RFC 907

HOST ACCESS PROTOCOL SPECIFICATION

July 1984

prepared for

Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, Virginia 22209

by

Bolt Beranek and Newman Laboratories
10 Moulton Street
Cambridge, Massachusetts 02238

Preface (Status of this Memo)

This document specifies the Host Access Protocol (HAP). Although HAP was originally designed as the network-access level protocol for the DARPA/DCA sponsored Wideband Packet Satellite Network, it is intended that it evolve into a standard interface between hosts and packet-switched satellite networks such as SATNET and TACNET (aka MATNET) as well as the Wideband Network. The HAP specification presented here is a minor revision of, and supercedes, the specification presented in Chapter 4 of BBN Report No. 4469, the "PSAT Technical Report". As such, the details of the current specification are still most closely matched to the characteristics if the Wideband Satellite Network. Revisions to the specification in the "PSAT Technical Report" include the definition of three new control message types (Loopback Request, Link Going Down, and NOP), a "Reason" field in Restart Request control messages, new Unnumbered Response codes, and new values for the setup codes used to manage streams and groups.

HAP is an experimental protocol, and will undergo further revision as new capabilities are added and/or different satellite networks are supported. Implementations of HAP should be performed in coordination with satellite network development and operations personnel.

Table of Contents

FIGURES

1 Introduction

The Host Access Protocol (HAP) specifies the network-access level communication between an arbitrary computer, called a host, and a packet-switched satellite network. The satellite network provides message delivery services for geographically separated hosts: Messages containing data which are meaningful to the hosts are submitted to the network by an originating (source) host, and are passed transparently through the network to an indicated destination host. To utilize such services, a host interfaces to the satellite network via an access link to a dedicated packet-switching computer, known as a Satellite Interface Message Processor (Satellite IMP or SIMP). HAP defines the different types of control messages and (host-to-host) data messages that may be exchanged over the access link connecting a host and a SIMP. 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.

The term "Interface Message Processor" originates in the ARPANET, where it refers to the ARPANET's packet-switching nodes. SIMPs differ from ARPANET IMPs in that SIMPs form a network via connections to a common multiaccess/broadcast satellite channel, whereas ARPANET IMPs are interconnected by dedicated point-to-point terrestrial communications lines. This fundamental difference between satellite-based and ARPANET-style networks results in different mechanisms for the delivery of messages from source to destination hosts and for internal network coordination. Additionally, satellite networks tend to offer different type of service options to their connected hosts than do ARPANET-style networks. These options are included in the Host Access Protocol presented here.

Several types of Satellite IMPs have been developed on a variety of processors for the support of three different packet-switched satellite networks. The original SIMP was employed in the Atlantic Packet Satellite Network (SATNET). It was developed from one of the models of ARPANET IMP, and was implemented on a Honeywell 316 minicomputer. The 316 SIMPs were succeeded in SATNET by SIMPs based on BBN C/30 Communications Processor hardware. The C/30 SIMPs have also been employed in the Mobile

Access Terminal Network (MATNET). The SATNET and MATNET SIMPs implement a network-access level protocol known as Host/SATNET Protocol. Host/SATNET Protocol is the precursor to HAP and is documented in Internet Experiment Note (IEN) No. 192. The Wideband Satellite Network, like SATNET, has undergone an evolution in the development of its SIMP hardware and software. The original Wideband Network SIMP is known as the Pluribus Satellite IMP, or PSAT, having been implemented on the BBN Pluribus Multiprocessor. Its successor, the BSAT, is based on the BBN Butterfly Multiprocessor. Both the PSAT and the BSAT communicate with their connected network hosts via HAP.

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 many of the topics addressed in this HAP specification can be found in BBN Report No. 4469, the "PSAT Technical Report".

The protocol used to provide sufficiently reliable message exchange over the host-SIMP link is assumed to be transparent to the network-access protocol defined in this document. Examples of such link-level protocols are ARPANET 1822 local and distant host, ARPANET VDH protocol, 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 SIMP and the host. Transmission is nonreliable in the sense that the protocol does not provide any guarantee of error-free sequenced delivery. To the extent that this functionality is required on the access link (e.g., non-collocated SIMP and host operating over a communication circuit), it must be supported by the link-level protocol below HAP. The flow control mechanism operates independently in each direction except that enabling or disabling the mechanism applies to both sides of the interface.

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 approximately 16K bits in length. Datagram messages provide the basic transmission service in the satellite network. Datagram messages transmitted by a host experience a nominal two satellite hop end-to-end network delay. (Note that this delay, of about 0.6 sec excluding access link delay, is associated with datagram transmission between hosts on different SIMPs. The transmission delay between hosts on the same SIMP will be much smaller assuming the destination is not a group address. See Section 3 and 6.2.) A datagram control header, passed to the SIMP by the host along with message text, determines the processing of the message within the satellite network independent of any previous exchanges.

Stream messages provide a one satellite hop delay (approximately 0.3 sec) for volatile traffic, such as speech, which cannot tolerate the delay associated with datagram transmission. Hosts may also use streams to support high duty cycle applications which require guaranteed channel bandwidth. 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 satellite 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 SIMP to which it is attached. These 16-bit logical addresses are provided in all Host-to-SIMP and SIMP-to-Host data messages.

Hosts may also be members of groups. Group addressing is provided primarily to support the multi-destination delivery required for conferencing applications. Like streams, group addresses are dynamically created and deleted by the use of setup messages exchanged between a host and the network. Membership in a group may consist of an arbitrary subset of all the permanent network hosts. A message addressed to a group address is delivered to all hosts that are members of that group.

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 SIMP and transfers from SIMP-to-SIMP over the satellite channel. The SIMP offers users a choice of network error protection options based on the network's ability to selectively send messages over the satellite channel at different coding rates. These forward error correction (FEC) options are referred to as reliability levels. Three reliability levels (low, medium, and high) are available to the host.

In addition to forward error correction, a number of checksum mechanisms are employed in the satellite network to add 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 SIMP 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 satellite channel error rate. 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 insure that SIMPs 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-level error control procedures (such as HDLC) will be used for this purpose.

Datagram and stream messages being presented to the network by a host may not be accepted for a number of reasons: priority too low, destination dead, lack of buffers in the source SIMP, etc. The host faces a similar situation with respect to handling messages from the SIMP. 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 SIMP'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 SIMP (and conversely). This allows the host (or SIMP) to retry refused messages at its discretion and can provide information useful for optimizing the sending of subsequent messages if the reason for refusals is also provided. The A/R mechanism can be disabled to provide a "pure discard" interface.

Each message submitted to the SIMP by a host is marked as being in one of four priority classes, from priority 3 (highest) through priority 0 (lowest). The priority class is used by the SIMP for arbitrating contention for scarce network resources (e.g., channel time). That is, if the network cannot deliver all of the offered messages, high priority messages will be delivered in preference to low priority messages. In the case of datagrams, priority level is used by the SIMP for ordering

satellite channel reservation requests at the source SIMP and message delivery at the destination SIMP. In the case of streams, priority is associated with the ability of one stream to preempt another stream of lower priority at setup time.

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-SIMP 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 control messages. Data messages are used to send information between network hosts. Control messages are exchanged between a host and the network to manage the local access link. HAP can also be viewed in terms of two distinct protocol layers, the message layer and the setup layer. The message layer is associated with the transmission of individual datagram messages and stream messages. The setup layer protocol is associated with the establishment, modification, and deletion of streams and groups. Setup layer exchanges are actually implemented as datagrams transmitted between the user host and an internal SIMP "service host."

Every HAP message consists of an integral number of 16-bit words. 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 (least significant) to fifteen (most 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. For HDLC and ARPANET VDH, each word of a HAP message is transmitted starting with the least significant bit (bit 0) and ending with the most significant bit (bit 15). The words of the message are transmitted from word 0 to word N. For ARPANET 1822 local and distant host interfaces, the order of bit transmission within each word is the reverse of that for HDLC and VDH, i.e., the transmission is from bit 15 to bit 0.

3 Datagram Messages

Datagram messages are one of the two types of message level data messages used to support host-to-host communication. Each datagram can contain up to 16,384 bits of user data. Datagram messages transmitted by a host to a host on a remote SIMP experience a nominal two satellite hop end-to-end network delay (about 0.6 sec), excluding delay on the access links. This network delay is due to the reservation per message scheduling procedure for datagrams which only allocates channel time to the message for the duration of the actual transfer. Since datagram transfers between permanent hosts on the same SIMP do not require satellite channel scheduling prior to data transmission, the network delay in this case will be much smaller and is determined strictly by SIMP processing time. Datagrams sent to group addresses are treated as if they were addressed to remote hosts and are always sent over the satellite channel. 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.

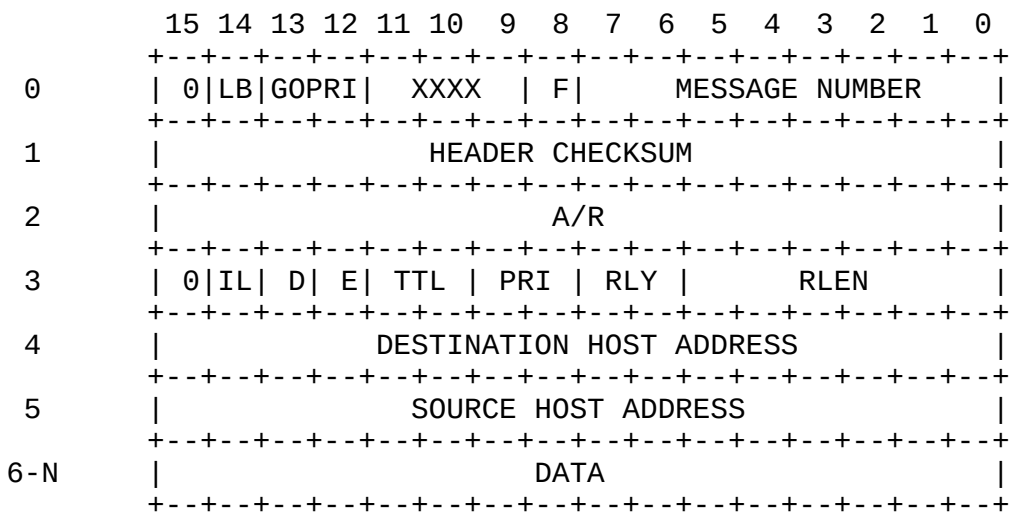

Figure 1 . DATAGRAM MESSAGE

0[15] Message Class. This bit identifies the message as a data message or a control message.

