3. SYSTEM OPERATION

3.1 Messages and Message-ids

Hosts communicate with each other via *regular messages*. A regular message may vary in length from 96 up to 8159 bits, the first 96 of which are control bits called the *leader*. The leader is also used for sending control messages between the Host and its IMP, in which case only the first 80 bits are used. The remainder of the message is the *data*, or the *text*.

For each regular message, the Host specifies a *destination*, consisting of IMP, Host, and *handling type*. These three parameters uniquely specify a *connection* between source and destination Hosts. The handling type gives the connection specific characteristics, such as priority or non-priority transmission (see below). Additional leader space has been reserved for a fourth parameter, to be used in future inter-network addressing. For each connection, messages are delivered to the destination in the same order that they were transmitted by the source.

For each regular message, the Host also specifies a 12-bit identifier, the *message-id*.* The message-id, together with the destination of the message, is used as the “name” of the message. The IMP will use this name to inform the Host of the disposition of the message. Therefore, if the Host refrains from re-using a particular message-id value (to a given destination) until the IMP has responded about that message-id, messages will remain uniquely identified and the Host can retransmit them in the event of a failure within the network.

After receiving a regular message from a Host connected to it, an IMP breaks the message into several packets (currently the maximum data bits/packet is 1008) and passes these through the network in the direction of the destination. Eventually, when all packets arrive at the destination, they are reassembled to form the original message and passed to the destination Host. The destination IMP returns a positive acknowledgment for receipt of the message to the source IMP, which in turn passes this acknowledgment to the source Host. This acknowledgment is called a *Ready for Next Message (RFNM)* and identifies the message being acknowledged by name. In some relatively rare cases, however, the message may be lost in the network due to an IMP failure; in such cases an *Incomplete Transmission* message will be returned to the source Host instead of a RFNM. Again, in this case, the message which was incompletely transmitted is identified by name.

* Until mid-1973 the first eight bits of the message-id field were called the “link”.

If a response from the destination IMP (either RFNM or *Incomplete Transmission*) is itself lost in the network, this condition will be detected by the source IMP, which will automatically inquire of the destination IMP whether the original message was correctly transmitted or not, and repeat the inquiry until a response is received from the destination IMP. This inquiry mechanism is timeout-driven, and each timeout period may be as little as 30 or as much as 45 seconds in length.

When a message arrives at its destination, the leader is modified to indicate the source Host, but the message-id field is passed through unchanged. Thus, in addition to providing message identification between a Host and its local IMP, the message-id can provide a means for Hosts to identify messages between themselves. For example, the message-id can be used for multiplexing several independent data streams, or for keeping track of the portions of a single data stream being sent "in parallel" through the network.

If the *priority* bit of the handling type is set, the message will be expedited through the network by being placed at the front of the various transmission queues it will encounter along the way. This can be useful for transactions requiring minimal delay (e.g., remote echoing or the exchange of control information) but should be used judiciously, since the more it is used the less effect each further use will have.

In order to prevent various types of deadlocks within the network, a source IMP must guarantee that the destination IMP will have enough storage to accept the message it is about to send. This is done by preceding each message with a short "request for buffer space" message. When the destination has enough buffer space to receive another message, it returns an "allocation" to the source IMP, which can then send the message it has been holding.

There are several situations in which an IMP may temporarily block* the transmission of a message from the source Host to the source IMP. In general, any such blockage will last for only a few milliseconds, but in some cases the blockage may be indefinite. In at least one such case the IMP will be unable to accept the remainder of a message from its Host until it frees buffer space by delivering some message to the Host (it is for this reason that half-duplex Host-IMP interfaces are prohibited). In all such cases, in order to prevent permanently hanging up transmission between the Host and the IMP, the source IMP will discard the message after a wait of about fifteen seconds and return a type 9 (sub-type 4) message (see Section 3. 4) to the Host, thus limiting the length of time that the interface will be blocked. Similarly, once a Host has

* By failing to provide *Ready-for-Next-Bit*, see Section 4.1.

begun to send the IMP a message, it must be prepared to deliver the entirety of that message to the IMP promptly. In particular, the IMP will discard any message that is not completely received from its Host in fifteen seconds and return a type 9 (sub-type 2) message to the Host (see Section 3.4).

