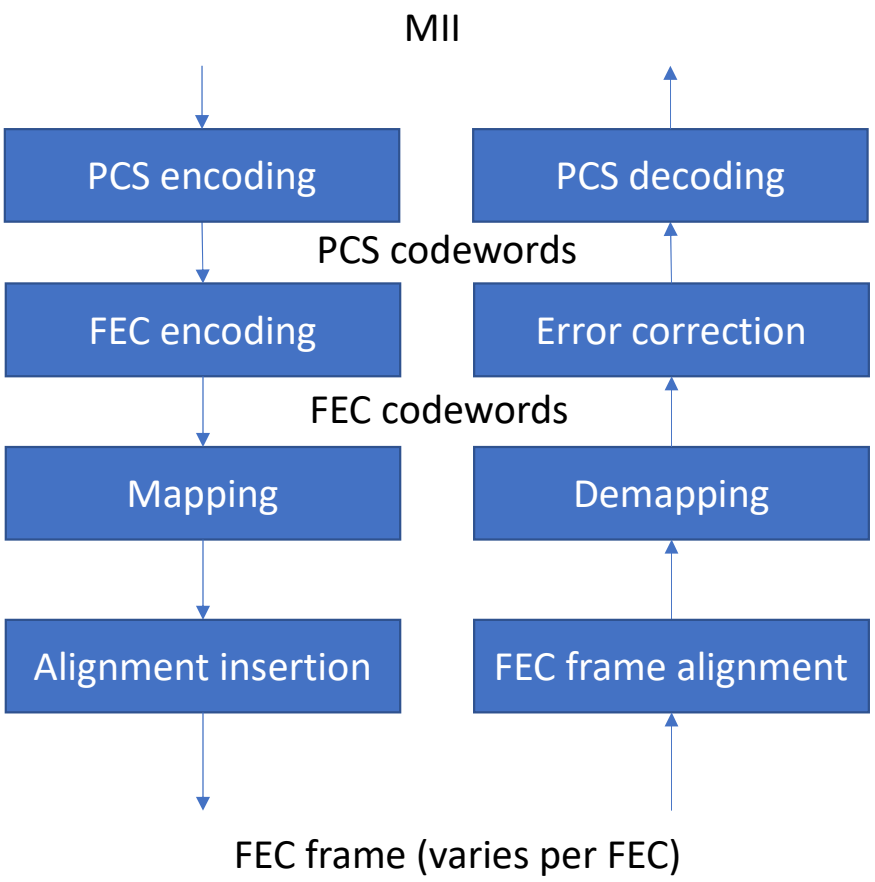
Describing the FEC frame and rate adaption for 800G and 1.6T PCS