0 = Data Message
1 = Control Message

0[14] Loopback Bit. This bit allows the sender of a message to determine if its own messages are being looped back. The host and the SIMP 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 SIMP

0[12-13] Go-Priority. In SIMP-to-Host messages, this field provides advisory information concerning the lowest priority currently being accepted by the SIMP. The host may optionally choose to provide similar priority information to the SIMP.

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

0[9-11] Reserved.

0[8] Force Channel Transmission Flag. This flag can be set by the source host to force the SIMP to transmit the message over the satellite channel even if the message contains permanent destination and source host addresses corresponding to hosts which are physically connected to the same SIMP.

0 = Normal operation
1 = Force channel transmission

0[0-7] 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. This field contains a checksum which covers words 0-5. It is computed as the negation of the 2's-complement sum of words 0-5 (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[15] Data Message Type. This bit identifies whether the message is a datagram message or a stream message.

0 = Datagram Message
1 = Stream Message

3[14] 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 a standard DoD Internet header. This field is passed transparently by the source and destination SIMPs for traffic between external satellite network hosts. This field is examined by internal SIMP hosts (e.g., the network service host) in order to support Internet operation.

0 = Internet
1 = Local

3[13] Discard Flag. This flag allows a source host to instruct the satellite 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[12] 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 SIMP-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[10-11] 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 satellite network before being deleted. Messages may be discarded by the network prior to this maximum elapsed time.

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

The Time-to-Live field is undefined in messages sent from a SIMP to a host.

3[8-9] 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-Low Priority
2 = Medium-High Priority
3 = High Priority

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

3[6-7] Reliability. The source host uses this field to specify the basic bit error rate requirement for the data portion of this message. The source SIMP uses this field to determine the satellite channel transmission parameters required to provide that bit error rate.

0 = Low Reliability
1 = Medium Reliability

2 = High Reliability
3 = Reserved

The Reliability field is undefined in messages sent from a SIMP to a host.

3[0-5] Reliability Length. This 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 six message header words and the first Reliability Length words of user data will be transmitted at Reliability=2 while the remainder of the user data will be transmitted at whatever reliability level is specified in field 3[6-7]. 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. The Reliability Length field is undefined in messages sent from a SIMP to a host.

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

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

6-N Data. This field contains up to 16,384 bits (1024 16-bit words) of user data.

4 Stream Messages

Stream messages are the second type of message level data messages. As noted in Section 2, streams exist primarily to provide a one satellite hop delay for volatile traffic such as speech. Hosts may also use streams to support high duty cycle applications which require guaranteed channel bandwidth.

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 associated with a recurring channel allocation within the satellite network. This fixed allocation imposes rather strict requirements on the host using the stream if efficient channel utilization is to be achieved. In particular, stream messages must match the stream allocation both in terms of message size and message interarrival time.

Within the bounds of its stream allocation, a host is permitted considerable flexibility in how it may use a stream. Although the priority, reliability, and reliability length of each stream message is fixed at stream creation 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 host stream. The format of stream messages is described in Figure 2.

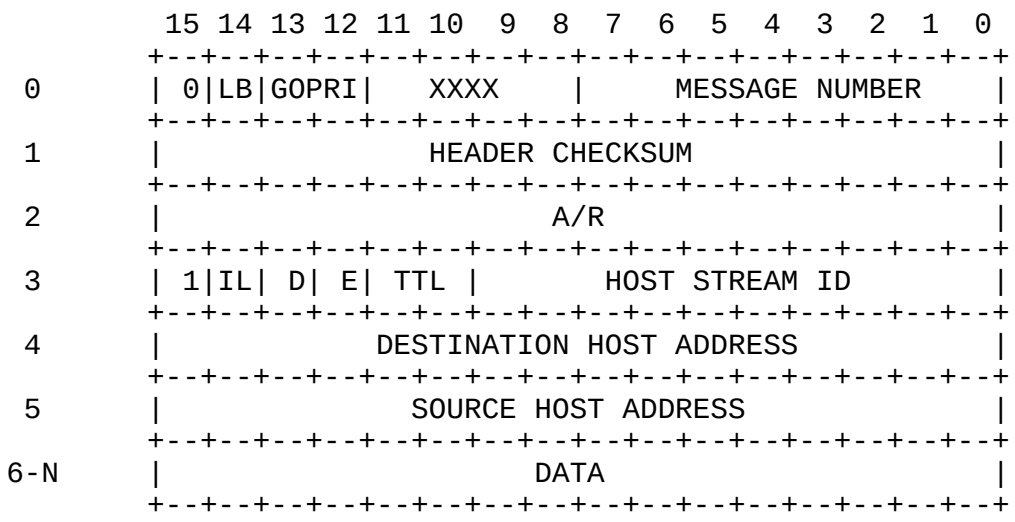

Figure 2 . STREAM MESSAGE

0[15] Message Class = 0 (Data Message).

0[14] Loopback Bit.

0[12-13] Go-Priority.

0[8-11] Reserved.

0[0-7] 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. Covers Words 0-5.

2[0-15] Piggybacked A/R.

3[15] Data Message Type = 1 (Stream).

3[14] Internet/Local Flag.

3[13] Discard Flag.

3[12] Data Error Flag.

3[10-11] Time-to-live Designator.

0 = Reserved
1 = 1 second
2 = Reserved
3 = Reserved

3[0-9] Host Stream ID. The service host uses this field to identify the host stream over which the message is to be sent by the SIMP. Host stream IDs are established at stream creation time via host exchanges with their network service host (see Section 6.1).

4[0-15] Destination Host Address.

5[0-15] Source Host Address.

6-N Data. This field contains up to 16,000 bits of user data (multiple of 16-bits).

5 Flow Control Messages

The SIMP 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. Both the host and the SIMP increment this number modulo 256 in successive messages they transmit. Up to 127 messages may be outstanding in each direction at any time. 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 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 not refused. Acceptance indications returned by the SIMP do not, however, imply a guarantee of delivery or even any assurance that the message will not be intentionally discarded by the network at a later time. 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. (Note that explicit acceptance of message number N does not imply that all of the unacknowledged outstanding messages have been received.) An analogous interpretation of refusal message number allows the receiver of a group of messages to reject them as a group assuming that they all are being refused for the same reason. As a further efficiency measure, HAP permits a block of A/R indications to be aggregated 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.

The format shown in Figure 3 is used both for piggybacking A/R indications on data messages (word 2), and for providing A/R information in separate control messages. When separate control messages are used to transmit A/R indications, the format shown in Figure 4 applies. 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.

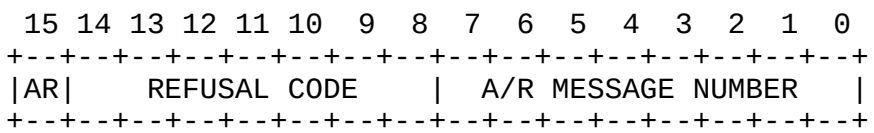

Figure 3 . ACCEPTANCE/REFUSAL WORD

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

0 = Acceptance
1 = Refusal

[8-14] Refusal Code. When the Acceptance/Refusal Type = 1, this field gives the Refusal Code.

0 = Priority not being accepted
1 = Source SIMP congestion
2 = Destination SIMP congestion
3 = Destination host dead
4 = Destination SIMP dead
5 = Illegal destination host address
6 = Destination host access not allowed
7 = Illegal source host address
8 = Message lost in access link
9 = Nonexistent 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 = Illegal reliability value
16 = Destination host congestion

[0-7] 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.

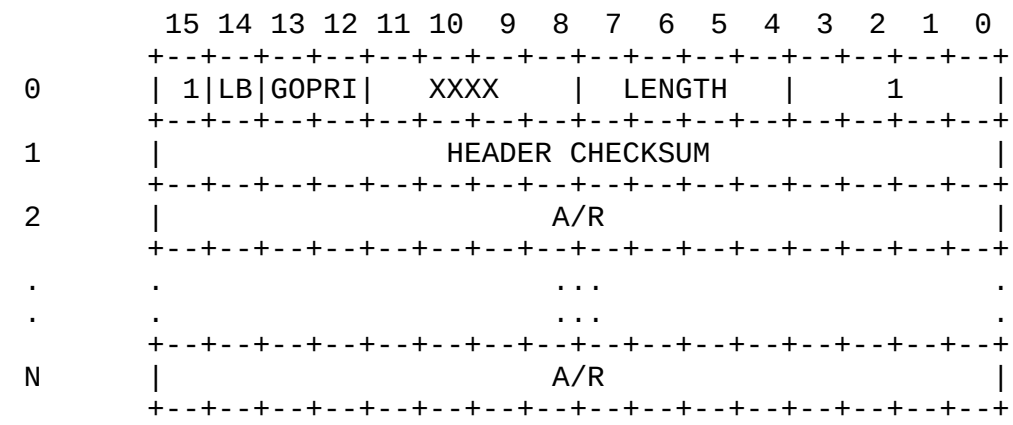

Figure 4 . ACCEPTANCE/REFUSAL MESSAGE

0[15] Message Class = 1 (Control Message).

0[14] Loopback Bit.

0[12-13] Go-Priority.

0[8-11] Reserved.

0[4-7] Message Length. This field contains the total length of this message in words (N+1).

0[0-3] Control Message Type = 1 (Acceptance/Refusal).

1[0-15] Header Checksum. The checksum covers words 0-N.

2[0-15] Acceptance/Refusal Word.

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

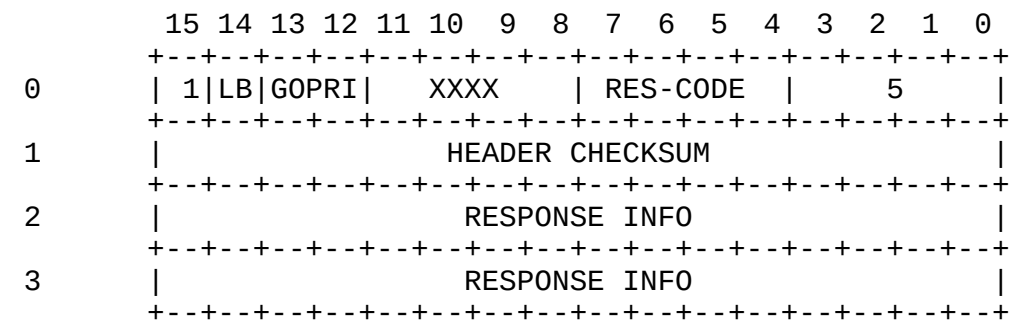

Figure 5 . UNNUMBERED RESPONSE

0[15] Message Class = 1 (Control Message).

