

IEEE p1394b
Standards Closure Action Team (SCAT)
Meeting

August 23rd, 25th 1999
Portland, Oregon, Downtown Marriott Hotel

23 August 1999

CHAIR: *David Wooten*
SECRETARY: *Steve Bard*
EDITOR: *Eric Hannah*

IEEE P1394b Task Group Chairs: **Copperheads:** *Max Bassler*, **PHY/Link:** *Sean Killeen/Tony Foster*;
Upstarts: *Colin Whitby-Strevens*; **Simulations:** *Jerry Hauck*; **UTP:** *Colin Whitby-Strevens*; **B-Low Power:**
Steve Bard; **Accelerations:** *Mike Teener/Alistair Coles/Colin Whitby-Strevens*; **B-Port:** *Alistair Coles/Mike*
Teener/Colin Whitby-Strevens

Review of action items from previous meetings:

Action Items Closed:

AI#1: Sean Killeen: The Standard Draft text needs to have a change for LREQ such that it contains a bit. The definition of which determines whether the request is beta-mode or indeterminate. <DONE>

AI#5: Steve Bard - Contact Eric Hannah with regard to the completion of Eric's action items contained within the SCAT Issues data base.< DONE>

AI#6: David Wooten will socialize on the general "B" reflector the notion of removing per port error count registers. If there is a lack of a scream response, these registers will be removed prior to the 0.8 draft. <David has not yet socialized this. Alistair's e-mail of xx-xx-99 makes this a moot point, therefore, per port error count registers will remain with a text change in the draft.>

A1#7: Bivabasu will submit to Colin changes required to the UPSTARTS code based upon P1394a 3.0 code changes (submitted to Colin around August 11th - sooner if possible). <Done>

#1: Alistair has some work to accomplish prior to the next SCAT meeting (see SCAT Issues #37 & #42).<#37 and #42 are Done >

Action Items carried from Previous Meetings:

Colin Whitby-Strevens: Must make certain that a multi-port node functioning as a leaf node and which is in standby will restore when a new connection is detected on one of its previously unconnected ports. In addition, if one of the ports were suspended and it resumes, the node restores. <WIP - Last part complete>

Colin Whitby-Strevens: Capture in the draft specification that the arbitration delay must be less than or equal to the data repeat delay. Table 8-14 in 1394a Draft 3.0 has been incorporated into the 1394b draft. However, it needs to be eye-balled and brought into ship-shape for 1394b (the "ship-shape" is the actual action item). <Table 14-6 and table 14-7 - this chapter was not reviewed at the Scotts Valley page turner and will be reviewed at the net page turner>

David Wooten will make sure that Linux, Sun, Apple and Microsoft software folks using ping to set gap count use of 1394a equations for determining the ping results. <MS -- ping and plugging in worst case delay that is seen into equation in 1995.

Sun - Assuming that all devices are connected in series and assuming 144ns PHY delay and 4.5M cables. Then using table in 1394a.

Apple - Determining actual number of hops between devices and assuming 144ns PHY delay and 4.5M cables and using the table in 1394a.

Linux - hasn't gotten around to setting the gap count yet. Will use formula in clause C..2.>

Colin will begin investigating the issue of symbols which may be deleted. The goal is to insure FIFOs do not become empty when they are supposed to be repeating packets. The FIFO may become empty due to clock frequency differences and related quantization effects. <This action item is, as admitted by Colin, serious deficit and must be completed.>

Terms for the unaware reader:

PIL Integrated PHY/Link silicon device

FOP Fan Out PHY

legacy link a link specifically targeted for use in 1394-1995 or 1394a environments (e.g. it does not contain any 1394b optimizations)

Monday, August 23, 1999

Action items from previous meetings reviewed (see above).

Jumped right into the SCAT issues table. Details of that activity has been documented in the various SCAT issue items found in the table.

Lengthy discussion surrounding issue #41 (so as to enable discussion of issue #40). Bottom line, there will be at least one quadlet delay between packets. This delay is created AUTOMATICALLY by guaranteeing two packet beginning symbols and two packet ending symbols for every packet. Colin will examine the 'C-Code' so as to make certain the 'C'-Code" guarantees this action. Sean will write text in the PHY-Link chapter that will detail this.

SCAT issue number 61 (deleted symbols) was created. Lengthy discussion regarding the insertion, deletion and propagation of delete-able symbols - both "may" and "must" symbols. Three action items were documented in the SCAT issue.

With regard to lack of recovery from loss of sync: With a bit error rate of 10^{12} then one might expect to see a single bit error about every three hours or so. It would take a sequence of several bit errors to cause a loss of sync - or a shifted bit in a frame. Mike Teener does not believe a shifted bit would occur. Alistair's B-Port, when detecting loss of sync, endeavors to recovery as quickly as possible. If it is not able to recover, it will disconnect and reconnect and endeavor to go through the start-up training sequence, etc.

Implement a connect debounce (approximately 320 milliseconds). During that time, apply toning. If, at the end of the debounce period, tone is still connected, begin speed, additionally, tone until speed signaling is detected and then begin speed signaling.

