

IEEE P1394b SCAT Meeting
Teleconference
April 22, 1999

Those Present on the telephone bridge:

Colin Whitby-Strevens (Zayante), Thang Le (HP), Eric Hannah (Intel), David Wooten (Compaq), Steve Bard (Intel), Dave Bruncker & John LaPota (Molex); Jerry Hauck (Zayante); Tony Foster (HP)

David Wooten informed the group that Sean Kileen will not be in attendance during the SCAT & Plenary meetings scheduled in San Jose next week.

Beta Only Connector

Representing Max Bassler (Copperheads) Dave Bruncker and John Lapota presented the circumstantial evidence surrounding Copperhead group pushback on a beta only connector. There are a couple of concerns: 1) SCAT may be creating requirements and constraints for a plug/socket which would not provide a way to mate with existing 1394 sockets/plugs; 2) perception that a plug/socket meeting the constraints of SCAT #18 is not easily designed or manufactured (e.g. it is much easier to modify existing plugs/sockets with a keying mechanism).

In response to number one: Yes, this is the goal of SCAT. There must not be a way to connect a Beta only node to either a 4-pin or 6-pin DS only node. The constraints for a keyed beta only plug/socket meet this requirement. There is, however, a method for providing interconnect capability between 4-pin and/or 6-pin DS only nodes and a bilingual 1394b node. It is called a cable – one end with the keyed beta-only plug and the other end with either a 4-pin or 6-pin plug. NOTE: A keyed beta-only plug does NOT insert into a NON-KEYED beta-only socket. Definition: A “keyed” beta-only socket/plug is a bilingual socket/plug.

Pushback from some plug/socket manufacturers reflects a concern regarding the ability to manufacturer a plug/socket which will meet the constraints outlined in SCAT #18 – some plug/socket manufacturers prefer to make a modification of the existing 6-pin connector to meet the needs of Beta/Bilingual plug/socket. However, there do exist other plug/socket manufacturers which believe it is best to start with a clean sheet of paper and begin with a new design for a beta/bilingual plug/socket for support of all 1394b speeds (S800 through S3200).

All requirements of the PC folks and the CE folks must be rolled into the new beta/bilingual plug/socket. It is believed by SCAT that the constraints provided in SCAT Table item #18 do precisely that.

It was noted that SCAT table item #18 is complemented by the plenary minutes for the meeting held march 22, 1999 in Tempe, Arizona at the Embassy Suites Hotel – reference the heading in the minutes: “More Connector/Cable Discussion”.

PIL/Fan-Out PHY LinkOn Discussion

Included here (for all to benefit) is a “reproduction” of a some comments from Jerry Hauck:

“To the point and in summary: I'm largely in agreement with you and think the LINK_ON works the way _you_ want it to. I simply had an observation that a next logical step (adding a link to the otherwise naked fan-out PHY) might get messy.

“Read on if you care:

“Stream of Confusion

The LINK_ON signal coming from the fanout PHY as defined Thursday works just fine for the scope of the application (call it App A): integrated link/phy connected to naked PHY. The

LINK_ON from the naked PHY can be used to wake up the integrated link/phy. The motivation for this is the observation that the integrated link/phy may not be able to provide sufficient power savings when suspended, and moving the phy keep-alive portion into the resume well may not be acceptable.

"My 'objection', if you want to call it that, came from considering a different, possibly pointless, application (call it App B): integrated link/phy connected to a multi-ported node (i.e., with an attached link and stuff above the link). It was unclear how one would provide the necessary LINK_ON functions in this case. Ignoring the integrated link/phy and considering only the multi-ported node for a moment, P1394a clearly defines the LINK_ON handshake: asserted when the link is 'not active' (not LPS or not LCtrl) upon receipt of LINK_ON packet or when an un-acknowledged interrupt is pending in the PHY. By definition then LINK_ON shall not be asserted whenever (LPS && LCtrl). Some early link designs (i.e., a specific OHCI system implementation I knew about in a former life) relied on this fact ... they knew that if they asserted LPS and LCtrl, then LINK_ON would be inactive and it would be safe to re-arm the edge-triggered NMI used to see LINK_ON. Based on this, I think it a bit dangerous to add any new conditions for the deassertion of LINK_ON.