Tom Huber, Nokia

Adding FEC to a PCS blockstream

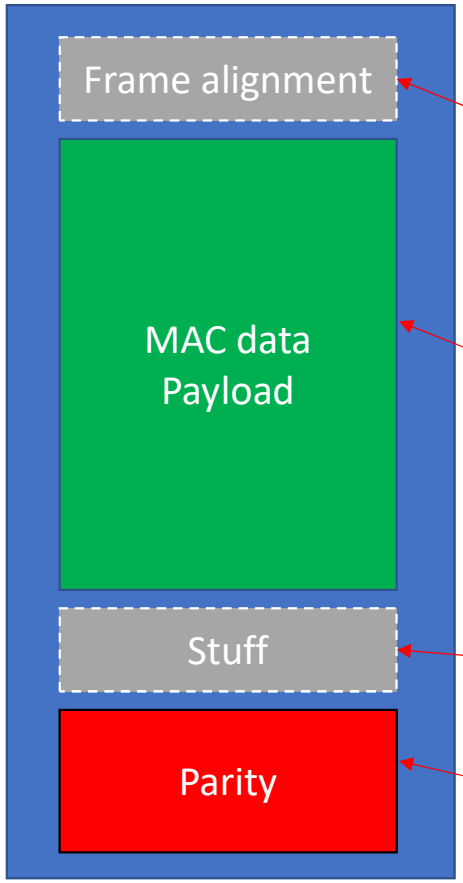
- PCS coding provides a way to reliably locate the start of a MAC frame
- FEC operates on a message consisting of a set of symbols and adds additional symbols to create a FEC codeword
  - The symbol used as the basis for constructing FEC codewords depends on the FEC code, and in general is unrelated to the PCS codeword symbols
    - E.g., with RS(544,514), we have 257b PCS codewords, but the FEC is based on 10-bit symbols; a FEC codeword has 514 10-bit symbols of MAC data and 30 parity symbols
- A FEC frame is needed to reliably locate the start of FEC codewords
  - For RS(544,514), the PCS alignment markers provide the FEC frame based on the assumption that rate adaptation was done based on Idle insertion/deletion in the PCS coding process and the convenience that an integer number of codewords fit in the FEC frame, so the mapping of FEC codewords to FEC frame can be synchronous (and thus finding the AMs means finding alignment to FEC codewords)
  - In the general case, the properties that are exploited for RS(544,514) will not hold for other FECs
    - There will need to be a mechanism for finding the FEC frame that is unrelated to the PCS lane alignment markers
    - The mapping of payload to FEC frame will be asynchronous, with both FEC codewords and stuff words in the payload area of the FEC frame and the mapping process providing the rate compensation
    - The number and location of stuff words may vary from one FEC frame to the next, so the encoded MAC payload is said to ‘float’ in the FEC frame

Generic illustration of PCS and FEC processing



- Rate adaptation is needed to compensate for clock differences between the MAC and PHY
  - PHY adds coding and other OH to the MAC signal (i.e., a fixed/known increase)
  - MAC and PHY may be differing clock domains in some designs
- Where should rate adaptation occur?
  - With RS(544,514), it is done in the PCS encoding/decoding via Idle insertion/deletion, and the mapping into the FEC frame is synchronous
  - With any other FEC, the mapping to the FEC frame will be asynchronous, so rate adaptation can be done via that mapping process, potentially making Idle-based rate adaptation unnecessary

Conceptual illustration of a generic FEC frame



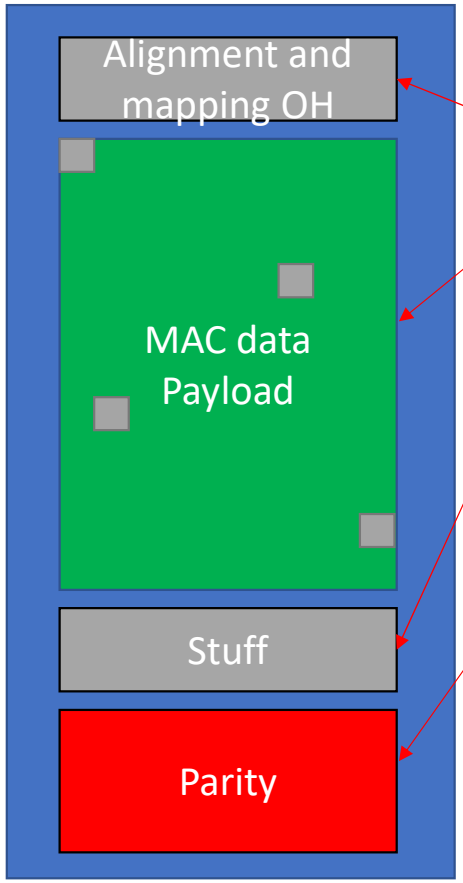
There must be a mechanism to find the start of FEC codewords and to deskew across lanes that form the FEC codeword in the case where multiple lanes are used

MAC frame data is encoded in this area. The coding must allow locating the start/end of MAC frames and signaling LF/RF when the link is down, indicating /E/ control characters, and /I/ or /LPI/ between packets

A variable amount of stuff bits or bytes may exist in the frame that can compensate MAC/PHY rate differences

The redundancy that the FEC adds is in this area.

