

# Error Handling for 100BASE-T1L

Jacobo Riesco  
Brian Murray  
Philip Curran

- ▶ At the July meeting the 802.3dg group approved a motion to adopt a constant latency MII to 8N/8N+1 encoding/decoding for 100BASE-T1L
  - See [Lo\\_3dg\\_01a\\_0724](#)
- ▶ At the September meeting a PCS for 100BASE-T1L was presented that included a simpler scheme for the constant latency MII to 8N/8N+1 encoding
  - See [Riesco\\_3dg\\_01a\\_09172024](#)
- ▶ With both of these proposals there is a limitation with respect to error handling compared to previous PHY technologies for speeds  $\leq 1$  Gbit/s and compared to multi-G PHY technologies
  - Note, an update to the 8N/8N+1 encoding scheme was presented to the October Interim meeting to address this issue, see [Riesco\\_3dg\\_02\\_10292024](#)

- ▶ Previous PHY technologies for speeds  $\leq 1$  Gbit/s that do not use  $8N/(8N+1)$  encoding use packet delimiters
  - Delimiters are added/removed by the PHY PCS so that
    - Errors anywhere within a packet are decoded as MII/GMII data reception with errors (RX\_DV = 1, RX\_ER = 1)
    - Errors during IPGs are decoded as MII/GMII false carrier indications (RX\_DV = 0, RX\_ER = 1, RXD = 0xE)
- ▶ Multi-G PHY technologies also use packet delimiters
  - Packet delimiters are defined in the permissible TX/RX encodings of the corresponding Media Independent Interfaces (and mapped in the PCS encoder/decoder)
    - Data valid status is generated at the Multi-G Reconciliation Sublayer (RS DATA\_VALID\_STATUS) based on those delimiters (see for example 46.1.7.5 for XGMII)
    - Enables the Multi-G Reconciliation Sublayers (RS) to differentiate if errors were received within a packet or not

# Receiver Errors Handling

- ▶ The  $8N/(8N+1)$  encoding/decoding adopted proposal does not use explicit start or end packet delimiters
- ▶ At the transmitter side, a packet may contain  $8N/(8N+1)$  encoded data and error control octets
  - Normal data transmission ( $TX\_EN = 1, TX\_ER = 0$ ) is encoded as data
  - Transmit error propagation ( $TX\_EN = 1, TX\_ER = 1$ ) is encoded as error control
- ▶ At the receiver, any errors will be decoded as error control octets
  - Since there are no packet delimiters, and a packet may start with an error, it is not possible in general to differentiate if errors occurred within a packet or not
    - $RX\_ER$  and  $RX\_DV$  must both be asserted when an error is decoded within a packet
    - But if the error occurred during the IPG, that may result in a received packet with errors
      - The MAC frameCheckError/alignmentError counters will be incremented incorrectly

# Proposed Error Handling in 100BASE-T1L

- ▶ It is desirable to differentiate if errors occurred within a packet or not
  - Prevents data reception with error being signalled on the MII for errors received outside a packet
    - The consequences might be minor for fully standard compliant MACs, but could cause unforeseen problems with existing MAC devices, referred to as MACs from now on
    - Most MACs supporting Clause 22 RS and MII are proven for older technologies like 100BASE-TX
      - 100BASE-T1L will have to interoperate with those MACs
- ▶ It is proposed to add start and end packet delimiter control codes to the 8N/(8N+1) encoding
  - Note that existing Cs/CD control codes already act as packet delimiters when a packet starts/ends on odd encoder cycles
- ▶ This is consistent with the use of start and end packet delimiters in almost all previous technologies; 100BASE-X, 1000BASE-X, 1000BASE-T, 100BASE-T1, 10BASE-T1L, 10BASE-T1S and all the Multi-G technologies
  - **This is necessary to work robustly with the huge range of MACs in the field**

# Questions ?