# P802.3cy unsatisfied comment I-118

Adee Ran

Cisco

# Comment and response

*Cl* **165** *SC* **165.1.4** *P* **40** *L* **51** # I-118

Ran, Adee — Cisco Systems, Inc.

*Comment Type* **TR** *Comment Status* **R**

"25GBASE-T1 signaling is performed by the PCS generating continuous code-group sequences"

The "continuous code-group sequences" seem to come from multi-pair PHYs. This PHY has a single pair, and uses a sequence of PAM4 symbols (item b in the list following this paragraph).

Also, in 165.3.2.2, P52 L29, and 165.3.2.3, P61 L50.

*SuggestedRemedy*

Change "continuous code-group sequences" to "a sequence of PAM4 symbols".

Change "code-groups" to "symbols" in the other two locations provided in the comment.

*Response* *Response Status* **U**

REJECT.

The term "continuous code-group sequences" is correct and has been used consistently for many PCS using block codes. This PHY uses both PAM4 and PAM2 signalling

# What is a “code group”

**1.4.255 code-group:** For IEEE 802.3, a set of encoded symbols representing encoded data or control information. For 100BASE-T4, a set of six ternary symbols that, when representing data, conveys an octet. For 100BASE-TX and 100BASE-FX, a set of five code-bits that, when representing data, conveys a nibble. For 100BASE-T2, a pair of PAM5×5 symbols that, when representing data, conveys a nibble. For 1000BASE-X, a set of ten bits that, when representing data, conveys an octet. For 1000BASE-T, a vector of four 8B1Q4 coded quinary symbols that, when representing data, conveys an octet. For 100BASE-T1, a set of ternary symbols that, when representing data, conveys three bits, as defined in 96.3. For 10BASE-T1L, a set of three ternary symbols that, when representing data, conveys a nibble, as defined in 146.3. (See IEEE Std 802.3, Clause 23, Clause 24, Clause 32, Clause 36, Clause 40, Clause 96, and Clause 146.)

“six ternary”, “five code-bits”, “a pair of PAM5x5”, “a set of ten bits”, “four 8B1Q4” – in all cases, a code group is more than one symbol of the modulation used; the term has been used mainly for PAM3 and PAM5 where a single symbol does not correspond to a vector of transmitted bits.

PAM4 symbols, which are used in many clauses, are not included in this list and are never referred to as “code groups”.

# 802.3cy uses PAM4 modulation

## 165.3.2.2 PCS Transmit function

(4th and 5th paragraphs)

After mapping the 25GMII transfers to 64B/65B blocks, the subsequent functions of the PCS Transmit process take L groups of 130 64B/65B blocks and append a 10-bit OAM field to each group. This forms the input to an L-interleaved RS-FEC which adds L × 900 parity bits. The resulting L × 9360 bits are then scrambled. These bits are then mapped, two at a time, into a PAM4 symbol. Transmit data-units are sent to the PMA service interface via the PMA_UNITDATA.request primitive.

In each symbol period, when communicating with the PMA, the PCS Transmit generates a PAM4 symbol that is transferred to the PMA via the PMA_UNITDATA.request primitive. The symbol period, T, is one period of the transmitter clock frequency, specified in 165.5.3.6.

A PAM4 symbol is not a code group (per the definition in 1.4.255)

# Where the term “code group” is used in 802.3cy

**165.1.4 Signaling**

25GBASE–T1 signaling is performed by the PCS generating continuous code-group sequences that the PMA transmits over a link segment meeting the specifications of 165.7. The signaling scheme achieves a number of objectives including:

**165.3.2.2 PCS Transmit function**
**(…)**

If a PMA_TXMODE.indication message has the value SEND_Z, PCS Transmit shall pass a vector of zeros at each symbol period to the PMA via the PMA_UNITDATA.request primitive.

If a PMA_TXMODE.indication message has the value SEND_T, PCS Transmit shall generate a sequence ($T_n$) defined in 165.3.5.1 to the PMA via the PMA_UNITDATA.request primitive. These code-groups are used for training mode and only transmit the values {–1, +1}.

**165.3.2.3 PCS Receive function**

The PCS Receive function shall conform to the PCS 64B/65B Receive state diagram in Figure 149–18 and Figure 149–19, and the PCS Receive bit ordering in Figure 165–7 including compliance with the associated state variables as specified in 165.3.7.2.2.

The PCS Receive function accepts received code-groups provided by the PMA Receive function via the parameter rx_symb. The PCS receiver uses knowledge of the encoding rules and PMA training alignment to correctly align the 65B RS-FEC frames. The received PAM4 symbols are demapped and descrambling is performed.

| | | | | | |
|---|---|---|---|---|---|
| PCT6 | If a PMA_TXMODE.indication message has the value SEND_N, the PCS is in the normal mode of operation and the PCS Transmit function uses a 65B coding technique to generate code-groups that represent data or control. | 165.3.2.2 | | M | Yes [ ] |

This does not match the text in 165.3.2.2

These instances correspond to text in clause 149 (see 149.1.4, 149.3.2.2, 149.3.2.3, and 149.11.4.2.1) which also uses PAM4 modulation and does not have code-groups.

The text in Clause 149 may be based on earlier clauses that did use code-groups (such as Clause 96).
Changing Clause 149 text should be done in maintenance.

# Suggested remedy

- In 165.1.4, change “code-group” to “symbol”.
- In 165.3.2.2, change as follows

If a PMA_TXMODE.indication message has the value SEND_T, PCS Transmit shall generate a sequence ($T_n$) defined in 165.3.5.1 to the PMA via the PMA_UNITDATA.request primitive. ~~These code-groups are used for training mode and only transmit the values~~ This sequence is used for training and contains only the symbols {–1, +1}.

- In 165.3.2.3, change “code-groups” to “symbols”.
- In 165.11.4.2.1, change as follows

If a PMA_TXMODE.indication message has the value SEND_N, the PCS is in the normal mode of operation and the PCS Transmit function uses a 65B coding technique specified in 149.3.2.2.4 to ~~generate code-groups that~~ represent data or control.