Host Access Protocol (HAP) Specification - Version 2

Status of this Memo

This memo describes the Host Access Protocol implemented in the
Terrestrial Wideband Network (TWBNET). It obsoletes most but not all
of RFC 907. This memo provides information for the Internet
community. It does not specify an Internet standard. Distribution
of this memo is unlimited.

Preface

This memo specifies the Host Access Protocol (HAP). HAP is a Network
layer (OSI Layer 3 lower) access protocol that was first implemented
about a decade ago for the DARPA/DCA sponsored Wideband Packet
Satellite Network (WBNET), the precursor of the current Terrestrial
Wideband Network (TWBNET). This version of the specification
obsoletes references [1] and [2] in addition to most of RFC 907.

HAP is a developmental protocol, and will be revised as new
capabilities are added and unused features are eliminated or revised.
One reason that HAP is being revised now is that, unlike the original
WBNET's satellite channel, the TWBNET's T1 fiber links are not a
broadcast medium. This has prompted some changes to the protocol
that will permit greater efficiency in a mesh topology network.
Another cause of revision is the need to make HAP able to support a
variety of OSI layer 3 upper protocols, such as DECNET Phase V, ST,
and CLNP, where before only Internet Protocol (IP) was used.
Appendix B describes how backward compatibility with the older IP-
only version of HAP is achieved. A third cause of protocol changes
is the desire to simplify interaction between ST2 protocol (RFC 1190)
agents and the TWBNET. This has mainly affected the way certain
setup errors are handled. These changes are expected to be backward
compatible. Appendix A describes two capabilities that may be added
to HAP in the future.

One of the protocol enhancements, "Group Streams", described in
reference [2] has been eliminated. There are no known applications
that use the feature. As described in Appendix A, a new mechanism,
to be called "shared streams", capable of providing equivalent
capabilities will be implemented if needed. Changes in [2] that have
been retained include various query/reply control messages that
permit a host to determine what resources it owns (mostly useful for

cleanup following a host reboot or crash).

This document assumes the reader is familiar with DoD internetworking
terminology.

1. Introduction

The Host Access Protocol (HAP) is a network layer protocol (as is
X.25). ("Network layer" here means ISO layer 3 lower, the protocol
layer below the DoD Internet Protocol (IP) layer [3] and above any
link layer protocol.) HAP defines the different types of host-to-
network control messages and host-to-host data messages that may be
exchanged over the access link connecting a host and the network
packet switch node. The protocol establishes formats for these
messages, and describes procedures for determining when each type of
message should be transmitted and what it means when one is received.

HAP has been implemented in the wide-area network called the
Terrestrial Wideband Network (TWBNET) [5] and in the routers and
other hosts that connect to TWBNET. The packet switch nodes that
compose the TWBNET are called Wideband Packet Switches (WPS).

Both the precursor to HAP, the Host/SATNET Protocol [6], used in the
Atlantic Packet Satellite Network (SATNET) and the Mobile Access
Terminal Network (MATNET [7]), and HAP, used in the original Wideband
Satellite Network (WBNET) [8], were originally designed to provide
efficient access to the single satellite channel each network used to
connect all sites. The HAP protocol designers reflected some of the
peculiarities of the single satellite channel environment in the HAP
protocol itself. The current Terrestrial Wideband Network (TWBNET)
utilizes T1-speed fiber connections between sites. Future networks
and TWBNET may use a combination of terrestrial connections and
satellite connections, and may have more than one of each. The HAP
protocol has been changed to accommodate these extensions.

Section 2 presents an overview of HAP. Details of HAP formats and
message exchange procedures are contained in Sections 3 through 10.
Further explanation of some of the topics addressed in this HAP
specification can be found in reference [1].

Any protocol employed to provide sufficiently reliable message
exchange over the Host-WPS link is assumed to be transparent to the
protocol defined in this document. Examples of such link-level
protocols are ARPANET 1822 local and distant host [9], ARPANET VDH
protocol [9], and HDLC.

2. Overview

   HAP can be characterized as a full duplex, nonreliable protocol with
   an optional flow control mechanism. HAP messages flow simultaneously
   in both directions between the WPS and the host. Transmission is
   nonreliable in the sense that the protocol does not provide any
   guarantee of error-free sequenced delivery. If error-free delivery
   on the host's access link is required, it must be provided by the
   link layer protocol below HAP. (Use of link layer protocols for this
   purpose is not within the scope of this document.) HAP's flow
   control mechanism operates independently in each direction, but the
   choice to enable flow control or not applies to both directions
   together.

   HAP supports host-to-host communication in two modes corresponding to
   the two types of HAP data messages, datagram messages and stream
   messages. Each type of message can be up to 2048 octets in length.
   The basic transmission service in the network is datagram service.
   Datagrams are variable length, unsequenced, independent, and delivery
   is not guaranteed. The HAP header of each datagram determines the
   processing of the message.

   On this datagram service base a "stream" service is built. Stream
   service provides network bandwidth guarantees, but requires explicit
   setup and teardown operations to allocate and deallocate network
   resources. Stream traffic is best suited for continuous media
   traffic, but may also be used to obtain the lowest possible network
   delay. Host streams are established by a setup message exchange
   between the host and the network prior to the commencement of data
   flow. Although established host streams can have their
   characteristics modified by subsequent setup messages while they are
   in use, the fixed allocation properties of streams relative to
   datagrams impose rather strict requirements on the source of the
   traffic using the stream. Stream traffic arrivals must match the
   stream allocation both in interarrival time and message size if
   reasonable efficiency is to be achieved. The characteristics and use
   of datagrams and streams are described in detail in Sections 3 and 4
   of this document.

   Both datagram and stream transmission in the network use logical
   addressing. Each host on the network is assigned a permanent 16-bit
   logical address which is independent of the physical port on the WPS
   to which it is attached. These 16-bit logical addresses are present
   in all Host-to-WPS and WPS-to-Host data messages.

   HAP supports multicast addressing via "groups". Multicast addressing
   is provided primarily to support the multi-destination delivery
   required for conferencing applications. Group addresses are

dynamically created and deleted by the use of setup messages
exchanged between a host and the WPS. Membership in a group may be
any arbitrary subset of the network hosts. A message addressed to a
group address is delivered to all hosts that are members of that
group, except the sender. Once a multicast address has been created,
any member host may use that address, not just the creator.

Although HAP does not guarantee error-free delivery, error control is
an important aspect of the protocol design. HAP error control is
concerned with both local transfers between a host and its local WPS
and transfers through the network to the destination(s). The WPS
offers users a choice of network error protection options based on
the network's ability to selectively send messages over its
transmission media at different forward error correction (FEC) rates.
These FEC options are referred to as reliability levels. Four
reliability levels (low, medium-low, medium-high, and high) are
available. The precise error rate provided by each reliability level
is not specified.

Various checksum and CRC mechanisms are employed in the network to
provide an error detection capability. A host has an opportunity
when sending a message to indicate whether the message should be
delivered to its destination or discarded if a data error is detected
by the network. Each message received by a host from the network
will have a flag indicating whether or not an error was detected in
that particular message. A host can decide on a per-message basis
whether or not it wants to accept or discard transmissions containing
data errors.

For connection of a host and WPS in close proximity, error rates due
to external noise or hardware failures on the access circuit may
reasonably be expected to be much smaller than the best network trunk
circuit error rates. Thus for this case, little is gained by using
error detection and retransmission on the access circuit. A 16-bit
header checksum is provided, however, to ensure that WPSen do not act
on incorrect control information. For relatively long distances or
noisy connections, retransmissions over the access circuit may be
required to optimize performance for both low and high reliability
traffic. It is expected that link layer error control procedures
(such as HDLC with retransmission) will be used for this purpose, but
use of a reliable link layer protocol is not within the scope of this
document.

Each datagram message submitted to the WPS by a host is marked as
being in one of three priority classes, from priority 2 (highest)
through priority 0 (lowest). The priority class is used by the WPS
for arbitrating contention for scarce network resources (e.g., link
bandwidth). That is, if the network cannot deliver all of the

offered messages, high priority messages will be delivered in
preference to low priority messages. Priority level affects the
order of access to intersite link bandwidth and the order of message
delivery at the destination WPS.

Each stream message also has three priority classes, from priority 2
(highest) through priority 0 (lowest). In addition, streams
themselves have three precedence classes, from precedence 2 (highest)
through precedence 0. A stream of higher precedence can preempt a
stream of lower precedence at setup time. Stream message priority
provides a mechanism for a low-bandwidth host to receive a high-
bandwidth stream and selectively discard messages marked as less
important by the sender. Stream message priority does not affect the
order of delivery of stream messages between the source and the
destination.

Datagram and stream messages being presented to the WPS by a host may
not be accepted for a number of reasons: priority too low,
destination dead, lack of buffers in the source WPS, etc. The host
faces a similar situation with respect to handling messages from the
WPS. To permit the receiver of a message to inform the sender of the
local disposition of its message, an acceptance/refusal (A/R)
mechanism is implemented. The mechanism is the external
manifestation of the WPS's (or host's) internal flow and congestion
control algorithm. If A/Rs are enabled, an explicit or implicit
acceptance or refusal for each message is returned to the host by the
WPS (and conversely). This allows the host (or WPS) to retry refused
messages at its discretion and can provide information useful for
optimizing the sending of subsequent messages when the reason for
refusals is also provided. The A/R mechanism can be disabled to
provide a "pure discard" interface. The host's choice to use the A/R
mechanism or not does not limit its ability to send and receive
messages to any other hosts.

While the A/R mechanism allows control of individual message
transfers, it does not facilitate regulation of priority flows. Such
regulation is handled by passing advisory status information (GOPRI)
across the Host-WPS interface indicating which priorities are
currently being accepted. As long as this information, relative to
the change in priority status, is passed frequently, the sender can
avoid originating messages which are sure to be refused.

HAP defines both data messages (datagram messages and stream
messages) and link control messages. Data messages are used to send
information between hosts on the network. Link control messages are
exchanged between a host and the WPS to manage the local access link.

Allocation of network resources, such as streams and groups, is

accomplished via an exchange of datagram messages, called Setups,
between the user host and an agent inside the WPS called the "Service
Agent." Setups are used to reserve, allocate, modify, free, and
deallocate network resources. Each allocated resource has a unique
identifier which, when placed in an appropriate field in a message
header, allows that message to use the resource. E.g., after an
exchange of Setups to create a group address, a message may be sent
to the group by placing the group address in the destination field of
that message. The Service Agent also permits a host to inquire about
resources it owns.

Every HAP message consists of an integral number of 16-bit words
(i.e., an even number of octets). The first several words of the
message always contain control information and are referred to as the
message header. The first word of the message header identifies the
type of message which follows. The second word of the message header
is a checksum which covers all header information. Any message whose
received header checksum does not match the checksum computed on the
received header information must be discarded. The format of the
rest of the header depends on the specific message type.

The formats and use of the individual message types are detailed in
the following sections. A common format description is used for this
purpose. Words in a message are numbered starting at zero (i.e.,
zero is the first word of a message header). Bits within a word are
numbered from zero (most significant) to fifteen (least significant).
The notation used to identify a particular field location is:

   <WORD#>{-<WORD#>} [<BIT#>{-<BIT#>}] <description>

where optional elements in {} are used to specify the (inclusive)
upper limit of a range. The reader should refer to these field
identifiers for precise field size specifications. Fields which are
common to several message types are defined in the first section
which uses them. Only the name of the field will usually appear in
the descriptions in subsequent sections.

Link-level protocols used to support HAP can differ in the order in
which they transmit the bits constituting HAP messages. The words of
the message are transmitted from word 0 to word N.

3. Datagram Messages

Datagrams are one of the two message types provided by HAP, as
described in the previous section. Because network resources are not
reserved in advance for datagram traffic, delivery of datagram
traffic is subject to greater delivery delays and delay variance than
stream traffic, and is subject to flow and congestion controls.

Datagram priority determines which packets are delivered or discarded
when network resources do not permit handling all of the presented
traffic. It is expected that datagram messages will be used to
support the majority of computer-to-computer and terminal-to-computer
traffic which is bursty in nature.

The format of datagram messages and the purpose of each of the header
control fields is described in Figure 1.

```
         0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
0       | 0|LB|GOPRI|     0     | F|    MESSAGE NUMBER   |
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
1       |                 HEADER CHECKSUM                |
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
2       |                      A/R                       |
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
3       | 0|IL| D| E| PRI | TTL | RLY |      RLEN         |
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
4       |             DESTINATION HOST ADDRESS           |
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
5       |               SOURCE HOST ADDRESS              |
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
6       |                  PROTOCOL ID                   |
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        |                                                |
7-N     :                      DATA                      :
        |                                                |
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
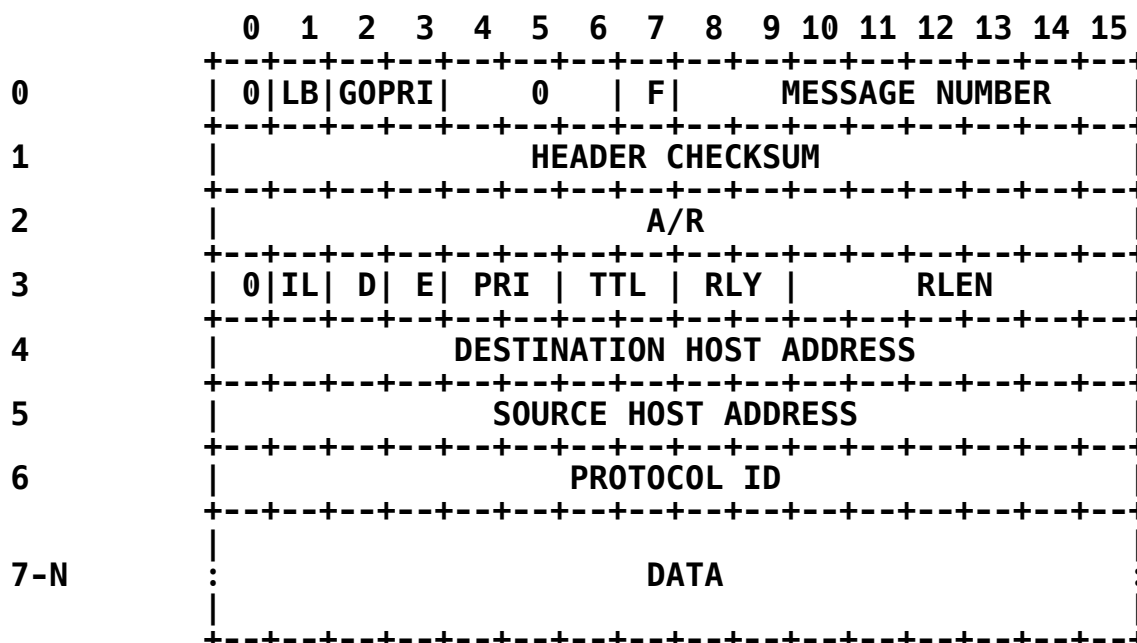
DATAGRAM MESSAGE
Figure 1

0[0]   Message Class. This bit identifies the message as a
            data message or a control message.

                0 = Data Message
                1 = Control Message

0[1]   Loopback indicator. This bit allows the sender of a
            message to determine if its own messages are being
            looped back. The host and the WPS each use different
            settings of this bit for their transmissions. If a
            message arrives with the loopback bit set equal to its

outgoing value, then the message has been looped.

            0 = Sent by Host
            1 = Sent by WPS

0[2-3]   Go-Priority. In WPS-to-Host messages, this field
          provides advisory information concerning the lowest
          priority currently being accepted by the WPS. The host
          may optionally choose to provide similar priority
          information to the WPS.

            0 = Low Priority
            1 = Medium Priority
            2 = High Priority
            3 = (Reserved.)

0[4-6]   Reserved. Must be zero.

0[7]   Reserved. Must be zero. Formerly used for WPS
          diagnostic purposes.

0[8-15]   Message Number. This field contains the identification
          of the message used by the acceptance/refusal (A/R)
          mechanism (when enabled). If the message number is
          zero, A/R is disabled for this specific message. See
          Section 5 for a detailed description of the A/R
          mechanism.

1[0-15]   Header Checksum. The checksum is the 2's-complement of
          the 2's-complement sum of words 0-6 (excluding the
          checksum word itself).

2[0-15]   Piggybacked A/R. This field may contain an
          acceptance/refusal word providing A/R status on traffic
          flowing in the opposite direction. Its inclusion may
          eliminate the need for a separate A/R control message
          (see Section 5). A value of zero for this word is used
          to indicate that no piggybacked A/R information is
          present.

3[0]   Data Message Type. This bit identifies whether the
          message is a datagram message or a stream message.

            0 = Datagram Message
            1 = Stream Message

3[1]   IL flag. Obsolete. Must be zero. (See Appendix B.)

3[2]   Discard Flag. This flag allows a source host to
            instruct the network (including the destination host)
            what to do with the message when data errors are
            detected (assuming the header checksum is correct).

                0 = Discard message if data errors detected.
                1 = Don't discard message if data errors detected.

            The value of this flag, set by the source host, is
            passed on to the destination host.

3[3]   Data Error Flag. This flag is used in conjunction with
            the Discard Flag to indicate to the destination host
            whether any data errors have been detected in the
            message prior to transmission over the destination's
            WPS-to-Host access link. It is used only if Discard
            Flag = 1. It should be set to zero by the source host.

                0 = No Data Errors Detected
                1 = Data Errors Detected

3[4-5]   Priority. The source host uses this field to specify
            the priority with which the message should be handled
            within the network.

                0 = Low Priority
                1 = Medium Priority
                2 = High Priority
                3 = (Reserved.)

            The priority of each message is passed to the
            destination host by the destination WPS.

3[6-7]   Time-to-Live Designator. The source host uses this
            field to specify the maximum time that a message should
            be allowed to exist within the network before being
            deleted. Elapsed time begins when the message has been
            received by the WPS from the source host (or is sent by
            a WPS agent) and is last checked when the message is
            queued for transmission out the I/O interface to the
            destination host. If a message is multicast, each copy
            is treated separately.

                0 = 1 seconds
                1 = 2 seconds
                2 = 5 seconds
                3 = 10 seconds

3[8-9]   Reliability. The source host uses this field to
            specify the basic bit error rate requirement for the
            data portion of this message. The source WPS uses this
            field to determine the trunk circuit transmission
            parameters and forward error correction level required
            to provide that bit error rate.

                0 = Low Reliability
                1 = Medium-Low Reliability
                2 = Medium-High Reliability
                3 = High Reliability

3[10-15]   Reliability Length. The source host uses this field to
            specify a portion of the user data which should be
            transmitted at the highest reliability level (lowest
            bit error rate). Both the HAP message header words and
            the first 2*<Reliability Length> octets of user data
            will be transmitted at high reliability while the
            remainder of the user data will be transmitted at
            whatever reliability level is specified in field 3[8-
            9]. The reliability length mechanism gives the user
            the ability to transmit private header information
            (e.g., IP and TCP headers) at a higher reliability
            level than the remainder of the data.

4[0-15]   Destination Host Address. This field contains the
            network logical address of the destination host.

5[0-15]   Source Host Address. This field contains the network
            logical address of the source host.

6[0-15]   Protocol ID. This field specifies the next higher
            level protocol. Protocol identifiers are assigned
            administratively, except 0 which is reserved, and are
            not part of this specification. See reference [10].

7-N   Data. This field contains up to 16,384 bits (2048
            octets) of user data, and must be an even number of
            octets.

4. Stream Messages

   Stream messages are the second message type provided by HAP, as
   described in Section 2. Streams provide guaranteed bandwidth between
   the source and destination(s), and provide the minimum delivery delay
   and delay variance available in the network. Streams are suitable
   for volatile traffic, such as speech, and for support of high duty
   cycle applications that require throughput guarantees.

Streams must be created before stream messages can flow from host to
host. The protocol to accomplish stream creation is described in
Section 6.1. Once established, a stream is allocated specific
network resources, such as bandwidth. Within the bounds of its
stream allocation, a host is permitted considerable flexibility in
how it may use the stream. Although the time to live, reliability,
and reliability length of each stream message is fixed at stream
setup time, the destination logical address can vary from stream
message to stream message.

A host can, therefore, multiplex a variety of logical flows onto a
single stream, as long as the stream was set up to reach all the
destination hosts. The format of stream messages is described in
Figure 2.

