Further Consideration on 100BASE-T1L PMA Training

Tingting Zhang

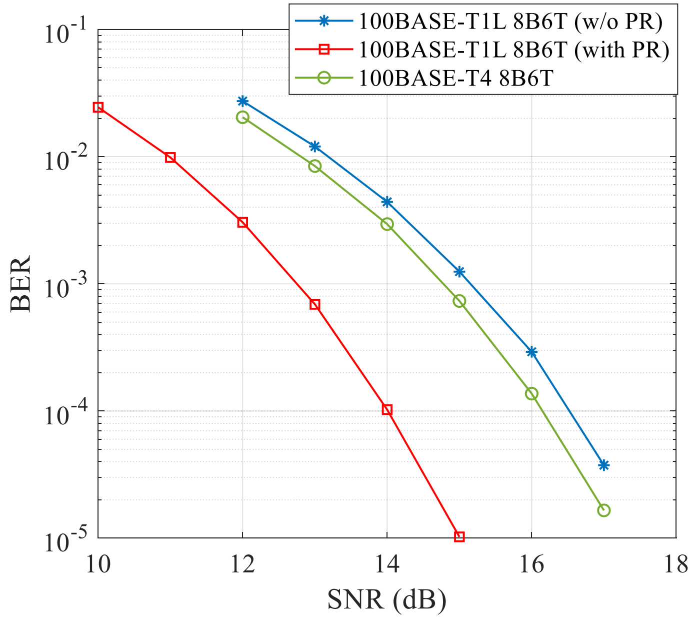
Huawei Technologies

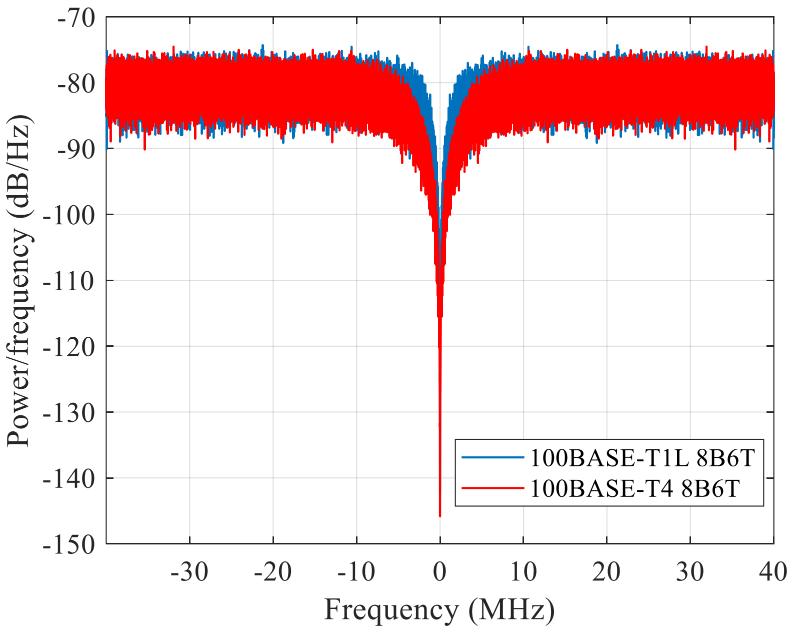
Introduction

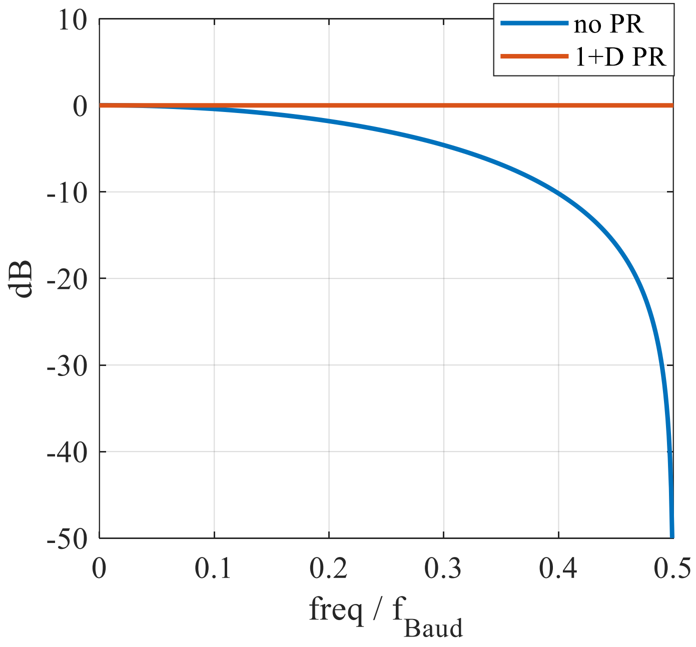
- Data-mode PCS is almost in place. PMA training has achieved consensus on using training frames with indicators and each training frame is composed of 16 partial PHY frames.
- PMA training with different formats and line coding (Riesco_3dg_01a_09172024.pdf, Tingting_3dg_02a_16_09_2024.pdf) has been discussed.
  - The main concerns of PAM3 include training efficiency and its impact on partial response (PR) equalization. In terms of PAM2, spectrum characteristics are more cared about.
- In this presentation, we consider PMA training based on PAM2 and two coding options further. The spectrum has no spur and is close enough to that of 8B6T PAM3.