Resolution was that any loss of synch on the bus would result in a bus reset with the bus being divided at the connection on which the loss of sync condition exists. Connections would drop back to disconnected accomplished by the port detecting the loss of sync. When it detects loss of sync, it will stop transmitting all together and go back to toning. This will cause the other end to get loss of synch and go back to toning.

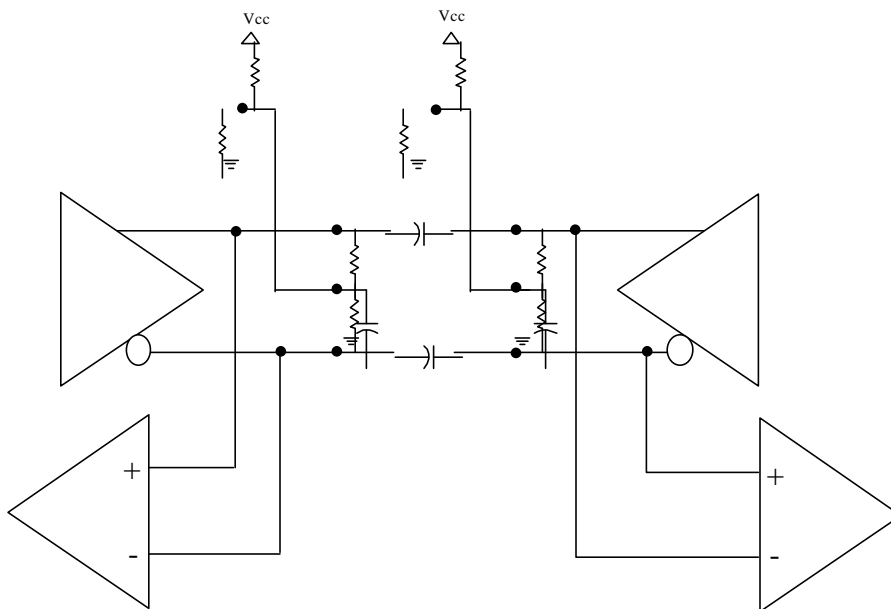
Dismissed at: 4:56 PM

Wednesday, August 25, 1999

The current draft has the side-band signals on the PIL/FOP interface are Manchester encoded - an effective 50 MHz clock. Therefore, what is recorded in the current draft will be changed.

These will probably have an EMI issues associated with them. There is goodness in making the side-band signals differential in nature.

After much discussion, the final proposal to consider was just two pins which are used in half-duplex communication mode and provide a differential signal. The diagram below represents a possible implementation of such a scheme:



The data transfer rate (bit rate) of this signal will be 6.144 MHz (base Rate/16). The data rate is slow enough to allow simple over-sampling for data recovery. Clock tolerance will be fairly wide.

We are going to have to look at this data rate a bit more. We can make the message handling between the PIL-FOP on the USB (unified side-band) if they are of fixed sizes - 12 bits would do. Add 1 for start bit and one for stop bit and two for bus turn-around, then we can end up having a 48-bit delay. At 6MHz, this would be about 4.26us worst case delay to get register 0 sent across. It would be preferred that register 0 be sent to the PIL within the amount of time that it takes for the FOP to send its first self-ID packet (about 1us). To do this, we will need to go faster by about a factor of 4 (24MHz).

If we have the optimization of marking PHY response packets as being the end of a sub-action we cannot mark the PHY response packet associated with a suspend or disable as the end of a sub-action. The capacitors in the lines can be eliminated if the bias levels of the portals are compatible. They can't be eliminated if galvanic isolation is required.

PIL FOP issues

Much discussion with regard to the PIL/FOP interface.

There needs to be an analogous "PHY/Link Interface" disable (e.g. LPS not asserted) for the "FOP/PIL" interface.

There are some concerns which must be worked out. A few of them are:

- 1) Loss of synch on PIL/FOP connection
- 2) When the FOP receives a standby request and the PIL connection is the only OTHER port active, does the FOP place the non-PIL port in standby. This was resolved. Answer is yes it does. The PIL port is not counted as an "active" port in any tests except the test where the FOP places its core in standby mode (the PIL port is considered the link connection for the FOP.)

The PHY port on the PIL may have the following properties:

- 1) It may be designed ONLY capable of operating as "standard" beta-only port
- 2) It may be designed as a Beta-only port with PIL-FOP capabilities.
- 3) It may be designed as a bilingual port with PIL-FOP capabilities.
- 4) It may be designed such that it may ONLY capable of operating as a "FOP/PIL" interface connected to a FOP
- 5) It may be designed to provide functionality for both "1" and "2" however its mode of operation shall only be either "1" or "2" (not both)

In any case, when connected to a FOP, the PIL must follow the rules appropriate for the interface

The PIL/FOP interface port on a FOP may have the following properties:

- 1) It may be designed to ONLY be capable of operating as "standard" beta-only port
- 2) It may be designed to function as a Beta only port as well as operate with a PIL
- 3) It may be designed to function as a bilingual port as well as operate with a PIL
- 4) It may be designed such that it may ONLY be capable of operating as a "FOP/PIL" interface when connected to a PIL
- 5) It may provide functionality for both "1" and "2" however its mode of operation shall only be either "1" or "2" (not both)

There are some unique properties of the PIL PHY port and the "Beta" port on the FOP when they are both functioning as a "FOP/PIL" interface.

