Cl **14** *SC* **14** *P* **15** *L* **5** # 101

Thaler, Pat Broadcom

Comment Type **TR** *Comment Status* **A**

Some text still implies that a type 10BASE-Te MAU is not a type 10BASE-T one, but it needs to be for backwards compatibility in places like autonegotiation. 10BASE-Te should be treated as a subtype of 10BASE-T.

SuggestedRemedy

In the title of Clause 14, change "and type 10BASE-Te" to "including type 10BASE-Te".

14.1.1 in the note say: "Support for both 10BASE-Te and non-10BASE-Te signal levels in a single device is not expected." Or you could use legacy 10BASE-T where you need to differentiate from 10BASE-Te.

14.1.1.3 - the first paragraph doesn't explicitly exclude 10BASE-Te. The paragraph needs language to exclude 10BASE-Te; either replace 10BASE-T with "10BASE-T except 10BASE-Te" or "legacy 10BASE-Te"

14.3, The additional sentence "This subclause also ..." is not needed since 10BASE-T includes 10BASE-Te when not otherwise specified.

14.3.1.2 the paragaph about insertion loss for a legacy 10BASE-T MAU needs to explicitly exclude 10BASE-Te.

This needs to be done for every time that there is a requirement that is different for 10BASE-Te. The paragraph near the beginning of 14.3.1.2 that contains the reference to Figure 14-7 is an example where it was done right.

Response *Response Status* **W**

ACCEPT IN PRINCIPLE.

In the title of Clause 14, change "and type 10BASE-Te" to "including type 10BASE-Te".

14.1.1 - Change note to read:
"Support for both 10BASE-T and 10BASE-Te signal levels in a single device is not required".

14.1.1.3 - Change text on page 16, line 5 from:
"The performance specifications of the 10BASE-T simplex …" to:
"The performance specifications of the 10BASE-T except 10BASE-Te simplex …"

14.3 - Delete additional sentence "This subclause also defines the ."

14.3.1.2 Change page 17, line 8 from:
"For a type 10BASE-T MAU, insertion ." to:
"For a type 10BASE-T MAU that is not a type 10BASE-Te MAU, insertion ."

Change text on page 18, line 34 from:
"data sequences for a type 10BASE-T MAU." to:
"data sequences for a type 10BASE-T MAU that is not a type 10BASE-Te MAU."

Change text on page 18, line 44 from:
". Figure 14-7 for 10BASE-T and ." to:
". Figure 14-5 for 10BASE-T except 10BASE-Te and ."

Change text on page 19, lines 12, 18 and 25 from:
". For 10BASE-T and ." to:
". For 10BASE-T except 10BASE-Te and ."

Change text on page 19, line 52 from:
". For a 10BASE-T MAU." to:
". For a 10BASE-T MAU that is not a 10BASE-Te MAU."

Cl **14** *SC* **14.10.3** *P* **21** *L* **12** # 97

Thaler, Pat Broadcom

Comment Type **ER** *Comment Status* **A**

Since the decision was that 10BASE-T includes 10BASE-Te, it is unclear whether a maker of a 10BASE-Te MAU also checks the 10BASE-T box yes.

SuggestedRemedy

Add text to the 10BASE-T entry that excludes 10BASE-TE.

Response *Response Status* **W**

ACCEPT.

Cl **36** *SC* **36.2.5.2.2** *P* **83** *L* **13** # 1

Barnette, James Vitesse Semiconducto

Comment Type **TR** *Comment Status* **A**

Branches from LPI_IDLE_D, LPI_K, RX_WAKE, and RX_WTF, are not sufficiently specified when multiple conditions occur simultaneously.

SuggestedRemedy

Branches from LPI_IDLE_D near line 13:

On the branch from LPI_IDLE_D to RX_LINK_FAIL, change the condition from "rx_ts_timer_done" to "signal_detect = OK * rx_ts_timer_done". On the branch from LPI_IDLE_D to off-page node F, change the condition from "xmit != DATA * SUDI(![/K28.5/])" to "signal_detect = OK * !rx_ts_timer_done * xmit != DATA * SUDI(![/K28.5/])". On the branch from LPI_IDLE_D to LPI_K, change the condition from "xmit = DATA * SUDI + SUDI([/K28.5/])" to "signal_detect = OK * !rx_ts_timer_done * (xmit = DATA * SUDI + SUDI([/K28.5/]))".

Branches from LPI_K near line 19:

On the branches from LPI_K to off-page nodes D, F, and C as well as the branch back to LP_IDLE_D, insert the condition "signal_detect = OK * <cond>" where <cond> is replaced by the previously-stated condition.

Branches from RX_WAKE near line 32:

On the branch to RX_WTF, insert the condition "signal_detect = OK * !(code_sync_status = OK * SUDI([/K28.5/]*EVEN)) * ..." into the condition for this branch. On the branch to RX_WAKE_DONE, insert the condition "signal_detect = OK * ..." into the condition for this branch.