One situation under which interface blocking will occur is when the source IMP must wait to receive an allocation from the destination IMP. Since a Host cannot send other messages into the network while its interface is blocked, it is desirable to expedite the "allocation" mechanism, and this is done in two different ways depending upon message length. For one-packet messages, the message itself is sent as its own request. Thus, if space is available, the message is immediately accepted and no additional delay is incurred. For multi-packet messages, when the destination IMP is about to return a RFNM it reserves storage in *anticipation* of the source Host's next message, and returns the allocation along with the acknowledgment. Thus, when the source IMP eventually sends its Host the RFNM, it is also implicitly informing it of the allocation now being available.[*] If the Host responds promptly with another message on that same connection (message-id is irrelevant), the message can be forwarded immediately, avoiding any set-up delay waiting for an allocation. If this allocation remains unused for about 125 ms, it is returned, unused, to the destination. Note that this mechanism applies only for messages longer than one packet (about 1103 bits, including leader).

The message processing (reassembly of packets into messages, allocation of buffer space, detection of lost messages, etc.) requires the IMP to perform a certain amount of bookkeeping on the flow of messages between each pair of communicating Hosts. In order to keep the amount of required table space within manageable bounds, the following two restrictions are imposed.

1. The maximum number of messages which a Host is permitted to have "in transit" on any connection is eight. In other words, if a Host attempts to transmit nine messages on any connection, the interface will be blocked by the IMP during transmission of the ninth message until a RFNM (or *Incomplete Transmission*) is returned for the first message. However, this rule does not prohibit one Host from having eight messages in transit to Host "A", eight more in transit to Host "B", etc., simultaneously.
2. When a Host wishes to establish a new connection with another Host, both source and destination IMPs must acquire a block of table space from pool of such blocks shared by

[*] In some (rare) cases the destination is unable to reserve storage immediately, and returns a RFNM without the reservation. Currently, the destination waits 1/2 second, attempting to reserve storage, before returning the RFNM without an accompanying reservation.

> all the Hosts local to each IMP. The source IMP must notify the destination of the need for the new connection, and the destination must reply with a confirmation that it has also acquired the table space. This action may result in a small additional delay before Host communication can begin. The pool will be sufficiently large to seldom interfere with a pair of Hosts wishing to communicate. In no case will Hosts be prevented from communicating because of lack of these resources. In the event that the Hosts on an IMP desire to simultaneously communicate with so many other Hosts that the pool would be exhausted, the space in the pool is quickly multiplexed in time among all the desired Host/Host conversations so that none is stopped although all are possibly slowed.

Section 3.7 describes an optional mechanism available to Hosts that wish to keep interface blocking to a minimum.

3.2 Establishing and Breaking Host/IMP Communications

Each IMP and Host interface has its own hardware Ready indicator. The Ready indicator in the standard Host/IMP interface will be on whenever the IMP is powered on and both the IMP program and the IMP hardware are determined to be working properly. The Ready indicator in the special Host interface should be on whenever the Host is powered on, the hardware is working properly, and the Host's Network Control Program (NCP) is running. If the Host temporarily neglects communications with the IMP, the Host's hardware Ready indicator should not go off. An off indication should mean only that something is broken or that communications have been willfully cut off for an extended period (cable removed, power shut off, routine maintenance programs running, batch processing with no network program running etc.).

In addition to the Ready indicator, the standard interface has a flip-flop, called the *Error* flip-flop, which remembers a not-ready indication from the Host or the IMP. This flip-flop is used to detect any momentary off condition on either the Host's ready line or the IMP's ready line. The flip-flop is cleared by the IMP program each time the program enables (i.e., prepares to receive) a new input from the Host and is tested by the program when the input is completed. The input is discarded if the Error flip-flop is turned on.

