Disposition of LReq, etc. in 1394a Rev1.4

Michael Shinkarovsky
Lucent Technologies
mshinkarovsky@lucent.com

Disposition of Fair/Priority Req.

- **Table 5-16 Fair/Priority::Receive** states: "Request always discarded if arbitration acceleration is not enabled."
- **Table 7-29 receive_actions()** takes `!enab_accel` into account ***only*** upon entrance to receive_actions().
- **Propose:** include enab_accel in **all** code clearing `breq` in receive_actions()

Disposition of Fair/Priority Req.

- **Old**:`if (bit_count > 8 && (breq == FAIR_REQ || breq == PRIORITY_REQ))`
- **New**:`if ((bit_count > 8 || !enab_accel) && (breq == FAIR_REQ || breq == PRIORITY_REQ))`
- **Old**:`if (!ack && (breq == FAIR_REQ || breq == PRIORITY_REQ))`
- **New**: `if ((!ack || !enab_accel) && (breq == FAIR_REQ || breq == PRIORITY_REQ))`

Disposition of Isoch Request

- **Table 5-16 Isochronous::Status** states: Request discarded if status indicates subaction gap.
- C code does not support this.
- Does not solve a real problem.
- PHY's have been forgiving on the timing of Isoch or Imm Req. acceptance in the past
- **Propose to remove** this new timing rule from Table 5-16.

Arb_Reset and Subaction_Gap Status

- **Clause 5.5** states: The PHY clears ARB_RESET_GAP and SUBACTION_GAP upon any transition out of state A0
- Not supported by C code
- Does not make sense for Idle:Request:Idle path
- Intent was to cancel “Gap” status on entrance to RX or TX.
- **Propose:** The PHY clears ARB_RESET_GAP and SUBACTION_GAP anytime Receive or Grant is asserted on Ctl

PHY Status vs. Link active

- **Clause 5.1** states: “any status information generated by the PHY while the interface is disabled shall be discarded and shall not cause a status transfer upon restoration of the interface.”
- **Clause 6.1** states: “If the link is active, *PHY_interrupt* is reported as S[3] in PHY status transfer, as specified by clause 5.5; otherwise a PHY interrupt shall cause LinkOn to be asserted.”
- Contradiction between 2 statements!

PHY Status vs. Link active

- **Propose**: Change clause 5.1 to state: "with the exception of *PHY_interrupt* PHY status, any status information generated by the PHY while the interface is disabled shall be discarded and shall not cause a status transfer upon restoration of the interface. *PHY_interrupt* PHY status shall cause LinkOn to be asserted and shall not cause a status transfer upon restoration of the interface."