

# P1394b SCAT

Done	27
Not yet done	13
Open	19
Recommend Accept	1
<b>Total</b>	<b>60</b>

---

<b>ID</b> 141	<b>Balloter</b>	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	Standby nodes and ports	<b>Status</b>	Done
<b>Owner</b>	CWS	<b>Clause</b> 4.7	<b>Page</b> 53 <b>Line</b> 0

**Problem description**

Standby is the state of a port, not a node

**Fix description**

---

<b>ID</b> 140	<b>Balloter</b>	CWS	<b>How submitted</b>	Post ballot	<b>Fixed in version</b>
<b>Title</b>	Active nodes and ports	<b>Status</b>	Done		
<b>Owner</b>	CWS	<b>Clause</b> 4.7	<b>Page</b> 53 <b>Line</b> 31		

**Problem description**

no such thing as an active node, only active ports.

**Fix description**

---

<b>ID</b> 187	<b>Balloter</b>	MS	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	spelling	<b>Status</b>	Done	
<b>Owner</b>	CWS	<b>Clause</b>	<b>Page</b> 171 <b>Line</b> 44	

**Problem description**

Resyn1

**Fix description**

---

<b>ID</b> 148	<b>Balloter</b>	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	Remove definition of BIAS_HANDSHAKE	<b>Status</b>	Done
<b>Owner</b>	CWS	<b>Clause</b> 11.3	<b>Page</b> 176 <b>Line</b> 8

**Problem description**

definition should be removed, and the text rolled into that of RECEIVE\_OK\_HANDSHAKE

**Fix description**



















<i>ID</i> 162	<i>Balloter</i>	<i>How submitted</i>	<i>Fixed in version</i>
<i>Title</i>	PHY Link handover	<i>Status</i>	Open
<i>Owner</i>	DW	<i>Clause</i>	12.6.3.2
		<i>Page</i>	208
		<i>Line</i>	13

***Problem description***

(text on p209, l10)

when phy hands bus over to link, spec leaves open the possibility that the link may drive CTL/DATA pins on cycle following PHY driving CTL and DATA to 0s. <"The link indicates ... asserting HOLD or TRANSMIT on the Ct[0:1] lines as soon as possible ...> This revives the old bug for isolated systems - that the phy and link drivers could collide for some brief overlap period. This would foul up the interface if one of the isolation networks were in place (both annex J and bus holders get broken by overlap).

Propose to reinvent having the link start by driving 0s to both CTL and DATA lines - see A spec, Figure 6-8 in section 6.5.

***Fix description***

<i>ID</i> 163	<i>Balloter</i>	<i>How submitted</i>	<i>Fixed in version</i>
<i>Title</i>	Upgrading requests	<i>Status</i>	Open
<i>Owner</i>	DW	<i>Clause</i>	12.6.3.4
		<i>Page</i>	209
		<i>Line</i>	33

***Problem description***

When a beta link makes a request of the PHY, the format of the request can be "upgraded" from unspecified to BETA. Later, when transmit\_actions.c executes, the link is granted using this "upgraded" format.

The link uses the format returned in the PH\_ARB\_confirmation to match it's outstanding requests so that it knows which has been granted. For example, assume the link made a NEXT\_EVEN S400 BETA request and then a CURRENT S400 unspecified request. If a grant for a BETA format is returned, the link knows to send the NEXT\_EVEN packet rather than the current packet. So the link is essentially checking the format.

Now if a format gets upgraded, the link may be confused or will need extra checking. For example, if a CURRENT S400 unspecified format is requested, and the grant returned is for ASYNC, S400, BETA ... the link will need to understand what has happened (that unspecified became beta). Is this what we intended to do?

***Fix description***

---

<b>ID</b> 164	<b>Balloter</b>	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	PHY cancellation of late link requests	<b>Status</b>	Open
<b>Owner</b>	DW	<b>Clause</b> 12.6.3.4	<b>Page</b> 209 <b>Line</b> 34

**Problem description**

PHY cancellation of "late" link requests (not sure where this comment should go)

Setup: At link in the even phase, link issues a next\_odd (Req #1) and sometime later issues a current (Req #2). Subsequently, the link receives from the PHY an ARB\_RST\_ODD followed by a GRANT. Assuming Req #1 and Req #2 have the same speed and format, the link can not determine whether the GRANT was for Req #1 or Req #2.

