CAUI-4 C2C Equalization control

In support of comments #22, #38, #39 and #41

Adee Ran, Intel Corporation

Comments #22, #38, #39

- Tables 83D-2 and 83D-3 define some possible tap settings, where a value of zero is written to c(1) and c(-1) respectively. Tables 83D-7 and 83D-8 are the normalized coefficient values only for these settings.
- We assume that other combinations are possible, with c(0) determined according to c(1) and c(-1), but it's not stated.
- Proposed remedies:
  - For #22 and #39, insert text before the last two sentences of the first paragraph of 83D.3.1.1 and change them:

The pre-cursor tap value c(-1) and the post-cursor tap value c(1) are controlled independently of each other. The cursor tap value c(0) is determined from c(-1) and c(1) so that the peak-to-peak transmit voltage is kept constant. When c(1) is set to 0, the pre-cursor equalization ratio R_pre for each setting of c(-1) is shown in Table 83D–2 where R_pre is defined in Equation (72-8). When c(-1) is set to 0, the post-cursor equalization ratio R_pst for each setting of c(1) is shown in Table 83D–3 where R_pst is defined in Equation (72-9).

  - Apply the suggested remedy to comment #38.

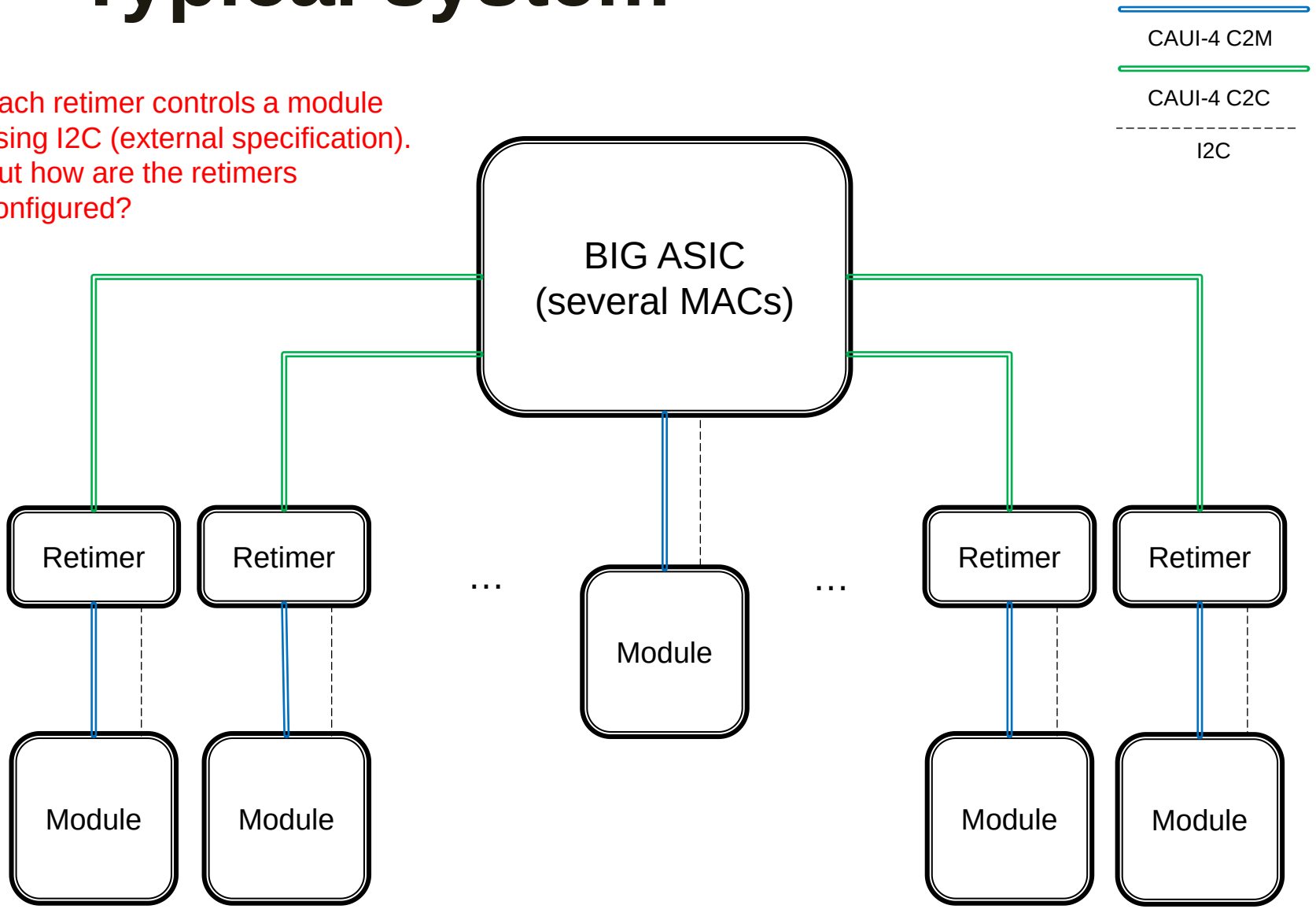
Rationale for Comment #41

- A system can comprise several CAUI-4 C2C links and multiple “chips”.
- Separate chips with CAUI-4 C2C (such as Retimers[1]) need to be programmed with transmitter equalization coefficients c(-1) and c(1) toward the CAUI-4 C2C partner.
- Settings depend on channel ➔ likely to differ between chips and possibly between lanes.
- All these settings would be easier to manage if stored centrally and communicated to the chips in a standardized way. MDIO is a natural choice.

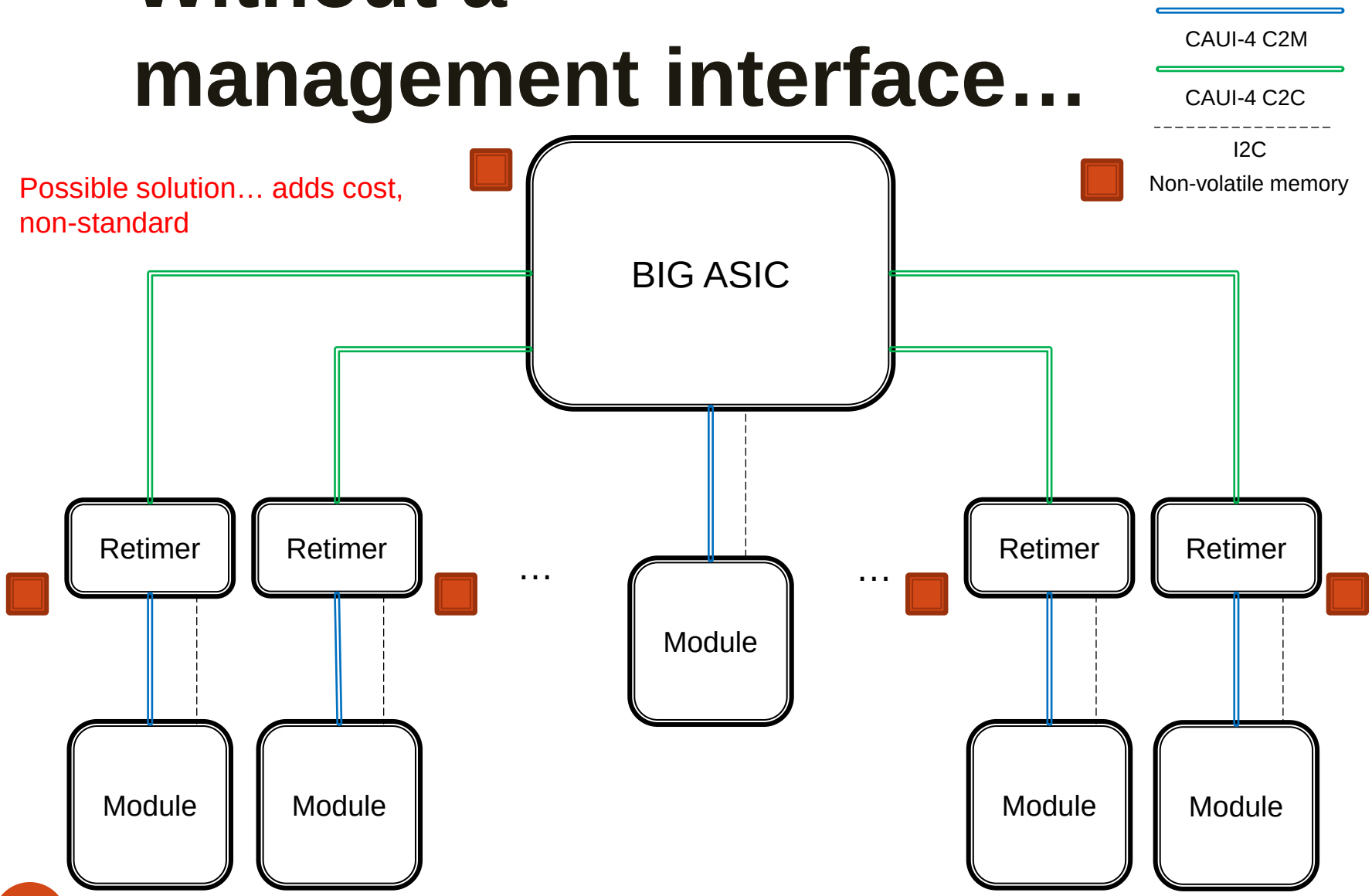
[1] Retimers will represent the class of MAC-less chips throughout this presentation

