CONGRUENT SOFTWARE, INC.
98 Colorado Avenue
Berkeley, CA 94707
(510) 527-3926
(510) 527-3856 FAX

FROM: Peter Johansson

TO: IEEE P1394a Ballot Response Committee

DATE: May 2, 1999

RE: Coordination requests from IETF IP1394 and P1394.1 working groups

---

As a consequence of discussions late in 1998 in these working groups (in which I was also an active participant), the Chairs have requested that some work items be transferred to the P1394a draft standard. Specifically:

- that the definition of the network protocol manager (NPM) in the Internet-Draft Draft-IETF-IP1394-IPv4-13 be generalized as the broadcast channel manager (BCM) and be standardized in P1394a instead of the IETF; and
- that a new format for asynchronous stream packets be defined to include *source_ID* in order to permit work in both the IETF and P1394.1 working groups to build upon a common standard.

That work has been incorporated into the draft of P1394a under review by the BRC, but there is also a desire by both the IETF and P1394.1 working groups to critically review the effort. Since the publication of 99-005r0 there has been additional discussion within the P1394.1 working group that has affected the broadcast channel manager (BCM) and the BROADCAST_CHANNEL register.

The rest of this document is an excerpt from the relevant portions of the most recent P1394a draft in preparation for recirculation ballot. The change bars are show the modifications since p1394a Draft 2.0, ***not*** those since 99-005r0.

8.1 Asynchronous stream packet format

The format of an asynchronous stream packet is identical to that of an isochronous stream packet, as specified by clause 6.2.3.1 of IEEE Std 1394-1995, and illustrated by figure 8-1.

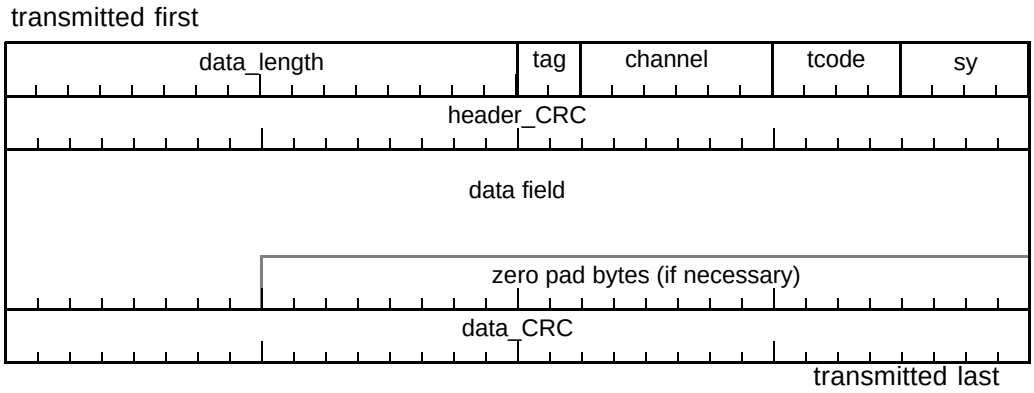

Figure 8-1 — Asynchronous stream packet format

The fields of an asynchronous stream packet shall conform to requirements of this standard and those specified in clause 6.2.4 of IEEE Std 1394-1995.

The ***data_length*** field shall specify the length in bytes of the data field in the asynchronous stream packet The number of bytes in the data field is determined by the transmission speed of the packet and shall not exceed the maximums specified by table 8-1 (which replaces table 6-4 in clause 6.2.2.3 of IEEE Std 1394-1995).

Table 8-1—Maximum data payload for asynchronous primary packets

Data rate	Maximum payload (bytes)	Comment
S25	128	TTL backplane
S50	256	BTL and ECL backplane
S100	512	Cable base rate
S200	1024	
S400	2048	
S800	4096	
S1600	8192	
S3200	16384	

The ***tag*** field shall have a value of zero, unformatted data, or one, global asynchronous stream packet (GASP) format; other values of *tag* are reserved for future standardization. When *tag* is zero the content and format of the data field are unspecified. Otherwise, when *tag* is one the format of the asynchronous stream packet is specified by clause 8.2.

The ***channel*** field shall identify the stream; the channel identified shall be allocated from the isochronous resource manager CHANNELS_AVAILABLE register.

NOTE—Unlike isochronous stream packets, which may continue to be transmitted for up to one second subsequent to a bus reset without channel reallocation, asynchronous stream packets may not be transmitted until their channel number(s) are reallocated.

The ***tcode*** field shall have a value of A_{16}. The new name for this transaction code value is stream packet; the context in which the packet is sent determines whether it is an asynchronous or isochronous stream packet.

The usage of any fields not specified above remains as described by IEEE Std 1394-1995.

8.2 Global asynchronous stream packet (GASP) format

Motivated by work on IEEE P1394.1, Serial Bus to Serial Bus Bridges, this standard defines an asynchronous stream packet format suitable for transport across a bridge from one Serial Bus to another.

The format of a global asynchronous stream packet is an extension of that specified by clause 8.1 which utilizes the first two quadlets of the packet's data payload. The GASP format is illustrated by figure 8-2.