Scenario A: Phy issued ARB\_RST\_ODD and GRANT after receiving Req #1 but before Req #2. Consequently, when Req #2 arrives, it is queued by the PHY.

Scenario B: Phy issued ARB\_RST\_ODD before receiving Req #1. Req #1 and Req #2 both received by PHY before it issues a GRANT. In this case, Req #2 has updated and replaced Req #1. After issuing the GRANT, no requests are pending in the PHY.

From the link's perspective, scenario A and scenario B look identical in terms of order of arriving indications from the PHY. But in scenario A, a request is still pending in the PHY and the link should not reissue, and in Scenario B no requests are pending and the link does need to re-issue.

Proposed fix is that after sending a GRANT to the link, the PHY discards all LREQs for the same "pipe" (async or isoch) as the GRANT until the link acknowledges the GRANT by taking possession of the PHY/Link interface. If so, then Scenario A would end up with no request pending in the PHY since it ignored Req #2 from the link.

**Fix description**

---

<b>ID</b> 165	<b>Balloter</b>	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	Remembering request format	<b>Status</b>	Open
<b>Owner</b>	DW	<b>Clause</b> 12.6.3.4	<b>Page</b> 209 <b>Line</b> 36

**Problem description**

Must remember requested format when returning grant

Even if PHY choses to upgrade an unspecified request format to a beta format, the grant issued to the parallel link must be tagged with the requested format, not the actual format. This allows the link to match the grant and LREQ exactly, and avoid some error inducing confusion in some small corner cases.

**Fix description**

---

<b>ID</b> 166	<b>Balloter</b>	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	standby_reset and missed_reset	<b>Status</b>	Open
<b>Owner</b>	CWS	<b>Clause</b> 14.1	<b>Page</b> 233 <b>Line</b> 19

**Problem description**

relationship between this and missed\_reset?

**Fix description**



---

<b>ID</b>	173	<b>Balloter</b>	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	Negotiated_speed		<b>Status</b>	Open
<b>Owner</b>	CWS	<b>Clause</b>	14.1	<b>Page</b> 237 <b>Line</b> 53

**Problem description**

No power\_reset value given for negotiated speed. C code / comments seem ambiguous on whether: value should start at 000, and go upwards as port is initialized, or value should start at max\_port\_speed, and go downwards as port is initialized

**Fix description**

Proposed response:-

Power reset value is implementation dependent, and negotiated\_speed is set only once speed negotiation is complete. See definition of port\_speed on p280 l35-37

Propose clarifying by

1) adding comment

p293 l38

// set Negotiated\_speed in port register map to port\_speed and

2) possibly also add some text into the description of S3:S0 on p266 l34 and S4:A0b on p267 l14 to say that this is the point at which Negotiated\_speed is set to port\_speed[xxx]

---

<b>ID</b>	175	<b>Balloter</b>	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	PH_DATA indication		<b>Status</b>	Open
<b>Owner</b>	MT	<b>Clause</b>	15.2.3.3	<b>Page</b> 247 <b>Line</b> 20

**Problem description**

There is an inconsistency in the description of Speed\_Code. Is the Speed\_Code valid with a phy data indication of PH\_DATA\_START or PH\_DATA\_BYTE?

**Fix description**

---

<b>ID</b>	176	<b>Balloter</b>	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	MIN_PKT_SPACING and MIN_PHY_PKT_SPACING		<b>Status</b>	Open
<b>Owner</b>	PhyDogs	<b>Clause</b>	15.4.7	<b>Page</b> 250 <b>Line</b> 14

**Problem description**

The two parameters Min\_Pkt\_Spacing, Min\_Phy\_Pkt\_Spacing are used in Figure 15-2 and its accompanying text - but never defined or even mentioned elsewhere in the document.

**Fix description**

---

<b>ID</b>	181	<b>Balloter</b>	<b>How submitted</b>	<b>Fixed in version</b>
<b>Title</b>	CONFIG_TIMEOUT		<b>Status</b>	Open
<b>Owner</b>	PhyDogs	<b>Clause</b>	15.6	<b>Page</b> 254 <b>Line</b> 45

**Problem description**

The value of CONFIG\_TIMEOUT needs to be adjusted to prevent false positive loop detects (always was possible in the Legacy draft, but the effect now is less benign).

**Fix description**

---