PAM3 impact on the PR equalization

- In the case of no partial response (PR), 100BASE-T4 8B6T outperforms 100BASE-T1L 8B6T, in terms of both BER performance and low-frequency suppression. 100BASE-T1L should support 8B6T with PR so that the coding gain can be achieved.
- The response of 1+D PR is significantly different from that without PR. If PMA training utilizes 8B6T, it is very difficult for the PR equalizer to operate stably, due to the significant difference between the target channel response.







PAM2 for PMA Training

- PAM2 has ~4dB lower SNR requirement than PAM3 and is more robust to noise and transmission impairments.
- Transmitting PAM2 during PMA training, the equalizer can operate in either PAM2 or PR (i.e. blind PAM3) mode easily.
- As long as the spectrum of the transmitted PAM2 signal is close enough to that of 8B6T PAM3, the risk of training failure when switching to 8B6T with PR is minimized. Consequently, the shining point of 100BASE-T1L 8B6T (i.e. coding gain) can be achieved.
- Overall, PAM2 is a better option for PMA training.

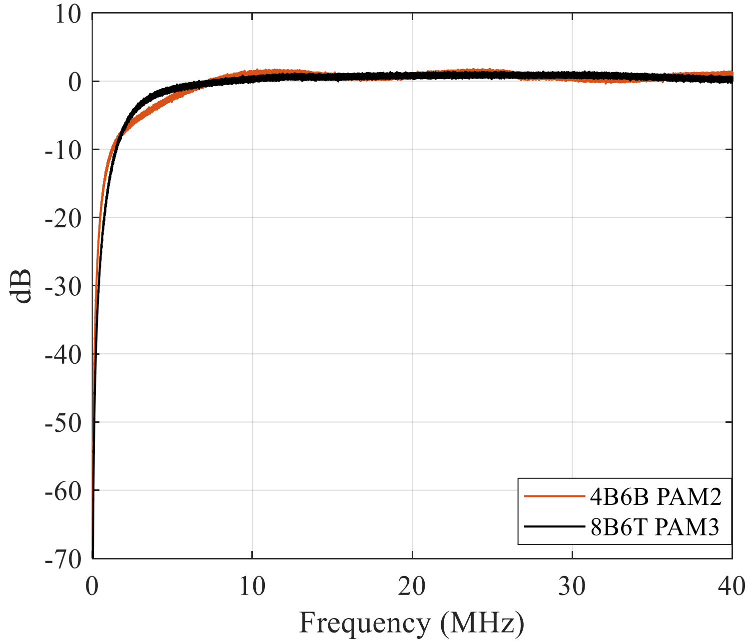
PAM2 Line Coding Option 1: 4B6B —— Code-groups

- There are 64 PAM2 6-tuples in total, and 42 non-negative disparity (NND) ones.
- Each of the 16 4-bit values input to the encoder is associated with one of the NND 6-tuples shown in the right table:
  - 10 6-tuples with disparity 0
  - 5 6-tuples with disparity 2
  - 1 6-tuples with disparity 4
- Each NND 6-tuple has a unique complementary code-group, which is generated by negating each element of the 6-tuple and not in the table.
  - To avoid spectrum spurs, pseudo random bit Sg_n is used to determine whether to negate NND 6-tuples **including the 0-disparity ones**. In 8B6T encoding, Sg_n only decides the sign of the positive 6-tuples.