0[14] Loopback Bit.

0[12-13] Go-Priority.

0[8-11] Reserved.

0[4-7] 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[0-3] Control Message Type = 5 (Unnumbered Response).

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

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 zero if no word 3
15, Word 2 of Loopback Request message

6 Setup Level Messages

Setup level protocol is provided to support the establishment, modification, and deletion of groups and streams in the packet satellite network. A host wishing to perform one of these generic operations interacts with the network service host (logical address zero). The service host 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 group or stream.

The communication between the host and the service host is implemented via special-purpose datagrams called setup messages. Each interaction initiated by a host involves a 3-way exchange where: (1) the user host sends a Request to the service host, (2) the service host returns a Reply to the user host, and (3) the user host returns a Reply Acknowledgment to the service host. This procedure is used to insure reliable transmission of 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 Reply Acknowledgment are identified by the Request ID that they contain. Hosts should generally expect a minimum delay of about two satellite round-trip times between the transmission of a setup Request to the SIMP and the receipt of the associated Reply. (Note that the Join Group Request and the Leave Group Request require only local communication between a host and its SIMP. The response time for these requests, therefore, is dependent solely on SIMP processing time and should be considerably shorter than two round-trip times.) This delay establishes a maximum rate at which changes can be processed by the SIMP. The user should receive a reply to a setup request requiring global communication within 2 seconds and to a setup request requiring local communication within 1 second. The host should respond to a SIMP Reply with a Reply Acknowledgment within 1 second.

Setup exchanges can also be initiated by the SIMP. SIMP-initiated setup messages are used to notify a host of changes in the status of an associated group or stream. Each notification involves a 2-way exchange where: (1) the service host sends a Notification to the user host, and (2) the user host returns a Notification Acknowledgment to the service host. In order to allow more than one Notification to be outstanding, each is assigned a unique Notification ID. The Notification Acknowledgment returned by the user host to the service host must contain the Notification ID.

The general format of every setup message is:

<DATAGRAM MESSAGE HEADER>
<OPTIONAL INTERNET HEADER>
<SETUP MESSAGE HEADER>
<SETUP MESSAGE BODY>

The service host accepts setup requests in either Internet or non-Internet format. Replies from the service host will be in the same form as the request, that is, Internet requests get Internet replies, and non-Internet requests get non-Internet replies.

The format of the combined datagram message header and setup message header is illustrated in Figure 6. The body of the setup messages depends on the particular setup message type. Stream request and reply messages are described in Section 6.1. Group request and reply messages are described in Section 6.2. To simplify the presentation in both of these sections, the setup messages are assumed to be exchanged between a local host and SIMP even though Internet group and stream setups are supported (see Figure 6). The format of notifications, which consists of only a single word beyond the basic setup header, is shown in Figure 7. Since the SIMP does not retain the optional Internet header information that can be included in setup requests, Internet notifications are not supported. The format of acknowledgment messages associated with request/reply and notification setups is illustrated in Figure 8.

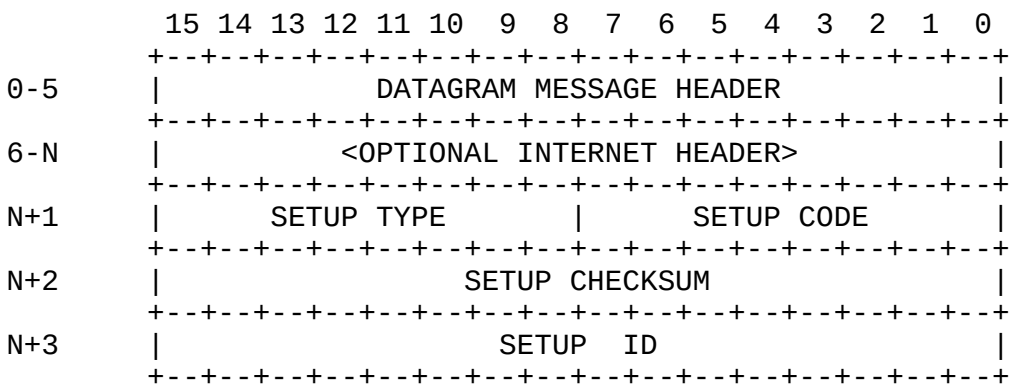

Figure 6 . SETUP MESSAGE HEADER

0-5 Datagram Message Header. Each setup message begins with the six word datagram message header (see Section 3).

6-N Internet Header (Optional). These fields, when present, conform to the DoD Standard Internet Protocol (IP). The Internet header size is a minimum of 10 words but can be longer depending on the use of optional IP facilities. (Internet notification messages are not supported.)

N+1[8-15] Setup Type. This field determines the type of setup message.

0 = Acknowledgment
1 = Request
2 = Reply
3 = Notification

N+1[0-7] Setup Code. For requests, this field identifies the

Request Type.

1 = Create group address
2 = Delete group address
3 = Join group
4 = Leave group
5 = Create stream
6 = Delete stream
7 = Change stream parameters
8 = Reserved

For 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 = Group joined
3 = Group left
4 = Stream changed
5 = Reserved
6 = Bad request type
7 = Reserved
8 = Network trouble
9 = Bad key
10 = Group address/stream ID nonexistent
11 = Not member of group/creator of stream
12 = Stream priority not being accepted
13 = Reserved
14 = Reserved
15 = Illegal interval
16 = Reserved
17 = Insufficient network resources
18 = Requested bandwidth too large
19 = Reserved
20 = Reserved
21 = Maximum messages per slot not consistent with slot size
22 = Reply lost in network
23 = Illegal reliability value

For Notifications, this field contains the Notification Type.

0 = Stream suspended
1 = Stream resumed
2 = Stream deleted
3 = Group deleted by host
4 = Group deleted by SIMP
5 = All streams deleted
6 = All groups deleted

For Acknowledgments, this field contains the Acknowledgment Type.

0 = Reply acknowledgment
1 = Notification acknowledgment

N+2[0-15] Setup Checksum. The checksum covers the three setup message header words and the setup message body data words. Setups received with bad checksums must be discarded.

N+3[0-15] Setup ID. This field is assigned by the host to uniquely identify outstanding requests (Request ID) and by the service host to uniquely identify outstanding notifications (Notification ID).

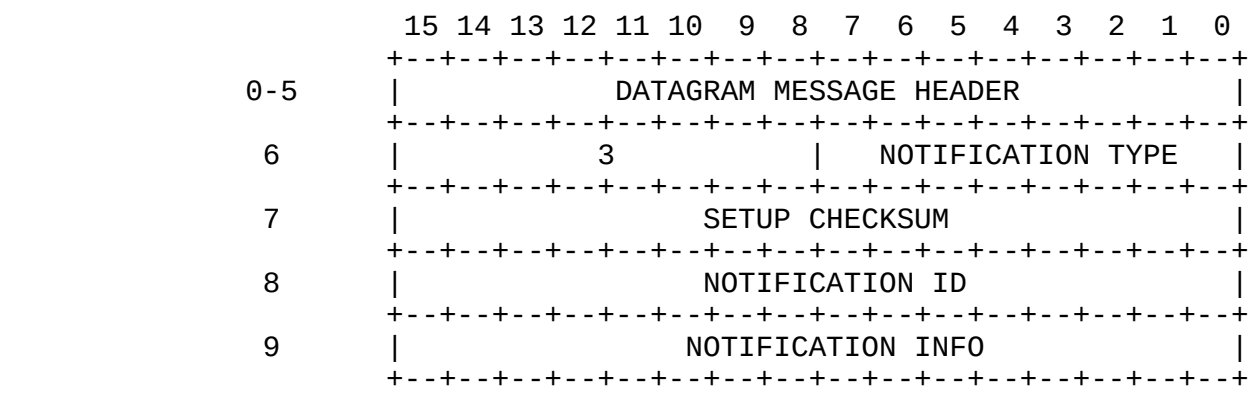

Figure 7 . NOTIFICATION MESSAGE

0-5 Datagram Message Header (see Section 3).

6[8-15] Setup Type = 3 (Notification).

6[0-7] Notification Type.

0 = Stream suspended
1 = Stream resumed
2 = Stream deleted
3 = Group deleted by host
4 = Group deleted by SIMP
5 = All streams deleted
6 = All groups deleted

7[0-15] Setup Checksum. Covers words 6-9.

8[0-15] Notification ID.

9[0-15] Notification Information. This field contains the 16-bit group address in the case of a group

notification (types 3 and 4) and the 10-bit host stream ID (right justified) in the case of a stream notification (types 0-2). This field is zero for Notification Types 5 and 6, which pertain to ALL streams and groups, respectively.

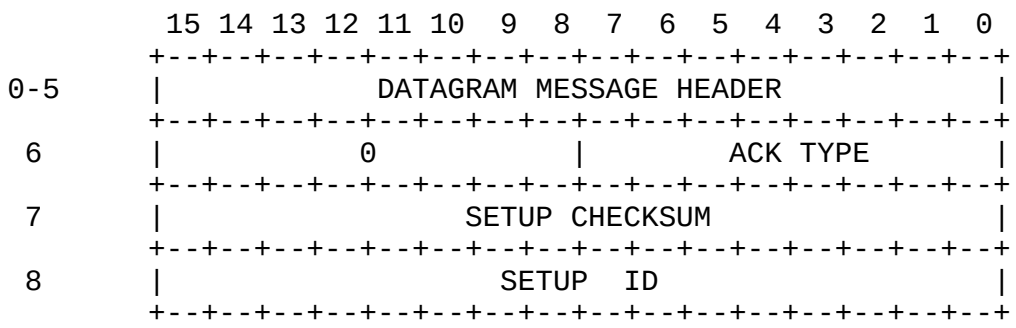

Figure 8 . SETUP ACKNOWLEDGMENT

0-5 Datagram Message Header.

6[8-15] Setup Type = 0 (Acknowledgment).

6[0-7] Acknowledgment Type.