```
            0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  0        | 0|LB|GOPRI|       0       |   MESSAGE NUMBER   |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  1        |                  HEADER CHECKSUM               |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  2        |                       A/R                      |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  3        | 1|IL| D| E| PRI |       HOST STREAM ID         |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  4        |              DESTINATION HOST ADDRESS          |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  5        |                SOURCE HOST ADDRESS             |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  6        |                    PROTOCOL ID                 |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           |                                                |
  7-N      :                       DATA                     :
           |                                                |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
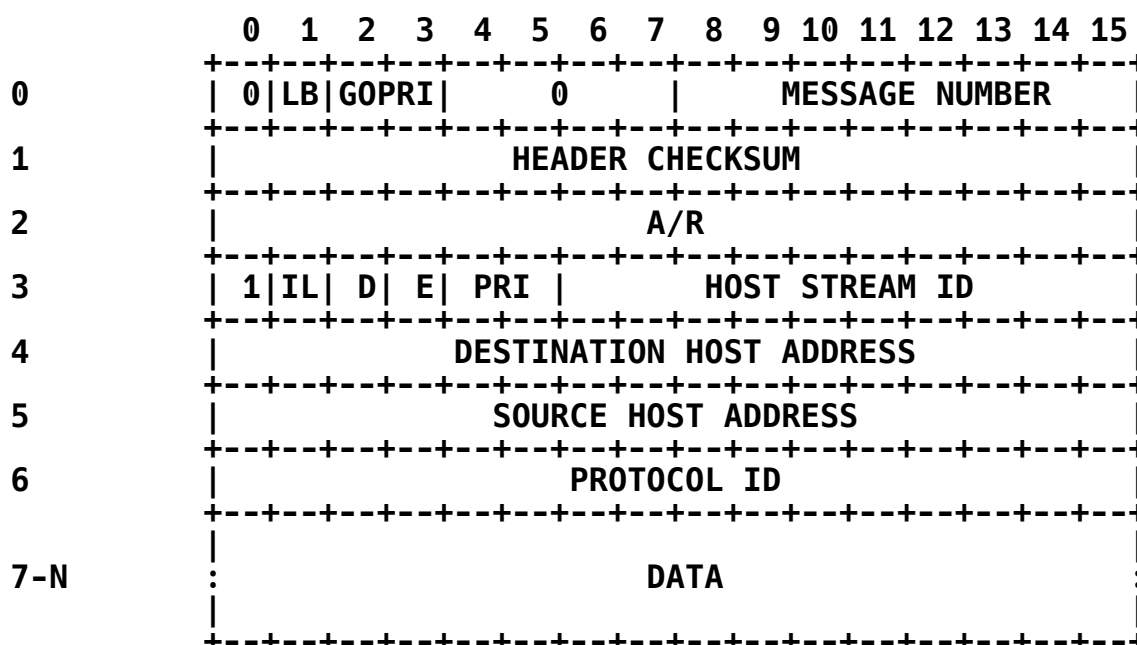
                          STREAM MESSAGE
                            Figure 2

  0[0]   Message Class = 0 (Data Message).

  0[1]   Loopback indicator.

  0[2-3]   Go-Priority.

0[4-7]   Reserved.

0[8-15]   Message Number. This field serves the same purpose as
            the message number field in the datagram message.
            Moreover, a single message number sequence is used for
            both datagram and stream messages (see Section 5).

1[0-15]   Header Checksum. (See datagram checksum for
            description.)

2[0-15]   Piggybacked A/R.

3[0]   Data Message Type = 1 (Stream).

3[1]   IL flag. Obsolete. Must be zero.

3[2]   Discard Flag.

3[3]   Data Error Flag.

3[4-5]   Stream message priority. Note that all stream messages
            have priority over any datagram message. Priority will
            not affect the order of stream message delivery.

                  0 = Low priority
                  1 = Medium priority
                  2 = High priority
                  3 = Reserved

3[6-15]   Stream ID. The WPS uses this field to identify the
            preallocated network resources (bandwidth allocations,
            queues, buffers, etc.) to use for delivery of the
            message. Streams and their identifying numbers (stream
            IDs) are established by an explicit Create Stream
            request (see Section 6.1).

4[0-15]   Destination Host Address.

5[0-15]   Source Host Address.

6[0-15]   Protocol ID.

7-N   Data. This field contains up to 16,384 bits (2048
            octets) of user data, and must be an even number of
            octets.

5. Flow Control Messages

   The WPS supports an acceptance/refusal (A/R) mechanism in each
   direction on the host access link. The A/R mechanism is enabled for
   the link by the host by setting a bit in the Restart Complete control
   message (see Section 8). Each datagram and stream message contains
   an 8-bit message number used to identify the message for flow control
   purposes. When the A/R mechanism is enabled, the message number is
   incremented modulo 256 in successive messages, skipping over message
   number zero (zero indicates that A/R's are disabled for that
   message). Up to 127 messages may be outstanding (awaiting acceptance
   or refusal) in each direction. If the receiver of a message is
   unable to accept the message, a refusal indication containing the
   message number of the refused message and the reason for the refusal
   is returned. The refusal indication may be piggybacked on data
   messages in the opposite direction over the link or may be sent in a
   separate control message in the absence of reverse data traffic.

   Acceptance indications are returned in a similar manner, either
   piggybacked on data messages or in a separate control message. An
   acceptance is returned by the receiver to indicate that the
   identified message was received from the host access link and was not
   refused. Acceptance indications returned by the WPS are not an end-
   to-end acknowledgement and do not imply any guarantee of delivery to
   the destination host(s), or even any assurance that the message will
   not be intentionally discarded by the network. They are sent
   primarily to facilitate buffer management in the host.

   To reduce the number of A/R messages exchanged, a single A/R
   indication can be returned for multiple (lower numbered) previously
   unacknowledged messages. Explicit acceptance of message number N
   implies implicit acceptance of outstanding messages with numbers N-1,
   N-2, etc., according to the definition of acceptance outlined above.
   Analogous interpretation of the refusal message number allows the
   receiver of a group of messages to reject them as a group when they
   all are being refused for the same reason. As a further efficiency
   measure, HAP permits aggregation of any mix of A/R indications into a
   single A/R control message. Such a message might be used, for
   example, to reject a group of messages where the refusal code on each
   is different.

   In some circumstances the overhead associated with processing A/R
   messages may prove unattractive. For these cases, it is possible to
   disable the A/R mechanism and operate the HAP interface in a purely
   discard mode. The ability to effect this on a link basis has already
   been noted (see Sections 2 and 8). In addition, messages with
   sequence number zero are taken as messages for which the A/R
   mechanism is selectively disabled. To permit critical feedback, even

when operating in discard mode, HAP defines an "Unnumbered Response"
control message. Flow control information, and other information
which cannot be sent as an A/R indication, is sent in an Unnumbered
Response control message. The format of this type of message is
illustrated in Figure 5.

The format shown in Figure 3 is used both for A/R indications that
are piggybacked on data messages (word 2), and for aggregated A/R
information in A/R control messages. The format of A/R control
messages is shown in Figure 4.

```
     0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
   |AR|     REFUSAL CODE     |   A/R MESSAGE NUMBER    |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
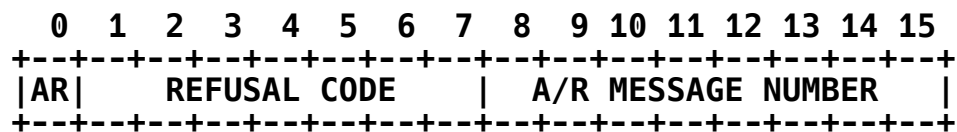
ACCEPTANCE/REFUSAL WORD
Figure 3

[0]   Acceptance/Refusal Type. This field identifies whether
              A/R information is an acceptance or a refusal.

                  0 = Acceptance
                  1 = Refusal

[1-7]   Refusal Code. When the Acceptance/Refusal Type = 1,
              this field gives the Refusal Code.

                   0 = Priority not being accepted
                   1 = Source WPS congestion
                   2 = Destination WPS congestion
                   3 = Destination host dead
                   4 = Destination WPS dead
                   5 = Illegal destination host address
                   6 = Destination host access not allowed
                   7 = Illegal source host address
                   8 = Message lost in access link
                   9 = Invalid stream ID
                  10 = Illegal source host for stream ID
                  11 = Message length too long
                  12 = Stream message too early
                  13 = Illegal control message type
                  14 = Illegal refusal code in A/R
                  15 = Can't implement loop

```
                    16 = Destination host congestion
                    17 = Delivery refused
                    18 = Odd byte length packet (not allowed)
                    19 = Invalid stream time-to-live value
                    20 = "Reliability length" exceeds message length
```

[8-15]   A/R Message Number. This field contains the number of
            the message to which this acceptance/refusal refers.
            It also applies to all outstanding messages with
            earlier numbers. Note that this field can never be
            zero since a message number of zero implies that the
            A/R mechanism is disabled.


```
            0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
0           | 1|LB|GOPRI|      0      |   LENGTH  |     1     |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
1           |                HEADER CHECKSUM                 |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
            |                                               |
2-N         :                    A/R's                      :
            |                                               |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
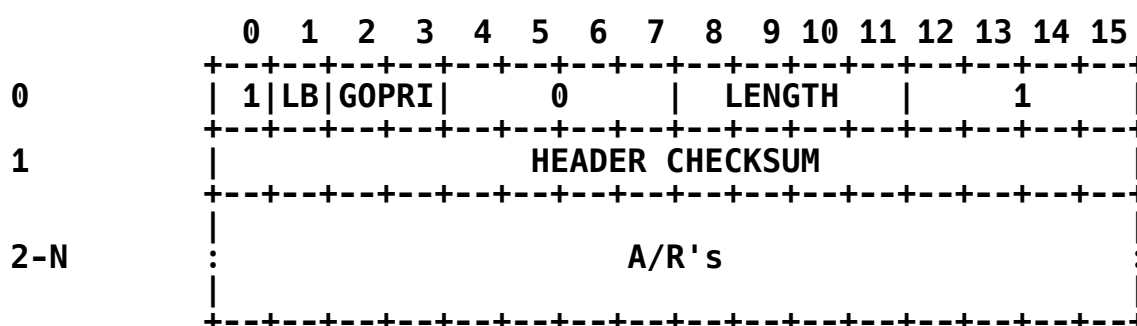
                    ACCEPTANCE/REFUSAL MESSAGE
                            Figure 4

0[0]   Message Class = 1 (Control Message).

0[1]   Loopback indicator.

0[2-3]   Go-Priority.

0[4-7]   Reserved.

0[8-11]   Message Length. This field contains the total length
            of this message in words (N+1).

0[12-15]   Control Message Type = 1 (Acceptance/Refusal).

1[0-15]   Header Checksum. The checksum is the 2's-complement of
            the 2's-complement sum of words 0-N (excluding the
            checksum word itself).

2[0-15]   Acceptance/Refusal Word.

3-N   Additional Acceptance/Refusal Words (optional).


```
             0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
0           | 1|LB|GOPRI|       0        | RES-CODE |    5    |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
1           |                 HEADER CHECKSUM                |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
2           |                 RESPONSE  INFO                 |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
3           |                 RESPONSE  INFO                 |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

                     UNNUMBERED RESPONSE
                         Figure 5

0[0]   Message Class = 1 (Control Message).

0[1]   Loopback indicator.

0[2-3]   Go-Priority.

0[4-7]   Reserved.

0[8-11]   Response Code.

                  3 = Destination unreachable
                  5 = Illegal destination host address
                  7 = Illegal source host address
                  9 = Nonexistent stream ID
                 10 = Illegal stream ID
                 13 = Protocol violation
                 15 = Can't implement loop

0[12-15]   Control Message Type = 5 (Unnumbered Response).

1[0-15]   Header Checksum. The checksum is the 2's-complement of
           the 2's-complement sum of words 0-3 (excluding the
           checksum word itself).

2[0-15]   Response Information. If Response Code is:

```
                3: Destination Host Address
                5: Destination Host Address
                7: Source Host Address
                9: Stream ID (right justified)
               10: Stream ID (right justified)
               13: Word 0 of offending message
               15: Word 0 of Loopback Request message
```

   3[0-15]   Response Information. If Response Code is:

```
               3,5,7, or 9: Undefined
               10: Source Host Address
               13: Word 3 of offending message, or 0 if no word 3
               15: Word 2 of Loopback Request message
```

6. The Service Agent

   Allocation of network resources, such as streams and groups, is
   accomplished via an exchange of datagram messages, called Setup
   messages, between the user host and the Service Agent (network
   address zero). Setup operations include reserving, allocating,
   modifying, freeing, and deallocating resources. The Service Agent
   causes the requested action to be carried out and serves as the
   intermediary between the user and the rest of the network. In the
   process of implementing the requested action, various network data
   bases are updated to reflect the current state of the referenced
   resource. The Service Agent also permits a host to inquire about
   resources it owns using Information Request and Information Reply
   messages.

   A setup interaction initiated by a host involves a 3-way exchange
   where: (1) the requesting host sends a Setup Request to the Service
   Agent, (2) the Service Agent returns a Setup Reply to the requesting
   host, and (3) the requesting host returns a Setup Acknowledgment to
   the Service Agent. This procedure is used to ensure reliable
   transmission of Setup Requests and Replies. In order to allow more
   than one Setup Request message from a host to be outstanding, each
   Request is assigned a unique Request ID. The associated Reply and
   subsequent Acknowledgment are identified by the Request ID that they
   contain. The requesting host should receive a reply to a setup
   request within 3 seconds. The actual delay will depend on the nature
   of the request and the topology of the network. For simple networks,
   the delay will often be less than one second. The requesting host
   should respond to a Reply with a Setup Acknowledgment within one
   second.

   Setup exchanges initiated by the Service Agent involve a two-way
   exchange where: (1) the Service Agent sends a Notification to

affected hosts, and (2) the hosts return a Setup Acknowledgment to
the Service Agent. Notifications are used to inform a host of
changes in the status of a network resource. In order to allow more
than one Notification to be outstanding, each is assigned a unique
Notification ID. The Setup Acknowledgment returned by the notified
host to the Service Agent must contain the Notification ID. The host
should respond within one second.

An information query is initiated by a host and involves a two-way
exchange where: (1) the host sends an Information Request message to
the Service Agent, and (2) the Service Agent sends back an
Information Reply. There is no acknowledgment mechanism, since this
request does not change any resource allocation. Furthermore, if
there is an error in the request, only one response will be sent by
the WPS, and the WPS will make no effort to check for or retransmit
lost responses. It is the responsibility of the host to wait a
certain amount of time and then determine that an unanswered
information request has been lost and to resend it. (The time
necessary to answer such a request is usually much less than one
second.) The WPS will return the message ID of the information
request in the information reply message.

The general format of all Service Agent messages is:

                        <DATAGRAM MESSAGE HEADER>
                         <SERVICE AGENT HEADER>
                            <MESSAGE BODY>

The Protocol ID field in the datagram message header must be
HAP_PROTO_SETUP (1) (see Appendix C) for messages sent to the Service
Agent and will be HAP_PROTO_SETUP in messages received from the
Service Agent. The Service Agent does not recognize or support use
of other higher level protocols (e.g., IP), in setup messages, and
will discard messages containing such headers.

Illustrations of message formats below show only the Service Agent
Header header and message body and do not include the datagram
message header. As a reminder that the datagram header is not
included, word offsets are prefixed with an "S".

The format of the Service Agent Header is illustrated in Figure 6.
The body of the message will depend on the particular message type.
Stream Request and Reply messages are described in Section 6.1.
Group Request and Reply messages are described in Section 6.2. The
format of Notifications is described in Section 6.3, and Setup
Acknowledgments are described in Section 6.4. Information Request
and Reply messages are described in Section 6.5.