Similarly, in branches from RX_WTF near line 36:

On the branch to RX_LINK_FAIL, insert the condition "signal_detect = OK * !(code_sync_status = OK * SUDI([/K28.5/]*EVEN)) * ..." into the condition for this branch. On the branch to RX_WAKE_DONE, insert the condition "signal_detect = OK * ..." into the condition for this branch.

Response *Response Status* **W**

ACCEPT IN PRINCIPLE.

Use changes as suggested for branches from LPI_IDLE_D and LPI_K. Use the following for the other two:

Branches from RX_WAKE near line 32:

On the branch to RX_WTF, insert the condition "signal_detect = OK * ..." into the condition for this branch. On the branch to RX_WAKE_DONE, insert the condition "signal_detect = OK * !rx_tw_timer_done * ..." into the condition for this branch.

Similarly, in branches from RX_WTF near line 36:

On the branch to RX_LINK_FAIL, insert the condition "signal_detect = OK * …" into the condition for this branch. On the branch to RX_WAKE_DONE, insert the condition "signal_detect = OK * !rx_wf_timer_done * …" into the condition for this branch.

Cl **36** *SC* **36.2.5.2.2** *P* **83** *L* **44** # 2

Barnette, James Vitesse Semiconducto

Comment Type **TR** *Comment Status* **A**

When state RX_QUIET is to be left via transition (signal_detect = FAIL * rx_tq_timer_done) entering state RX_LINK_FAIL (via the "I" connector) signal "rx_quiet" is not set back to FALSE.

In case this condition (and transition) is ever met rx_quiet will never be set to FALSE again. A receiver would never be able to get data again since the receiver (e.g. a deserializer) would be powered down all the time - only a reset would help.

SuggestedRemedy

When entering state RX_LINK_FAIL signal "rx_quiet" must be reset (rx_quiet <= FALSE; this would be an additional assignment to the already existing assignment "rx_lpi_active <= FALSE").

Response *Response Status* **W**

ACCEPT.

Cl **45** *SC* **45.2.7.13a** *P* **128** *L* **25** # 99

Thaler, Pat Broadcom

Comment Type **TR** *Comment Status* **A**

The bit assignments still aren't right. Bits 3 through 1 of the register should map to U3 through U1 of the U field. I.e. each bit in the register should map to the corresponding bit of the U field. This was agreed in the resolution of my comment 416 on the first ballot and in the response to 193 in the first recirculation.

This comment also applies to 45.2.7.14a which should use the same mapping.

SuggestedRemedy

Change the mapping of bits 3 through 1 to U3 through U1 respectively in both tables.

I would also prefer that the resolution in response to 416 be fully implemented - the register bits 0 through 15 should map to U0 through U15 (all bits apply to Clause 73 and only bits 0 through 10 apply to Clause 28) with the unused values reserved. That allows the mapping for the register to U bits to be established now for when additional bits are added latter.

Response *Response Status* **W**

ACCEPT IN PRINCIPLE.

Change 3:1 to U3 through U1 to rectify editorial mistake implementing comment #193.
Make the change in both tables: 45-157a and 45-157b

Add a new paragraph after the current one in 45.2.7.13a:

Bits 10:0 of register 7.60 map to bits U10 through U0 respectively of the unformatted next page following a EEE technology message code as defined in 28C.12. Bits 15:0 of register 7.60 map to bits U15 through U0 respectively of the unformatted next page following a EEE technology message code as defined in 73A.4. Devices using Clause 28 autonegotiation may ignore bits defined for Clause 73 autonegotiation and devices using Clause 73 autonegotiation may ignore bits defined for Clause 28 autonegotiation.

Cl **55** *SC* **55.3.2.2.9a** *P* **189** *L* **13** # 100

Thaler, Pat Broadcom

Comment Type **TR** *Comment Status* **A**

Most of the clean-up of terminology for LPI and EEE has been done, but there are still a few cases where the EEE capability is referred to as low power idle.

in the state machine definitions of clause 55, "When the low power idle function is <not> supported," appears a number of times including in 55.3.5.2.4 Functions where ther is no low power idle function. These should all refer to EEE which is the name of the optional capability.

SuggestedRemedy

If low power idle is not supported should be "If EEE is not supported".
"the low power idle function" shoudl be "EEE"

Check for any other instances of supported being applied to low power idle or LPI and correct. LPI is the signal and LPI mode is the state where that signal is used. EEE is the optional capability.

Response *Response Status* **W**

ACCEPT IN PRINCIPLE.

"If low power idle is not supported" should be "If EEE capability is not supported".
"the low power idle function" should be "EEE capability"

Also make changes on page 179, 195, 196, 206.

Cl **74** *SC* **74.7.4.8** *P* *L* # 102

Thaler, Pat Broadcom

Comment Type **TR** *Comment Status* **A**

The response to 384 on the first Working Group ballot has not been fully implemented. FEC does not have "frames", it has blocks

SuggestedRemedy

All instances of "frame" in Claause 74 should be replaced with "block".

Response *Response Status* **W**

ACCEPT.

Change "frame" to "block" at the following locations:

Page 233, line 11, 15 and 19.