Typical system

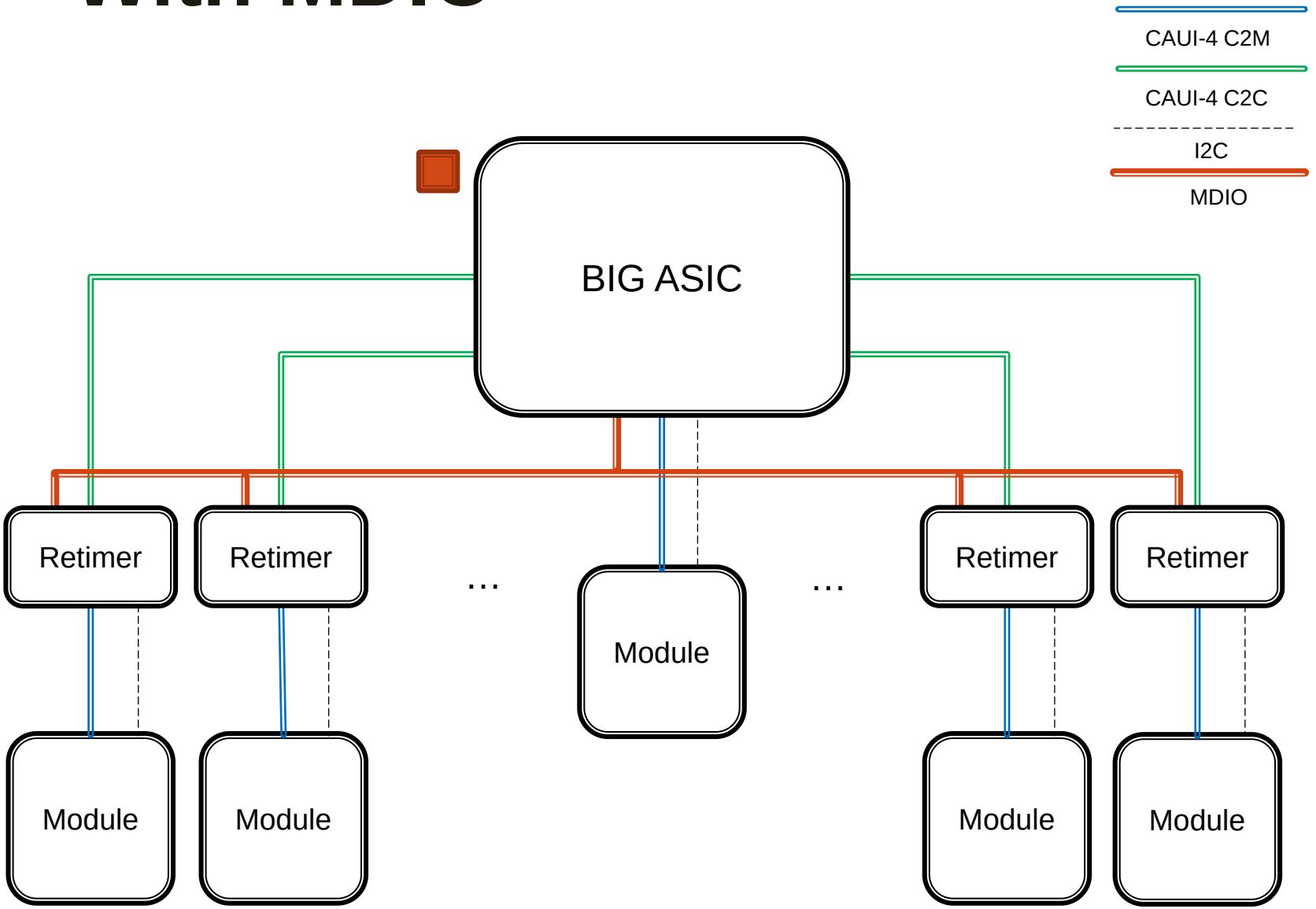
Each retimer controls a module using I2C (external specification). But how are the retimers configured?