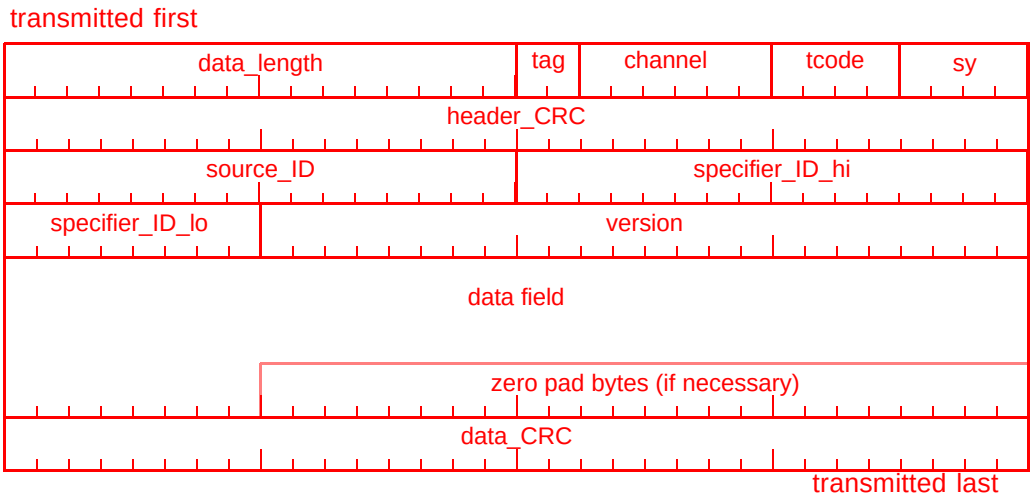

Figure 8-2 — Global asynchronous stream packet (GASP) format

Except as specified below, the definition and usage of the fields in figure 8-2 is contained within clause 8.1.

The ***tag*** field shall have a value of one.

The ***sy*** field is reserved for future standardization and shall be zero.

The ***source_ID*** field shall specify the node ID of the sending node and shall be equal to the most significant 16 bits of the sender's NODE_IDS register.

The ***specifier_ID*** field shall contain a 24-bit organizationally unique identifier (OUI) assigned by the IEEE Registration Authority Committee (RAC). The owner of the OUI (company, accredited standards organization or industry group) shall be responsible to define the meaning and usage of the remainder of the data payload in the stream packet.

The meaning and usage of the ***version*** field shall be defined by the owner of *specifier_ID*.

9.17 ~~NETWORK_CHANNELS~~ BROADCAST_CHANNEL register

~~The CSR architecture reserves a range of addresses in initial register space for bus-dependent uses; clause 8.3.2.3 in IEEE Std 1394-1995 defines registers within that range for use by Serial Bus. This supplement defines a new register, NETWORK_CHANNELS, within that address space whose format and usage is to be specified by the Internet Engineering Task Force (IETF).~~ This register is optional and if implemented shall be a quadlet register located at offset 234_{16} within initial register space ~~FFFF F000 0234$_{16}$ within initial node space~~. Any node that implements the BROADCAST_CHANNEL register shall be isochronous resource manager-capable. All isochronous resource manager-capable nodes should implement the BROADCAST_CHANNELS register.

The BROADCAST_CHANNEL register permits the broadcast channel manager to communicate the channel number assigned for asynchronous stream broadcast to other nodes on Serial Bus. This register also provides a mechanism used by the election protocol for the broadcast channel manager. The BROADCAST_CHANNEL register shall support quadlet read and write requests, only. The definition is given by figure 9-5 below.

Figure 9-5—BROADCAST_CHANNEL format

The most significant bit (a constant one) differentiates the presence of the BROADCAST_CHANNEL register from the value (all zeros) that may be returned when this register's address is read at node(s) that do not implement the register.

NOTE—Nodes compliant with this supplement return an address error response when unimplemented addresses are accessed— but some implementations are known to complete such requests with *resp_OK* and response data of zeros.

The ***valid*** bit (abbreviated as *v* above), when set to one, indicates that the *channel* field contains meaningful information. Nodes shall not transmit stream packets that specify this channel while the *valid* bit is zero.

The ***channel*** field shall be initialized by the broadcast channel manager (see clause 9.22) to identify the channel number shared by all nodes for asynchronous stream broadcast.

9.23 Determination of the root (cable environment)

IEEE Std 1394-1995 clause 8.4.2.6 requires the bus manager or, in the absence of a bus manager, the isochronous resource manager, to set the root node's `force_root` variable to TRUE if the root is cycle master-capable.[1] The `force_root` variable is cleared to FALSE at all other nodes; this promotes stability across bus resets since it increases the likelihood that the same node becomes the root after subsequent bus resets. Although no other rationale is given for setting a particular node's `force_root` variable true, IEEE Std 1394-1995 does not explicitly forbid the use of `force_root` for other purposes. This standard establishes additional policy for the use of the `force_root` variable to influence the selection of the root node during the tree identify process that follows a bus reset.