0 = Reply acknowledgment
1 = Notification acknowledgment

7[0-15] Setup Checksum. Covers words 6-8.

8[0-15] Setup ID. This is either a Request ID or a Notification ID.

6.1 Stream Setup Messages

Hosts use streams to support high duty cycle applications and applications requiring a one satellite hop network transmission delay. Host streams must be set up before stream data messages can flow. The stream setup messages defined by HAP are Create Stream Request, Create Stream Reply, Delete Stream Request, Delete Stream Reply, Change Stream Parameters Request, and Change Stream Parameters Reply. The use of these messages is illustrated in the scenario of exchanges between a host and its local SIMP shown in Figure 9 where the host establishes a stream, sends some data, modifies the stream characteristics, sends some more data, and finally closes down the stream.

It is worthwhile noting that the setup exchanges in Figure 9 are completely between the host originating the stream and its local SIMP. Other SIMPs and hosts are essentially unaware of the existence of the stream. Stream messages received by a destination host are, therefore, processed identically to datagram messages. (All SIMPs must, of course, be aware of the channel allocation associated with a host stream in order to perform satellite channel scheduling.) Not illustrated, but implicit in this scenario, are the optional A/R indications associated with each of the stream setup messages.

Figure 9 . STREAM EXAMPLE

Host streams have six characteristic properties which are selected at stream setup time. These properties, which apply to every message transmitted in the stream, are: (1) slot size, (2) interval, (3) reliability, (4) reliability length, (5) priority, and (6) maximum messages per slot. To establish a stream, the host sends the Create Stream Request message illustrated in Figure 10 to the SIMP. After the satellite network has processed the Create Stream Request, the SIMP will respond to the host with a Create Stream Reply message formatted as shown in Figure 11. Assuming that the reply code in the Create Stream Reply is zero indicating 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 six characteristic properties should be modified. All of the properties except the stream interval can be modified using the Change Stream Parameters Request message. The format of this command is illustrated in Figure 12. After the network has processed the Change Stream Parameters Request, the SIMP will respond by sending a Change Stream Parameters Reply to the host with the format shown in Figure 13. A host requesting a reduced channel allocation should decrease its sending rate immediately without waiting for receipt of the Change Stream Parameters 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 of 4 indicating that the requested change has taken effect.

When the host which created the host stream determines that the stream is no longer needed and the associated satellite channel allocation can be freed up, the host sends its local SIMP a Delete Stream Request message formatted as indicated in Figure 14. After the network has processed the Delete Stream Request, the SIMP will respond by sending a Delete Stream Reply to the host with the format shown in Figure 15.

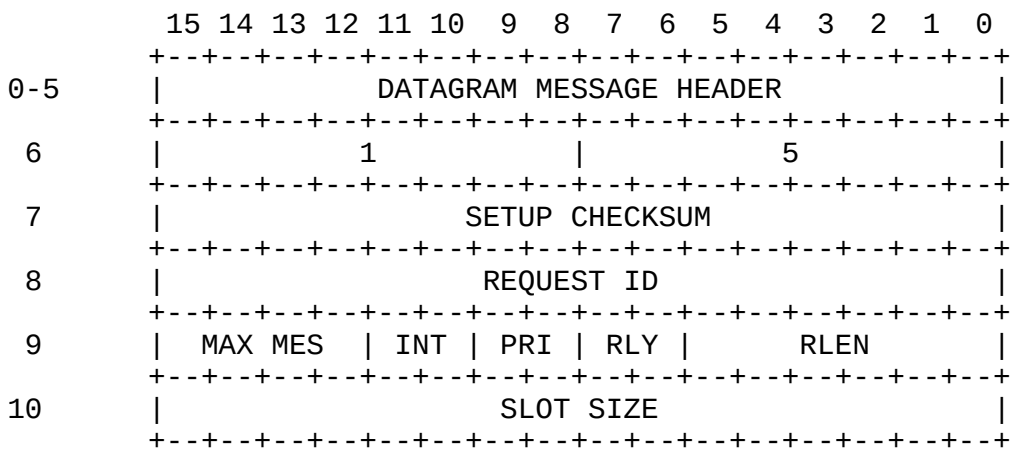

Figure 10 . CREATE STREAM REQUEST

0-5 Datagram Message Header.

6[8-15] Setup Type = 1 (Request).

6[0-7] Request Type = 5 (Create Stream).

7[0-15] Setup Checksum. Covers words 6-10.

8[0-15] Request ID.

9[12-15] Maximum Messages Per Slot. This field specifies the the maximum number of stream messages that will ever be delivered to the SIMP by the host for transmission in one stream slot.

9[10-11] Interval. This field specifies the interval, in number of 21.2 ms frames, between stream slots.

0 = 1 frame
1 = 2 frames
2 = 4 frames
3 = 8 frames

As an example, an interval of 4 frames corresponds to an allocation of Slot Size words every 85 ms.

9[8-9] Priority. This field specifies the priority at which all messages in the host stream should be handled.

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

9[6-7] Reliability. This field specifies the basic bit-error rate requirement for the data portion of all messages in the host stream.

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

9[0-5] 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.

10[0-15] Slot Size. This field specifies the slot size in 16-bit words of stream message text. Stream message header words are excluded from this count. The host can partition this allocation on a slot-by-slot basis among a variable number of messages as long as the maximum number of messages per slot does not exceed MAX MES.

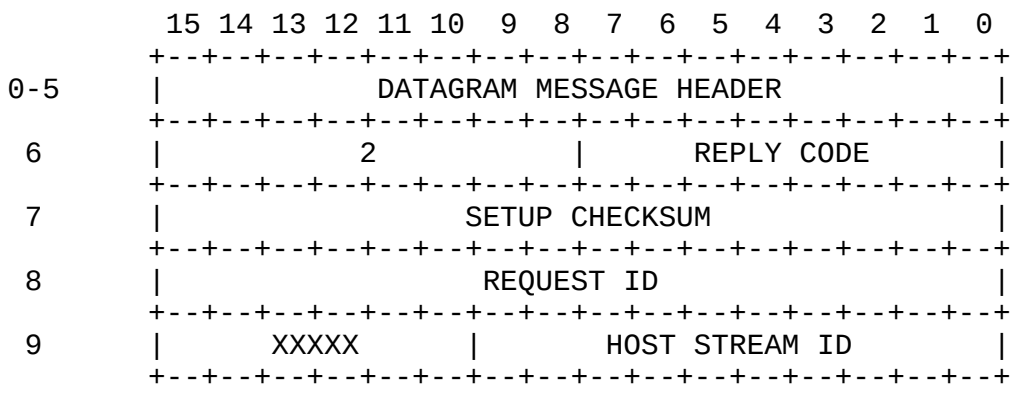

Figure 11 . CREATE STREAM REPLY

0-5 Datagram Message Header.

6[8-15] Setup Type = 2 (Reply).

6[0-7] Reply Code.

0 = Stream created
8 = Network trouble
12 = Stream priority not being accepted
17 = Insufficient network resources
18 = Requested bandwidth too large
21 = Maximum messages per slot not consistent with slot size
22 = Reply lost in network
23 = Illegal reliability value

7[0-15] Setup Checksum. Covers words 6-9.

8[0-15] Request ID.

9[10-15] Reserved.

9[0-9] Host Stream ID. This field contains a host stream ID assigned by the network. It must be included in all stream data messages sent by the host to allow the SIMP to associate the message with stored stream characteristics and the reserved satellite channel time.

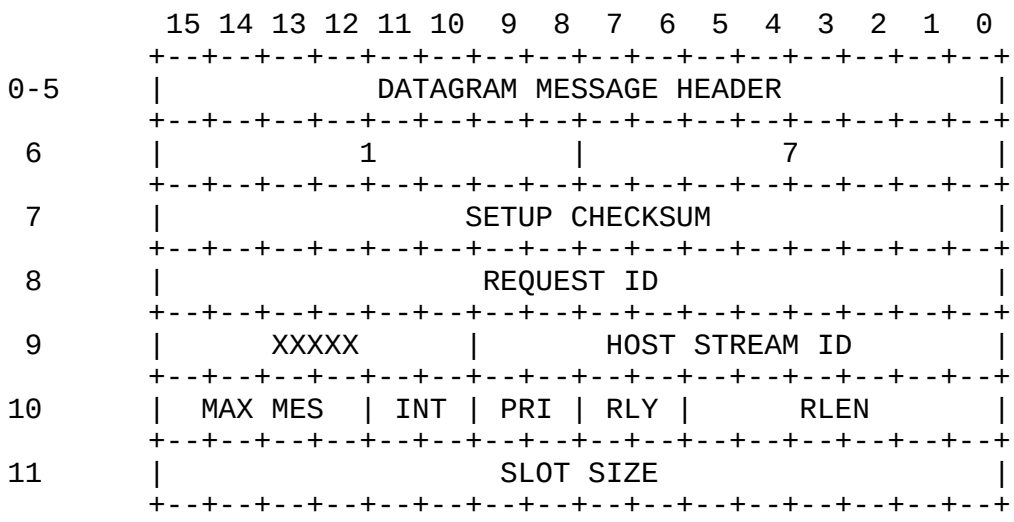

Figure 12 . CHANGE STREAM PARAMETERS REQUEST

0-5 Datagram Message Header.

6[8-15] Setup Type = 1 (Request).

6[0-7] Request Type = 7 (Change Stream Parameters).

7[0-15] Setup Checksum. Covers words 6-11.

8[0-15] Request ID.

9[10-15] Reserved.

9[0-9] Host Stream ID.

10[12-15] New Maximum Messages Per Slot.

10[10-11] Interval. This field must specifiy the same interval as was specified in the Create Stream Request message for this stream.

10[8-9] New Priority.

10[6-7] New Reliability.

10[0-5] New Reliability Length.

11[0-15] New Slot Size.