4 bits input	**NND PAM2 6-tuples**					
0000	-1	1	-1	1	-1	1
0001	-1	-1	1	1	-1	1
0010	-1	1	1	1	1	1
0011	1	-1	1	-1	1	1
0100	-1	1	-1	1	1	-1
0101	1	1	1	-1	1	-1
0110	-1	1	1	-1	-1	1
0111	-1	1	-1	-1	1	1
1000	1	1	1	1	-1	-1
1001	-1	-1	-1	1	1	1
1010	-1	-1	1	-1	1	1
1011	-1	-1	1	1	1	-1
1100	1	1	-1	1	1	-1
1101	-1	1	1	-1	1	-1
1110	-1	1	1	1	-1	-1
1111	1	1	-1	-1	1	1

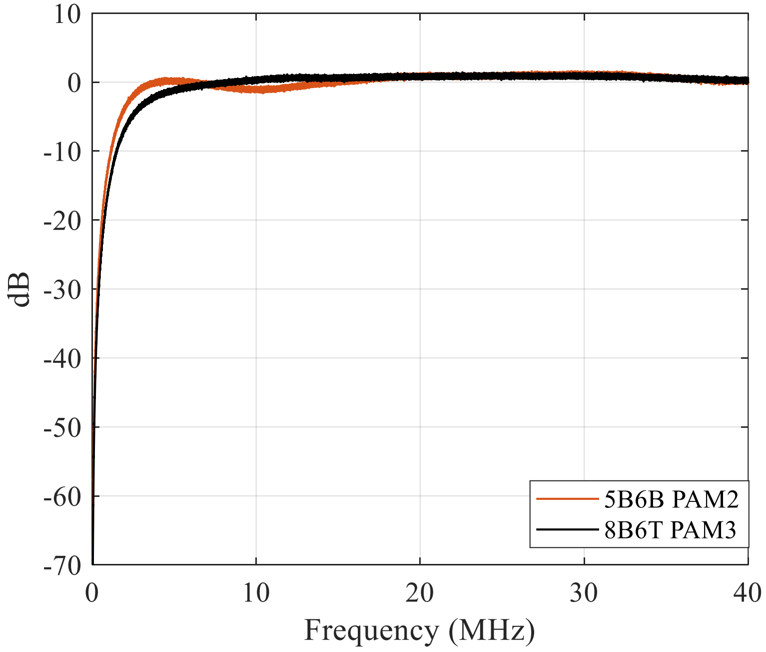
PAM2 Line Coding Option 1: 4B6B —— Control Running Disparity

- The running disparity (RD) at the transmitter is controlled as follows:
  - If both RD and the disparity change of the 6-tuple associated with the 4-bit value are positive, then the 6-tuple is negated before transmission.
  - If RD is zero, **OR** the 4-bit value is associated with a **6-tuple with zero disparity**, then a pseudo random bit Sg_n determines whether to negate the 6-tuple before transmission.
  - RD is recomputed after transmission of each 6-tuple.
- The spectrum of 4B6B PAM2 has no spurs and is close enough to that of 8B6T PAM3. The equalizer can easily fine adjust once PAM3 starts.