```
                 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S0      |     MESSAGE TYPE      |          CODE         |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S1      |                    CHECKSUM                   |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S2      |                   MESSAGE ID                  |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
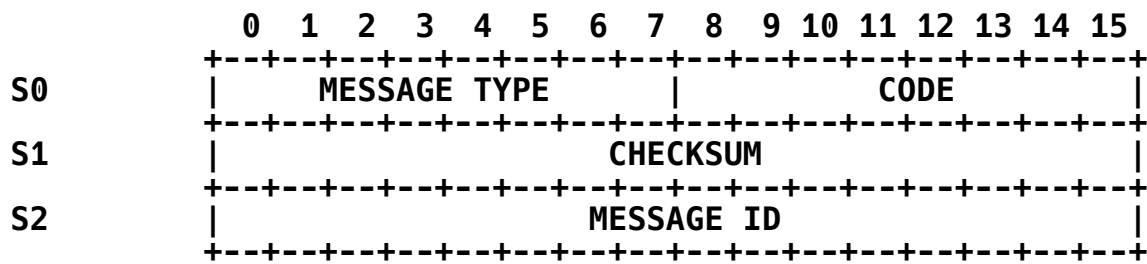
                          SERVICE AGENT HEADER
                              Figure 6

    S0[0-7]   Message Type. This field determines the type of
               message.

                   0 = Setup Acknowledgment
                   1 = Setup Request
                   2 = Setup Reply
                   3 = Notification
                   4 = Information Request
                   5 = Information Reply

    S0[8-15]   Code. For Setup Requests, this field identifies the
               request type.

                   1 = Create group (multicast) address
                   2 = Delete group address
                   3 = Join group
                   4 = Leave group
                   5 = Create stream
                   6 = Delete stream
                   7 = Change stream
                   8 = Create shared stream
                   9 = Delete all streams owned by this host
                  10 = Add member to group
                  11 = Remove member from group

               For Setup Replies, this field provides the Reply Code.
               Some of the Reply Codes can be returned to any setup
               request and others are request specific.

                   0 = Group or stream created
                   1 = Group or stream deleted
                   2 = Host added to group
                   3 = Host deleted from group
                   4 = Stream changed

```
                    5 = (Reserved)
                    6 = Request type invalid or unsupported
                    7 = (Reserved)
                    8 = Network trouble
                    9 = Bad group key
                   10 = Group address/stream ID nonexistent
                   11 = Not member of group/not creator of stream
                   12 = Stream precedence not being accepted
                   13 = (Reserved)
                   14 = (Reserved)
                   15 = (Reserved)
                   16 = Unable to add all the new hosts
                   17 = Insufficient network resources
                   18 = Requested bandwidth too large
                   19 = (Reserved)
                   20 = (Reserved)
                   21 = Maximum messages per interval too small
                   22 = Reply lost in network
                   23 = Illegal priority or precedence value
                   24 = Invalid address provided
```

For Notifications, this field contains the Notification Type. (See Section 6.3.)

For Setup Acknowledgments, this field contains the Acknowledgment Type. (See Section 6.4.)

For Information Requests, this field contains the request type. (See Section 6.5.)

For Information Replies, this field contains the reply type. (See Section 6.5.)

S1[0-15] Checksum. The checksum is the 2's-complement of the 2's-complement sum of the words in the Service Agent Header (excluding the checksum word itself) and the message body. Messages received with bad checksums must be discarded.

S2[0-15] Message ID. This field is assigned by the host to uniquely identify outstanding requests (Request ID) and by the Service Agent to uniquely identify outstanding notifications (Notification ID).

6.1. Stream Setup Messages

Streams provide a means of reserving network resources for the delivery of traffic at a specified maximum throughput to a specified

list of recipients. Traffic sent via a stream has priority over all
non-stream traffic, and is delivered with the minimum end-to-end
delay possible. Hosts use streams to support applications that have
predictable traffic loads (such as packet voice or video or other
continuous media traffic) or that require minimum transmission delay
and lowest delay variance. Streams are typically used for traffic
flows of moderate to long duration, where the cost of performing a
stream Setup is acceptable.

Streams must be set up before stream data messages can flow. The
stream setup messages, each of which has a Request and a Reply, are
Create Stream, Delete Stream, Change Stream, and Delete All Streams.
(Create Shared Stream Request is a planned future addition to the
protocol.) The use of these messages is illustrated in the scenario
of exchanges between a host and the Service Agent shown in Figure 7
where the host establishes a stream, sends some data, modifies the
stream characteristics, sends some more data, and finally closes down
the stream. Not illustrated, but implicit in this scenario, are the
optional A/R indications associated with each of the stream Setup
messages.

```
                                     Service      Other
                           Host       Agent       hosts

        Create Stream Request      ---------->
        Create Stream Reply        <----------
        Reply Acknowledgment       ---------->
        Stream Messages            -------------------->
             :    :
        Change Stream Request      ---------->
        Change Stream Reply        <----------
        Reply Acknowledgment       ---------->
        Stream Messages            -------------------->
             :    :
        Delete Stream Request      ---------->
        Delete Stream Reply        <----------
        Reply Acknowledgment       ---------->
```
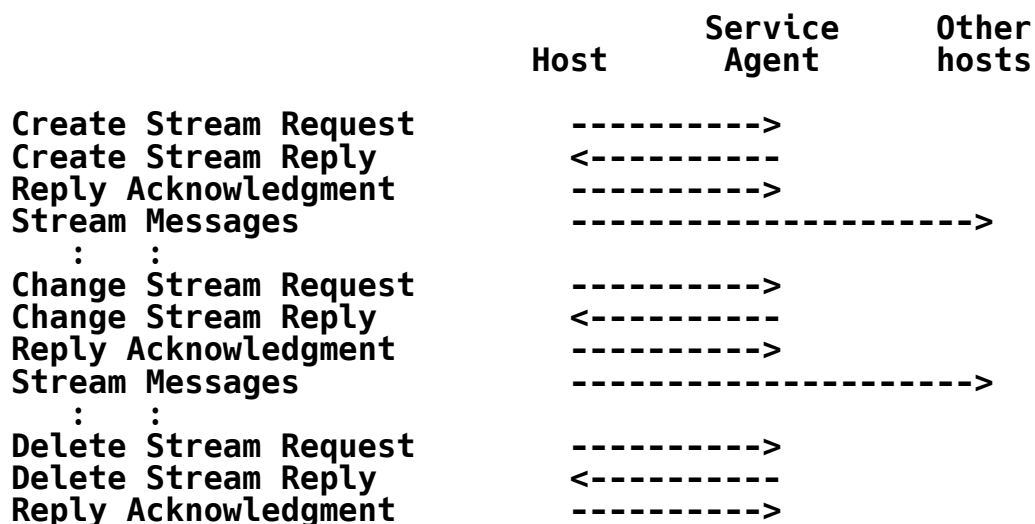
STREAM EXAMPLE
Figure 7

Streams have eight characteristic properties which are selected at
stream setup time. These properties are: (1) data words per time
interval, (2) time interval, (3) reliability, (4) reliability length,
(5) precedence, (6) maximum messages per interval, (7) the list of
recipients, and (8) the set of other streams with which this stream
shares resources. To establish a stream, the host sends the Create

Stream Request message (Figure 8) to the Service Agent. After the
network has processed the Create Stream Request, the Service Agent
will reply with a Create Stream Reply message (Figure 9). If the
reply code in the Create Stream Reply indicates that the stream has
been created successfully, the host may proceed to transmit stream
data messages after sending a Reply Acknowledgment.

During the lifetime of a stream, the host which created it may decide
that some of its characteristic properties should be modified. All
but one of the properties can be modified using the Change Stream
Request message (Figure 10). The one property that cannot be changed
is whether or not the stream is willing to share its resources with
other streams. After the network has processed the Change Stream
Request, the Service Agent will respond by sending a Change Stream
Reply (Figure 11) to the host. A host requesting a reduced channel
allocation should decrease its sending rate immediately without
waiting for receipt of the Change Stream Reply. A host requesting an
increased allocation should not proceed to transmit according to the
new set of parameters without first having received a Reply Code
indicating that the requested change has taken effect.

When the host no longer needs the stream it created, it should first
stop sending traffic via the stream and then send the Service Agent a
Delete Stream Request message (Figure 12). After the network has
processed the Delete Stream Request, the Service Agent will respond
by sending a Delete Stream Reply (Figure 13) to the host.

If the host has crashed or restarted, it may no longer know what
streams it owns. The host may use an Information Request (see
Section 6.5) to determine what streams it owns, or the host may use a
Delete All Streams Request (Figure 14) to discard whatever stream
resources it may own. The format for the Delete All Streams Reply is
shown in Figure 15.

Note that streams, like all other resources allocated by the Service
Agent, may be reclaimed by the network if unused. Currently, if no
traffic is sent to a stream in a 6 minute interval, and if the owner
of the steam is down or unreachable, the stream may be deleted.

```
                    0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S0        |          1           |          5            |
                   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S1        |                SETUP CHECKSUM                 |
                   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S2        |                  REQUEST ID                   |
                   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S3        |  MAX MES  | PRE | INT | RLY |      RLEN       |
                   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S4        |            DATA WORDS PER INTERVAL            |
                   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S5        |                  INTERVAL                     |
                   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S6        |          0           | ADDRESS LIST LENGTH    |
                   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
                   |                                               |
         S7-SN     :           DESTINATION ADDRESS LIST            :
                   |                                               |
                   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
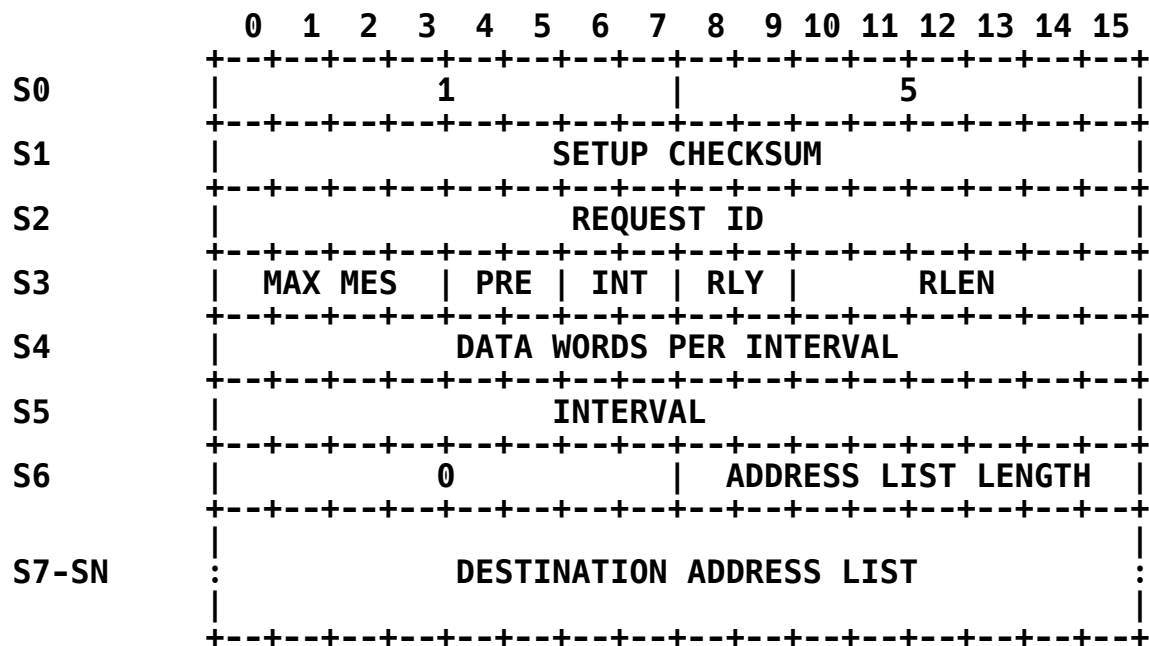
CREATE STREAM REQUEST
Figure 8

S0[0-7]   Setup Type = 1 (Request).

S0[8-15]   Request Type = 5 (Create Stream).

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

S3[0-3]   Maximum Messages Per Interval (1-15). This field
           specifies the maximum number of stream messages the
           host will deliver to the WPS in any single stream
           interval.

S3[4-5]   Precedence. This field specifies the precedence of the
           stream. When there are insufficient network resources
           to support all the requested streams, requests for
           higher precedence streams will preempt existing lower
           precedence streams, and requests for streams with
           insufficient precedence will be rejected. Medium
           precedence is recommended as the default choice.

```
                    0 = Low Precedence
                    1 = Medium Precedence
                    2 = High Precedence

    S3[6-7]     Interval.  This field specifies the interval, in
                multiples of 21.22 milliseconds.  (For backward
                compatibility only.  New applications should use 3.
                Use of this field to specify an interval is being
                phased out.)

                    0 =  21.22 milliseconds
                    1 =  42.44 milliseconds
                    2 =  84.88 milliseconds
                    3 =  use interval in word S5

    S3[8-9]     Reliability.  This field specifies the basic bit-error
                rate requirement for the data portion of all messages
                in the stream.  The exact error rate obtained by each
                choice is not specified.

                    0 = Low Reliability
                    1 = Medium-Low Reliability
                    2 = Medium-High Reliability
                    3 = High Reliability

    S3[10-15] Reliability Length.  This field specifies how many
                words beyond the stream message header should be
                transmitted at maximum reliability for all messages in
                the host stream.

    S4[0-15]    Data words per interval.  This field specifies the
                maximum number of 16-bit words of this stream's data
                the network will need to carry during each interval,
                not counting HAP stream message header words.  The
                stream data may be carried in however many messages (up
                to MAX MES) in each interval the host chooses.

    S5[0-15]    Interval (125 microsecond units).  This field specifies
                the time interval over which the <data words per
                interval> data in <max mes> messages will be sent.  For
                backward compatibility, an interval of 0 selects an
                interval of 169.76 milliseconds.  This field is ignored
                unless the INT field is 3.

    S6[0-7]     Reserved.  Must be zero.

    S6[8-15]    Destination address list length.  This field specifies
                the number of entries in the Destination Address List
```

                       field. Allowed values are 1-8.

     S7-SN   Destination address list. This list must specify, at
                   least indirectly, all the intended recipients of this
                   stream's traffic. At least one destination address
                   must be supplied. Any valid network address,
                   specifically including group addresses, may be used
                   (except the Service Agent's address, 0). Messages sent
                   in the stream are not limited to using the HAP
                   addresses listed. E.g., if the list consists of only
                   group address G, and host A is a member of G, a stream
                   message may be sent to A, which was not in the list.

   Caution: Group membership is only evaluated at setup time. Changes
   in group membership do not cause the stream to be modified.

   Caution: Stream creation involves allocation of specific network
   resources along specific routes for delivery of that traffic. A
   stream message sent to hosts other than those specified via Setup
   will probably be undeliverable. A stream message to a group address
   that has gained new members since the stream's last Setup may be
   undeliverable to the new members.

```
              0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
       S0    |         2         |         REPLY CODE        |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
       S1    |                   SETUP CHECKSUM              |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
       S2    |                     REQUEST ID                |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
       S3    |        0          |          STREAM ID        |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
       S4    |        0          |     ADDRESS LIST LENGTH   |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
             |                                               |
     S5-SN   :                   ADDRESS LIST                :
             |                                               |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

                         CREATE STREAM REPLY
                             Figure 9

     S0[0-7]   Setup Type = 2 (Reply).

S0[8-15] Reply Code. Any reply other than "Stream created"
          means the stream was not created.

                0 = Stream created
                8 = Network trouble
               12 = Stream precedence not being accepted
               17 = Insufficient network resources
               18 = Requested bandwidth too large
               21 = Max. messages per interval too small
               22 = Reply lost in network
               23 = Illegal precedence value
               24 = Invalid destination address in list

S1[0-15] Setup Checksum. (See setup header description.)

S2[0-15] Request ID.

S3[0-5]   Reserved. Must be zero.

S3[6-15] Stream ID. This field contains a stream ID assigned by
          the network. It must be included in all stream data
          messages sent by the host to allow the WPS to associate
          the message with stored stream characteristics and the
          resources reserved for that stream's traffic.

S4[0-5]   Reserved. Must be zero.

S4[6-15] Address list length. The number of entries in the
          Address List field.

