Improving 800GBASE-ER PCS PTP accuracy (supporting comments 254-256, 302, 303, 356, 457, 458)

Tom Huber, Nokia

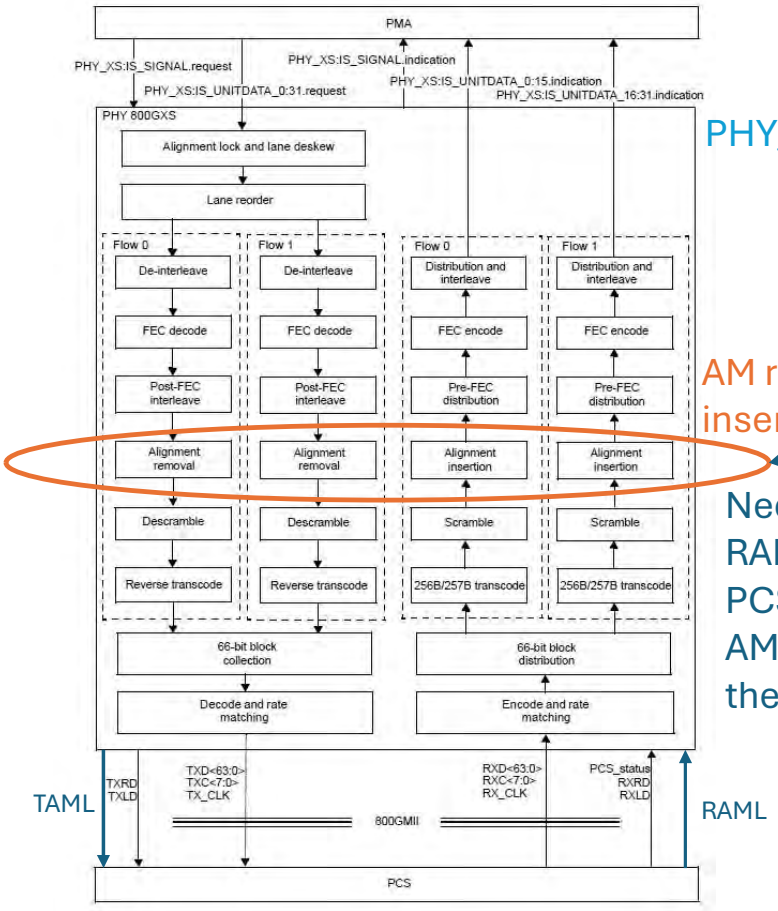
Supporters

- Gary Nicholl, Cisco
- Andras DeKoos, Microchip
- Ulf Parkholm, Ericsson
- Mike Sluyski, Cisco
- Steve Gorshe, Microchip
- Peter Sinn, Alphawave

Problem statement
Comments 254-256, 302, 303, 457, 458

- The baseline in sluyski_3dj_01a_2405 describes new overhead (JC7-JC9) for the 800GBASE-ER1 PCS to convey the location of the alignment markers between the source and sink nodes for cases where the 800GBASE-ER1 PCS connects to an 800GXS
- This baseline cannot be implemented entirely within the 800GBASE-ER1 PCS:
  - The functions for removing and inserting AMs (and the corresponding rate adaptation) are in the PHY_XS rather than the 800GBASE-ER1 PCS
  - The information about where the AMs would be relative to the GMP frame is known only to the 800GBASE-ER1 PCS
- As such, it will be necessary to add new functions to the PHY_XS in clause 171 and new signals between the XS and 800GBASE-ER1 PCS (i.e., to the PCS service interface) in addition to adding functions to the 800GBASE-ER1 PCS
- The PHY_XS and the sublayer to which it connects (in this case, 800GBASE-ER1 PCS) are implemented in the same device, so this is an issue of “how to write the spec in a self-consistent manner”, not a concern with implementation feasibility of the baseline proposal