PAM2 Line Coding Option 2: 5B6B

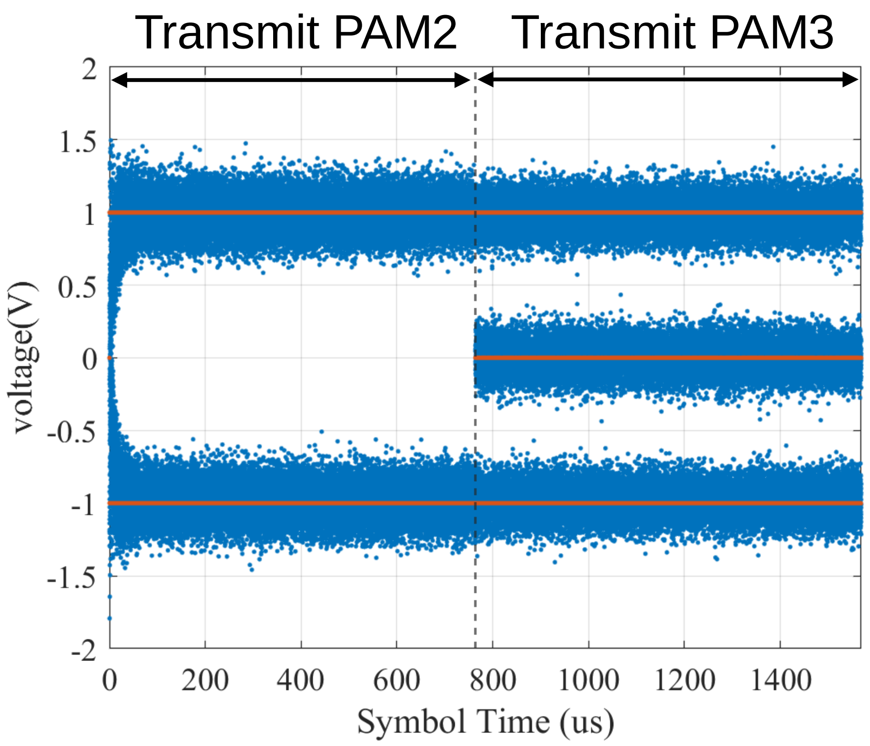
- Each of the 32 5-bit values from the encoder is associated with one of the NND 6-tuples shown in the right table, which consists of 11, 15, and 6 6-tuples with disparity changes of 0, 2, and 4 respectively.
- The RD control rule for 5B6B is the same as 8B6T. **The zero-disparity 6-tuples are not negated.**
- The PSD of 5B6B PAM2 is close enough to that of 8B6T PAM3. The target channel response is approximately the same, and the equalizer can easily fine adjust once PAM3 starts.

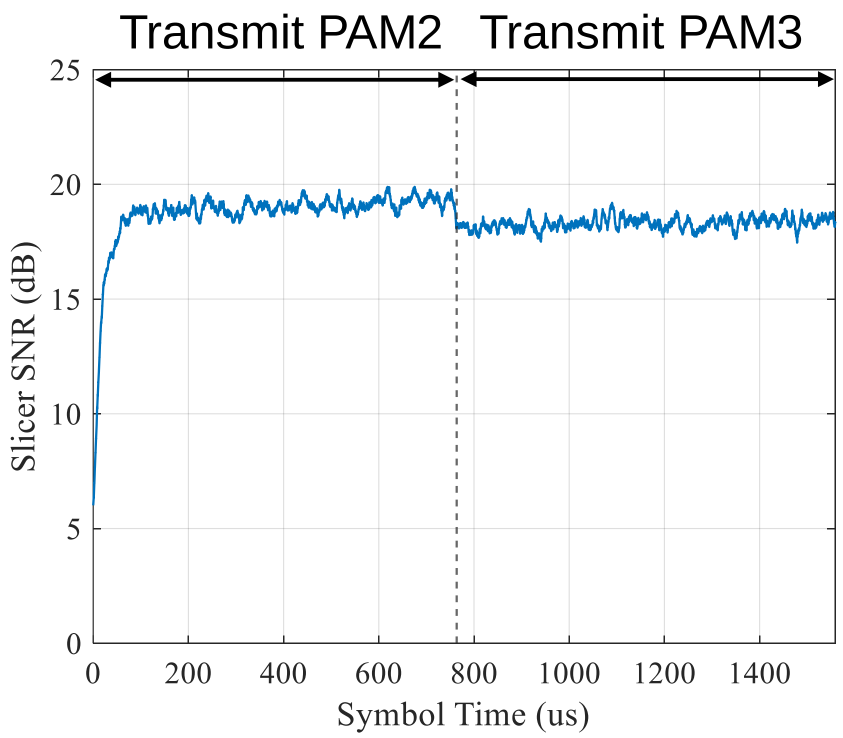


5 bits input	NND PAM2 6-tuples					
00000	1	-1	1	1	1	1
00001	1	-1	-1	-1	1	1
00010	1	1	1	1	-1	1
00011	1	1	-1	1	1	-1
00100	-1	1	1	1	1	1
00101	1	1	-1	-1	1	1
00110	1	1	-1	1	1	1
00111	1	-1	-1	1	-1	1
01000	1	1	1	1	-1	-1
01001	-1	-1	1	1	1	1
01010	1	1	1	-1	1	-1
01011	-1	1	-1	1	1	-1
01100	-1	1	1	-1	1	-1
01101	1	-1	1	1	1	-1
01110	-1	1	1	1	-1	1
01111	1	-1	1	1	-1	-1
10000	1	1	-1	1	-1	1
10001	-1	1	1	1	1	-1
10010	1	1	1	1	1	-1
10011	1	-1	1	-1	1	1
10100	1	1	1	-1	1	1
10101	-1	1	-1	-1	1	1
10110	1	-1	-1	1	1	1
10111	-1	1	1	1	-1	-1
11000	1	-1	1	-1	-1	1
11001	-1	1	-1	1	-1	1
11010	-1	1	1	-1	1	1
11011	1	1	1	-1	-1	1
11100	-1	1	-1	1	1	1
11101	-1	1	1	-1	-1	1
11110	1	-1	-1	1	1	-1
11111	1	-1	1	1	-1	1