S5-SN   Address list. This contains the destination addresses
          from the Create Stream Request that were invalid or
          unreachable. Unreachable destinations are listed as a
          group if every member of the group was unreachable, or
          individually otherwise; i.e., group addresses are
          expanded and the unreachable members are included in
          the list. The list of unreachable destinations will be
          truncated, if needed, to limit this Reply to a single,
          maximum length HAP message.

```
                   0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           S0    |        1        |              7              |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           S1    |              SETUP CHECKSUM                   |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           S2    |               REQUEST ID                      |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           S3    |        0        |         STREAM ID           |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           S4    | MAX MES | PRE | INT | RLY |       RLEN        |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           S5    |          DATA WORDS PER INTERVAL             |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           S6    |                 INTERVAL                      |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           S7    |        0        |    ADDRESS LIST LENGTH      |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
                 |                                               |
           S8-SN :           DESTINATION ADDRESS LIST           :
                 |                                               |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

                       CHANGE STREAM REQUEST
                             Figure 10

S0[0-7]   Setup Type = 1 (Request).

S0[8-15]   Request Type = 7 (Change Stream).

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

S3[0-5]   Reserved. Must be zero.

S3[6-15]   Stream ID.

S4[0-3]   New Maximum Messages Per Interval.

S4[4-5]   New Precedence.

S4[6-7]   New Interval selection.

S4[8-9]   New Reliability.

S4[10-15] New Reliability Length.

S5[0-15] New Data Words Per Interval.

S6[0-15] New Interval (ignored unless INT = 3).
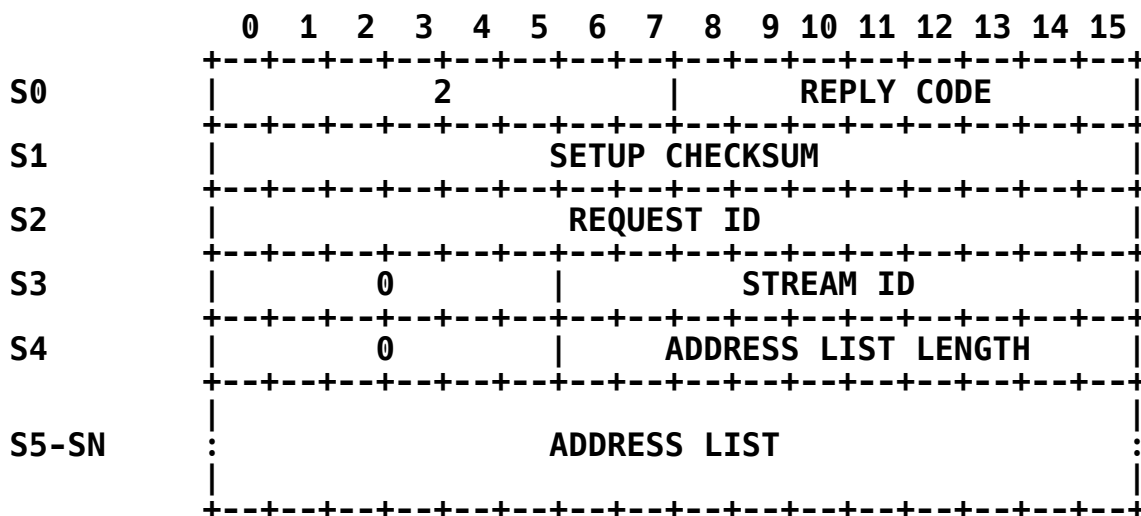
S7[0-7]   Reserved. Must be zero.

S7[8-15] Destination Address List length. This field specifies
          the number of entries in the new Destination Address
          List. Allowed values are 0-8. Use zero (indicating no
          addresses in the list) to avoid changing the list of
          recipient hosts.

S8-SN   New Destination Address List. The new, complete, list
          of recipient hosts. Membership of group addresses is
          evaluated at setup execution time. Subsequent changes
          in group membership do not cause the stream to be
          modified. Note that using the same destination address
          list in the Change Stream Request as was used in the
          Create Stream Request can result in a change in the
          list of recipient hosts if membership in a group has
          changed.

```
              0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S0        |         2         |         REPLY CODE        |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S1        |                  SETUP CHECKSUM               |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S2        |                   REQUEST ID                  |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S3        |         0         |    ADDRESS LIST LENGTH    |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
              |                                               |
    S4-SN     :                 ADDRESS LIST                 :
              |                                               |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
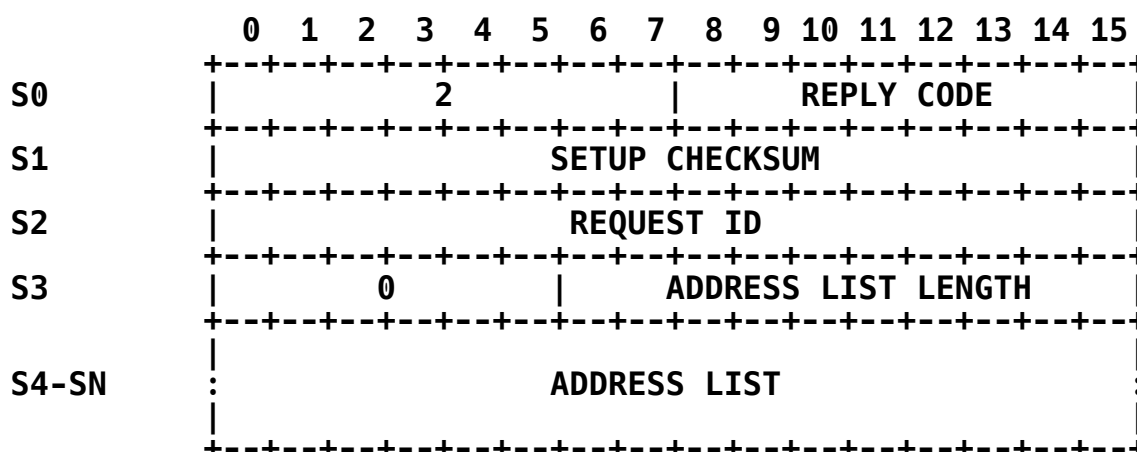
                        CHANGE STREAM REPLY
                            Figure 11

S0[0-7]   Setup Type = 2 (Reply).

S0[8-15]   Reply Code. The number in parentheses indicates the
           processing phase at the time of the error (see Caution
           below). Phase zero and phase one errors leave the
           stream unchanged; errors from later phases may leave
           the stream partially modified.

                  4 = Stream changed
                  8 = (1) Network trouble
                 10 = (0) Stream ID nonexistent
                 11 = (0) Not creator of stream
                 12 = (0) Stream precedence not being accepted
                 16 = (3) Unable to add all the new recipients
                 17 = (2) Insufficient network resources
                 18 = (2) Requested bandwidth too large
                 21 = (0) Maximum messages per interval too small
                 22 = (2) Reply lost in network
                 23 = (0) Illegal precedence value
                 24 = (0) Invalid destination address in list

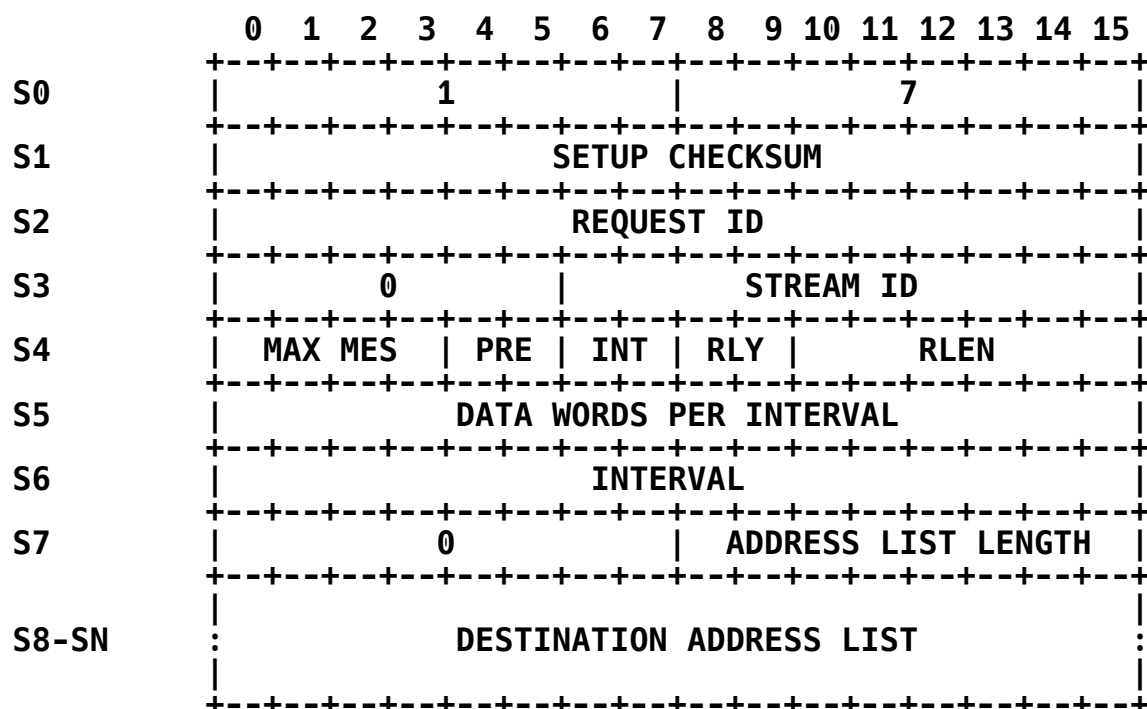S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

S3[0-5]   Reserved. Must be zero.

S3[6-15]   Address list length. This field specifies the number
           of addresses in the Address List.

S4-SN   Address list. This contains the destination addresses
           from the Change Stream Request that were invalid (phase
           0 errors) or unreachable (phase 3 errors). Unreachable
           destinations are listed as a group if every member of
           the group was unreachable, or individually otherwise;
           i.e., group addresses are expanded and the unreachable
           members are included in the list. The list of
           unreachable destinations will be truncated, if needed,
           to limit this Reply to a single, maximum length HAP
           message.

Caution: The Change Stream Reply will indicate failure if any
aspect of the requested changes did not occur. However, the
stream may have been partially modified. Processing is performed
in the following phases:
    0: check for invalid requests;
    1: drop former recipients that are not in the latest list;
    2: increase or decrease the stream's bandwidth allocation
       (decreases are normally successful); then
    3: extend the stream to any new recipients.

If phase 2 fails, phase 3 is not performed, the Reply Code will
indicate an error and the stream parameters will be unchanged.
If phase 3 fails, the Address List will contain the destinations,
if any, from the latest list that the stream does not reach.
Phase 1 only fails if the stream has been suspended (see
Notifications) or the WPS is experiencing network connectivity
problems.

```
           0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S0  |          1           |          6             |
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S1  |              SETUP CHECKSUM                   |
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S2  |                REQUEST ID                     |
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S3  |        0          |         STREAM ID         |
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
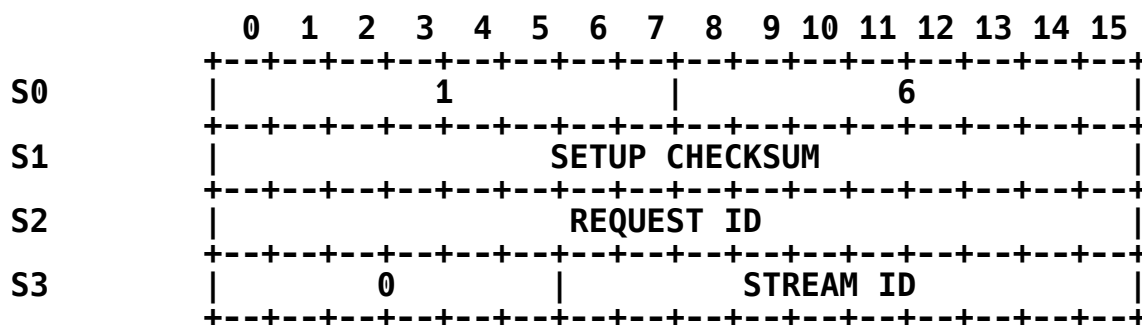
DELETE STREAM REQUEST
Figure 12

S0[0-7]   Setup Type = 1 (Request).

S0[8-15] Request Type = 6 (Delete Stream).

S1[0-15] Setup Checksum. (See setup header description.)

S2[0-15] Request ID.

S3[0-5]   Reserved. Must be zero.

S3[6-15] Stream ID.

```
              0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S0        |           2           |       REPLY CODE      |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S1        |                   SETUP CHECKSUM              |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S2        |                    REQUEST ID                 |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
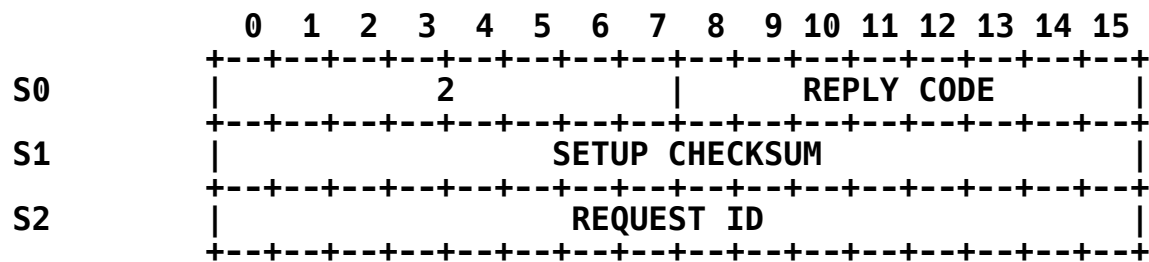
DELETE STREAM REPLY
Figure 13

S0[0-7]   Setup Type = 2 (Reply).

S0[8-15]   Reply Code. If the request was valid, the Service
           Agent will have marked the stream for deletion even if
           the stream resources have not actually been deleted
           yet.

                1 = Stream deleted
               10 = Stream ID nonexistent
               11 = Not creator of stream

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.


```
              0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S0        |           1           |           9           |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S1        |                   SETUP CHECKSUM              |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S2        |                    REQUEST ID                 |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
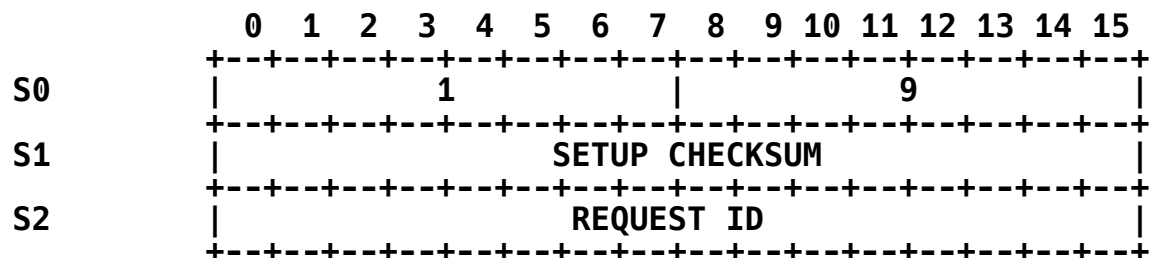
DELETE ALL STREAMS REQUEST
Figure 14

S0[0-7]   Setup Type = 1 (Request).

S0[8-15]   Request Type = 9 (Delete All Streams).

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

```
            0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
S0          |          2          |       REPLY CODE        |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
S1          |                SETUP CHECKSUM                 |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
S2          |                  REQUEST ID                   |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
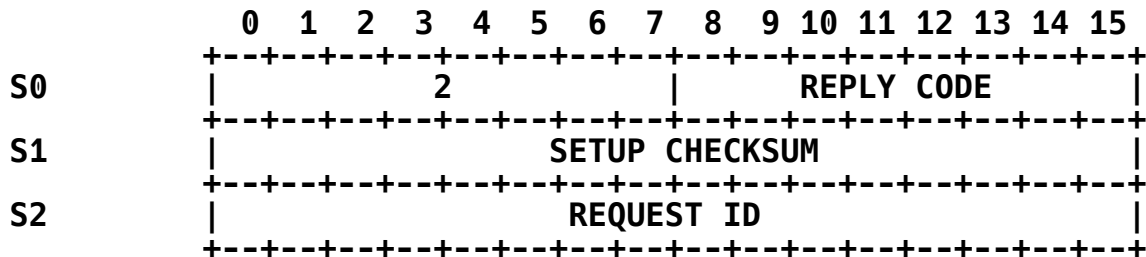
DELETE ALL STREAMS REPLY
Figure 15

S0[0-7]   Setup Type = 2 (Reply).

S0[8-15]   Reply Code. The Service Agent will have marked all of
           the host's streams for deletion, even if the stream
           resources have not actually been deleted yet.

               1 = Streams deleted

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

6.2. Group Setup Messages

Group (multicast) addressing allows a host to send the same message
to N different hosts without having to send N copies of the message.
The network duplicates the message as required. In addition to
reducing the burden on the originating host, multicasting reduces the
load on the network because the network no longer has to carry the
duplicates along the common portions of the paths between the source
and destinations. Multicasting is particularly recommended for
multi-site conferencing and distributed simulations.

Group addresses are dynamically created and deleted via setup
messages exchanged between the hosts and the Service Agent.
Membership in a group may be any arbitrary subset of the network
hosts. A datagram message or stream message addressed to a group is
delivered to all hosts that are members of that group (exception:
stream messages sent to a group address that includes hosts the

stream was not set up to reach). The group setup messages, each of
which has a Request and a Reply, are Create Group, Delete Group, Join
Group, Leave Group, Add Group Member, and Remove Group Member.

Figure 16 shows a typical use of group setup messages. The figure
illustrates a scenario of exchanges between three hosts and the
Service Agent. In the scenario one host, Host A, creates a group
which is joined by hosts B and C. The hosts then exchange some data
messages using the group address. Note that multicast messages are
not returned to their originator. Hosts A and C then leave the
group, and Host B decides to delete the group. As in the scenario in
Section 6.1, A/R indications have been omitted for clarity.

Part of the group creation procedure involves the Service Agent
returning to the creating host a 48-bit key along with the 16-bit
group address. The creating host must pass the key along with the
group address to other hosts that want to join the group. These
other hosts must supply the key along with the group address in their
Join Group Requests. The key is used by the network to authenticate
these operations and thereby minimize the probability that unwanted
hosts will deliberately or inadvertently become members of the group.
The procedure used by a host to distribute the group address and key
is not within the scope of HAP.