A picture is worth a thousand (or at least a slide full of) words



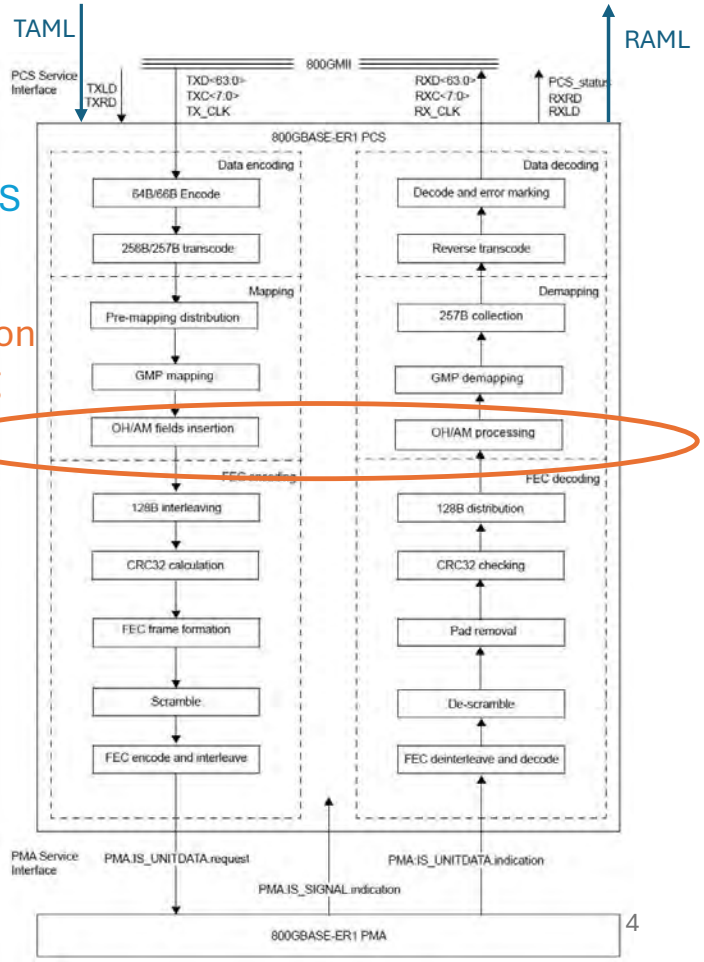
PHY_XS

800GBASE-ER PCS

JC7-JC9 insertion and processing

AM removal and insertion

Need to define signals (TAML and RAML) between the XS and ER1 PCS to enable communication of AM location information between these functions



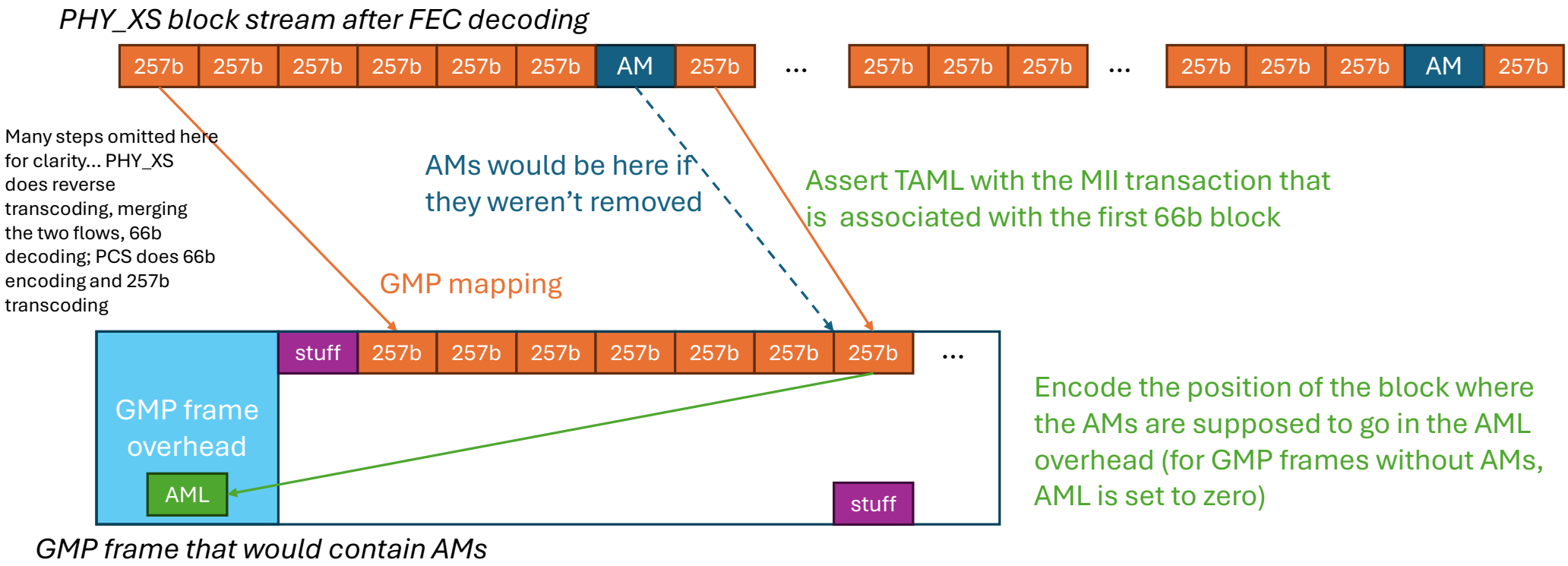
High level summary of the mechanism (1)