To establish communication, a Host should simply send its message to the IMP. The operational IMP program will process any message transmitted from the Host. The Host must always send at least three NOP messages[*] to the IMP whenever either the Host or the IMP Ready line is turned on, for the reasons described below.

One reason is that the Host-to-IMP NOP message contains information as to how much leader padding is to be contained in regular Host-to-IMP and IMP-to-Host messages. Also, until old-style leader formats (Appendix A) are no longer used, this NOP informs the IMP of the style of leader the Host is using.

Another reason is that in general, when the Host Ready indicator goes off, the IMP program will be either receiving or waiting (in an input command) to receive a message from the Host. Upon resumption of transmission by the Host, the IMP will unwittingly append the new information to the unfinished input. Upon completion of the message, the IMP program will note that the Error flip-flop is on and thus discard the entire message. To guarantee that a useful new message is not thereby discarded, the first message sent by the Host after its Ready indicator comes on should be a discardable *NOP* message. The special interface should have a similar Error flip-flop, and the Host's Network Control Program should be designed to use this flip-flop in a similar manner.

When the Host Ready indicator comes on, it will generally alternate few times between on and off (due to relay contact bounce—see Section 4.4) before setting solidly on. The Host should delay an appropriate period to permit its ready indicator to stabilize before starting output or preparing for input. Failure to do so may cause incorrect data to be taken from or sent to the IMP.

A Host may go down, thus halting network traffic to itself from other Hosts, in either of two ways: by turning off its ready indicator (hard down), or by failing to accept messages from the IMP (tardy down). In either case, the IMP will mark the Host as dead and see to it that any attempt to communicate with the Host results in a Destination Dead response.

The IMP program tests the Host Ready indicator (not the Error flip-flop) every half-second. If the program ever finds this ready indicator off, the Host will be marked dead (hard down) and the IMP will discard old messages for transmission to the Host and will set up 3 *NOP* messages followed by type a 10 message for transmission to the Host. Both the IMP and the Host must discard any *NOP* messages that are recognized as such. (A *NOP* message that is appended to an unfinished message may not be recognized, but it will be discarded as discussed above.)

[*] See Section 3.3.

The IMP follows the above procedures when the Host Ready indicator is off momentarily or for an extended period. The following steps are taken by the IMP when its own indicator has gone off.

1. The Error flip-flop is turned on. This action will cause the first incoming message from the Host to be discarded.

2. Old messages for transmission to the Host are discarded.

3. The IMP Ready indicator is turned on.

4. Sufficient *NOP* messages are placed on the output queue to the Host to cover the period of relay bounce and insure correct transmission of at least one NOP.

5. A Type 10 message is placed on the output queue to the Host.

The Host should employ a similar procedure whenever its own Ready indicator has gone off, except that old messages for transmission to the IMP need not necessarily be discarded.

In order to not tie up network resources for an inordinate amount of time, Hosts must be prepared to accept messages from the network promptly. In particular, any given message will be discarded if it resides on a queue to the Host for more than thirty seconds. (With the current IMP system, this requires that the Host must read its interface at the rate of about 1,500 bits/second, averaged across about twenty seconds.) If the Host does not meet this constraint, the IMP will:

1. Declare the Host to be “tardy down”.

2. Discard all messages pending on the queues to the Host.

3. Momentarily drop its ready line (thus setting the error flip-flop). This is done because a component failure in the interface may have caused the handshaking procedure (see Section 4.2) to get out of step, which would have the same effect as the Host merely being tardy. “Flapping” the ready line insures that the interfaces are synchronized.

4. Place some NOP’s and a type 10 message on the queue to the Host.

The Host will be declared up the next time that it sends a message to the IMP or accepts a message from the IMP. The Host must send at least three NOP messages to the IMP if it is aware that it has been declared tardy, since the error flip-flop will cause the first Host-to-IMP message to be discarded. (Alternatively, the Host could bring down its own ready line; the IMP would then proceed as though the Host were in a hard down, rather than continuing to treat the Host as though it were in a tardy down.)