"Now consider the addition of the integrated link/PHY for which we want to proxy LINK_ON ... as we defined it Thursday, it would not be deasserted until communication had been established with the integrated combo. Given that this is a new 'deassertion' term for LINK_ON and contrary to my warning above, it makes me a bit queasy.

"Another uncertainty for me is that it may be likely for the multi-ported node to report the 'L' bit as active. If so, will the remote fax machine or whatever know to hit the multi-ported node with a LINK_ON packet? Or if the multi-ported PHY has all of its ports suspended, but the link is still active ... should we cause a LINK_ON when a resume arrives? Currently, we don't since LPS && LCtrl is true in this scenario.

"Consequently, App B at first glance has some issues which aren't completely addressed with the LINK_ON proposal. But there is no point in addressing these issues if App B isn't needed by the market ...

"So are either of these Apps (A or B) really vital or necessary? App A was questionable to me ... Mr. Bard was arguing for a beta-only P1394b to ease chipset integration (of a multi-ported phy?) well before the fan-out PHY concept was publicly suggested. And knowing his love for mobile, standby, and suspend type things, I had foolishly assumed that he had signed up to placing the keep-alive toning in the resume well. I also personally liked the model of having a fully functional (including suspend/resume with power savings) single port beta-only PHY integrated with the link, particularly for a mobile market. The decision to add a fan-out PHY then only hinged on the need to 'bridge to legacy' or the need to fan-out. One wouldn't add a fan-out PHY just to achieve power conversation of the core logic chipset. But I supposed the other argument for App A is that most applications will want more connections than the chipset provides or will want a bi-lingual connection for legacy CE. As such, the 'normal' application will include a fan-out PHY and, as such, why not ease the integration burden on the chipset and allow the whole of the integrated link/phy to be on the same power well? Seems reasonable. But then App B sounds more compelling as well. Since we've determined an external PHY is needed for most of our system configurations, wouldn't it be great to share that PHY logic with the HDD (probably a bad example) or some other internal peripheral that hasn't yet made the jump to an integrated PHY/LINK? Maybe this is a bunch of hooey given that these 'other' peripherals won't be soldered down and the notion of a cabled P1394a style PHY/Link interface is bunk. I concluded from all of this that we may just say 'no' to App B which clears the way for the proxy LINK_ON functionality."

Jerry's "B-Application" would result in normal suspend/resume and/or standby/restore.

A fan-out PHY identifies a static port as the “PHY/Link” interface to the PIL (it just happens to be beta only signaling).

A “trusted-port” (a dedicated port which becomes the “logical” PHY/Link interface when connected to a PIL) would recognize PHY Extended Command packets to do PHY register write with response OK == 1. PHY Register Write Extended Commands coming through on any other port would result in a response packet returning OK== 0.

There are problematic issues associated with software when trying to determine when the PIL has and does not have a fan-out PHY. Issues such as: one node-ID or two when a fan-out PHY is present; b) software transparency between a PIL with and a PIL without a fan-out PHY.

Loop Healing

Jerry will deliver a presentation at SCAT next week.

Misc

Tony pointed out that better socializing needs done in communicating precisely what 1394PE (e.g. “PIL”) is and is NOT. The issue is word is getting out on the street that 1394b is only available as an integrated solution and implementers MUST integrate the entire PHY and link – it is not understood that integration is necessary ONLY for S1600 (and above) – e.g. S800 1394b continues to be able to be implemented using discrete PHY and Link silicon.

Actions Items:

AR: Steve Bard/Colin Whitby-Strevens: Extract from Jim Skidmore the algorithm a branch node would use to calculate the node-ID of a leaf node in standby as it performs the proxy process for the leaf node.

AR: Steve Bard – open discussion on the matter of software support for 1394PE and the fan-out-PHY with John Fuller.

AR: SCAT – bring John Fuller into the fan-out PHY discussion during one of the SCAT meetings next week.

AR: Steve Bard to send to B-reflector the information contained in the last Plenary meeting minutes regarding the cable solutions for legacy interconnect when a beta only plug/socket (with key capability) is created).