In the figure below, the network Service Agent is pictured as a
single entity for simplicity.

```
                                  Service   Host  Host  Host
                                  Agent      A     B     C
         Create Group Request           |<-------|
         Create Group Reply             |------->|
         Reply Acknowledgment           |<-------|
                 :   :
         Distribute Group Adr & Key              |---->|
         Distribute Group Adr & Key              |---------->|
                 :   :
         Join Group Request (C)         |<-------------------|
         Join Group Reply               |------------------->|
         Reply Acknowledgment           |<-------------------|
         Join Group Request (B)         |<-------------|
         Join Group Reply               |------------->|
         Reply Acknowledgment           |<-------------|
                 :   :
         Data Message 1 (A to B and C)           |---->|---->|
         Data Message 2 (B to A and C)           |<----|---->|
         Data Message 3 (C to A and B)           |<----|<----|
                 :   :
         Leave Group Request (C)        |<-------------------|
         Leave Group Reply              |------------------->|
         Reply Acknowledgment           |<-------------------|
         Leave Group Request (A)        |<-------|
         Leave Group Reply              |------->|
         Reply Acknowledgment           |<-------|
         Delete Group Request           |<-------------|
         Delete Group Reply             |------------->|
         Reply Acknowledgment           |<-------------|
```

                         GROUP EXAMPLE
                          Figure 16

   An alternative method of adding and removing group members is the use
   of Add Group Member and Remove Group Member. These setup requests
   allow hosts that are already members of the group to add or delete
   other hosts.

   The Setup requests Join Group, Leave Group, Add Group Member, Remove
   Group Member, and Delete Group are authenticated using the 48-bit
   key. Leave Group and Remove Group Member will remove a host from the
   group membership list but will not alter the existence of the group.
   Delete Group expunges all knowledge of the group from the network.
   HAP permits any host with the proper key to delete the group at any
   time. Thus, group addresses can be deleted even if the host which
   originally created the group has left the group or has crashed.
   Moreover, groups may exist for which there are currently no members

because each member has executed a Leave while none has executed a
Delete. It is the responsibility of the hosts to coordinate and
manage the use of group addresses.

Note that group addresses, like all other resources allocated by the
network, may be reclaimed by the network if unused for too long.
Currently, if no traffic is sent to the group address in a 6 minute
interval, the network may delete the group and notify all members
that the group no longer exists.

The Create Group Request (Figure 17) is used to establish a multicast
address. After the network has processed the Create Group Request,
the Service Agent will respond by sending a Create Group Reply
(Figure 18) to the host.

A host may become a member of a group, once it knows the group
address and the 48-bit key, by sending the Service Agent the Join
Group Request message (Figure 19). The Service Agent will respond to
the Join Group Request with a Join Group Reply (Figure 20). The host
which creates a group automatically becomes a member of that group
without any need for an explicit Join Group Request.

A member host may add another host to the group by sending the
Service Agent the Add Group Member Request message (Figure 21). The
Service Agent will respond with an Add Group Member Reply (Figure
22).

At any time after becoming a member of a group, a host may choose to
drop out of the group. To do this, the host sends the Service Agent
a Leave Group Request (Figure 23). The Service Agent will respond
with a Leave Group Reply (Figure 24).

One member host may expel another member of the group by sending the
Service Agent the Remove Group Member Request message (Figure 25).
The Service Agent will respond with a Remove Group Member Reply
(Figure 26).

A host can delete an existing group via a Delete Group Request
(Figure 27). The Service Agent will respond with a Delete Group
Reply (Figure 28). The Service Agent will also send the other
members of the group, if any, a notification that the group has been
deleted (see Section 6.3).

```
                0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S0         |           1           |           1           |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S1         |                  SETUP CHECKSUM               |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S2         |                   REQUEST ID                  |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

CREATE GROUP REQUEST
Figure 17

S0[0-7]   Setup Type = 1 (Request).

S0[8-15]   Request Type = 1 (Create Group).

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

```
                0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S0         |           2           |      REPLY CODE       |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S1         |                 SETUP CHECKSUM                |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S2         |                   REQUEST ID                  |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S3         |                  GROUP ADDRESS                |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S4         |                      KEY                      |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S5         |                      KEY                      |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S6         |                      KEY                      |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

CREATE GROUP REPLY
Figure 18

S0[0-7]   Setup Type = 2 (Reply).

S0[8-15] Reply Code.

                 0 = Group created
                 8 = Network trouble
                17 = Insufficient network resources
                22 = Reply lost in network

S1[0-15] Setup Checksum. (See setup header description.)

S2[0-15] Request ID.

S3[0-15] Group Address. This field contains the 16-bit
          multicast address that any group member may use to
          reach the other group members. Multicast addresses are
          dynamically assigned by the network.

S4-S6   Key. This field contains a 48-bit key assigned by the
          network which is associated with the group address. It
          must be provided for subsequent Join Group, Leave
          Group, Add Group Member, Remove Group Member, and
          Delete Group requests which reference the group
          address.

```
            0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S0     |           1           |           3           |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S1     |                  SETUP CHECKSUM               |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S2     |                   REQUEST ID                  |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S3     |                  GROUP ADDRESS                |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S4     |                      KEY                      |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S5     |                      KEY                      |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S6     |                      KEY                      |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S7     |                      0                | MGP |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
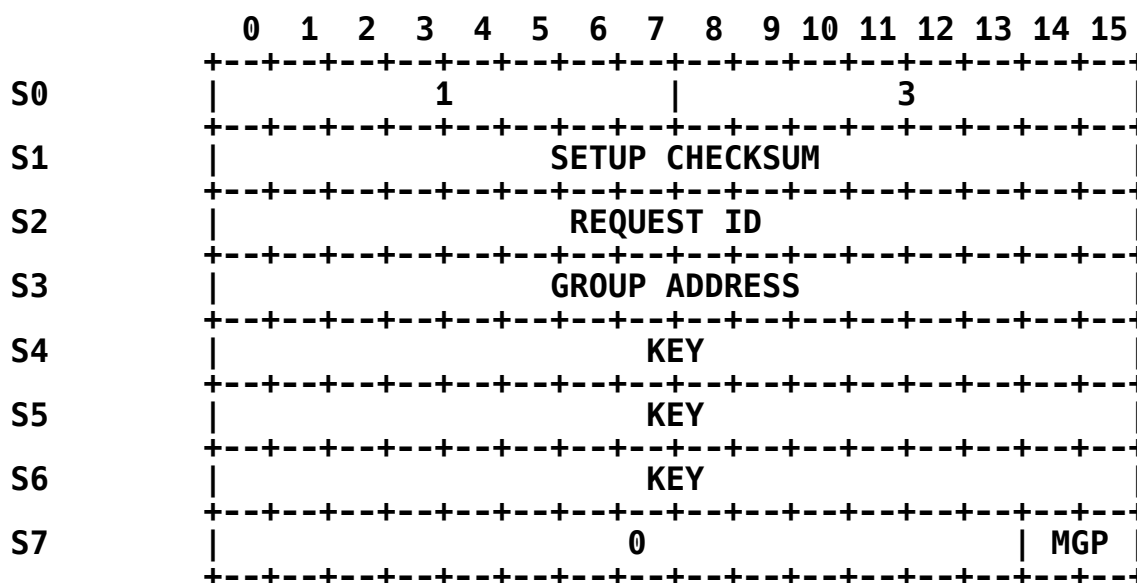
                      JOIN GROUP REQUEST
                         Figure 19

S0[0-7]   Setup Type = 1 (Request).

S0[8-15]   Request Type = 3 (Join Group).

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

S3[0-15]   Group Address. This is the group that the host wishes
           to join. Upon successfully joining the group, the host
           may send messages to the group and will receive
           messages sent to the group when those messages have a
           priority of MGP or higher.

S4-S6   Key. This is the key associated with the group
           address.

S7[0-13]   Reserved. Must be zero.

S7[14-15] Minimum group message priority. The host will not
           receive messages sent to the group that have a message
           priority less than MGP. Send another Join Group
           Request message to change the minimum priority.


```
           0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S0    |         2            |       REPLY CODE        |
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S1    |                 SETUP CHECKSUM                 |
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S2    |                  REQUEST ID                    |
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

JOIN GROUP REPLY
Figure 20

S0[0-7]   Setup Type = 2 (Reply).

S0[8-15]   Reply Code.

                2 = Host added to group
                9 = Bad key
               10 = Group address nonexistent
               17 = Insufficient network resources

    S1[0-15] Setup Checksum. (See setup header description.)

    S2[0-15] Request ID.


```
                   0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S0       |           1           |           10          |
                  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S1       |                   SETUP CHECKSUM              |
                  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S2       |                     REQUEST ID                |
                  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S3       |                   GROUP ADDRESS               |
                  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S4       |                       KEY                     |
                  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S5       |                       KEY                     |
                  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S6       |                       KEY                     |
                  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S7       |                   HOST ADDRESS                |
                  +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
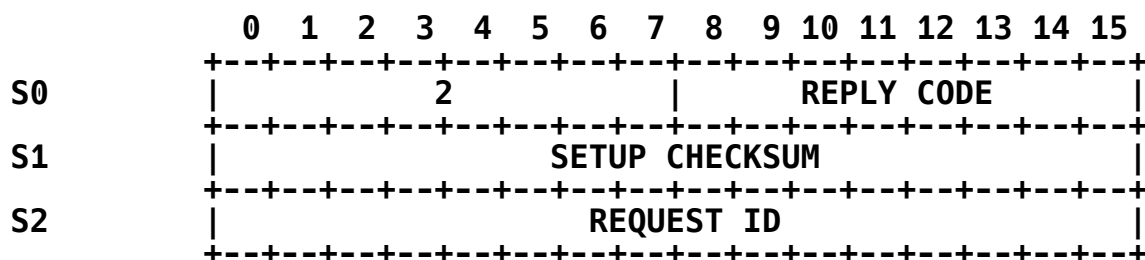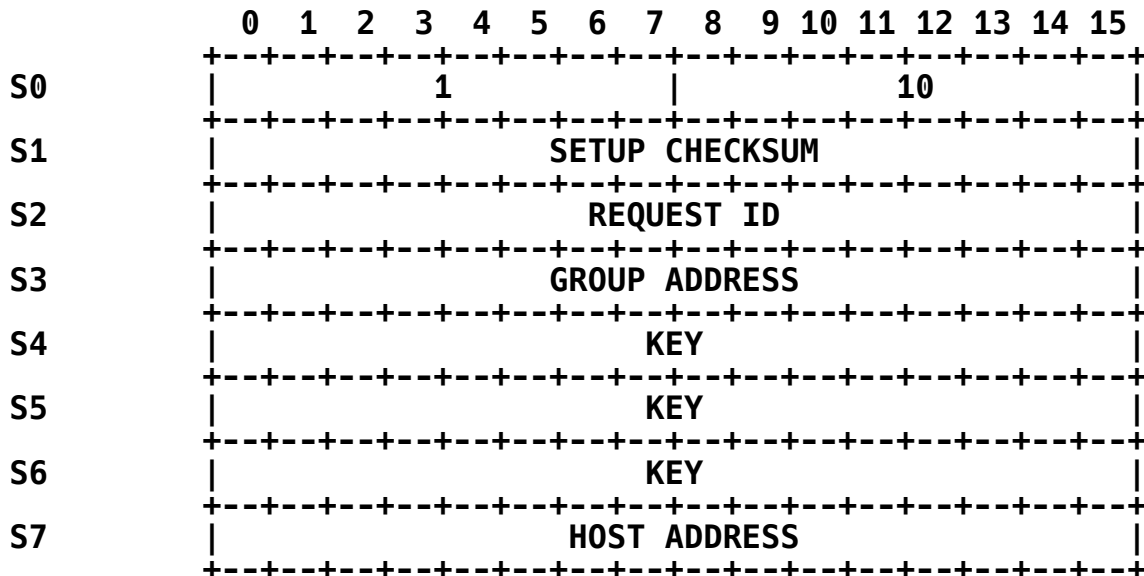
                        ADD GROUP MEMBER REQUEST
                              Figure 21

    S0[0-7]   Setup Type = 1 (Request).

    S0[8-15] Request Type = 3 (Join Group).

    S1[0-15] Setup Checksum. (See setup header description.)

    S2[0-15] Request ID.

    S3[0-15] Group Address. This is the group the host will join.
              Upon successfully joining the group, the host may send
              messages to the group and will receive messages sent to
              the group by other hosts (the initial minimum priority
              will be 0).

    S4-S6   Key. This is the key associated with the group
              address.

    S7[0-15] Host address. The network address of the host to add

to the group.

```
          0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
 S0       |           2           |      REPLY CODE       |
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
 S1       |                  SETUP CHECKSUM               |
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
 S2       |                   REQUEST ID                  |
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

ADD GROUP MEMBER REPLY
Figure 22

S0[0-7]   Setup Type = 2 (Reply).

S0[8-15]   Reply Code.

            2 = Host added to group (or was already a member)
            9 = Bad key
           10 = Group address nonexistent
           11 = Requestor is not a member of the group
           17 = Insufficient network resources
           22 = Reply lost in network
           24 = Host address was invalid

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

```
                0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S0     |           1           |           4           |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S1     |                   SETUP CHECKSUM              |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S2     |                    REQUEST ID                 |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S3     |                   GROUP ADDRESS               |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S4     |                       KEY                     |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S5     |                       KEY                     |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S6     |                       KEY                     |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
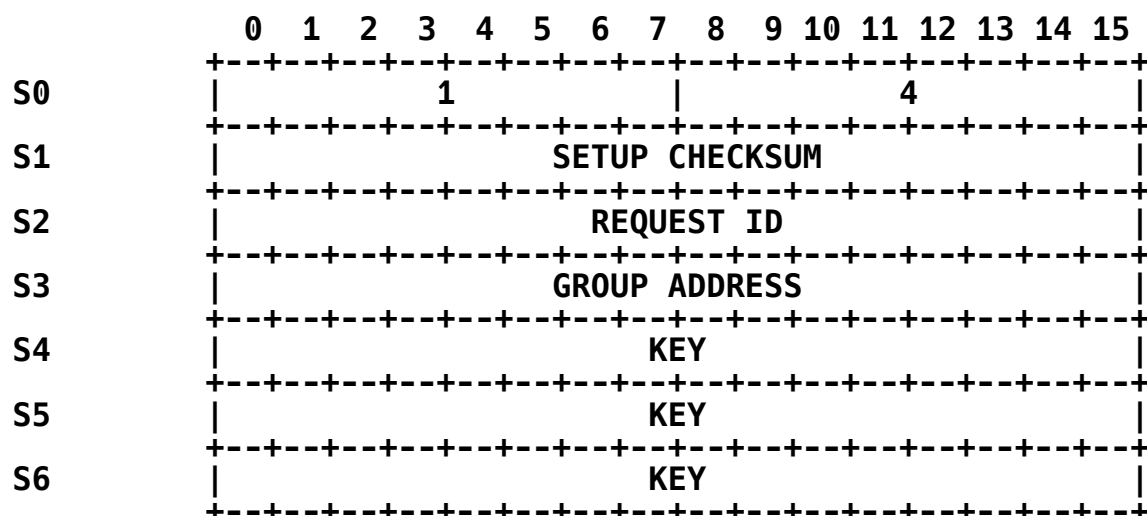
LEAVE GROUP REQUEST
Figure 23

S0[0-7]   Setup Type = 1 (Request).

S0[8-15] Request Type = 4 (Leave Group).

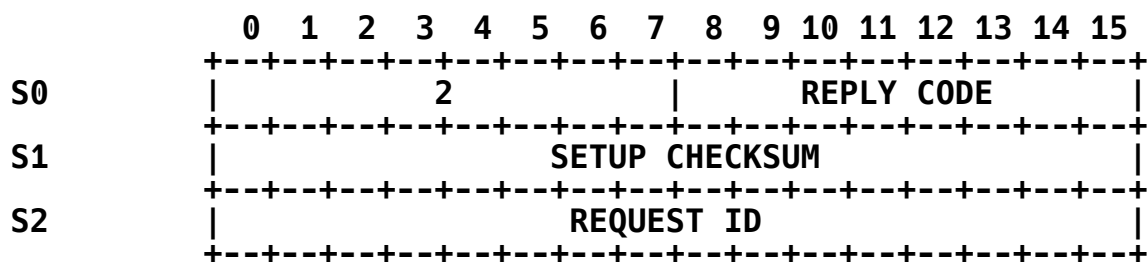S1[0-15] Setup Checksum. (See setup header description.)

S2[0-15] Request ID.

S3[0-15] Group Address. This is the group that the host wishes
          to cease being a member of. After leaving the group,
          the host will cease receiving messages sent to the
          group and will be unable to send to the group.

S4-S6   Key. This is the key associated with the group
          address.

```
              0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S0   |           2           |       REPLY CODE      |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S1   |                  SETUP CHECKSUM               |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S2   |                   REQUEST ID                  |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
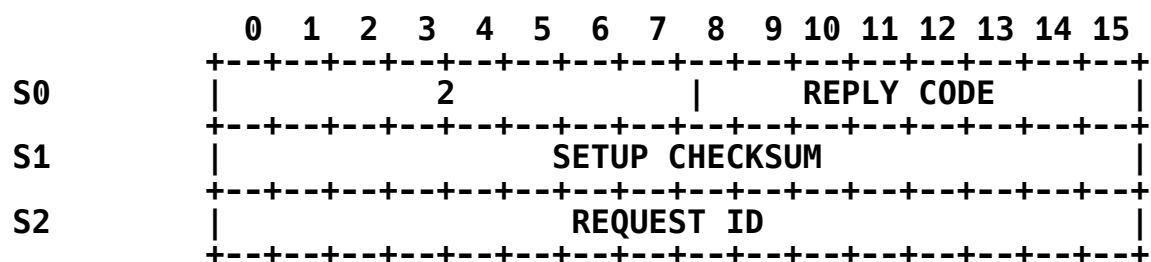
                        LEAVE GROUP REPLY
                            Figure 24

     S0[0-7]   Setup Type = 2 (Reply).