Without a management interface…



With MDIO


CAUI-4 C2M
CAUI-4 C2C
I2C
MDIO
BIG ASIC
Retimer
Retimer
...
Module
...
Retimer
Retimer
Module
Module
Module
Module

Notes

- Each transmitter has 4 setting for c(−1), 6 settings for c(1) – making 24 combinations (c(0) always complements so that c(0) − c(−1) − c(1) = 1).
- There is a possibility for multiple retimers in the path.
  - If each one is a separate MDIO manageable device (MMD), they can be programmed separately.
  - It is possible to have two sets of parameters per device – one for transmit direction, one for receive direction (as defined in clause 83).
- Setting can be written by the MDIO host (SME) e.g. during system boot sequence.

Proposed change

- Allocate MDIO registers for controlling CAUI-4 C2C equalization in a PMA/PMD device.

Bits	Name	Description	R/W
15:5	Reserved	Read always 0, writes ignored	RO
4:2	Transmit equalizer post-cursor setting	See next slide	RW
1:0	Transmit equalizer pre-cursor setting		RW

- For the Pre-cursor field, all binary values are valid, selecting c(−1) values (shown in next slide)
- For the Post-cursor field, the binary values 000 to 101 are valid, selecting c(1) values (shown in next slide); 110 and 111 are reserved values (unspecified effect)

Bit field interpretation

Pre-cursor field

Binary value	c(-1) value
00	0
01	-0.05
10	-0.1
11	-0.15

Post-cursor field

Binary value	c(1) value
000	0
001	-0.05
010	-0.1
011	-0.15
100	-0.2
101	-0.25
110	Reserved
111	Reserved

Address allocation

- 8 separate addresses are needed – one per lane (0 to 3) and direction (transmit/receive).
  - If the transmit direction is a CAUI-4 C2M, the “Transmit direction” registers have no effect.
  - If the receive direction is attached to the PCS, the “Receive direction” registers have no effect.
- “CAUI-4 chip-to-module recommended CTLE” is currently at 1.169, but adjacent addresses are allocated.
- Currently 1.176 to 1.199 are reserved (available).
- Suggested allocation:

Address	Register name
1.180	CAUI-4 Chip-to-chip Transmitter Equalization, Receive Direction, Lane 0
1.181	CAUI-4 Chip-to-chip Transmitter Equalization, Receive Direction, Lane 1
1.182	CAUI-4 Chip-to-chip Transmitter Equalization, Receive Direction, Lane 2
1.183	CAUI-4 Chip-to-chip Transmitter Equalization, Receive Direction, Lane 3
1.184	CAUI-4 Chip-to-chip Transmitter Equalization, Transmit Direction, Lane 0
1.185	CAUI-4 Chip-to-chip Transmitter Equalization, Transmit Direction, Lane 1
1.186	CAUI-4 Chip-to-chip Transmitter Equalization, Transmit Direction, Lane 2
1.187	CAUI-4 Chip-to-chip Transmitter Equalization, Transmit Direction, Lane 3

Backup

Comment #22

Cl 83D *SC* 83D.3.1.1 *P* 148 *L* 10 # 22

Ran, Adee Intel

Comment Type T *Comment Status* D

The current text specifies minimum equalization support using two tables - one for pre-cursor and one for post-cursor - where the tables include 4 and 6 settings respectively. It is not clear how many settings are required altogether.

I assume the intent is that each of the 4 settings for c(-1) implied from table 83D-2 can be used with each of the 6 settings for c(1) implied from table 83D-3, with c(0) set to complement the peak-to-peak value. Tha would make exactly 24 possible settings.

This should be specified clearly.

The same combinations of settings should be used in calculation of COM, where tables 83D-7 and 83D-8 describe the actual coefficients in some of the settings.

SuggestedRemedy

Proposed remedy to be presented.

Proposed Response *Response Status* O

Comment #38

Cl 83D *SC* 83D.4 *P* 151 *L* 16 # 38

Healey, Adam LSI Corporation

Comment Type T *Comment Status* D

Table 83D-7 and Table 83D-8 leave some ambiguity as to how the transmitter equalizer may be configured. Is it required that a CAUI-4 chip-to-chip transmitter use either pre-cursor or post-cursor equalization but not both? Assuming that this is not the case, if one wants to configure the transmitter to use pre-cursor setting 2 and post-cursor setting 4, c(-1) cannot be simultaneously -0.1 and 0, c(1) cannot be simultaneously 0 and -0.2, and what should c(0) be?

Table 83D-7 and Table 83D-8 simply implement the rule that c(0) = 1-|c(-1)|-|c(1)|. This is already stated in 93A.1.4.2 and does not need to be repeated here.

SuggestedRemedy

Delete Table 83D-7 and Table 83D-8. Change Table 83D-6 as follows.