Equalizer in the traditional mode – 500m, 7mVrms noise

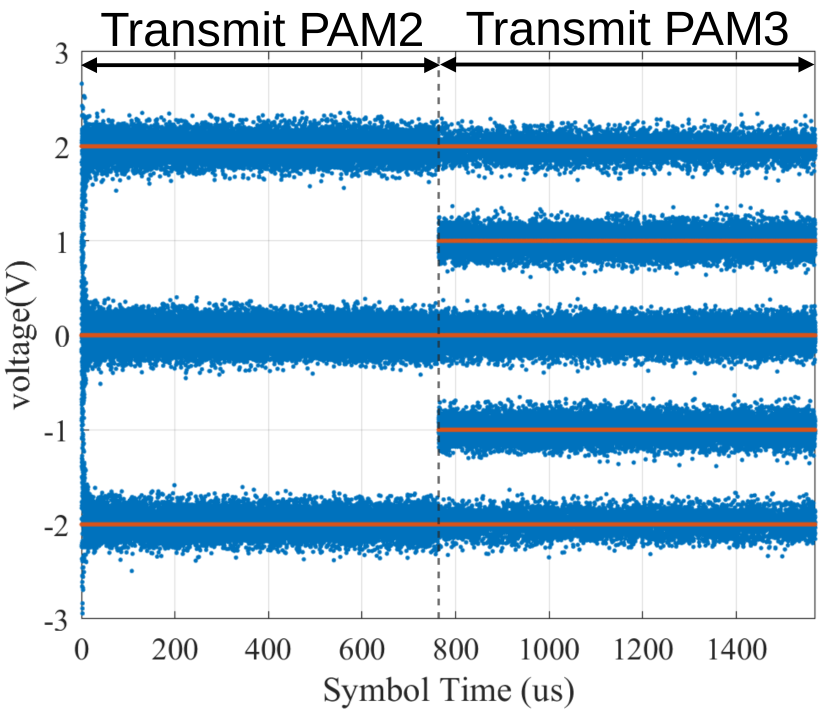
- Time-domain simulation model refers to Tingting_3dg_01a_16_07_2024.pdf
- In the case of 4B6B PAM2 transmission over 500m channel, the FFE/DFE equalizer in the normal mode converges within 200us. Switching from 4B6B PAM2 to 8B6T PAM3, the equalizer works very stable.
- The SNR convergence speed for PAM2 is 10x faster than PAM3 (Murray_3dg_01_05132024.pdf).