     S0[8-15]   Reply Code.

                     3 = Host deleted from group
                     9 = Bad key
                    10 = Invalid group address
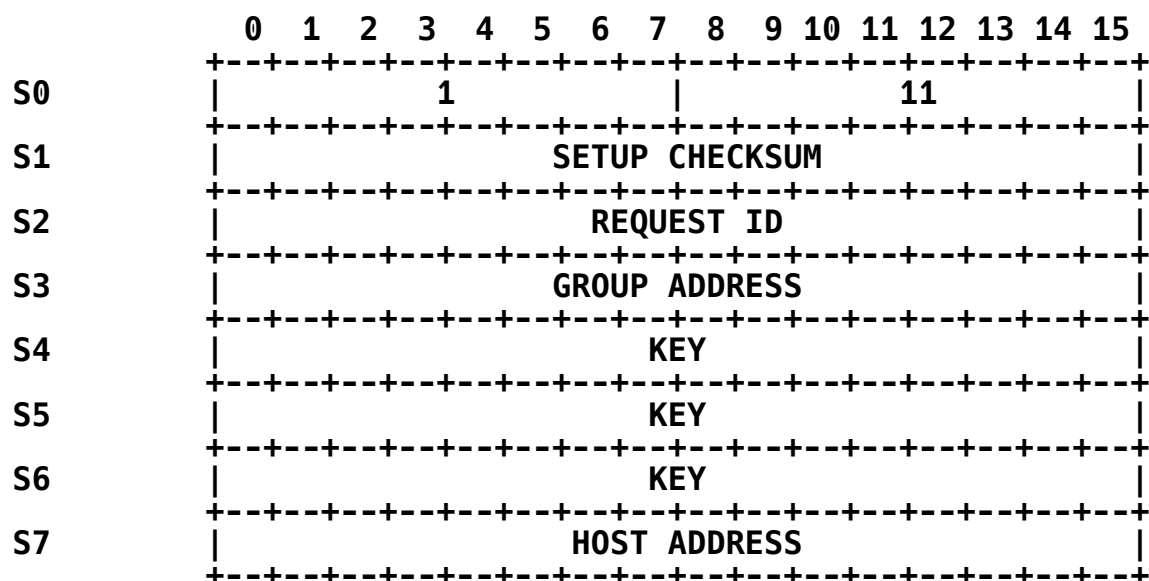   11 = Not member of group
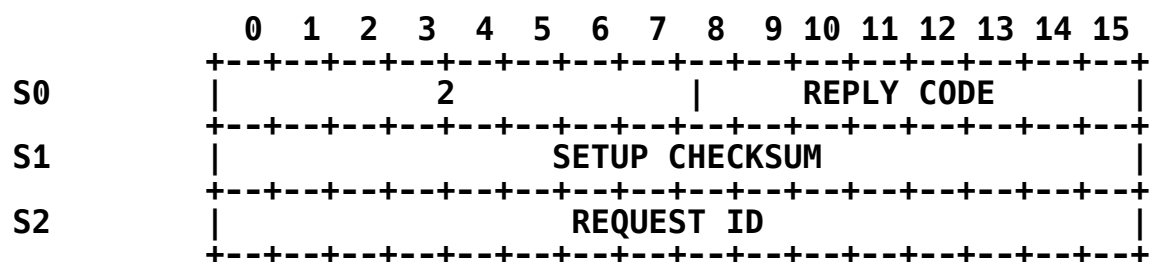   17 = Insufficient network resources

     S1[0-15]   Setup Checksum. (See setup header description.)

     S2[0-15]   Request ID.

```
               0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S0     |           1           |           11          |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S1     |                 SETUP CHECKSUM                |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S2     |                   REQUEST ID                  |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S3     |                 GROUP ADDRESS                 |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S4     |                     KEY                       |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S5     |                     KEY                       |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S6     |                     KEY                       |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S7     |                 HOST ADDRESS                  |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

REMOVE GROUP MEMBER REQUEST
Figure 25

S0[0-7]   Setup Type = 1 (Request).

S0[8-15]   Request Type = 4 (Leave Group).

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

S3[0-15]   Group Address. This is the group from which the host
           should be removed. After leaving the group, that host
           will cease receiving messages sent to the group and
           will be unable to send to the group.

S4-S6   Key. This is the key associated with the group
           address.

S7[0-15]   Host address. The network address of the host to
           remove from the group.

```
                0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S0     |           2           |      REPLY CODE       |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S1     |                 SETUP CHECKSUM                |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         S2     |                   REQUEST ID                  |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

                     REMOVE GROUP MEMBER REPLY
                            Figure 26

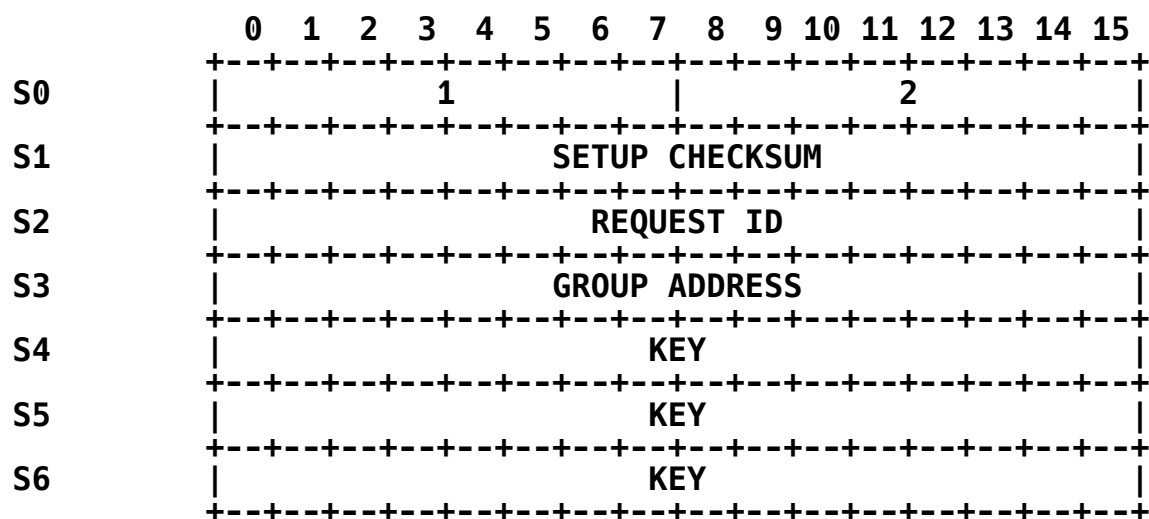   S0[0-7]   Setup Type = 2 (Reply).

    S0[8-15]   Reply Code.

                    3 = Host deleted from group (or was not a member)
                    9 = Bad key
                   10 = Invalid group address
                   11 = Requestor is not a member of the group
                   17 = Insufficient network resources
                   22 = Reply lost in network
                   24 = Host address was invalid
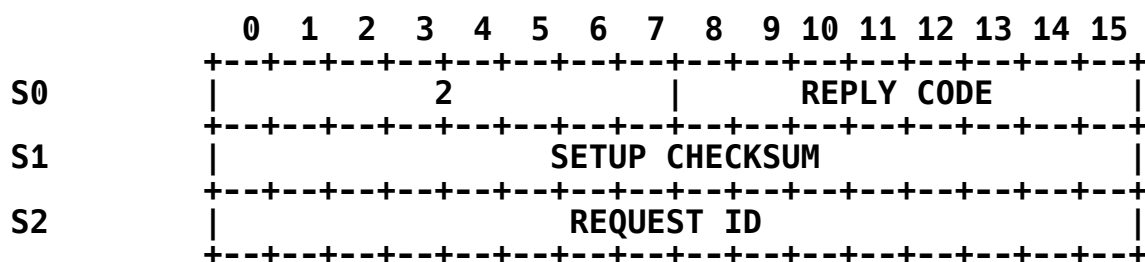
    S1[0-15]   Setup Checksum. (See setup header description.)

    S2[0-15]   Request ID.

```
             0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S0    |           1           |           2           |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S1    |                    SETUP CHECKSUM             |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S2    |                     REQUEST ID                |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S3    |                    GROUP ADDRESS              |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S4    |                       KEY                     |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S5    |                       KEY                     |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S6    |                       KEY                     |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

DELETE GROUP REQUEST
Figure 27

S0[0-7]   Setup Type = 1 (Request).

S0[8-15]   Request Type = 2 (Delete Group).

S1[0-15]   Setup Checksum. (See setup header description.)

S2[0-15]   Request ID.

S3[0-15]   Group Address. This is the multicast address to
           delete. If the group is deleted, the other remaining
           members of the group, if any, will be notified of the
           group's deletion.

S4-S6   Key.

```
                 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S0        |           2           |       REPLY CODE      |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S1        |                   SETUP CHECKSUM              |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      S2        |                     REQUEST ID                |
                +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

                         DELETE GROUP REPLY
                            Figure 28

    S0[0-7]   Setup Type = 2 (Reply).

    S0[8-15]   Reply Code.

                    1 = Group deleted
                    8 = Network trouble
                    9 = Bad key
                   10 = Invalid group address
                   17 = Insufficient network resources
                   22 = Reply lost in network

    S1[0-15]   Setup Checksum. (See setup header description.)

    S2[0-15]   Request ID.

6.3. Notifications

   Notifications are Setup exchanges initiated by the WPS to inform a
   host of changes in the status of a network resource. The format of
   Notification messages is shown in Figure 29.

```
             0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S0      |        3        |            CODE             |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S1      |                     CHECKSUM                  |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S2      |                  NOTIFICATION ID              |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S3      |                 NOTIFICATION INFO             |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
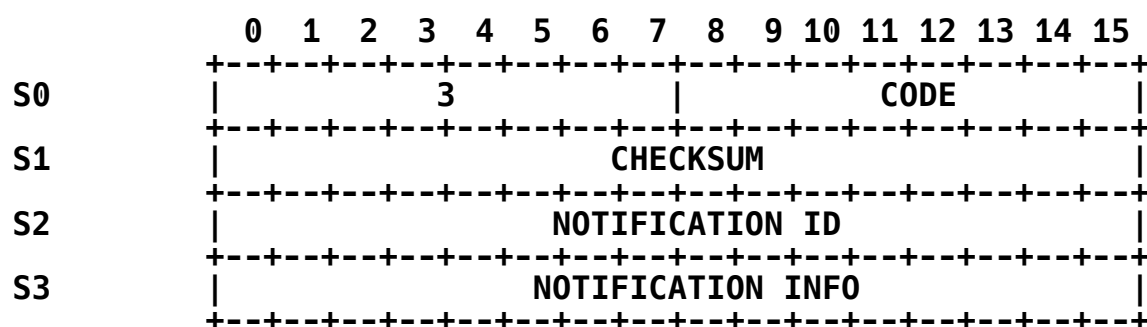
NOTIFICATION MESSAGE
Figure 29

S0[0-7]   Message Type = 3 (Notification).

S0[8-15]   Code. This indicates what the Notification signifies.

                0 = Stream suspended
                1 = Stream resumed
                2 = Stream deleted
                3 = Group deleted by a host
                4 = Group deleted by network
                5 = All streams deleted
                6 = All groups deleted
                7 = Group changed by a host
                8 = Group changed by network

S1[0-15]   Checksum. (See Service Agent Header description.)

S2[0-15]   Notification ID.

S3[0-15]   Notification Information.

            For notification types 0, 1, and 2, NOTIFICATION INFO
            contains the following:


```
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    S3  |        0        |            stream ID         |
        +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
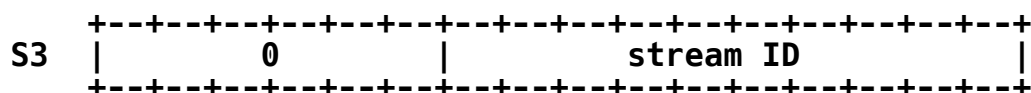
            For notification types 3, 4, 7, and 8, NOTIFICATION
            INFO contains the following:

```
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S3 |                  group address                |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

For notification types 5 and 6, which refer to all
streams or groups, NOTIFICATION INFO is zero.

6.4. Setup Acknowledgments

The host must acknowledge receipt of Setup Replies and Notifications
from the Service Agent, as described earlier. The format for the
Setup Acknowledgment message is shown in Figure 30.

```
            0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S0 |           0           |         CODE          |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S1 |                    CHECKSUM                    |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S2 |                   MESSAGE ID                   |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

SETUP ACKNOWLEDGMENT
Figure 30

S0[0-7]   Message Type = 0 (Acknowledgment).

S0[8-15] Code. This field indicates the type of acknowledgment.

          0 = Reply acknowledgment
          1 = Notification acknowledgment

S1[0-15] Checksum. (See Service Agent Header description.)

S2[0-15] Message ID. This is either a Request ID or a
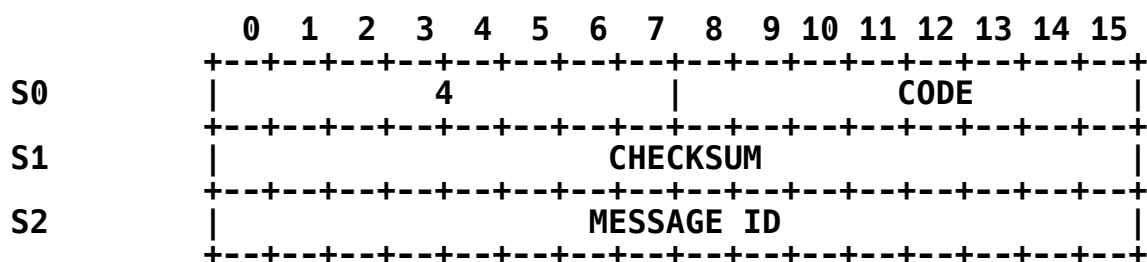   Notification ID.

6.5. Information Request / Reply Messages

The host may obtain information about WPS state and about what
resources the WPS currently has allocated for the host by sending an
Information Request message to the Service Agent. The Information
Reply that is returned will enable the host to determine 1) what

resources the WPS has allocated to the host, and 2) the current state
of the network and, possibly, certain network parameters. This
allows the host to refrain from trying to use resources it no longer
has, and to regain information it may have lost on its network
resources. This communication also informs the host of the network
state so that it may make priority and routing decisions.

Each Information Request (Figure 31) and Information Reply (Figure
32) message deals with a single type of resource at a time. The
header of the Information Reply message contains the number of
entries within the message, the number of 16-bit words in each entry,
and an instance of the appropriate information structure for each
resource the Information Reply message describes. These information
structures are described in Figures 33 and 34.

Future versions of the HAP protocol may permit queries about network
connectivity, estimated delay to a specified destination address
under specified conditions, etc. This is a section of the protocol
that is likely to expand in the future. Extensions are expected to
be backward compatible provided implementors do not hard code the
size of the returned information entries.

```
                0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S0     |           4           |         CODE          |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S1     |                    CHECKSUM                   |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        S2     |                   MESSAGE ID                  |
               +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

                     INFORMATION REQUEST MESSAGE
                              Figure 31
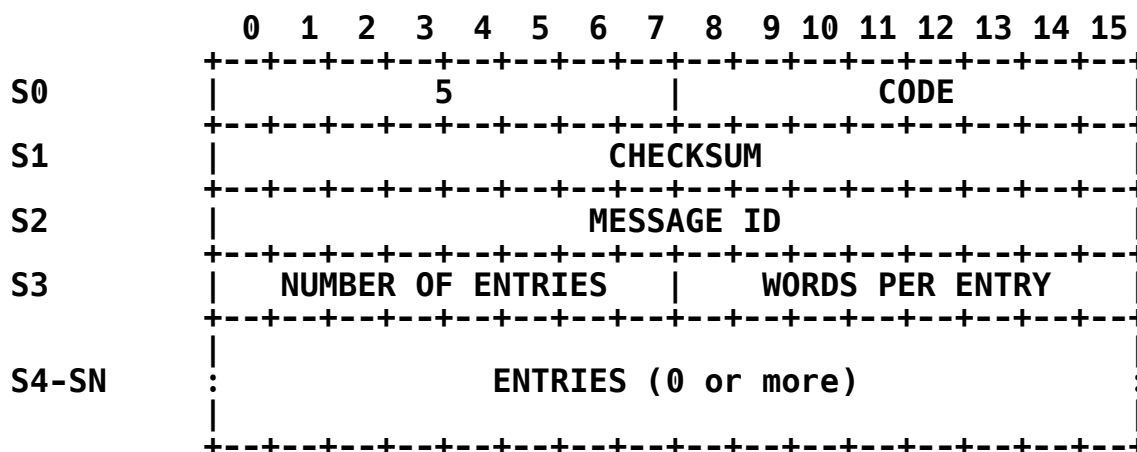

    S0[0-7]   Message type = 4 (Information Request).

    S0[8-15]   Code. This field identifies the Information Request
               Type.

                     1 = streams owned by host
                     2 = groups to which the host belongs

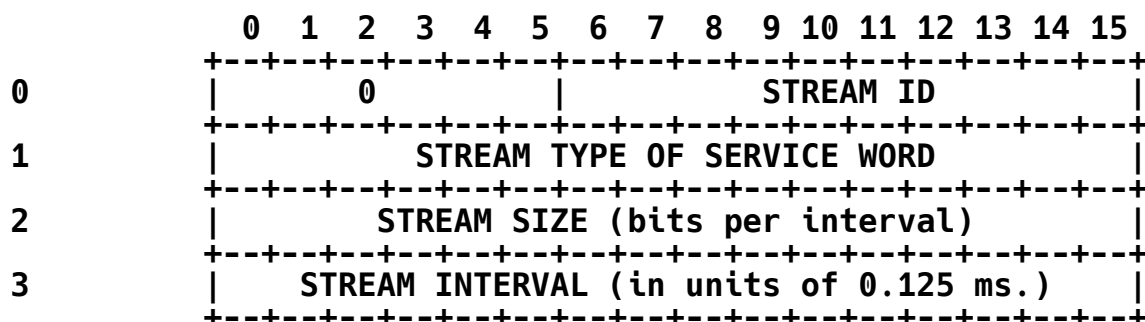   S1[0-15]   Checksum. (See Service Agent Header description.)

S2[0-15] Message ID. This field is assigned by the host to
          uniquely identify outstanding requests (Request ID).
          This ID is copied into Information Replies by the
          Service Agent.