The PIL (when its PHY port is operating in "FOP/PIL" interface mode):

- 1) Shall not be root (do this by not allowing it to participate in tree-ID (e.g. the PIL comes up knowing that it is operating as a "FOP/PIL" interface))
- 2) Never generates self ID
- 3) does not process link-on packet.
- 4) Does not make use of Gap Count
- 5) The configuration packet is not used by PIL, therefore, the "R" bit in a PHY - configuration packet is not used.
- 6) Resume command -- handled by FOP. Does not resume PIL port
- 7) PIL need not do anything with LTP PHY packets.
- 8) LTS is not sent on PIL/PHY port

The FOP (when its beta port that is connected to the PIL is operating in "FOP/PIL" interface mode):

- 1) Never proxies self ID for PIL: - the PIL never enters "Standby".
- 2) Returns zeros when a read attempt is made on any of the PIL PHY port registers (remote register reads are processed by the FOP and nothing is read from the PIL). When the FOP is designed such that the PIL port can be used as a normal 1394b port, the port registers for the PIL port can't be accessed by a remote register access command

- 3) Reports the port connected to the PIL PHY port as either disconnected or not implemented. Colin prefers not present (00), so does David Wooten.
- 4) Shall use the highest numbered port on the FOP (Jerry thinks this does not need to be specified. At least one person disagrees with Jerry) as the beta port that connects to the PHY port on the PIL.
- 5) Never sends self-ID grant to PIL
- 6) DOES process a link-on packet.

Ping packet does not get processed by PIL but FOP MAY have to do something special to the ping response to accommodate the presence of the PIL. The other thing that is going to be looked at is the PHY delay value reported by the FOP. This may have to be adjusted so that the PING results can be used to set gap counts when PIL is present.

Remote port commands:

- port disable -- not done in PIL or on PIL port on FOP
- suspend -- not done in PIL or on PIL port on FOP
- clear fault conditions -- not done in PIL or on PIL port on FOP
- enable port -- not done in PIL or on PIL port on FOP
- resume port -- not done in PIL or on PIL port on FOP
- standby port -- not done in PIL or on PIL port on FOP
- restore port -- not done in PIL or on PIL port on FOP
- remote response packets never generated by PIL

The following symbols are not sent in either direction on PIL-FOP interface.

- Ident_Done
- Child_Notify
- Parent_Notify
- Standby
- Disable_Notify
- Suspend
- Attach_request

Those in attendance at the IEEE P1394b SCAT meeting (August 23rd and/or August 25th):

23✓	25✓	Name	Company	Email	Phone
✓	✓	Bard, Steve	Intel	Steve.bard@intel.com	503-264-2923
✓		Bassler, Max	Molex	Mbassler@molex.com	630-527-4490
		Brunker, Dave	Molex	Dbnker@molex.com	(630) 527-2622
		Coles, Alistair	HP	anc@hplb.hpl.hp.com	+44 117 922 8750
		Maarten de Vries			
		Dorsey, Chris	ST Microelectronics	christopher.dorsey@st.com	972-466-7850
		Fasano, Lou	IBM	fasano@us.ibm.com	914-892-8904
		Fidler, Mark	Hewlett-Packard		
		Foster, Tony	Hewlett-Packard	tony_foster@hp.com	(916) 785-1092
✓		Hannah, Eric	Intel	eric.hannah@intel.com	408-765-4441
✓	✓	Hauck, Jerry	Zayante, Inc.	jhauck@zayante.com	510-668-1006
		Kakihara, Toshio	IBM	Kakihara@jp.ibm.com	(507) 252-5227
		Kanhere, Prashant	Zayante, Inc.	prashant@zayante.com	510-668-1773
✓		Killeen, Sean	SSL	sean.killeen@ssl.ie	+353 1 402 5700
		Le, Thang	Hewlett-Packard	tl@rose.hp.com	(916) 785-4667
		Lopata, John	Molex	Jlopata@molex.com	(630) 579-4110
		Marazas, Gerald	IBM	marazas@us.ibm.com	919-543-6892
		McDonnell, Edward	HP Labs	emcd@hplb.hpl.hp.com	117-922-8942 (UK)
		Richard Scheel	Sony	Richard.scheel@am.sony.com	(408) 982-5834
		Selander, Carl	IBM		
		Michael Smith	ControlNet	Michael@controlnet.com	(408) 341-1428
		Bivabasu Sarkar	EnThink, Inc.		
		Takayuki Nyu	NEC		
✓	✓	Teener, Michael Johas	Zayante	mike@zayante.com	831-461-4901
		Washburn, Bill	IBM		
✓	✓	Whitby-Strevens, Colin	Zayante	colin@zayante.com	831-461-4948
✓	✓	Wooten, David	Compaq	david.wooten@compaq.com	281-518-7231
		Tadahiro Yoshida	Panasonic		