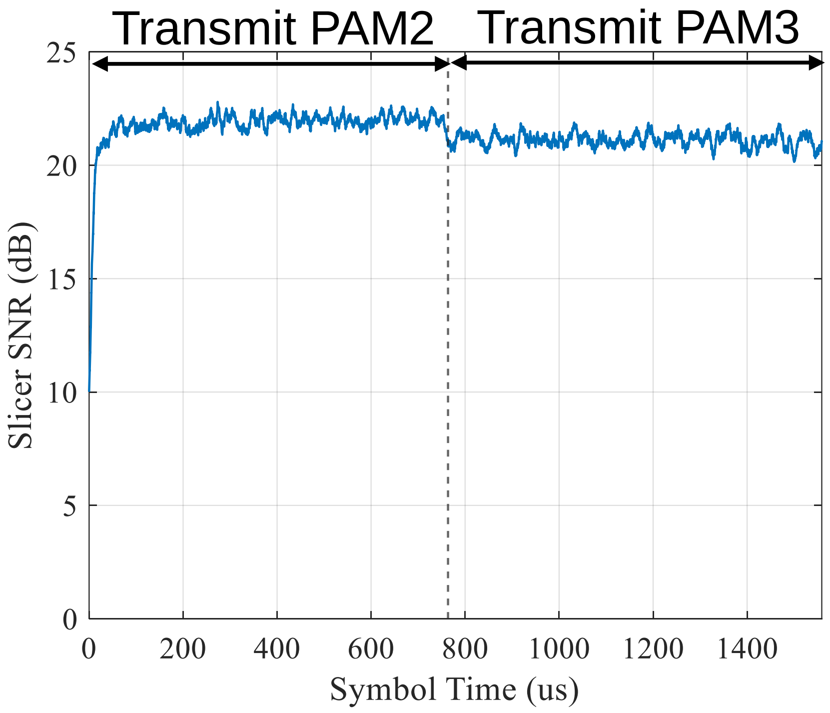




Equalizer in the PR mode – 500m, 7mVrms noise

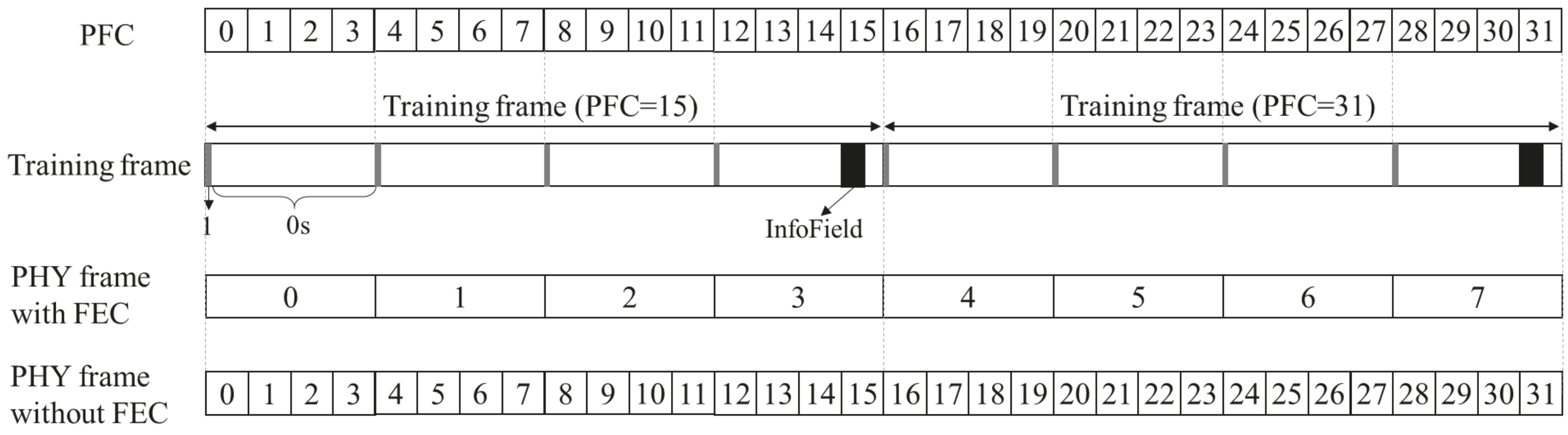
- Only change the FFE/DFE equalizer mode to the PR mode. This means blind PAM3 equalization for the transmitted PAM2 signal and blind PAM5 equalization for PAM3 signal.
- Transmitting PAM2 signal also ensures fast PR equalizer convergence (less than 200us).
- Switching from 4B6B PAM2 to 8B6T PAM3, the PR equalizer works very stable.





PMA Training Frame

- Each training frame is composed of 16 partial PHY frames. Each partial PHY frame has 128 bits. InfoField is embedded in the last partial frame.
- The scrambler and training frame synchronization impacts training efficiency. The premise for this is to correctly delimit the 6-tuples. The 1st bit of every four partial frames is inverted, and the inverted bit positions are aligned with the delimiters and FEC frame.
  - With the knowledge of the delimiter coding rule, the receiver can reliably and efficiently get the 6-tuple boundary, similar as the alignment marker locking.



PMA Training Scrambler

- The training frame uses the same scrambler bits $Sy_n[3:0]$ as data mode, regardless of the line coding. In the case of 4B6B, the scrambled bits $ST_n[3:0]$ and pseudo random bit Sg_n are used to generate PAM2 6-tuples. If 5B6B is selected, bits $ST_n[3:0]$, Sg_n, and $Sx_n[0]$ are input to the encoder.