If the Host has advance warning that it will be going down, it may use the *Host Going Down* message (see Section 3.3) to inform the IMP of its status (i.e., the reason for and duration of the down). Transmission of this message from the Host to the IMP will not cause the IMP to declare the Host down; the IMP will store the status information for use during the next Host down. When the Host comes up again, the status information stored in the IMP will be discarded.

The set of events described above is summarized in Table 3-1. Suggestions for Host use of the Ready indicators are contained in Appendix B.

EVENT: STATE	READY LINE UP	READY LINE DOWN	SEND OR RECEIVE HOST MESSAGE	RECEIVE *HOST GOING DOWN*	HOST TARDY
UP:		HARD DOWN		SET STATUS	TARDY DOWN
HARD DOWN:	UP, CLEAR STATUS		UP, CLEAR STATUS*	UP, SET STATUS*	
TARDY DOWN:		HARD DOWN	UP, CLEAR STATUS	UP, SET STATUS	

*This can’t, of course, happen without the Ready line being up, but the IMP might detect the input or output before detecting the change in Ready line status.

Table 3-1 Transitions Between Host States

3.3 Host-to-IMP Leader Format

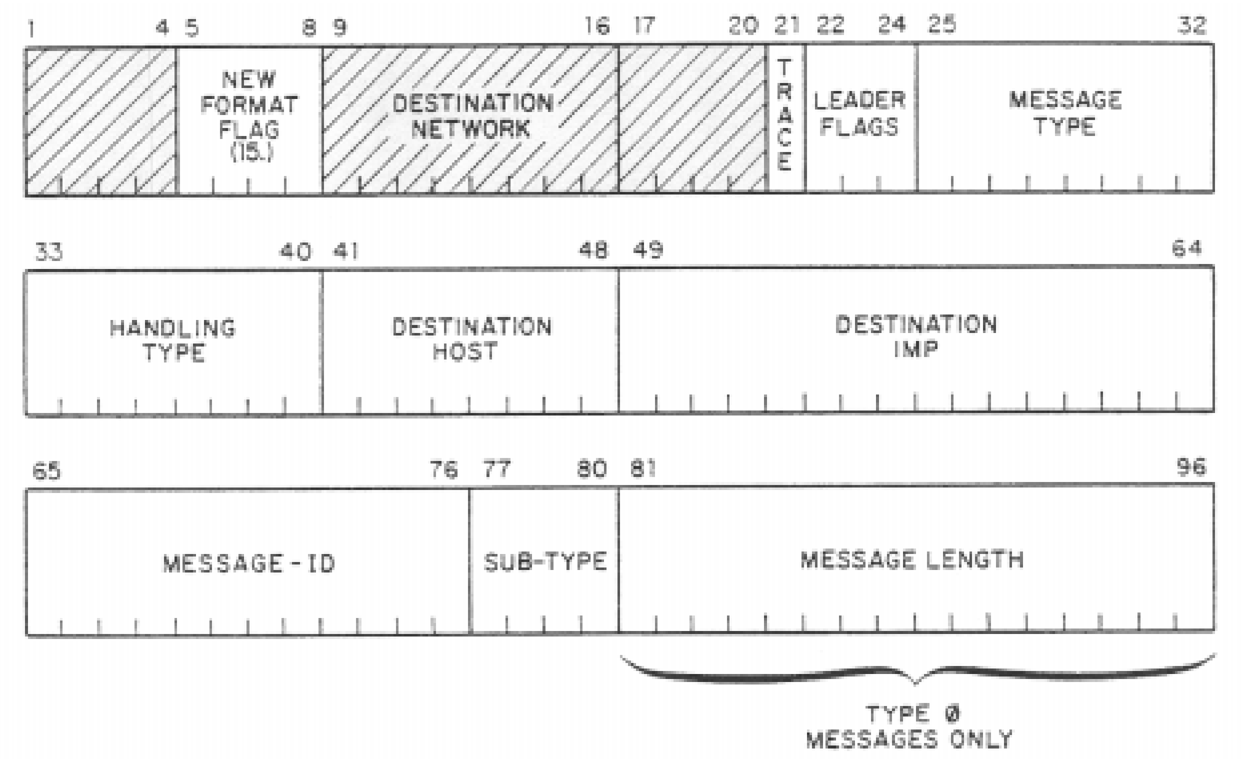

Figure 3-1 Host-to-IMP Leader Format

Bits 1–4 Unassigned—

Must be zero.