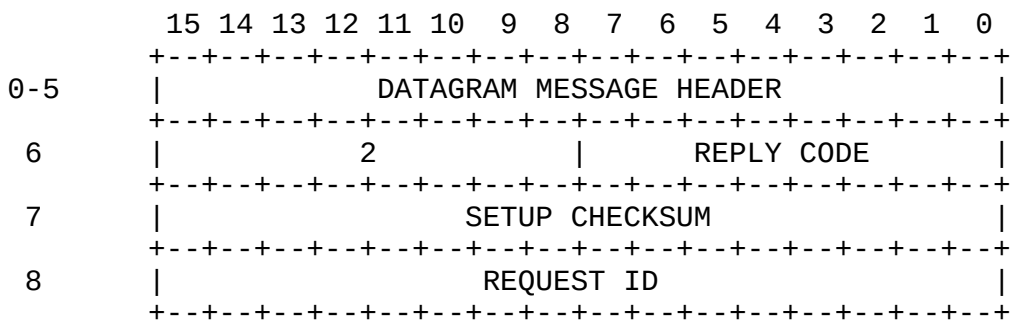

Figure 13 . CHANGE STREAM PARAMETERS REPLY

0-5 Datagram Message Header.

6[8-15] Setup Type = 2 (Reply).

6[0-7] Reply Code.

```
 4 = Stream changed
 8 = Network trouble
10 = Stream ID nonexistent
11 = Not creator of stream
12 = Stream priority not being accepted
15 = Illegal interval
17 = Insufficient network resources
18 = Requested bandwidth too large
21 = Maximum messages per slot not consistent with
     slot size
22 = Reply lost in network
23 = Illegal reliability value
```

7[0-15] Setup Checksum. Covers words 6-8.

8[0-15] Request ID.

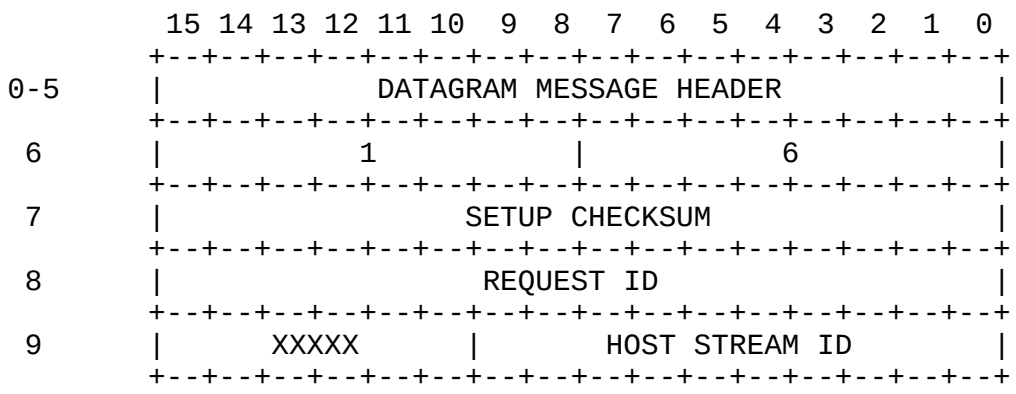

Figure 14 . DELETE STREAM REQUEST

0-5 Datagram Message Header.

6[8-15] Setup Type = 1 (Request).

6[0-7] Request Type = 6 (Delete Stream).

7[0-15] Setup Checksum. Covers words 6-9.

8[0-15] Request ID.

9[10-15] Reserved.

9[0-9] Host Stream ID.

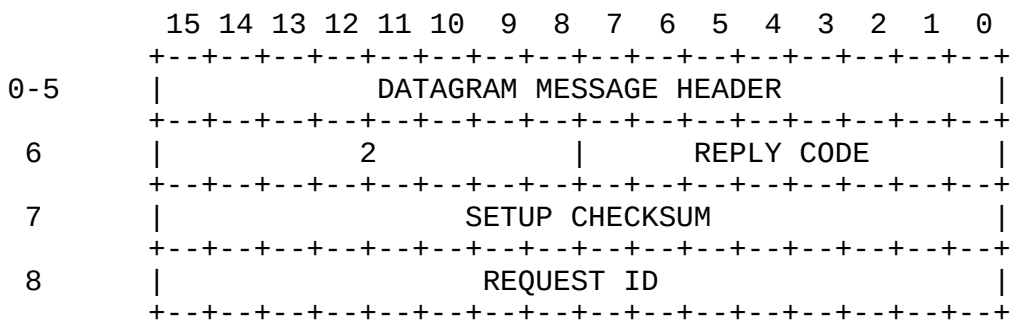

Figure 15 . DELETE STREAM REPLY

0-5 Datagram Message Header.

6[8-15] Setup Type = 2 (Reply).

6[0-7] Reply Code.

```
 1 = Stream deleted
 8 = Network trouble
10 = Stream ID nonexistent
11 = Not creator of stream
17 = Insufficient network resources
22 = Reply lost in network
```

7[0-15] Setup Checksum. Covers words 6-8.

8[0-15] Request ID.

6.2 Group Setup Messages

Group addressing allows hosts to take advantage of the broadcast capability of the satellite network and is primarily provided to support the multi-destination delivery required for conferencing applications. Group addresses are dynamically created and deleted via setup messages exchanged between hosts and the network. Membership in a group may consist of an arbitrary subset of all the permanent network hosts. A datagram message or stream message addressed to a group is always sent over the satellite channel and delivered to all hosts that are members of that group. The group setup messages are Create Group Request, Create Group Reply, Delete Group Request, Delete Group Reply, Join Group Request, Join Group Reply, Leave Group Request, and Leave Group Reply.

The use of group setup messages is shown in Figure 16. The figure illustrates a scenario of exchanges between two hosts and their local SIMPs. In the scenario one host, Host A, creates a group which is joined by a second host, Host B. After the two hosts have exchanged some data mesages addressed to the group, Host B decides to leave the group and Host A 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 host returning a 48-bit key along with a 16-bit group address to the host creating the group. The creating host must pass the key along with the group address to the other hosts which it wants as group members. 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.

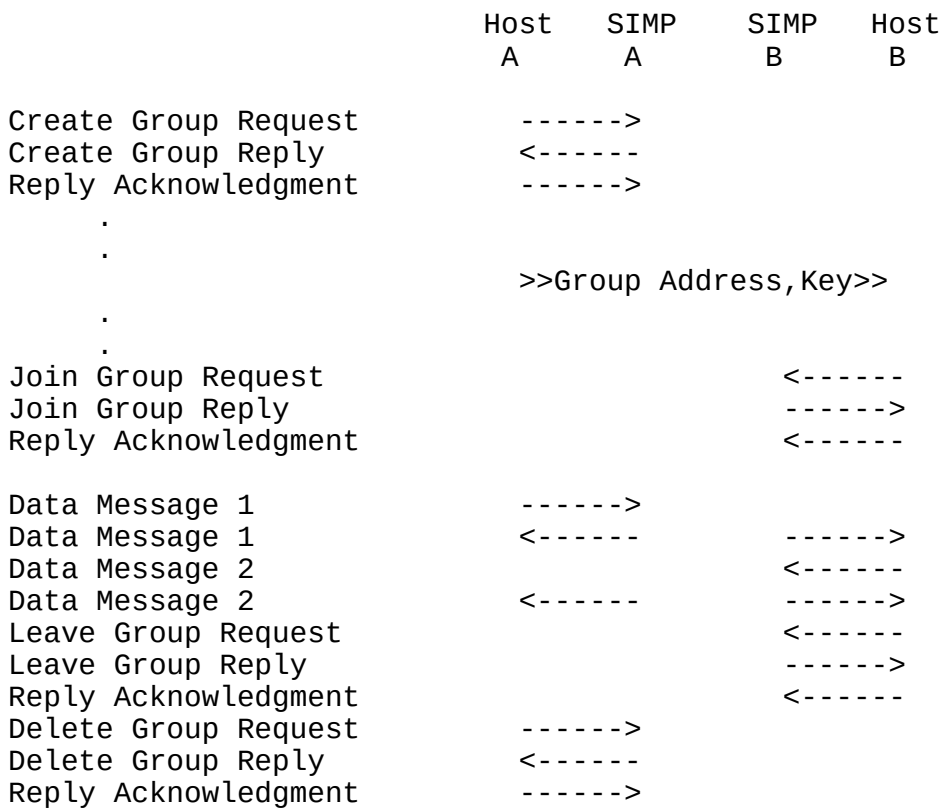

Figure 16 . GROUP EXAMPLE

Any host no longer wishing to participate in a group may choose to drop out. This can be accomplished by either a Leave or a Delete. Both Leave and Delete operations are authenticated using the 48-bit key. Leave is a local operation between a host and its SIMP which removes the requesting host from the group membership list but does not alter the global existence of the

group. A Delete, on the other hand, expunges all knowledge of the group from every SIMP in the network. HAP will permit any member of a group 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 groups.

The Create Group Request message sent to the service host to establish a group address is illustrated in Figure 17. After the network has processed the Create Group Request, the service host will respond by sending a Create Group Reply to the host as illustrated in Figure 18.

A host may become a member of a group once it knows the address and key associated with the group by sending the service host the Join Group Request message shown in Figure 19. The service host will respond to the Join Group Request with a Join Group Reply formatted as indicated in Figure 20. The host which creates a group automatically becomes a member of that group without any need for an explicit Join Group Request.

At any time after becoming a member of a group, a host may choose to drop out of the group. To effect this the host sends the service host a Leave Group Request formatted as shown in Figure 21. The service host will respond to the Leave Group Request with a Leave Group Reply formatted as shown in Figure 22.

Any member of a group can request that the service host delete an existing group via a Delete Group Request. The format of the Delete Group Request setup message is illustrated in Figure 23. After the network has processed the Delete Group Request, the service host will respond to the host with a Delete Group Reply formatted as illustrated in Figure 24.

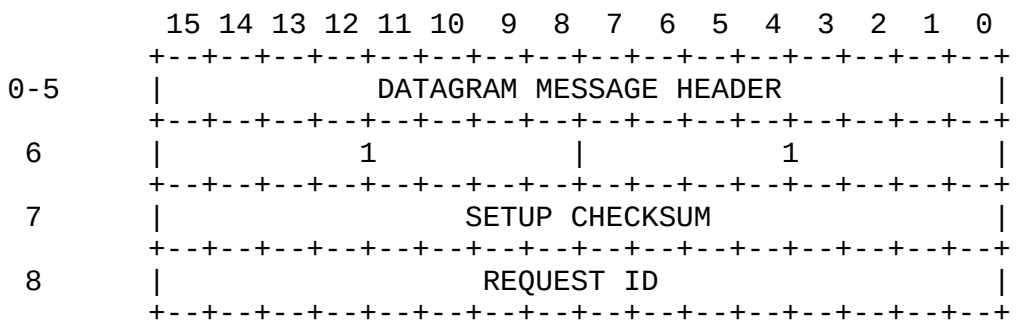

Figure 17 . CREATE GROUP REQUEST

0-5 Datagram Message Header.