Sg_n is generated as:

$$Sg_n = Scr_n[1] \hat{} Scr_n[5].$$

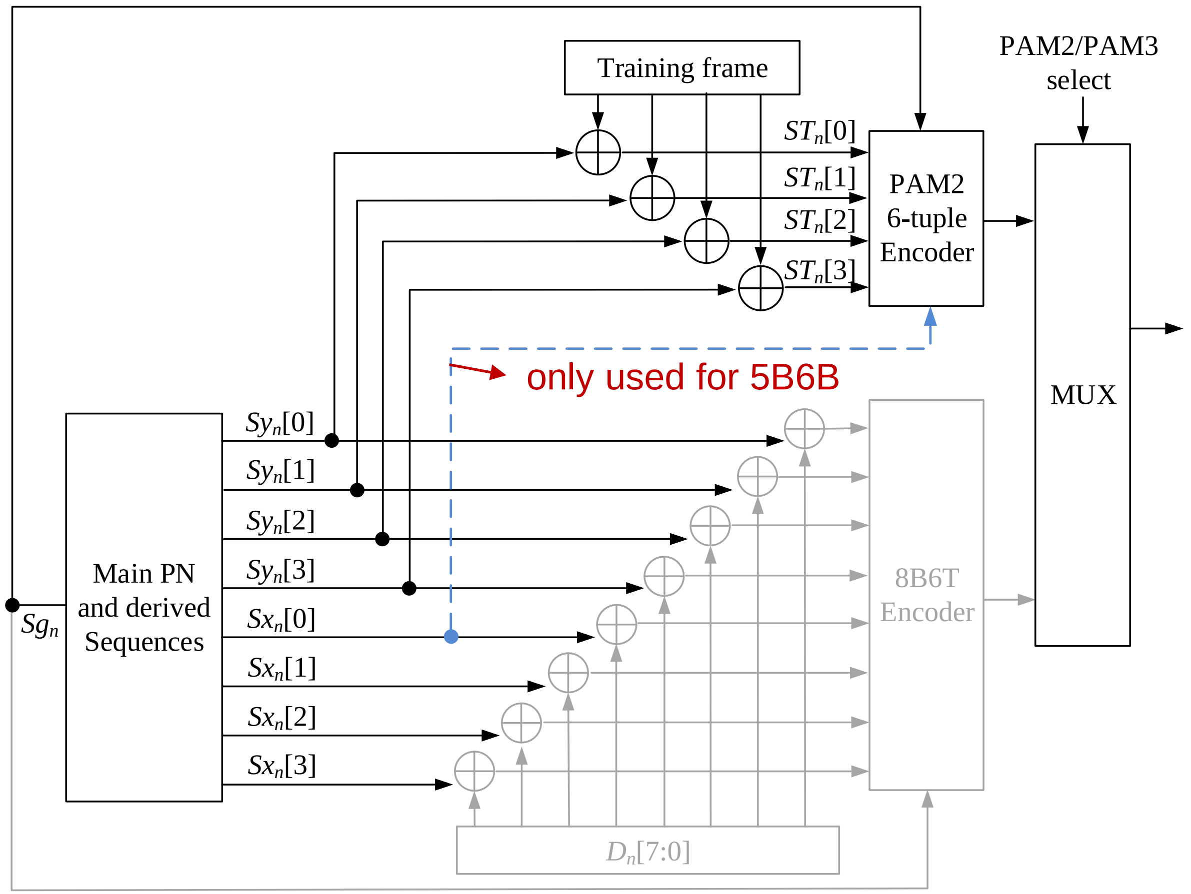
The scrambled bits $ST_n[3:0]$ can be expressed as:

$$ST_n[0] = \begin{cases} Sx_n[0] \oplus \text{InfoField}_{(4n \bmod 128)} & 480 \le (n \bmod 512) \le 503 \\ Sx_n[0] \oplus 1 & \text{else if } (n \bmod 128) = 0 \\ Sx_n[0] & \text{otherwise} \end{cases}$$

$$ST_n[1] = \begin{cases} Sx_n[1] \oplus \text{InfoField}_{(4n+1 \bmod 128)} & 480 \le (n \bmod 512) \le 503 \\ Sx_n[1] & \text{otherwise} \end{cases}$$

$$ST_n[2] = \begin{cases} Sx_n[2] \oplus \text{InfoField}_{(4n+2 \bmod 128)} & 480 \le (n \bmod 512) \le 503 \\ Sx_n[2] & \text{otherwise} \end{cases}$$

$$ST_n[3] = \begin{cases} Sx_n[3] \oplus \text{InfoField}_{(4n+3 \bmod 128)} & 480 \le (n \bmod 512) \le 503 \\ Sx_n[3] & \text{otherwise} \end{cases}$$