A node's `force_root` variable shall not be set TRUE unless the node is at least as capable as the current root. A node's `force_root` variable should not be set TRUE unless the node is more capable than the current root.

The first provision permits the current root's `force_root` variable to be set TRUE, as required by IEEE Std 1394-1995.

The second provision necessitates a definition of "more capable." This standard creates an ordering of root node capabilities, which may be extended by future IEEE standards. The most basic of root node capabilities is inherent in all nodes whether or not link and transaction layers are implemented (see IEEE Std 1394-1995 clause 8.3.1.1). Additional capabilities may be implemented by transaction-capable nodes and are enumerated below in increasing order of capability:

a) Cycle master-capable. When an isochronous resource manager or bus manager is present, the root shall be cycle master-capable. A cycle master-capable node may be positively identified by the *cmc* bit in the bus information block or its presence may be inferred by the observation of cycle start packets; and
b) Broadcasts channel manager-capable. A broadcast channel manager-capable node may be identified by the presence of a BROADCAST_CHANNEL register whose most significant bit is one.

In order for this scheme to be extensible, it is crucial that nodes compliant with this and future standards adhere to the second provision and not set their own `force_root` variable TRUE unless they are more capable than the current root. This is particularly important in the case where the current root's `force_root` variable is cleared to FALSE in order that another node become the root. The former root should not attempt to reestablish itself as the root if the new root is at least as capable as the former root. It is likely that the new root is more capable and that the former root is unable to detect it's enhanced capabilities.

Clause 9.25 conforms to these requirements; the only reason a candidate broadcast channel manager is permitted to set its own `force_root` variable TRUE is because it implements all root node capabilities defined by this standard. If a future IEEE standard specifies new functionality for which it is desirable that a particular node be the root, that node shall be both cycle master- and broadcast channel manager-capable in addition to its new capabilities.

[1] IEEE Std 1394-1995 also describes how the bus manager or isochronous resource manager selects a different, cycle master-capable node and sets its `force_root` variable TRUE if the current root cannot function as the cycle master— this process eventually produces a root whose `force_root` variable is TRUE.

9.25 Determination of the broadcast channel manager (cable environment)

The presence of a broadcast channel manager permits all nodes on a local Serial Bus to share a single channel for asynchronous stream broadcast. The broadcast channel manager shall be responsible to allocate a channel number from the CHANNELS_AVAILABLE register and communicate the channel number to other interested nodes on the same bus.

Subsequent to a Serial Bus reset a single broadcast channel manager shall be determined by a distributed algorithm executed by all the broadcast channel manager-capable nodes. The algorithm is straightforward: a broadcast channel manager-capable node eventually shall be selected as the isochronous resource manager, at which point it is also selected as the broadcast channel manager. The steps in the algorithm are as follows:

a) A broadcast channel manager-capable node shall also be a contender for the role of isochronous resource manager. The *C* (contender) and *L* (link active) bits in its self-ID packet shall be set to one;

b) Subsequent to a bus reset, isochronous resource manager contention takes place during the self-identification process specified by IEEE Std 1394-1995;

c) A broadcast channel manager-capable node that wins the contention process is the broadcast channel manager and shall proceed with g). Other broadcast channel manager-capable node(s) not selected as the isochronous resource manager (hereafter referred to as candidates) shall continue with d);

d) A candidate broadcast channel manager that fails to win the contention process shall determine whether or not the isochronous resource manager is also the broadcast channel manager, but it shall delay before the attempt. The delay time shall be at least 15 ms * (*irm_ID* - *candidate_ID*), measured from the completion of the self-identify process that follows bus reset; *irm_ID* and *candidate_ID* are the physical IDs of the isochronous resource manager and the candidate broadcast channel manager, respectively. If the *valid* bit of the candidate's BROADCAST_CHANNEL register is set to one before the delay time elapses, the isochronous resource manager is also the broadcast channel manager and the algorithm terminates;

e) Otherwise, once the delay time elapses, a candidate broadcast channel manager shall read the BROADCAST_CHANNEL register at the isochronous resource manager. If the register is implemented, the broadcast channel manager exists and the algorithm terminates. The candidate should use the information read from the BROADCAST_CHANNEL register to initialize the *valid* bit and *channel* field in its own BROADCAST_CHANNEL register;

f) If the current isochronous resource manager does not implement the BROADCAST_CHANNEL register, a new isochronous resource manager that is also broadcast channel manager-capable shall be selected. A candidate broadcast channel manager shall set its own `force_root` variable TRUE, clear those of all other nodes to FALSE and then initiate a bus reset. The broadcast channel manager selection algorithm resumes with a);

g) The broadcast channel manager shall allocate channel number 31 from its own CHANNELS_AVAILABLE register and then update its own BROADCAST_CHANNEL register with the allocated channel number and set the *valid* bit to one. The broadcast channel manager shall then write the updated value of the entire register to all the BROADCAST_CHANNEL registers implemented by any of the nodes on the bus. Either a broadcast write request or a series of write requests addressed to individual nodes shall be used to propagate the information.