TRANSMITTER CONTROL FOR THE AUI-C2C INTERFACES (D1.1 COMMENT #59)

Adee Ran, Intel

May 2020

Supporters

- Howard Heck, Intel
- Phil Sun, Credo
- Upen Reddy Kareti, Cisco
- Rick Rabinovich, Keysight Technologies
- Arthur Marris, Cadence

Comment

Cl **120F** *SC* **120F.1** *P* **202** *L* **31** # 59

Ran, Adee Intel

Comment Type **T** *Comment Status* **D**

"If implemented, the transmitter equalization feedback mechanism described in 120D.3.2.3 may be used to identify an appropriate setting"

As presented in ran_3ck_adhoc_02_021920, that mechanism supports the equalizer that was specified in the original CAUI-4 C2M (Annex 83D), which has only 3 taps with 5% coefficient resolution. The PAM4 AUIs defined in 802.3.bs (120D.3.1.5) and re-used in 802.3cd have kept this structure. However, we now have a 5-tap equalizer with a finer resolution. Even if pre-cursor tap c(-3) is removed as suggested in 120F.3.1.4 it would not be identical to the FFE in Annex 83D.

Therefore, re-using this method for 100GAUI-1 is impossible and new method should be defined. Possible solutions include a training protocol as in the PMD control function, new management variables and registers, or combinations of the two approaches.

SuggestedRemedy

A presentation with possible solutions is planned.

Why?

- To enable a simple AUI-C2C receiver, some of the equalization must be done in the transmitter.
- Optimal Tx setting is channel and device dependent, so has to be controlled.
- The control interface from previous AUIs is inadequate for the specified Tx equalizer with 5 (or possibly 4) taps and fine steps.
  - Something else has to be defined.
- The control function in clause 136 (and re-used in 162) defines the necessary information that should be passed between receiver and transmitter. It is a good foundation for equalization control.

Protocol or registers?

- The dilemma was discussed in previous presentation ran_3ck_adhoc_02_021920
- There are arguments for either method.
- Both are described below as options for implementation.

Option A

Register-based control with no start-up protocol

- Add a new top level subclause in annex 120F (e.g. 120F.5) titled “Transmitter control function”.
  - Define a pair of registers (“request” and “response”) in each direction (“local” and “remote”) per lane (total 2*2*4=16 registers).
  - Content of registers based on the PMD control function “request” and “status” fields. For management, the “local” registers are RO, and the “remote” registers are writeable.
  - Effect of writing to the “remote” registers is as specified in clause 136 for incoming training frames.
  - The “local” register updates (requests and responses) are relayed to the link partner’s corresponding “remote” registers by management.
  - The value “PAM2” is not allowed for the modulation and precoding fields (local/remote, request/status).

Details

- Benefits of a register-based control function
  - **No pattern generator or protocol logic is required** in the AUI-C2C
- Implications
  - **Initial NRZ signaling cannot be used**; bad equalization setting can prevent CDR lock.
  - **Management is tasked with reading/writing registers in sequence**, in all lanes, in order to “bring up” the AUI-C2C link.
  - There is **no well-defined startup sequence**. Debugging management flows may be difficult.
  - New text in the standard (instead of referring to a known function…)

Option B

Training protocol (with mapped MDIO registers)

- Add a new top level subclause in annex 120F (e.g. 120F.5) titled “Transmitter control function”.
  - Define a AUI-C2C transmit function with two operational modes, TRAINING and DATA, as in 162.8.2.
  - Each lane of the AUI-C2C interface shall implement the control function identical to the one specified in 162.8.11.
  - Update the PMA service interface in 120.3 such that the PMA:IS_SIGNAL.indication primitive is generated based also on the TRAINING/DATA mode.
  - Use the optional MDIO register mapping of Table 162–7. A device with both AUI-C2C and a KR/CR PMD shall have a separate set of control/status registers for each interface.

Details

- Benefits of a training protocol
  - There is a **well-defined startup sequence**.
  - **Initial NRZ signaling can be used** to achieve CDR lock with unknown channel and initial equalization setting.
  - **Autonomous training:** Each lane of the AUI-C2C has a state diagram logic which is responsible for managing the protocol and bringing up the AUI-C2C link without management intervention. Simplifies management.
  - **Management override:** If management has MDIO or equivalent access to both devices, training can be disabled, and then management can use the training control registers to optimize the TxEQ, as in 120D.3.2.3.
- Implication
  - **Training pattern generator and decoder are required** in the AUI-C2C endpoints.

Comparison

- Option B enables fallback to option A.
  - Enables different solutions in different products.
- Implication of option A is adding startup protocol encoder/decoder logic in retimers.
  - Assuming control registers are implemented anyway, the additional logic is a small impact.

Next steps

- Option A or option B? or something else?
- Are the descriptions above sufficient for implementation (with editorial license)?