Bits 5–8 New Format Flag—

These bits are always set to the value 15. This permits the IMP to distinguish between new-style and old-style (Appendix A) leaders.

Bits 9–16 Destination Network—

For future use, these bits must always be zero.

Bits 17–20 Unassigned—

Must be zero.

Bit 21 Trace—

If equal to one, the message is designated for tracing as it proceeds through the network so that reports of this message's transit through the network may be sent to a trace destination (see Section 5.5).

Bits 22–24 Leader Flags—

Bits 23 and 24 are currently unassigned but are reserved for future network use and must be zero. Bit 22 is available as a destination Host flag, its meaning, if any, being assigned by that Host. The only Host with a preassigned meaning is the IMP Teletype Fake Host. If the bit is one, the message will be printed on the Teletype as a sequence of octal numbers, each representing one 16-bit IMP word. If equal to zero, then the message will be printed as a sequence of ASCII characters.*

Bits 25–32 Message Type—

0. *Regular Message*—All Host-to-Host communication occurs via regular messages. Sub-types (bits 77–80):

    0. *Standard, Non-Refusable*. Interface blocking will occur if any resource needed to send the message is not immediately available.

    1. *Refusable*** Used to minimize the number of times the interface may be blocked. If any resource needed to send the message is not available, the message is discarded, and the Host is notified via a type 11, 12, or 13 Host-to-IMP control message. In the case of a type 12 (*Refused, will notify*) response, the IMP is committed to also sending a type 14 (*Ready*) when the resource does become available.

    2. *Get Ready*** (see Section 3.7). Similar to *Refusable* (above), except only the leader, rather than the full message, is sent in to the IMP. If all necessary resources are immediately available, the Host is notified via a Type 14 message.

    3. *Uncontrolled*—(see Section 3.6). The IMP will perform no message-control functions for this type of message.

    4–15. Unassigned.

* The IMP's internal ASCII character set is listed in Appendix E.

** The non-blocking Host interface (see Section 3.7) is not yet implemented.

1. *Error Without Message Identification*—The Host program detected an error in a previous IMP-to-Host message and had to assume that the leader was garbled. Sub-types:

   0. Host's error flip-flop was set during transmission of the message.

   1. Host received a message less than 80 bits.

   2. Host received a message of an unassigned type (3, 15–255).

   3–15. Unassigned.

2. *Host Going Down*—It is assumed that as the time for the Host to (voluntarily) go down approaches, the Host itself will send warning messages to its network users. Just before going down, the Host should send the Host-Going-Down message to its IMP. The Host should then (if it can) continue to accept messages from the IMP for a period of 5 or 10 seconds, to allow messages already in the network to reach it. The IMP will store the Host-Going-Down message and return it to any source Host along with Destination (Host) Dead messages. The IMP will try to preserve this message over IMP reloads where appropriate. The NCC will be able to manually update the stored copy of this message in response to a phone call from the Host site in the event the Host is going to be down longer than it said or if it did not have time to give warning before going down.

   Bits 65–76 (the message-id field) of the Host-Going-Down message give the time of the Host's coming back up, bit-coded as follows:

   Bits 65–67: the day of the week the Host is coming back up. Monday is day 0 and Sunday is day 6.

   Bits 68–72: the hour of the day, from hour 0 to hour 23, that the Host is coming back up.

   Bits 73–76: the five minute interval, from 0 to 11, in the hour that the Host is coming back up.

   All three of the above are to be specified in Universal Time (i.e., G.M.T.). The Host may indicate that it will be coming back up more than a week away by setting bits 65–76 all to ones. Setting all bits 65–75 to one and bit 76 to zero means it is unknown when the Host is coming back up.