```
              0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S0      |         5          |          CODE            |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S1      |                  CHECKSUM                      |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S2      |                  MESSAGE ID                    |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     S3      |   NUMBER OF ENTRIES    |    WORDS PER ENTRY    |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
             |                                                |
     S4-SN   :            ENTRIES (0 or more)                 :
             |                                                |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

                 INFORMATION REPLY MESSAGE
                        Figure 32

S0[0-7]   Message type = 5 (Information Reply).

S0[8-15] Code. This field identifies the Information Reply
          Type.

                1 = streams owned by host
                2 = groups to which the host belongs
                3 = error in Information Request message
                4 = network trouble
                5 = access not allowed

S1[0-15] Checksum. (See Service Agent Header description.)

S2[0-15] Message ID. This field is assigned by the host in the
          Information Request message to uniquely identify
          outstanding requests. This ID is copied into the
          Information Reply message by the Service Agent.

S3[0-7]   Number of entries included in the Information Reply
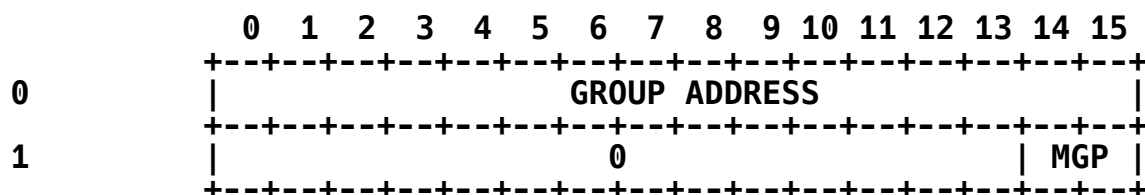   message.

S3[8-15]   Number of 16-bit words per entry.

S4-SN   Zero or more instances of either the stream information
            or group information structure.

```
             0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
0           |        0        |           STREAM ID          |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
1           |            STREAM TYPE OF SERVICE WORD          |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
2           |            STREAM SIZE (bits per interval)      |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
3           |        STREAM INTERVAL (in units of 0.125 ms.)  |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

STREAM INFORMATION
Figure 33

```
             0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
0           |                  GROUP ADDRESS                  |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
1           |                     0                  | MGP |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

GROUP INFORMATION
Figure 34

7. Host Access Link Monitoring

   While the access link is operating, statistics on traffic load and
   error rate are maintained by the host and WPS. Once a second, the
   host and WPS exchange this information via Status messages (Figure
   35). This periodic exchange of Status messages permits both ends of
   the link to monitor flows in both directions. The WPS also reports
   these monitoring statistics to the Network Operations Center (NOC).
   If either host or WPS fails to receive Status messages for ten
   seconds, the link will be restarted (see Section 8).

The link restart procedure initializes all internal WPS counts and
statistics for that link to zero. As data and control messages are
processed, counts are updated to reflect the total number of messages
sent, messages received correctly, and messages received with
different classes of errors since the last link restart. Whenever a
Status message arrives, a snapshot is taken of the local WPS counts.
The local receive counts, in conjunction with a sent count contained
in the received Status message, permits the computation of traffic
statistics in the one second update interval assuming that the set of
counts at the time of the previous monitoring report have been saved.
By including in the Status message sent (in the opposite direction)
the receive counts and the received sent count that was used with
them, the transmitting end of the access link as well as the
receiving end can determine the link performance from sender to
receiver.

```
              0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     0       | 1|LB|GOPRI|              0              |   0   |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     1       |                 HEADER CHECKSUM                |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     2       |               MOST RECENT A/R SENT             |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     3       |                 STREAM CAPACITY                |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     4       |                   TIMESTAMP                    |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     5       |                      SBU                       |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     6       |                      STU                       |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     7       |                      RNE                       |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     8       |                      RWE                       |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     9       |                      BHC                       |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    10       |                      HEI                       |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

STATUS MESSAGE
Figure 35

    0[0]   Message Class = 1 (Control Message).

    0[1]   Loopback indicator.

    0[2-3]   Go-Priority.

    0[4-11]   Reserved. Must be zero.

    0[12-15]   Control Message Type = 0 (Status).

    1[0-15]   Header Checksum. The checksum is the 2's-complement of
                the 2's-complement sum of words 0-10 (excluding the
                checksum word itself).

    2[0-15]   Most Recent A/R Sent. This field is a duplicate of the
                most recent acceptance/refusal word. It is included in
                the periodic Status message in case previous
                transmissions containing A/R information were lost.

    3[0-15]   Stream Capacity. When sent by the WPS, this field
                indicates how much stream capacity is unused, in units
                of data bits per millisecond. There is no guarantee
                that a request for a stream of this size will succeed.
                Since available capacity depends directly on a variety
                of parameters that can be selected by the user, the
                value of this field is the maximum capacity that could
                be achieved if existing streams were expanded at low
                reliability. This field is not meaningful in messages
                sent from the host to the WPS and must be set to zero.

    4[0-15]   Timestamp. This field indicates the time that the
                Status message was generated. When sent by a WPS, the
                time is in units of seconds since the last link
                restart. The host should also timestamp its messages
                in units of seconds.

    5[0-15]   Sent By Us. Count of messages sent by us since the
                last link restart (not including this one).

    6[0-15]   Sent To Us. Count of messages sent to us since the
                last link restart. This is the count from word 5 of
                the last Status message received.

    7[0-15]   Received, No Errors. This is the count of messages
                received without errors (since the last link restart)
                at the time that the last Status message was received.

    8[0-15]   Received With Errors. This is the count of messages

                    received with errors (since the last link restart) at
                    the time the last Status message was received.

        9[0-15]   Bad Header Checksums. This is the count of messages
                    received with bad header checksums (since the last link
                    restart) at the time the last Status message was
                    received.

        10[0-15]   Hardware Error Indication. This is the count of
                    messages received with hardware CRC errors or hardware
                    interface error indications (since the last link
                    restart) at the time the last Status message was
                    received.

8. Initialization

   The Host Access Protocol uses a number of state variables that must
   be initialized in order to function properly. These variables are
   associated with the send and receive message numbers used by the
   acceptance/refusal mechanism and the statistics maintained to support
   link monitoring. Link initialization should be carried out when a
   machine is initially powered up, when it does a system restart, when
   the ON state (see below) times out, when a loopback condition times
   out (see Section 9), or whenever the link transitions from non-
   operational to operational status.

   Initialization is accomplished by the exchange of Restart Request
   (RR) and Restart Complete (RC) messages between a host and a WPS.
   Either end (or both ends) may send an initial RR, and both ends must
   have sent and received an RC message in order to declare the link up.
   Because the RC message is a reply (to an RR or RC), receipt of an RC
   message by both ends guarantees that the physical link is operating
   in both directions. The initialization state diagram that must be
   implemented by both WPS and host is shown in Figure 36. Five states
   are identified in the state diagram:

     OFF   Entered upon recognition of a requirement to restart.
                The interface in the Host or WPS can recognize this
                requirement itself or be forced to restart by receipt
                of an RR message from the other end while in the ON
                state.

     INIT   Local state variables have been initialized but no RC
                messages have yet been sent or received. If receipt of
                an RR initiated the restart, or if an RR has been
                received since this restart began, send an RC
                (optional, reduces startup time). Otherwise, send an
                RR to alert the other end of the restart.

RR-SNT   A request to reinitialize (RR) has been sent to the
           other end, but no RR or RC messages have been received.

RC-SNT   An RC has been sent to the other end in response to an
           RR. The interface is waiting to receive an RC.

ON   RC messages have been both sent and received. Local
           counters have been zeroed. Data and control messages
           can now be exchanged between the WPS and host.

All states have 10-second timeouts (not illustrated) which return the
protocol to the OFF state. The occurrence of any events other than
those indicated in the diagram are ignored.
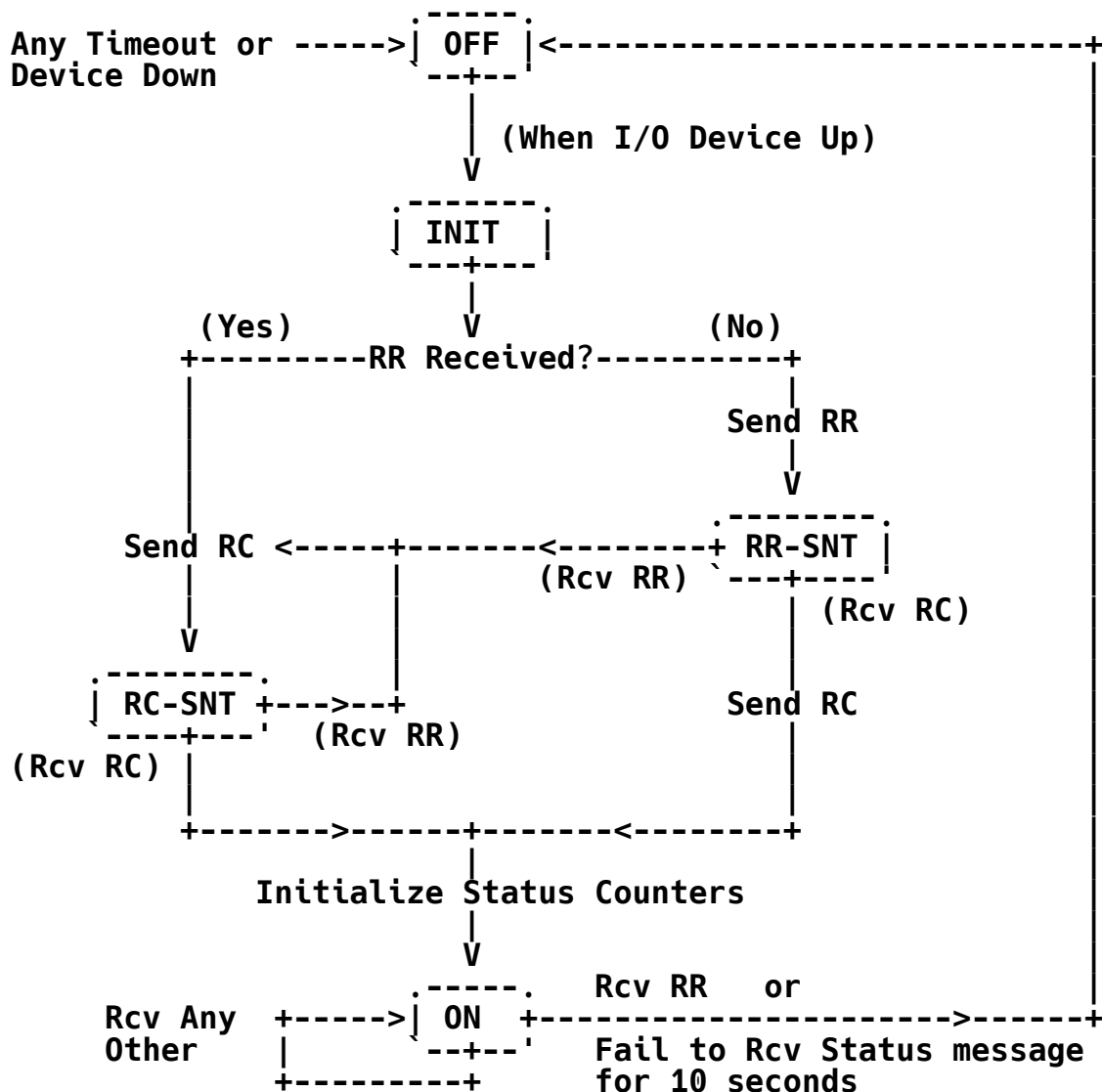
```
                              .-----.
 Any Timeout or ----->|  OFF  |<----------------------------+
 Device Down          --+--                                  |
                        |                                    |
                        |  (When I/O Device Up)              |
                        V                                    |
                     .-------.                               |
                    | INIT  |                                |
                     ---+---                                 |
                        |                                    |
        (Yes)           V              (No)                  |
         +---------RR Received?---------+                    |
         |                              |                    |
         |                           Send RR                 |
         |                              |                    |
         |                              V                    |
   Send RC <-----+-------<---------+ RR-SNT |                |
         |       |         (Rcv RR) `---+----'               |
         |       |                      |  (Rcv RC)          |
         V       |                      |                    |
    .---------.  |                   Send RC                 |
   | RC-SNT +--->--+                    |                    |
    ----+---'  (Rcv RR)                 |                    |
(Rcv RC) |                              |                    |
         |                              |                    |
         +-------->-------+-------<--------+                 |
                         |                                   |
             Initialize Status Counters                     |
                         |                                   |
                         V                                   |
                      .-----.    Rcv RR    or                |
 Rcv Any   +----->|  ON  +----------------------->------+
 Other     |       `--+--'  Fail to Rcv Status message
           +---------+     for 10 seconds
              HAP LINK RESTART STATE DIAGRAM
                      Figure 36
```

   The Restart Request control message (Figure 37) is sent by either a
host or a WPS when it wishes to restart a link. The Restart Request
causes all the monitoring statistics reported in the Status Message
to be reset to zero and stops all traffic on the link in both
directions. The Restart Complete message (Figure 38) is sent in
response to a received Restart Request or Restart Complete to
complete link initialization. The Restart Complete carries a field
used by the host to enable or disable the acceptance/refusal

mechanism for the link being restarted (see Section 5). After the
Restart Complete is processed, traffic may flow on the link.

The allocation and state of network resources (streams and groups)
are separate from the state of the host's access link(s) to the WPS.
The Information Request message (see Section 6.5) may be used by a
host to determine what resources it has. If the "SL" bit is set in
the Restart Complete message from the WPS, and if the host believes
it has resources allocated to it, the host is strongly encouraged to
use an Information Request to verify that it still has its resources.

```
                  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
   0             | 1|LB|    0     |VERSION |      0    |    3     |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
   1             |                  HEADER CHECKSUM               |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
   2             |                   HOST ADDRESS                 |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
   3             |                   LINK NUMBER                  |
                 +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

RESTART REQUEST
Figure 37

0[0]   Message Type = 1 (Control Message).

0[1]   Loopback indicator.

0[2-4]   Reserved. Must be zero.

0[5-7]   HAP version number. Use 1. Use of zero invokes
          backward compatibility code (see Appendix B).

0[8-11]   Reserved. Must be zero.

0[12-15] Control Message Type = 3 (Restart Request).

1[0-15]   Header Checksum. The checksum is the 2's-complement of
          the 2's-complement sum of words 0-3 (excluding the
          checksum word itself).

2[0-15]   Host Address. The WPS inserts the primary network
          address of the host. The host may insert any of its

network addresses in this field (hosts may have more
than one logical address per physical port). The WPS
will only bring up the HAP link if the host address is
valid for the port being used.

3[0-15]   Link Number. This field contains the sender's
          identification of the physical link being used. This
          information is used to identify the link when reporting
          errors to the Network Operations Center (NOC).

```
            0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
0          | 1|LB|   0     |VERSION |  0  |SL|AR|    4      |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
1          |                HEADER CHECKSUM                |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
2          |                 HOST ADDRESS                  |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
3          |                 LINK NUMBER                   |
           +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

RESTART COMPLETE
Figure 38

0[0]   Message Type = 1 (Control Message).

0[1]   Loopback indicator.

0[2-4]   Reserved. Must be zero.

0[5-7]   HAP version number. Use 1. Use of zero invokes
          backward compatibility code (see Appendix B).

0[8-9]   Reserved. Must be zero.

0[10]   Service loss alert (boolean) (WPS to host only; host
          must send zero). If the WPS has any reason to believe
          that the resources allocated to the host may not match
          what the host believes is allocated, SL is set to one.
          If SL is one, a host that believes it owns any resources
          is strongly encouraged to use an Information Request to
          verify that the resources are still allocated. SL will
          be one the first time a link is brought up after a WPS
          is restarted, and may be set in other cases.

```
0[11]      Acceptance/Refusal Control.  This bit is used by the
           host to enable or disable the acceptance/refusal
           mechanism for all traffic on the link.

              0 = Disable acceptance/refusal
              1 = Enable acceptance/refusal

0[12-15] Control Message Type = 4 (Restart Complete).

1[0-15]  Header Checksum.  Covers words 0-3.

2[0-15]  Host Address.