6[8-15] Setup Type = 1 (Request).

6[0-7] Request Type = 1 (Create Group).

7[0-15] Setup Checksum. Covers words 6-8.

8[0-15] Request ID.

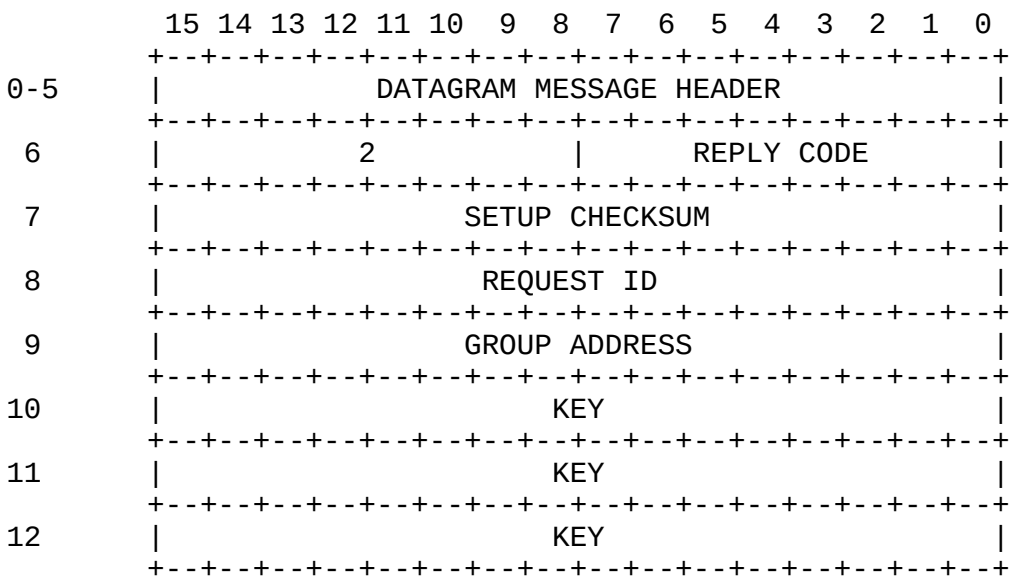

Figure 18 . CREATE GROUP REPLY

0-5 Datagram Message Header.

6[8-15] Setup Type = 2 (Reply).

6[0-7] Reply Code.

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

7[0-15] Setup Checksum. Covers words 6-12.

8[0-15] Request ID.

9[0-15] Group Address. This field contains a 16-bit logical address assigned by the network which may be used by the host as a group address.

10-12 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, Leave, and Delete requests which reference the group address.

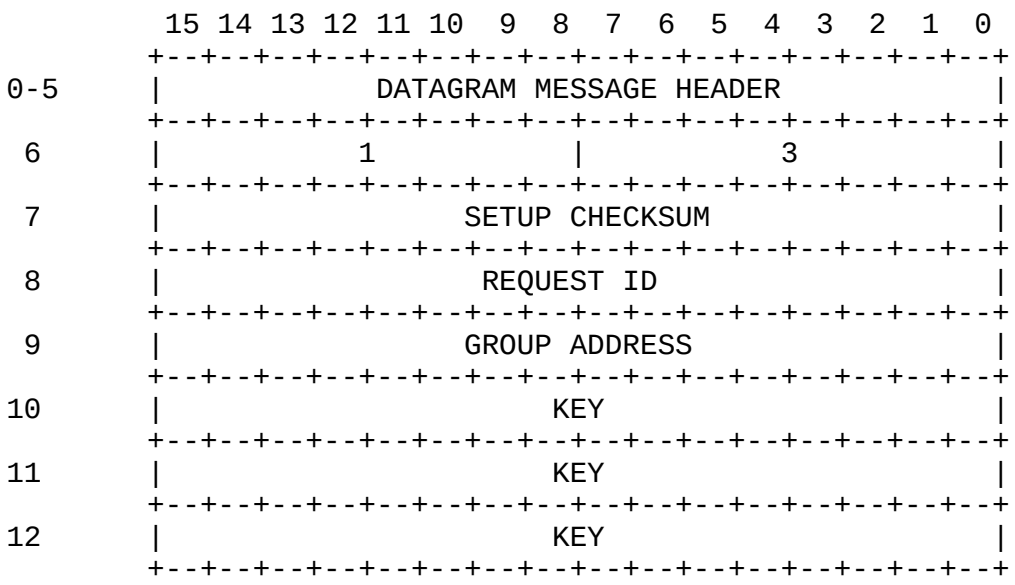

Figure 19 . JOIN GROUP REQUEST

0-5 Datagram Message Header.

6[8-15] Setup Type = 1 (Request).

6[0-7] Request Type = 3 (Join Group).

7[0-15] Setup Checksum. Covers words 6-12.

8[0-15] Request ID.

9[0-15] Group Address. This is the logical address of the group that the host wishes to join.

10-12 Key. This is the key associated with the group

address.

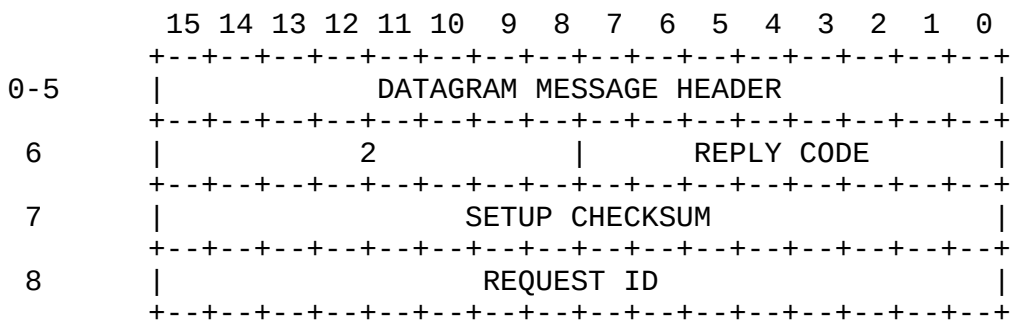

Figure 20 . JOIN GROUP REPLY

0-5 Datagram Message Header.

6[8-15] Setup Type = 2 (Reply).

6[0-7] Reply Code.

2 = Group joined
9 = Bad key
10 = Group address nonexistent
17 = Insufficient network resources

7[0-15] Setup Checksum. Covers words 6-8.

8[0-15] Request ID.

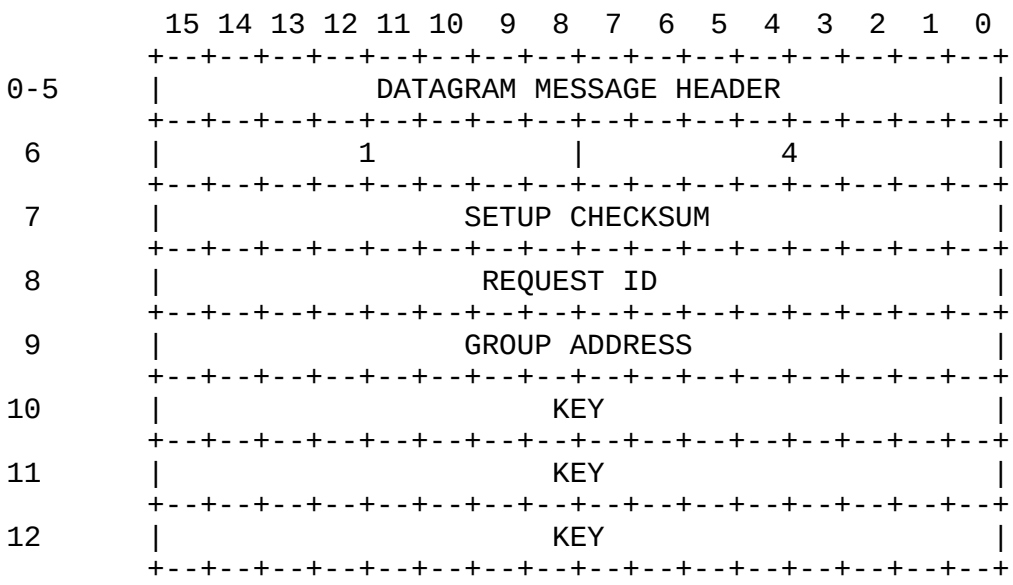

Figure 21 . LEAVE GROUP REQUEST

0-5 Datagram Message Header.

6[8-15] Setup Type = 1 (Request).

6[0-7] Request Type = 4 (Leave Group).

7[0-15] Setup Checksum. Covers words 6-12.

8[0-15] Request ID.

9[0-15] Group Address. This is the logical address of the group that the host wishes to leave.

10-12 Key. This is the key associated with the group

address.

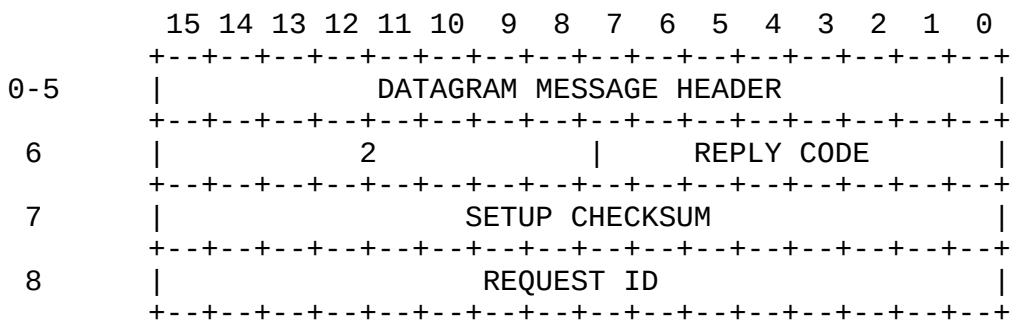

Figure 22 . LEAVE GROUP REPLY

0-5 Datagram Message Header.

6[8-15] Setup Type = 2 (Reply).

6[0-7] Reply Code.

3 = Group left
9 = Bad key
10 = Group address nonexistent
11 = Not member of group
17 = Insufficient network resources

7[0-15] Setup Checksum. Covers words 6-8.

8[0-15] Request ID.