Illustration of SC-FEC frame from 802.3ct



64 bytes (first 16 columns of each of 4 rows) are frame alignment and mapping overhead

15200 bytes of MAC frame data and variable stuffing from GMP

32 bytes of fixed stuff (a vestige of the frame originally being designed for OTN and reused for 100GBASE-ZR)

1024 bytes of parity information

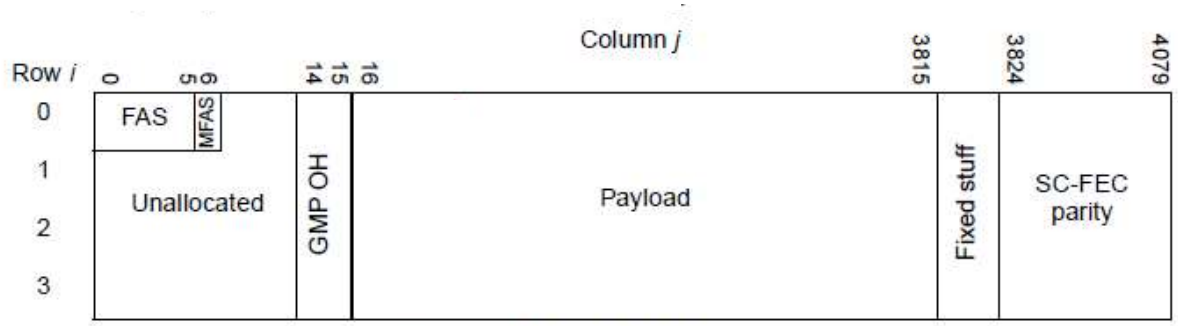
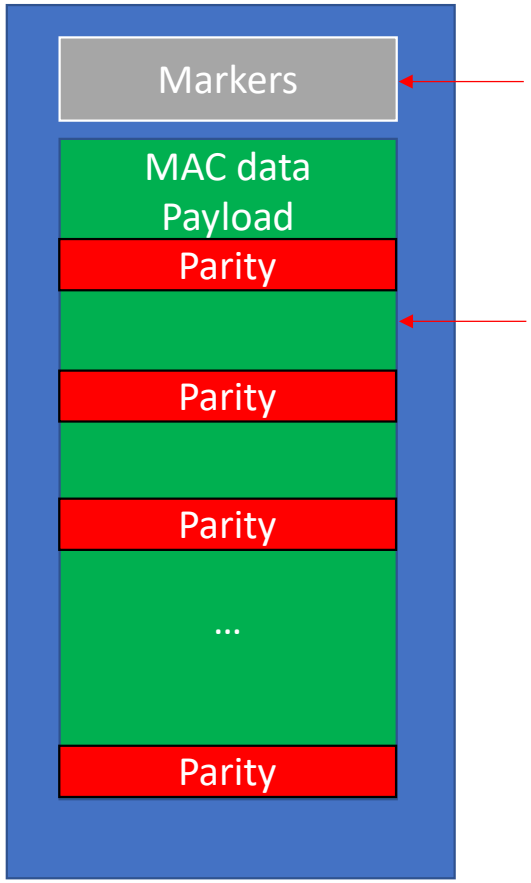