- The PCS service interface is driven by MII transactions (64 data bytes plus control information), therefore the signaling between the PHY_XS and 800GBASE-ER1 PCS should be as well
- Transmit direction
  - PHY_XS signals to 800GBASE-ER1 PCS (via the TAML signal) that AMs were removed before this MII transaction
  - The 800GBASE-ER1 PCS encodes each MII transaction as a 66b block; it keeps track of which 66b block is associated with the AMs based on the TAML signal
  - In each GMP frame that the PCS creates:
    - If there are no AMs, TAML is encoded as all zeros
    - If there are AMs, the position of the AMs relative to the GMP frame is computed and encoded in TAML

High level summary of the mechanism (2)

- Receive direction
  - PCS extracts the overhead from each GMP frame:
    - If it is all zeros, there are no AMs associated with that GMP frame
    - If it is not all zeros, the PCS signals to the PHY_XS (via the RAML signal) that AMs are to be inserted before this MII transaction
  - The PHY_XS by default inserts AMs based on its own count of blocks being transmitted
  - A state machine is needed to synchronize the PHY_XS block counter to the RAML signal
    - If there is never a RAML signal asserted, the PHY_XS works as it always has
    - If RAML is asserted, the PHY_XS aligns its counter to RAML
    - Once it has locked to RAML, the state machine needs to allow a small amount of jitter in the RAML signal due to possible Idle insertion/deletion by the PCS or XS
      - In practice this won't happen, this is an artifact of how the XS is defined

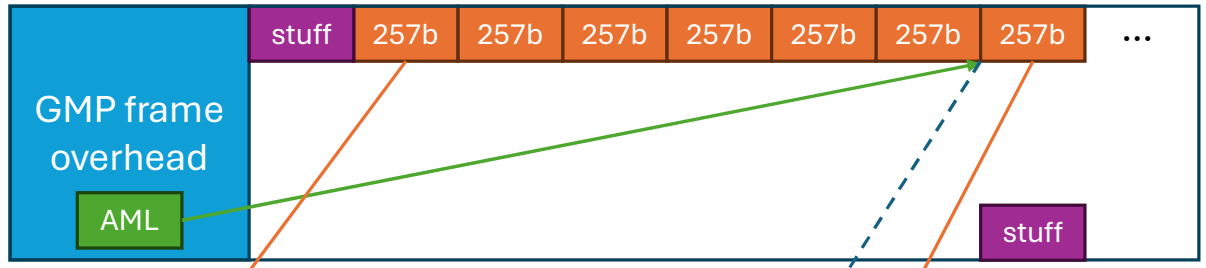
Another picture worth a few slides of words… Transmit direction



The figure shows the case where the AMs happened to precede a 66b block that was first in a 257b block. In practice this will always be the case. In theory, the 66b block could be elsewhere in the 257b block if the PHY_XS or PCS does any rate adaptation.

Another picture worth a few slides of words… Receive direction

GMP frame that would contain AMs



AMs are supposed to go before the first 66b block of this 257b block, so assert RAML with that MII transaction

GMP de-mapping

As in the transmit figure, many steps omitted here for clarity

PHY_XS block stream AM insertion

Insert AMs before the block where RAML is asserted

Interworking considerations

- Neither source nor sink supports enhanced PTP accuracy
  - The source is transmitting zeros in the overhead, the sink does not process the overhead and inserts AMs based on its own block count
  - The link works without enhanced PTP accuracy
- Source supports enhanced PTP accuracy, sink does not
  - The source inserts the overhead as described, the sink does not process it and inserts AMs based on its own block count
  - The link works without enhanced PTP accuracy
- Source does not support enhanced PTP accuracy, sink does
  - The source always transmits zeros in the overhead, the sink extracts that and never asserts RAML, so it inserts AMs on its own block count
  - The link works without enhanced PTP accuracy

Naming of the new overhead in the PCS frame

- While the baseline uses the names JC7-JC9, this is awkward for two reasons:
  - These bytes carry information about the AM location, which is not related to the justification control information in JC1-JC6 that is used for mapping the Ethernet signal into the 800GBASE-ER1 PCS frame (aka the FlexO-8e frame defined by ITU-T)
  - The 3 bytes compose a single field, in the order JC8-JC7-JC9, which is awkward to describe
- It would be better to use a different nomenclature that better identifies the purpose of the overhead and puts the bytes in order, e.g., AML1, AML2, AML3