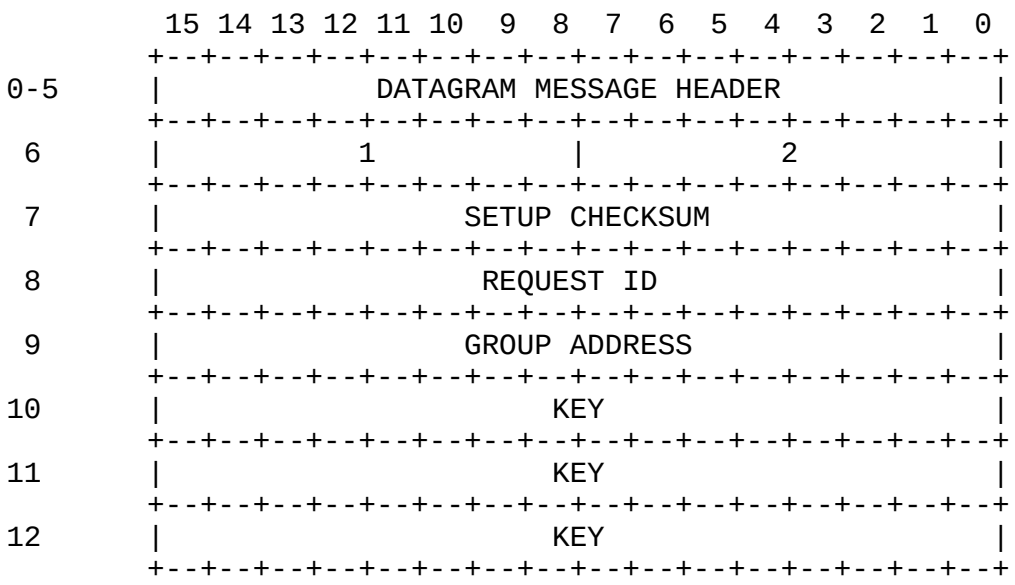

Figure 23 . DELETE GROUP REQUEST

0-5 Datagram Message Header.

6[8-15] Setup Type = 1 (Request).

6[0-7] Request Type = 2 (Delete Group).

7[0-15] Setup Checksum. Covers words 6-12.

8[0-15] Request ID.

9[0-15] Group Address.

10-12 Key.

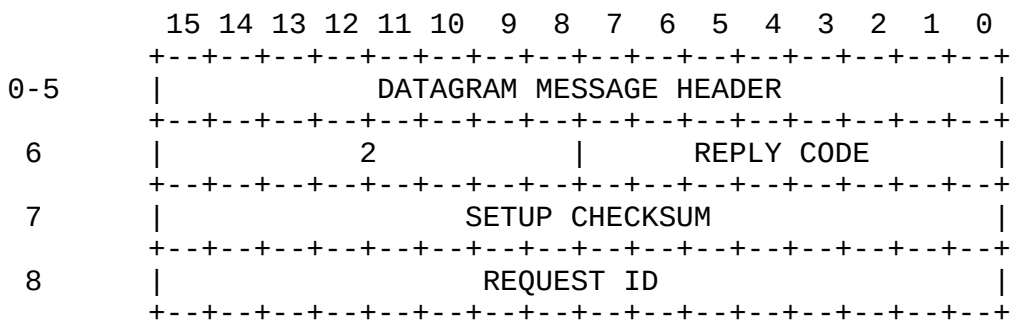

Figure 24 . DELETE GROUP REPLY

0-5 Datagram Message Header.

6[8-15] Setup Type = 2 (Reply).

6[0-7] Reply Code.

1 = Group deleted
8 = Network trouble
9 = Bad key
10 = Group address nonexistent
11 = Not member of group
17 = Insufficient network resources
22 = Reply lost in network

7[0-15] Setup Checksum. Covers words 6-8.

8[0-15] Request ID.

7 Link Monitoring

While the access link is operating, statistics on traffic load and error rate are maintained by the host and SIMP. The host and SIMP must exchange status messages once a second. Periodic exchange of status messages permits both ends of the link to monitor flows in both directions. Status messages are required to support monitoring by the Network Operations Center (NOC).

The link restart procedure (see Section 8) initializes all internal SIMP 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 SIMP 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. The format of the Status control message is illustrated in Figure 25.

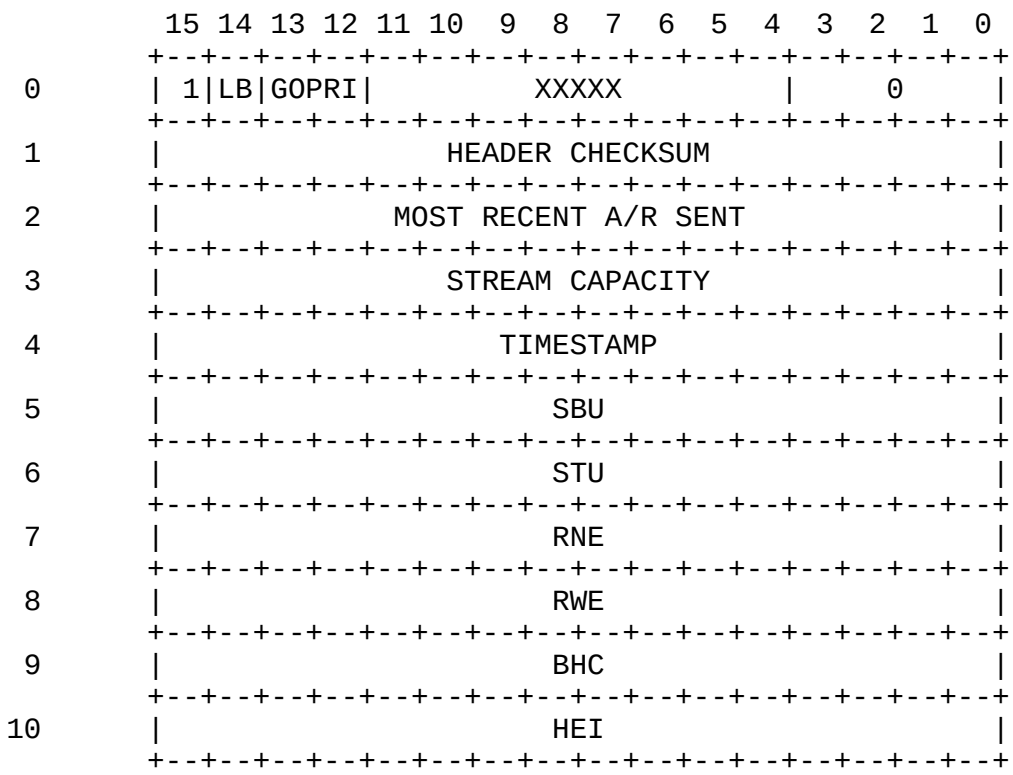

Figure 25 . STATUS MESSAGE

0[15] Message Class = 1 (Control Message).

0[14] Loopback Bit.

0[12-13] Go-Priority.

0[4-11] Reserved.

0[0-3] Control Message Type = 0 (Status).

1[0-15] Header Checksum. Covers words 0-10.

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 SIMP, this field indicates how much stream capacity is unused, in units of data bits per frame. 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 host streams were expanded at low reliability. This field is undefined in messages sent from the host to the SIMP.

4[0-15] Timestamp. This field indicates the time that the status message was generated. When sent by a SIMP, 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 SIMP. The state diagram in Figure 26 shows the sequence of events during initialization. Both SIMP and host must implement this state diagram if deadlocks and oscillations are to be avoided. This particular initialization sequence requires both sides to send and receive the Restart Complete message. Because this message is a reply (to a Restart Request or Restart Complete), its receipt guarantees that the physical link is operating in both directions. Five states are identified in the state diagram:

OFF — Entered upon recognition of a requirement to restart. The device 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 and local counters have been zeroed but no restart control messages have yet been sent or received.

RR-SNT — A request to reinitialize (RR) has been sent to the other end but no restart control messages have yet been received.

RC-SNT — A reply (RC) has been sent to the other end in response to a received reinitialization request

(RR). The device is waiting for a reply (RC).

ON Reply (RC) messages have been both sent and received. Data and control messages can now be exchanged between the SIMP 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.

The Restart Request control message illustrated in Figure 27 is sent by either a host or a SIMP when it wishes to restart a link. The Restart Request causes all the monitoring statistics to be reset to zero and stops all traffic on the link in both directions. The Restart Complete message illustrated in Figure 28 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.

