0>
```

Interleaved FEC Tx – AM Values (am_txmapped)

- **Want AM's to remain intact so need to counter-act the symbol interleaving that occurs on the way to the PMA**

```
for all k=0 to 31
  for all j=0 to 1
    if isEven(k)
      am_txmapped<40k+20j+9:40k+20j> = am_txpayloads<2j,10k+9:10k>
      am_txmapped<40k+20j+19:40k+20j+10> = am_txpayloads<2j+1,10k+9:10k>
    else
      am_txmapped<40k+20j+9:40k+20j> = am_txpayloads<2j+1,10k+9:10k>
      am_txmapped<40k+20j+19:40k+20j+10> = am_txpayloads<2j,10k+9:10k>
```

Heavily
Leverages
Clause
119

Interleaved FEC Tx – AM Values (am_txmapped)

```
am_txmapped<9:0>        = am_txpayloads<0,9:0>
am_txmapped<19:10>      = am_txpayloads<1,9:0>
am_txmapped<29:20>      = am_txpayloads<2,9:0>
am_txmapped<39:30>      = am_txpayloads<3,9:0>
am_txmapped<49:40>      = am_txpayloads<1,19:10>
am_txmapped<59:50>      = am_txpayloads<0,19:10>
am_txmapped<69:60>      = am_txpayloads<3,19:10>
am_txmapped<79:70>      = am_txpayloads<2,19:10>
am_txmapped<89:80>      = am_txpayloads<0,29:20>
am_txmapped<99:90>      = am_txpayloads<1,29:20>
am_txmapped<109:100>    = am_txpayloads<2,29:20>
am_txmapped<119:110>    = am_txpayloads<3,29:20>
am_txmapped<129:120>    = am_txpayloads<1,39:30>
am_txmapped<139:130>    = am_txpayloads<0,39:30>
am_txmapped<149:140>    = am_txpayloads<3,39:30>
am_txmapped<159:150>    = am_txpayloads<2,39:30>
am_txmapped<169:160>    = am_txpayloads<0,49:40>
am_txmapped<179:170>    = am_txpayloads<1,49:40>
am_txmapped<189:180>    = am_txpayloads<2,49:40>
am_txmapped<199:190>    = am_txpayloads<3,49:40>
<SNIP>
```

```
<SNIP>
am_txmapped<1209:1300>  = am_txpayloads<0,309:300>
am_txmapped<1219:1210>  = am_txpayloads<1,309:300>
am_txmapped<1229:1220>  = am_txpayloads<2,309:300>
am_txmapped<1239:1230>  = am_txpayloads<3,309:300>
am_txmapped<1249:1240>  = am_txpayloads<1,319:310>
am_txmapped<1259:1250>  = am_txpayloads<0,319:310>
am_txmapped<1269:1260>  = am_txpayloads<3,319:310>
am_txmapped<1279:1270>  = am_txpayloads<2,319:310>
am_txmapped<1284:1280>  = 5'b00101 or 5'b11010 (alternating)
```

Interleaved FEC Tx – AM Spacing

- **AM spacing same as 91.5.2.6**
- **Formation of tx_scrambled_am based on pairs of codewords**
    - 10280 bits at a time

Heavily Leverages Clause 91

One group of aligned and reordered alignment markers are mapped every 20 × 16384 66-bit blocks. This corresponds to 81920 x 257-bit blocks. This group of aligned and reordered alignment markers is called the "alignment marker group".

The alignment marker group am_txmapped<1284:0> shall be inserted so it appears in the output stream every 81 920 × 257-bit blocks. The variable tx_scrambled_am<10279:0> is constructed in one of two ways. Let the set of vectors tx_scrambled_i<256:0> represent consecutive values of tx_scrambled<256:0>.

For a 10280-bit block with an alignment marker group inserted:

tx_scrambled_am<1284:0> = am_txmapped<1284:0>

For all i=0 to 34

tx_scrambled_am<257i+1541:257i+1285> = tx_scrambled_i<256:0>

For a 10280-bit block without an alignment marker group:

For all i=0 to 39

tx_scrambled_am<257i+256:257i> = tx_scrambled_i<256:0>

Interleaved FEC Tx – AM Insertion

- **Insertion of AM's uses diagram similar to the one found in 119.2.4.4.1 and 119.2.4.4.2**

For each 10280-bit block with an alignment marker group inserted, the first 257-bit block inserted after am_txmapped shall correspond to the four 66-bit blocks received on PCS lanes 0, 1, 2, and 3 that immediately followed the alignment marker on each respective lane.

Alignment marker repetition rate is shown in Figure 300-TBD.