High level modifications to clause 171 (to be implemented with editorial license)

- In 171.1.1, modify the 4th bullet to note the exception (when connected to an 800GBASE-ER1 PCS, there is an additional function)
- In 171.3
  - Add a new exception for the new functions when connected to an 800GBASE-ER PCS
  - Update figure 171-2 to include the new function and signals at the service interface to the 800GBASE-ER1 PCS (graphically indicating that the new function and service interface component are conditional)
- In 171.3.3
  - Define a new signal for the AM location information
- Insert new 171.6a describing the AM location function (extraction of the information and mapping to the new signal for the tx, and taking the information from the signal and using it to control AM insertion for the rx)

High level modifications to clause 186.2 (to be implemented with editorial license)

- Update figure 186-3 to include the new TAML signal at the PCS service interface (and graphically indicate that it is only present when connecting to a PHY_XS)
- Update figure 186-6 to include AML1-AML3
- Add text in 186.2.4.6.10 to describe the use of the TAML signal to determine what to put in AML1-AML3 of each GMP frame:
  - GMP frames without AMs have all zeros in AML1-AML3
  - GMP frames with AMs have the position of the AMs encoded
- Add text in 186.2.5.6.5 to describe extraction of the AML1-AML3 information and how it determines the assertion of the RAML signal

Should the new functions be mandatory? Comment 356

- Comment 356 proposes making the use of the enhanced PTP accuracy feature mandatory
- There is some appeal to that:
  - Functions that can be turned on or off, but should be set the same way in both source and sink nodes can create interoperability issues
    - But per slide 9, that is not really a concern here
  - The payload type in the GMP frame overhead would have a more consistent meaning if the enhanced PTP overhead is always present
    - But the OH can be defined as “always used” without making the feature the overhead supports mandatory; if the feature is off, the overhead is populated with all zeros in the transmitter and ignored by the receiver

Counterarguments

- Not all applications for 20 km or 40 km links require the same degree of PTP accuracy that the MOPA application requires; a vendor shouldn't be forced to add the feature if they only target markets that don't need it
- As was shown on slide 9, there are no interworking concerns if the user misconfigures the two endpoints
  - The only issue is that they won't get the desired PTP accuracy, so if configurability is allowed, it might be useful to have some way to detect a misconfiguration (e.g., if you're configured to use the feature, and you never see RAML asserted, the other end must not be using the feature, so an alarm could be raised)
- Some implementations will support both 800GBASE-ER1 and OIF 800ZR (which does not currently use the feature), so it will be configurable; the question is how it is configurable:
  - Are there OIF vs 802.3 modes for the module, with the feature always on in 802.3 mode?
  - Is the feature configurable within the context of 802.3?

PHY_XS perspective

- The new functions in the PHY_XS and the new TAML/RAML signals between the XS and 800GBASE-ER1 PCS are effectively optional since they are not relevant when the PHY_XS is connected to anything other than the 800GBASE-ER1 PCS
  - It would be possible to specify the PHY_XS and new signals in a way that has them always present (no harm in always generating the AM location signal, and the TAML/RAML would be unconnected if the PCS is not an 800GBASE-ER1 PCS), but it's not clear that is what we should do
  - It seems 'cleaner' to specify the PHY_XS such that the new functions and signals are only active if the PHY_XS is connected to a PCS that could use them (such that an implementer not concerned with 800GBASE-ER1 can safely ignore those features/signals)

800GBASE-ER1 PCS perspective

- The 800GBASE-ER1 PCS in practice will probably always use an XS, but it is not required to do so in the standard
  - Moreover, the two ends of a link are not required to both use an XS or not use an XS
  - The new functions and TAML/RAML signals must be specified in a way that allows for any valid combination of implementation choices on the two ends of the link
- It will be necessary to specify the AML1-AML3 insertion in a way that is harmless when there is no XS present, so effectively it is ‘mandatory’ to put a valid value in the OH, but ‘optional’ for that value to be information that will improve PTP accuracy

What to do about mandatory vs. optional

- Don’t explicitly make the AM location function mandatory or optional; the function can be defined in a way that allows interworking when one end supports it and the other does not
- Pick a direction with respect to configuration:
  1. Leave it up to implementations to decide if they want to allow the function to be enabled/disabled (and if they want to report mismatches in configuration or not)
  2. Explicitly provide MDIO registers to enable configuration within the 800GBASE-ER1 PCS (and use those to also control the behavior in the PHY_XS, since it is implemented in the same device)