3[0-15]  Link Number.
```

9. Loopback Control

The Host Access Protocol provides a Loopback Request control message
which can be used by a WPS or a host to request the remote loopback
of its HAP messages. Such requests are usually the result of
operator intervention for purposes of system fault diagnosis. For
clarity in the following discussion, the unit (WPS or host)
requesting the remote loopback is referred to as the "transmitter"
and the unit implementing (or rejecting) the loopback is referred to
as the "receiver".

When the host access link is remotely looped, all HAP messages will
be returned, unmodified, over the access link by the receiver.
(Messages that are too long to be valid HAP messages may be discarded
instead of being returned.) The receiver will not send any of its
own messages to the transmitter while it is implementing the loop.
WPS-generated messages are distinguished from host-generated messages
by means of the Loopback indicator that is in every HAP message
header.

Two types of remote loopback may be requested: loopback at the
receiver's interface hardware and loopback at the receiver's I/O
driver software. HAP does not specify the manner in which the
receiver should implement these loops; additionally, some receivers
may use interface hardware which is incapable of looping the
transmitter's messages, only allowing the receiver to provide
software loops. A receiver may not be able to interpret the
transmitter's messages as it is looping them back. If such
interpretation is possible, however, the receiver will not act on any
of the transmitter's messages other than requests to reinitialize the
WPS-host link (Restart Request (RR) control messages; see Section 8.)

When a receiver initiates a loopback condition in response to a

loopback request, it makes an implicit promise to maintain the
condition for the duration specified in the Loopback Request message.
However, if an unanticipated condition such as a system restart
occurs in either the transmitter or the receiver, the affected unit
will try to reinitialize the WPS-host link by sending an RR message
to the other unit. If the RR message is recognized by the other
unit, a link initialization sequence can be completed. This will
restore the link to an unlooped condition even if the specified loop
duration has not yet expired. If a receiver cannot interpret a
transmitter's RR messages, and in the absence of operator
intervention at the receiver, the loop will remain in place for its
duration.

HAP does not specify the characteristics of any loopback conditions
that may be locally implemented by a given unit. An example of such
a condition is that obtained when a WPS commands its host interface
to loop back its own messages. If such local loop conditions also
cause the reflection of messages received from the remote unit, the
remote unit will detect the condition via the HAP header Loopback
indicator.

A specific sequence must be followed for setting up a remote
loopback. It begins after the HAP link has been initialized and a
decision is made to request a remote loop. The transmitter then
sends a Loopback Request message (Figure 39) to the receiver and
waits for either (1) a 10-second timer to expire, (2) a "Can't
implement loop" Unnumbered Response message from the receiver, or (3)
one of its own reflected messages. If event (1) or (2) occurs the
request has failed and the transmitter may, at its option, try again
with a new Loopback Request message. If event (3) occurs, the remote
loopback condition has been established. While waiting for one of
these events, messages from the receiver are processed normally.
Note that RR messages arriving from the receiver during this time
will terminate the loopback request.

When a receiver gets a Loopback Request message, it either implements
the requested loop for the specified duration, or returns a "Can't
implement loop" response without changing the state of the link. The
latter response would be returned, for example, if a receiver is
incapable of implementing a requested hardware loop. A receiver
should initiate reinitialization of the link with an RR message(s)
whenever a loopback condition times out.

There is one asymmetry that is required in the above sequence to
resolve the (unlikely) case where both WPS and host request a remote
loopback at the same time. If a WPS receives a Loopback Request
message from a host while it is itself waiting for an event of type
(1)-(3), it will return a "Can't implement loop" response to the host

and will continue to wait. A host in the converse situation,
however, will abort its loopback request and will instead act on the
WPS's loopback request.

```
              0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
0             | 1|LB|GOPRI|       0       | LOOP TYPE |    8     |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
1             |                 HEADER CHECKSUM                  |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
2             |                  LOOP DURATION                   |
              +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```
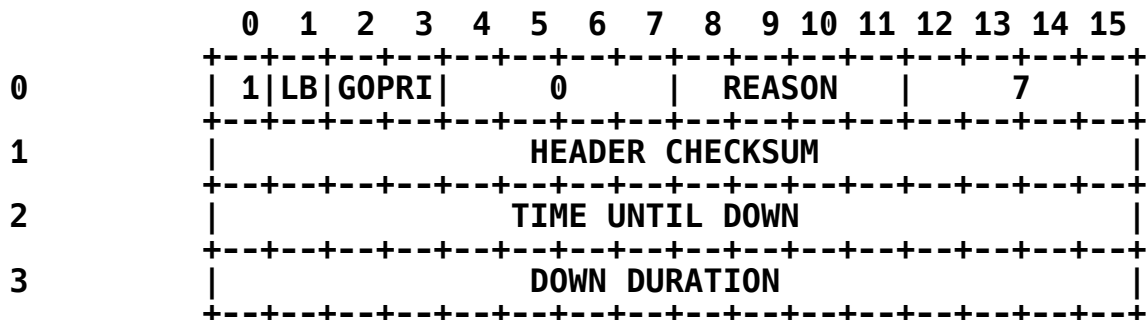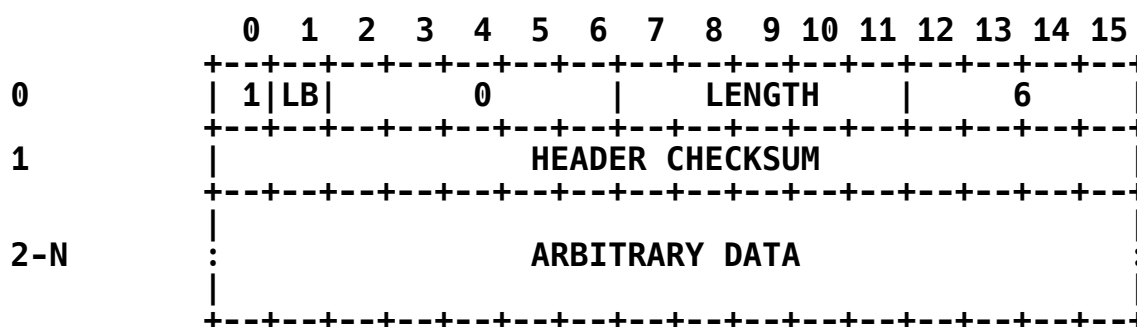
LOOPBACK REQUEST
Figure 39

0[0]   Message Type = 1 (Control Message).

0[1]   Loopback indicator.

0[2-3]   Go-Priority.

0[4-7]   Reserved. Must be zero.

0[8-11]   Loop Type. This field indicates the type of loop that
            is being requested as follows:

                 0 = Undefined
                 1 = Loop at interface (hardware loop)
                 2 = Loop at driver (software loop)
                 3-15 = Undefined

0[12-15]   Control Message Type = 8 (Loopback Request).

1[0-15]   Header Checksum. The checksum is the 2's-complement of
            the 2's-complement sum of words 0-2 (excluding the
            checksum word itself).

2[0-15]   Loop Duration. The transmitter of a Loopback Request
            message uses this field to specify the number of
            seconds that the loop is to be maintained by the
            receiver.

10. Other Control Messages

Before a WPS or a host voluntarily disables a WPS-host link, it
should send at least one Link Going Down control message (Figure 40)
over that link. HAP does not define the action(s) that should be
taken by a WPS or a host when such a message is received; informing
the Network Operations Center (NOC) and/or the network users of the
impending event is a typical course of action. Note that each Link
Going Down message only pertains to the WPS-host link that it is sent
over; if a host and a WPS are connected by multiple links, these
links may be selectively disabled.

A No Operation (NOP) control message (Figure 41) may be sent at any
time by a WPS or a host. A NOP message contains up to 32 words of
arbitrary data which are undefined by HAP. NOP messages may be
required in some cases to clear the state of the WPS-host link
hardware.

```
              0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  0          | 1|LB|GOPRI|       0      |  REASON  |    7    |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  1          |                 HEADER CHECKSUM                |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  2          |                 TIME UNTIL DOWN                |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
  3          |                 DOWN DURATION                  |
             +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

LINK GOING DOWN
Figure 40

0[0]   Message Type = 1 (Control Message).

0[1]   Loopback indicator.

0[2-3]   Go-Priority.

0[4-7]   Reserved. Must be zero.

0[8-11]   Reason. This field is used by the WPS or the host to
            indicate the reason for disabling this WPS-host link as
            follows:

```
                 0 = Cancel previous notice, not going down
                 1 = Unspecified reason
                 2 = Scheduled PM
                 3 = Scheduled hardware work
                 4 = Scheduled software work
                 5 = Emergency restart
                 6 = Power outage
                 7 = Software breakpoint
                 8 = Hardware failure
                 9 = Not scheduled up
                10 = Last warning:  The WPS or host will disable
                     the link in 10 seconds
                11-15 = Undefined
```

0[12-15]   Control Message Type = 7 (Link Going Down).

1[0-15]   Header Checksum. The checksum is the 2's-complement of
           the 2's-complement sum of words 0-3 (excluding the
           checksum word itself).

2[0-15]   Time Until Down. This field specifies the amount of
           time remaining until the WPS or host disables the link
           (in minutes). An entry of zero indicates that there is
           less than a minute remaining.

3[0-15]   Down Duration. This field specifies the amount of time
           that the WPS-host link will be down (in minutes). An
           entry of zero indicates that the down duration will be
           less than a minute. An entry of -1 (all bits set)
           indicates an indefinite down duration.

```
          0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
0         | 1|LB|        0       |    LENGTH    |    6    |
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
1         |                 HEADER CHECKSUM               |
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
          |                                               |
2-N       :                 ARBITRARY DATA                :
          |                                               |
          +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

                        NO OPERATION (NOP)
                            Figure 41

```
   0[0]        Message Type = 1 (Control Message).

   0[1]        Loopback indicator.

   0[2-6]      Reserved.  Must be zero.

   0[7-11]     Length.  The number of words of arbitrary data.

   0[12-15]    Control Message Type = 6 (NOP).

   1[0-15]     Header Checksum.  The checksum is the 2's-complement of
               the 2's-complement sum of words 0-N (excluding the
               checksum word itself).

   2-N         Arbitrary Data.  Up to 32 words of data may be sent.
               The data are undefined by HAP.
```

11. Appendix A -- Future Extensions

The extensions to HAP described below are included to provide
additional context for the understanding of HAP's current
capabilities, as well as suggest how HAP may be enhanced in the
future to provide better support for multi-site conferencing. These
capabilities are not supported by TWBNET.

One change under consideration is the addition of a "conference"
resource, which would own some number of streams and groups and
improve the network's ability to meet the needs of video conference
users. A single request to modify the "conference", such as to add a
new member, would result in modifying all the streams in the
conference to include the new member, modifying the conference's
primary group address to add the new member, etc., in a single
network operation. Such a capability would not only simplify
conference resource management for hosts, but also reduce the number
of network setup operations, permit more nearly "atomic" decisions of
whether a particular conference modification is possible, and reduce
the problem of recovery if modification is not possible.

Another change under consideration is the addition of "shared
streams." This capability would allow hosts to share a single
allocation of network bandwidth (and other resources) wherever the
streams shared a common communication path. Hosts using a shared
stream must be willing to restrict their total transmission rate to
the rate of the shared bandwidth. Multi-site conferences could use
such a capability to avoid allocating full bandwidth for voice data
for all conference members. Instead, bandwidth for, say, four active
voices at once could be allocated and shared, and voice messages
would only be lost when more than four people tried to talk at once.

The Create Shared Stream Request would use a different request code
than Create Stream Request, and the setup message would likely
contain at least one additional field to identify the set of shared
streams. Change and Delete Stream requests could be used for both
shared and non-shared streams.

12. Appendix B -- Backward compatibility

The WPS will support the use of HAP version 0 by hosts until all
hosts have upgraded to version 1. The WPS determines which HAP
version the host is using by examining the Restart Request and/or
Restart Complete control messages sent by the host to the WPS. If
the host initiates a restart and thus sends both a Restart Request
and a Restart Complete, and if the HAP version numbers in the two
messages differ, the version number in the Restart Complete will
prevail. The WPS will always set the version number to 1. If the
host sends 0 in the version number field, version 0 compatiblity mode
will be invoked.

Version 0 of HAP did not contain the PROTOCOL ID field in the
datagram and stream message headers. Instead, the IL bit in the Type
of Service word was used to indicate the presence or absence of an
Internet Protocol (IP) header (any version number) following the HAP
header. This is the original description of that bit:

   3[1]   Internet/Local Flag. This flag is set by a source host to
           specify to a destination host whether the data portion of
           the message contains an Internet Protocol (IP) header [3].
           This field is passed transparently by the source and
           destination WPSen for traffic between network hosts. This
           field is examined by WPS Agents in order to support
           Internet operation.

                   0 = Internet
                   1 = Local

Conversion Algorithms

Link control messages (e.g., Restart Request) do not require
conversion. Datagram and stream messages sent by or to a host
running HAP version 0 will be converted by the WPS. Message
conversion will probably cause the maximum throughput of hosts using
HAP version 0 to be somewhat lower than that of hosts using HAP
version 1.

HAP version 0 used the IL bit in the HAP Type of Service word to
indicate the presence or absence of an IP header. Version 1 uses the
Protocol ID field. To convert host-to-WPS messages, the IL bit will

be cleared, and the protocol ID field will be inserted, with the
value indicated:

```
     IL was    Destination   Protocol ID set to:
     ------    -----------   -------------------
       0           any       HAP_PROTO_IP   (0x800)
       1       Service Agent  HAP_PROTO_SETUP (1)
       1          other       HAP_PROTO_NONE  (0)
```

   To convert WPS-to-host messages, the protocol ID field will be
   deleted, and the IL bit will be set by:
             IL = (protocol_id was HAP_PROTO_IP) ? 0 : 1;

   HAP_PROTO_IP (see Appendix C) will be used for IP "versions" 3
   (GG protocol), 4 (IP), and 5 (ST).

The datagram message header fields TTL and PRI have been swapped in
HAP version 0 compared to version 1. The conversion code swaps the
contents of these two fields for hosts running version 0.

The stream message header field TTL in HAP version 0 was replaced by
the PRE field in version 1. Since the only permitted value of TTL
was 1, and it is a valid PRE value, no conversion is necessary.

In HAP version 0, messages between a host and the Service Agent were
allowed to contain Internet Protocol headers. No hosts use that
capability, so no provision will be made to accommodate IP headers in
Setups between hosts and the Service Agent.

In version 0, the Restart Request control message contained a "reason
for restart" field. That field was ignored in all current
implementations and has been eliminated in version 1.

Current implementations expect the WPS to insert an "incarnation
count" in bits 5-10 of the first word of both Restart Request and
Restart Complete messages. This functionality has been replaced by
the "SL" bit in the Restart Complete message in version 1.
Compatibility code will be added if needed, but it is expected that
none will be needed.

13. Appendix C -- HAP Protocol ID Assigned Numbers

   This section lists the values of the PROTOCOL ID field. This part of
   the specification will be obsolete when a version of the Assigned
   Numbers RFC containing HAP protocol ID numbers is issued.

   HAP adopts the Ether-type numbers in the 1500-65535 range. Protocol
   IDs 256-511 identify ISO protocols. Zero indicates the absence of a

higher level protocol header. Other protocol IDs are reserved for
future assignment.

```
     Protocol ID      Indicates
     -----------      ---------
          0           No higher level protocol
          1           For Network Service Agent messages
        2-255         Reserved
       256-511        ISO protocol identifier + 256
      512-1499        Reserved
     1500-65535       Identical to Ether-type [10].
```

                 HAP PROTOCOL ID NUMBERS
                        Figure 42

REFERENCES

   1. Falk, G., Groff, S., Koolish, R., and W. Milliken, "PSAT
      Technical Report", BBN Technical Report No. 4469, Chapter 4, May
      1981.

   2. Rees, T., Editor, "A Host Access Protocol Specification", BBN
      Laboratories, Inc., May 1987. (A revision of RFC 907 that was
      distributed to DARPA and the WBNET user community but not
      resubmitted as an RFC.)

   3. Postel, J., Editor, "Internet Protocol - DARPA Internet Program
      Protocol Specification", RFC 791, USC/Information Sciences
      Institute, September 1981.

   4. Topolcic, C., Editor, "Experimental Internet Stream Protocol,
      Version 2 (ST-II)", RFC 1190, Bolt Beranek and Newman, Inc.,
      October 1990.

   5. Edmond, W., Seo, K., Leib, M., and C. Topolcic, "The DARPA
      Wideband Network Dual Bus Protocol", Proceedings of ACM SIGCOMM
      '90, pages 79-89, September 24-27, 1990.

   6. "Host/SATNET Protocol", Internet Engineering Note (IEN) 192, July
      1981.

   7. Evenchik, L., McNeill, D., Bressler, R., Owen, A., Rice, Jr., R.,
      Trout, G., Pavey, C., Damer, R., Deckelman, F., and T. Hughes,
      "MATNET, An Experimental Navy Shipboard Satellite Communications
      Network", Proceedings of INFOCOM '82, pages 3-11, March 30 -
      April 1, 1982.

   8. Falk, G., Groff, J., Milliken, W., Nodine, M., Blumenthal, S.,
      and W. Edmond, "Integration of Voice and Data in the Wideband
      Packet Satellite Network", IEEE Journal on Selected Areas in
      Communications, Vol. SAC-1, No. 6, December 1983.

   9. "Interface Message Processor: Specifications for the
      Interconnection of a Host and an IMP", BBN Technical Report No.
      1822, October 1980.

   10. Reynolds, J., and J. Postel, "Assigned Numbers", RFC 1060,
       USC/Information Sciences Institute, March 1990.

Security Considerations

   Security issues are not discussed in this memo.

Author's Address

   Winston Edmond
   Bolt Beranek and Newman, Inc.
   Network Technologies Department
   10 Moulton Street
   Cambridge, Massachusetts 02138

   Phone: (617) 873-3000

   EMail: wbe@bbn.com