```
---------------------------------------------         ----------------------------
|     2 codewords            |  2 codewords  |         | 2 codewords               |
---------------------------------------------         ----------------------------
|           |                |               |         |           |               |
| am_mapped | tx_scrambled   |  tx_scrambled |  * * *  | am_mapped | tx_scrambled  |
| 5x257-bit |  35x257-bit    |   40x257-bit  |         | 5x257-bit |  35x257-bit   |
| blocks    |    blocks      |    blocks     |         | blocks    |    blocks     |
|           |                |               |         |           |               |
---------------------------------------------         ----------------------------
 \                                                    /
  \                                                  /
   \                                                /
       81920 x 257-bit blocks = 4096 codewords
```

Heavily
Leverages
Clause
119

Interleaved FEC Tx – Pre-FEC distribution

- **Pre-FEC distribution same as 119.2.4.5**

For all i=0 to 513

mA<(513-i)> = tx_scrambled_am<(20i+9):(20i)>

mB<(513-i)> = tx_scrambled_am<(20i+19):(20i+10)>

Identical
to
Clause
119

Interleaved FEC Tx – Reed-Solomon encoder

- **Reed-Solomon encoder like 119.2.4.6**

The PCS shall implement an RS(544,514) based FEC encoder. The PCS distributes a group of 40 × 257-bit blocks from tx_scrambled_am on a 10-bit round robin basis into two 5140-bit message blocks, mA and mB, as described in 300.5.2.7. These are then encoded using RS(544,514) encoder into codeword A and codeword B, respectively.

Identical to Clause 119

Interleaved FEC Tx – Codeword Interleaving

- **Interleaving and symbol distribution like 119.2.4.7**
  - Except 4 FEC lanes for 100GBASE-R PCS
    - Compared to 8 PCS lanes for 200GBASE-R or 16 PCS lane for 400GBASE-R

Heavily Leverages Clause 119

Once the data has been FEC encoded, two FEC codewords (cA<543:0> and cB<543:0>) are interleaved on a 10-bit basis before the data is distributed to each FEC lane.

The interleaving of two codewords for the 100GBASE-R PCS shall follow this procedure:

```
For all k=0 to 271
For all j=0 to 1
  if even(k)
    tx_out<4k+2j> = cA<543-2k-j>
    tx_out<4k+2j+1> = cB<543-2k-j>
  else
    tx_out<4k+2j> = cB<543-2k-j>
    tx_out<4k+2j+1> = cA<543-2k-j>
```

Interleaved FEC Tx – Symbol distribution

Once the data has been Reed-Solomon encoded and interleaved, it shall be distributed to 4 FEC lanes, one 10-bit symbol at a time, from the lowest to the highest FEC lane. The first bit transmitted from each 10-bit symbol is bit 0.

```
tx_out<0>     = cA<543>
tx_out<1>     = cB<543>
tx_out<2>     = cA<542>
tx_out<3>     = cB<542>
tx_out<4>     = cB<541>
tx_out<5>     = cA<541>
tx_out<6>     = cB<540>
tx_out<7>     = cA<540>
tx_out<8>     = cA<539>
tx_out<9>     = cB<539>
tx_out<10>    = cA<538>
tx_out<11>    = cB<538>
tx_out<12>    = cB<537>
tx_out<13>    = cA<537>
tx_out<14>    = cB<536>
tx_out<15>    = cA<536>
<SNIP>
```

```
<SNIP>
tx_out<1080>  = cA<3>
tx_out<1081>  = cB<3>
tx_out<1082>  = cA<2>
tx_out<1083>  = cB<2>
tx_out<1084>  = cB<1>
tx_out<1085>  = cA<1>
tx_out<1086>  = cB<0>
tx_out<1087>  = cA<0>
```

P802.3ck – Interleaved FEC Rx – Changes from CL91

- **Assume new Clause 300 for Interleaved FEC**
  - 300.5.3 Receive function
- **Sections that would be different from Clause 91 are following:**
  - 300.5.3.1 Alignment lock and deskew
    - Based on 119.2.5.1 Alignment lock and deskew
  - 300.5.3.2 Lane reorder and de-interleave
    - Based on 119.2.5.2 Lane reorder and de-interleave
  - 300.5.3.3 Reed-Solomon decoder
    - Same as 119.2.5.3 Reed-Solomon decoder
  - 300.5.3.4 Post FEC interleave
    - Same as 119.2.5.4 Post FEC interleave
  - 300.5.3.5 Alignment marker removal
    - Based on 119.2.5.5 Alignment marker removal

P802.3ck – Interleaved FEC Rx – CL91-based functions

- **Sections that could be directly used from Clause 91 are following:**
  - 300.5.3.6 256B/257B to 64B/66B transcoder
    - Same as 91.5.3.5 256B/257B to 64B/66B transcoder
  - 300.5.3.7 Block distribution
    - Same as 91.5.3.6 Block distribution
  - 300.5.3.8 Alignment marker mapping and insertion
    - Same as 91.5.3.7 Alignment marker mapping and insertion

Thank You!