PAM2 6-tuple Generation

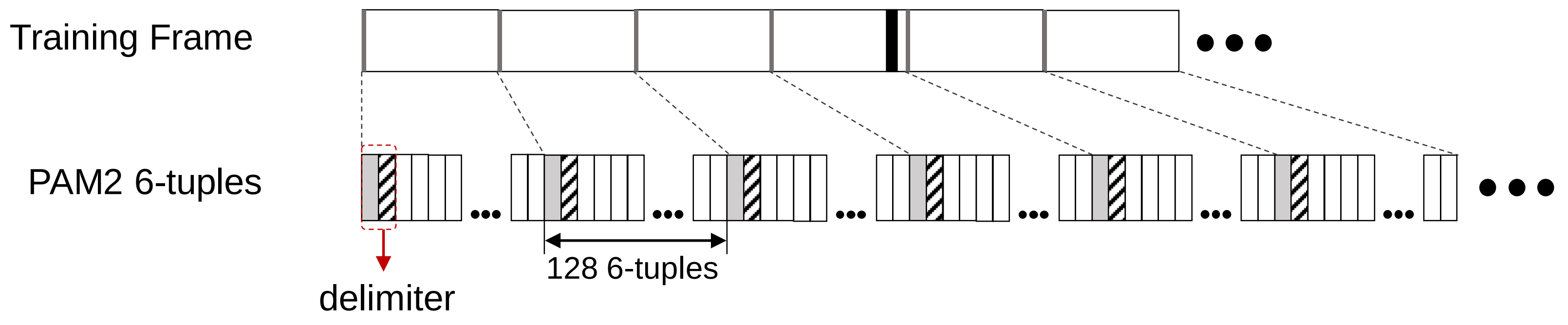
- If 4B6B is used, the PAM2 6-tuple (tx_code_group) is generated as:

$$tx\_code\_group = \begin{cases} delimiter & (n \bmod 128) < 2 \\ Encoder\_4B6B(ST_n[3:0], Sg_n) & \text{otherwise} \end{cases}$$

- Alternatively, if 5B6B is used, the generated PAM2 6-tuple is expressed as:

$$tx\_code\_group = \begin{cases} delimiter & (n \bmod 128) < 2 \\ Encoder\_5B6B(ST_n[3:0], \boldsymbol{Sx_n}[\mathbf{0}], Sg_n) & \text{otherwise} \end{cases}$$

- The delimiter consists of two zero-disparity PAM2 6-tuples (-1, 1, 1, 1, -1, -1) and (-1, 1, 1, -1, -1, 1). To reduce the complexity of delimiter validity checker in the receiver, whether to negate the 2nd 6-tuple before transmission follows the same operation for the 1st one for each delimiter.



6-tuple Synchronization

- Each training frame is encoded to 512 PAM2 6-tuples, including 4 equally-spaced delimiters.
- With the knowledge of the delimiter coding rule, the receiver can reliably get the 6-tuple boundary in a similar way as the alignment marker locking.
  - The received 12-bit block is regarded as valid delimiter, if all 12 bits match the corresponding bits of the delimiter.
  - If the number of valid delimiters at expected positions is more than 7 for 1024 received 6-tuples, the 6-tuple boundary is correct and reliable. Otherwise, slip the boundary.
  - In this way, the 6-tuple synchronization can be finished within 76.8us on average for BER less than 10^{-5}. False synchronization occurs once per every $5.888*10^{12}$ years (larger than the universe age).
- Reliable 6-tuple delimiting allows fast scrambler synchronization, as the scrambler can directly use the decoded bits from the 6-tuples after the delimiter for initialization.

Conclusion

- The PAM2 training sequence enables a more robust DSP with 10x faster convergence speed than PAM3 signal. The PAM2 spectrum after 4B6B or 5B6B line coding has no spurs and is close to the 8B6T PAM3 spectrum.

- This allows the equalizer to work stably when the transmitted signal is switched from PAM2 to PAM3, regardless of the operation mode (traditional or PR). Therefore, the risk of training failure is minimized while the coding gain brought by the PR equalization is achieved.

- Either line coding allows the reuse of the data-mode scrambler.

- The delimiters in the transmitted PAM2 signal help the remote receiver reliably recognize the 6-tuple boundary and realize fast scrambler synchronization.

Q & A