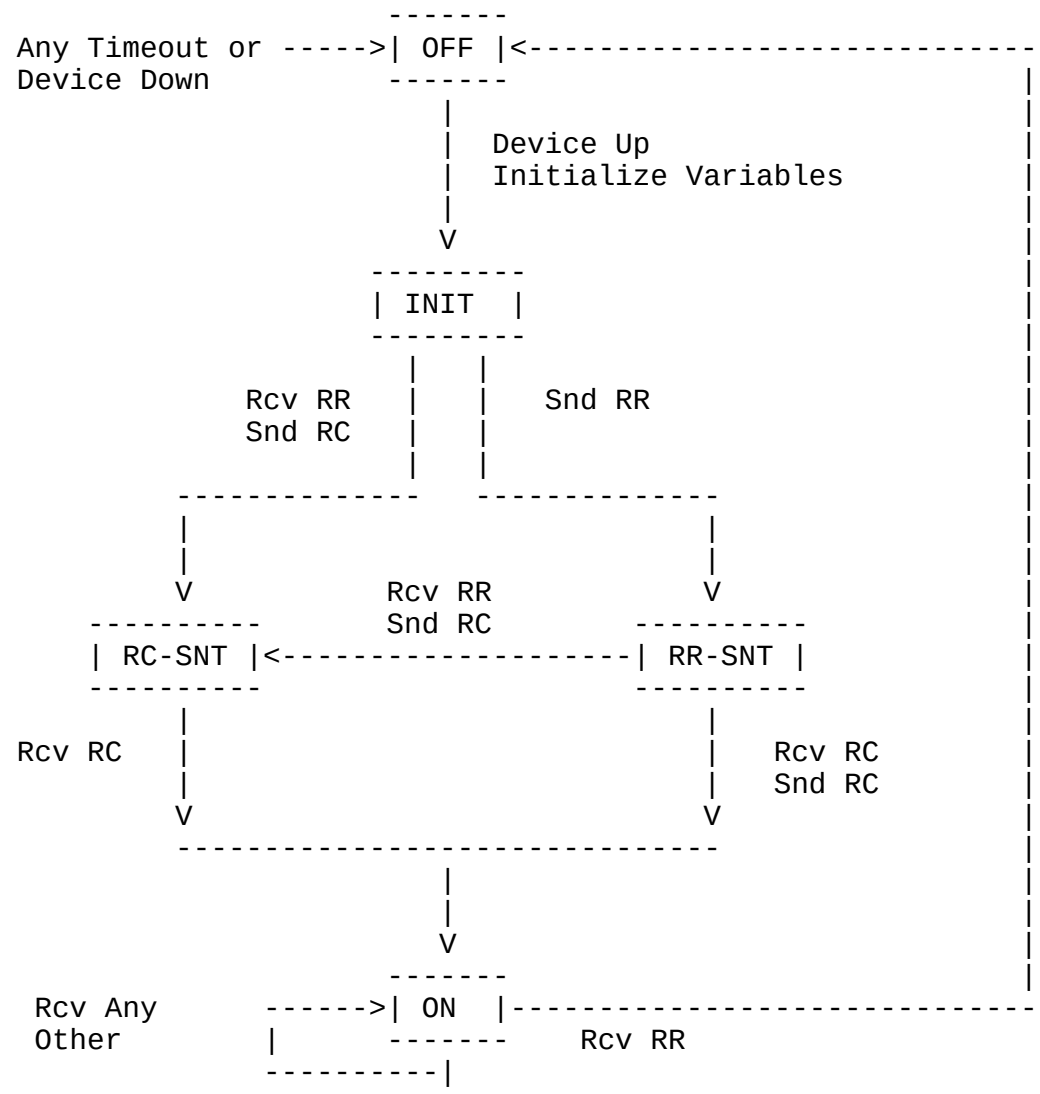

Figure 26 . HAP LINK RESTART STATE DIAGRAM

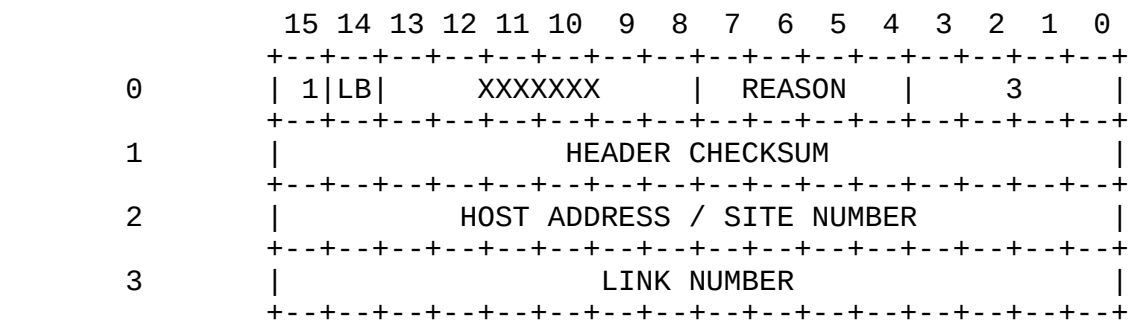

Figure 27 . RESTART REQUEST

0[15] Message Type = 1 (Control Message).

0[14] Loopback Bit.

0[8-13] Reserved.

0[4-7] Reason. This field is used by the SIMP or the host to indicate the reason for the restart as follows:

0 = power up
1 = system restart
2 = link restart
3 = link timeout
4 = loopback timeout

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

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

2[0-15] Host Address / Site Number. The host inserts its satellite network address in this field. The SIMP validates that the host address is correct for the port

being used. When sent by the SIMP, this field will contain the SIMP site number.

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).

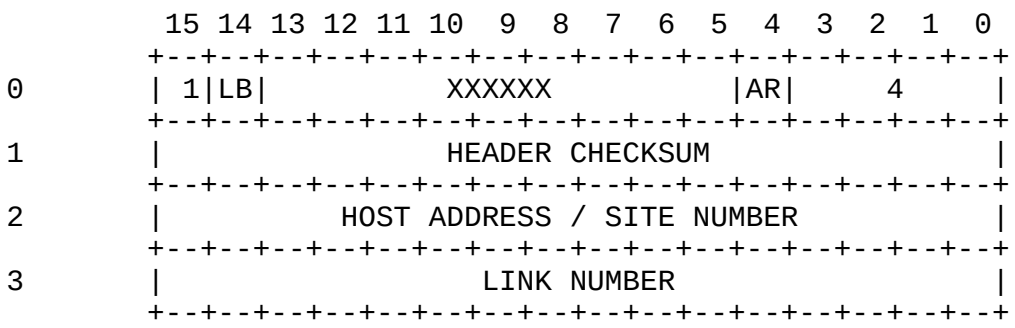

Figure 28 . RESTART COMPLETE

0[15] Message Type = 1 (Control Message).

0[14] Loopback Bit.

0[5-13] Reserved.

0[4] 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[0-3] Control Message Type = 4 (Restart Complete).

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

2[0-15] Host Address / Site Number.

3[0-15] Link Number.

9 Loopback Control

The Host Access Protocol provides a Loopback Request control message which can be used by a SIMP 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 (SIMP 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". The format of a Loopback Request control message is illustrated in Figure 29.

When a transmitter is remotely looped, all of its HAP messages will be returned, unmodified, over the access link by the receiver. The receiver will not send any of its own messages to the transmitter while it is implementing the loop. SIMP-generated messages are distinguished from host-generated messages by means of the Loopback Bit 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 SIMP-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 SIMP-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 SIMP 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 Bit.

A specific sequence must be followed for setting up a remote loopback condition. 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 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 SIMP and host request a remote loopback at the same time. If a SIMP 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 SIMP's loopback request.

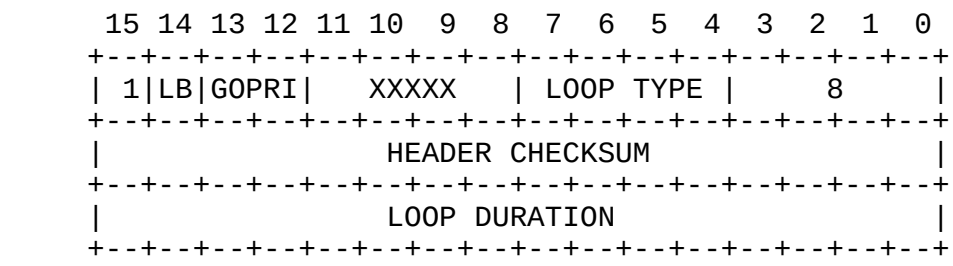

Figure 29 . LOOPBACK REQUEST

0[15] Message Type = 1 (Control Message).

0[14] Loopback Bit.

0[12-13] Go-Priority.

0[8-11] Reserved.

0[4-7] 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[0-3] Control Message Type = 8 (Loopback Request).

1[0-15] Header Checksum. Covers words 0-2.

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 SIMP or a host voluntarily disables a SIMP-host link, it should send at least one Link Going Down control message over that link. The format of such a message is illustrated in Figure 30. HAP does not define the action(s) that should be taken by a SIMP 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 SIMP-host link that it is sent over; if a host and a SIMP are connected by multiple links, these links may be selectively disabled.

A No Operation (NOP) control message may be sent at any time by a SIMP or a host. The format of such a message is illustrated in Figure 31. 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 SIMP-host link hardware.

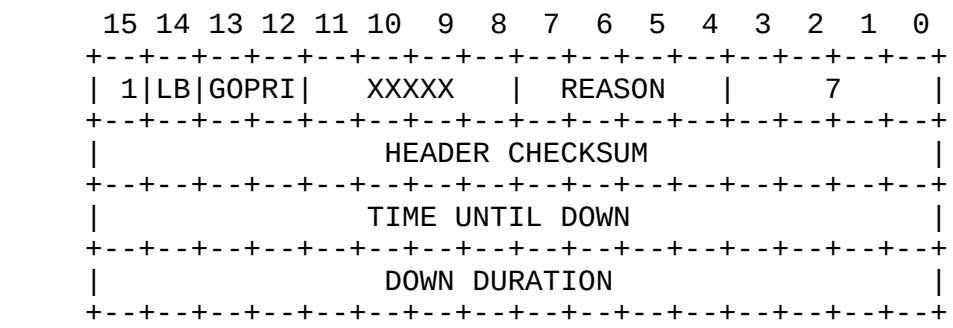

Figure 30 . LINK GOING DOWN

0[15] Message Type = 1 (Control Message).

0[14] Loopback Bit.

0[12-13] Go-Priority.

0[8-11] Reserved.

0[4-7] Reason. This field is used by the SIMP or the host to indicate the reason for disabling this SIMP-host link as follows:

0 = NOT going down: Cancel previous Link Going Down message
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 SIMP or host is disabling the link in 10 seconds
11-15 = Undefined

0[0-3] Control Message Type = 7 (Link Going Down).

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

2[0-15] Time Until Down. This field specifies the amount of time remaining until the SIMP 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 SIMP-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.

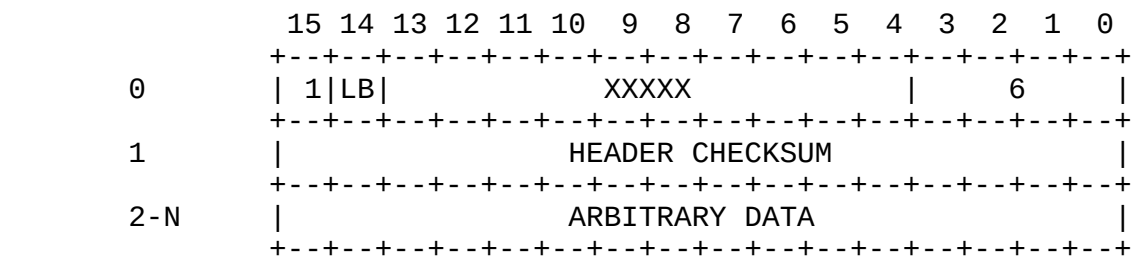

Figure 31 . NO OPERATION (NOP)

0[15] Message Type = 1 (Control Message).

0[14] Loopback Bit.

0[4-13] Reserved.

0[0-3] Control Message Type = 6 (NOP).

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

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