Transmitter equalizer, pre-cursor coefficient | c(-1)
Minimum value | -0.15 | --
Maximum value | 0 | --
Step size | 0.05 | --

Transmitter equalizer, post-cursor coefficient | c(1)
Minimum value | -0.25 | --
Maximum value | 0 | --
Step size | 0.05 | --

Proposed Response *Response Status* O

Comment #39

Cl 83D *SC* 83D.3.1.1 *P* 148 *L* 10 # 39

Healey, Adam LSI Corporation

Comment Type T *Comment Status* D

The requirements for R_pre and R_pst are ambiguous. R_pre is affected by the by the value of c(1) and R_pst is affected by the value of c(-1). The text cites 72.7.1.11 which includes specific conditions for the measurement of R_pre (c(1) disabled or zero) and R_pst (c(-1) disabled or zero). However, Table 83D-2 states the R_pre requirement for 4 settings with no regard to the post-cursor equalization setting. Is it necessary to maintain the +/-12.5% tolerance on R_pre over all of the post-cursor equalization settings? This is not a requirement for 100GBASE-KR4 and should not be a requirement for CAUI-4 chip-to-chip.

Also, starting at page 148, line 11, it is stated that the "minimum pre-cursor equalization R_pre supported is shown in Table 83D-2..." Table 83D-2 specifies ranges and not minimum values.

SuggestedRemedy

Change the last two sentences of the first paragraph of 83D.3.1.1.

"The pre-cursor equalization ratio R_pre for each pre-cursor tap setting is shown in Table 83D–2 where R_pre is defined in Equation (72-8) and the post-cursor tap setting is 0. The post-cursor equalization ratio R_pst for each post-cursor tap setting is shown in Table 83D–3 where R_pst is defined in Equation (72-9) and the pre-cursor tap setting is 0."

Proposed Response *Response Status* O

MDIO message format

Table 45–202—Extensions to management frame format for indirect access

	Management frame fields							
Frame	**PRE**	**ST**	**OP**	**PRTAD**	**DEVAD**	**TA**	**ADDRESS / DATA**	**IDLE**
Address	1...1	00	00	PPPPP	EEEEE	10	AAAAAAAAAAAAAAAA	Z
Write	1...1	00	01	PPPPP	EEEEE	10	DDDDDDDDDDDDDDDD	Z
Read	1...1	00	11	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z
Post-read-increment-address	1...1	00	10	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z

- OP: the type of transaction being performed by the frame. A <00> pattern indicates that the frame payload contains the address of the register to access. A <01> pattern indicates that the frame payload contains data to be written to the register whose address was provided in the previous address frame. A <11> pattern indicates that the frame is read operation. A <10> pattern indicates that the frame is a post-read-increment-address operation.
- PRTAD: Port address (Ethernet port), enabling up to 32 ports per SME
- DEVAD: Device address (managed component), enabling up to 32 devices (MMDs; e.g. repeaters) per Ethernet port

MDIO connectivity

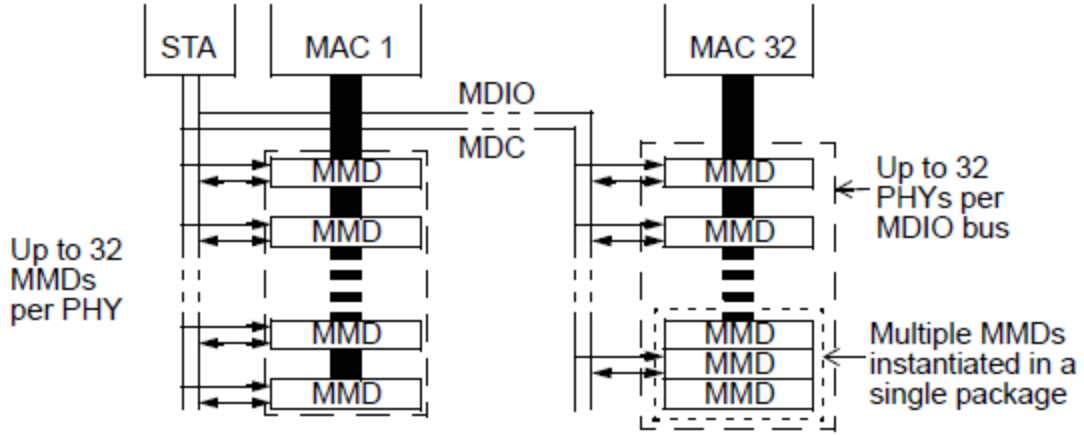


Figure 45–1—DTE and MMD devices