Figure 153–3—SC-FEC frame

Illustration of RS(544,514) FEC frame per clause 119

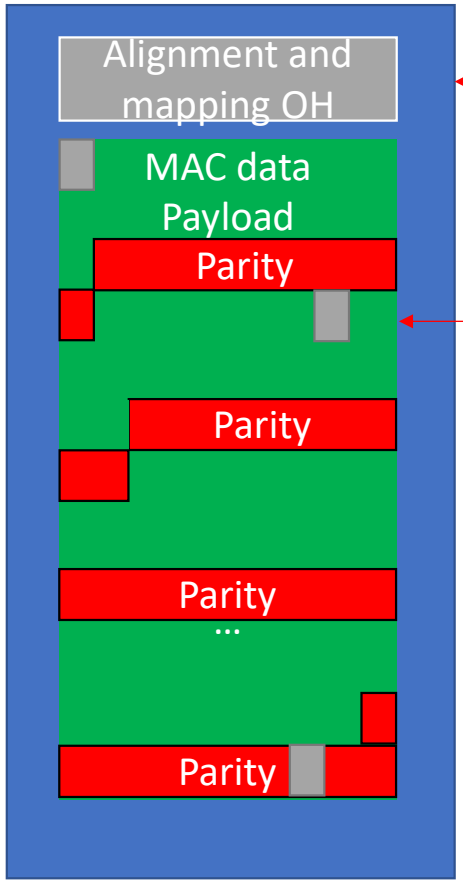


Alignment markers are inserted at the start of each set of 4096 FEC codewords for 200G and every 8192 FEC codewords for 400G. AMs are padded to be an integer number of 257b blocks. The AMs serve as the FEC frame alignment, allowing the Rx to correctly delineate FEC codewords

FEC codewords consist of 514 10-bit symbols (corresponding to twenty 257b blocks) and thirty 10-bit parity symbols

There are no stuff words in the payload area because rate adaption was done in the PCS coding via Idle insertion/deletion (i.e., the mapping of codewords into the payload area is synchronous). However, it would be possible to use an asynchronous stuffing mechanism instead (i.e., do rate adaptation as part of the mapping into the FEC frame, rather than as part of the PCS coding). This would require some overhead space for the mapping OH (e.g., GMP overhead requires ~6 bytes of overhead) that would presumably be available in the AMs.

Illustration of RS(544,514) FEC frame with GMP mapping



Frame size based on desired ratio of alignment and mapping OH to payload area (e.g., for 200G, the frame is 81920 blocks, with 4 of those carrying AMs). Payload bit rate must be rate large enough to accommodate clock differences between MAC and PHY clocks.

Payload area divided into words of a convenient size. Mapping OH determines which words are stuff and which are payload. Stuff blocks for rate adaptation are distributed throughout the payload. No Idle-based rate adaptation in the PCS coding is necessary.

This is only an example - other possibilities exist for designing this FEC frame (e.g., separate MAC data payload and parity areas like SC-FEC frame uses, etc.)

Alignment markers

- In 200G/400G, the alignment markers serve two purposes – PCS lane alignment and FEC frame alignment
  - This leads to a complex description of how to create the AMs such that the right values appear on the right lanes after distribution to FEC lanes is performed, along with a need to exclude the AMs from other processes like FEC and scrambling
  - The AM structure was inherited from 100GBASE-R, where use of FEC was optional for the initial PHYs
- In the more general case, a FEC frame can have an alignment mechanism that does not depend on the PCS lane AMs, potentially enabling simpler PCS lane AMs that are processed post-FEC

Proposal for FEC, rate adaptation, and AMs

- All 800G and 1.6T PCS should be specified based on the idea of mapping the PCS-coded payload into a FEC frame
  - For interfaces using RS(544,514) FEC, the existing behavior of rate adaptation in the PCS coding and synchronous mapping into the RS(544,514) FEC frame that uses AMs for frame alignment could be retained, if that is desirable, even if the description methodology is changed to more clearly describe the existence of the FEC frame
  - Other interfaces using other FECs, the PCS could avoid Idle-based rate compensation and rely on the mapping into the FEC frame
  - Alternatively, asynchronous mapping into all FEC frames could be specified, and Idle-based rate adaptation could be eliminated for all 800G and 1.6T PHYs
- FEC frame alignment and PCS lane alignment should be separated, with simpler AMs (processed post-FEC) used for PCS lane alignment

OTN mapping reference point

- Since all PHYs will use FEC, the OTN mapping reference should be specified as 257b blocks rather than 66b blocks
- As a consequence, there is no need for PCS process diagrams to show 66b coding and 257b transcoding as separate steps
  - Diagrams can simply say ‘257b coding/257b decoding’.
  - The text description of that process could still point to clause 119 (which points to clause 82 for 66b coding and describes the transcoding to 257b) to avoid the risk of specifying a direct-to-257b coding in a manner that is not consistent with clause 119
    - Gives implementations more freedom since there is no need to expose the 66b coding
    - May require different description of Idle-based rate adaptation if that is retained for RS(544,514) FEC

Thanks!