Bits 77–80 (the sub-type field) of the Host-Going-Down message should be used by the Host to specify the reason it is going down. These bits are coded (in octal) as follows:

Value	***Meaning***
0–4	Reserved for IMP use
5	Schedule P.M.
6	Scheduled Hardware Work
7	Scheduled Software Work
10	Emergency Restart
11	Power Outage
12	Software Breakpoint
13	Hardware Failure
14	Not scheduled up
15	Unspecified Reason
16–17	Currently Unused

3. *Unassigned*.

4. *NOP*—The IMP will discard this message, which is intended for use during initialization of IMP/Host communication. Bits 77–80 (the sub-type field) contain the number of 16-bit words of padding (9 max.) that the Host wishes to send and receive on type 0 messages. This padding occurs immediately after the leader (starting at bit 97) and is provided as a convenience for Hosts for which the combined Host/IMP (IMP/Host) and Host/Host leaders would otherwise not be an integral number of memory words. A simple rule for the Host to follow is to send three *NOP* messages whenever the Host or the IMP has been down either voluntarily or involuntarily.

5. *Unassigned*.

6. *Unassigned*.

7. *Unassigned*.

8. *Error with Message Identification*—The Host detected an error in a previous IMP-to-Host message after the leader was correctly received; e.g., the message was too long, or the IMP Error flip-flop was set after transmission of the first packet of a multiple packet

message but before the end of the message. A message of this type will have a leader whose assigned bits are identical to the assigned bits in the leader of the message in error except that the message type bits will be changed to have value 8.

9–255. *Unassigned*.

Bits 33–40 Handling Type—

This field is bit-coded to indicate the transmission characteristics of the connection desired by the Host.

Bit 33: *Priority*—Most messages should have this bit set to zero; messages with this bit set to one will be treated as priority messages (see Section 3.1).

Bits 34–37: Currently unassigned, must be zero.

Bits 38–40: *Maximum Message Size*[*]

The maximum size (in packets) of any message the Host expects to send on the connection (#packets = (#bits in message – 96)/1008). This number is expressed as (maximum # of packets – 1) and ranges from 1 (2 packets max) to 7 (8 packets max). A value of zero indicates the default maximum which is 8 packets. It is to the advantage of the Host to specify this quantity as accurately as possible, since it enables the destination IMP to make the most efficient allocation of reassembly space. On the other hand, messages that must remain in strict sequence must all have the same handling type. Multiple connections between two Hosts, each with a different maximum message size, should be used only when there are large differences in the maxima and strict sequencing is not required. A message whose length exceeds the specified maximum will be discarded and type 9, subtype 1 will be returned to the Host.

Bits 41–48 Destination Host—

Identify the particular Host at an IMP site. Host numbers 252–255 are reserved for use by the IMP's "fake" Hosts (see Section 5).

[*] Until this is implemented by the IMP, the default value of 0 should be used by the Host.

Bits 49–64 Destination IMP—

Identify the IMP site

Bits 65–76 Message-id—

Host-specified identification supplied in all type 0 and 8 messages. Also used in type 2 (Host-Going-Down) message.

Bits 77–80 Sub-type—

Used by message types 0, 2, 4, and 8.

Bits 81–96 Message Length—

This field is used for type 0 messages only and specifies the length (in bits) of the message, exclusive of leader, leader padding and hardware padding. The only use that the IMP makes of this field is the *Get Ready* (Sub-type 2) message where it is used to determine if the message is single or multi-packet. If a zero length is given in a *Get Ready* message, a multi-packet length is assumed.

The following table shows which non-constant fields are used by each valid message type.

	Message Type				
Fields	0	1	2	4	8
Trace	x				
Leader Flags	x				
Message Type	x	x	x	x	x
Handling Type	x			x	
Destination Host	x			x	
Destination IMP	x			x	
Message-id	x		x	x	
Sub-type	x		x	x	x
Message Length	x				

3.4 IMP-to-Host Leader Format

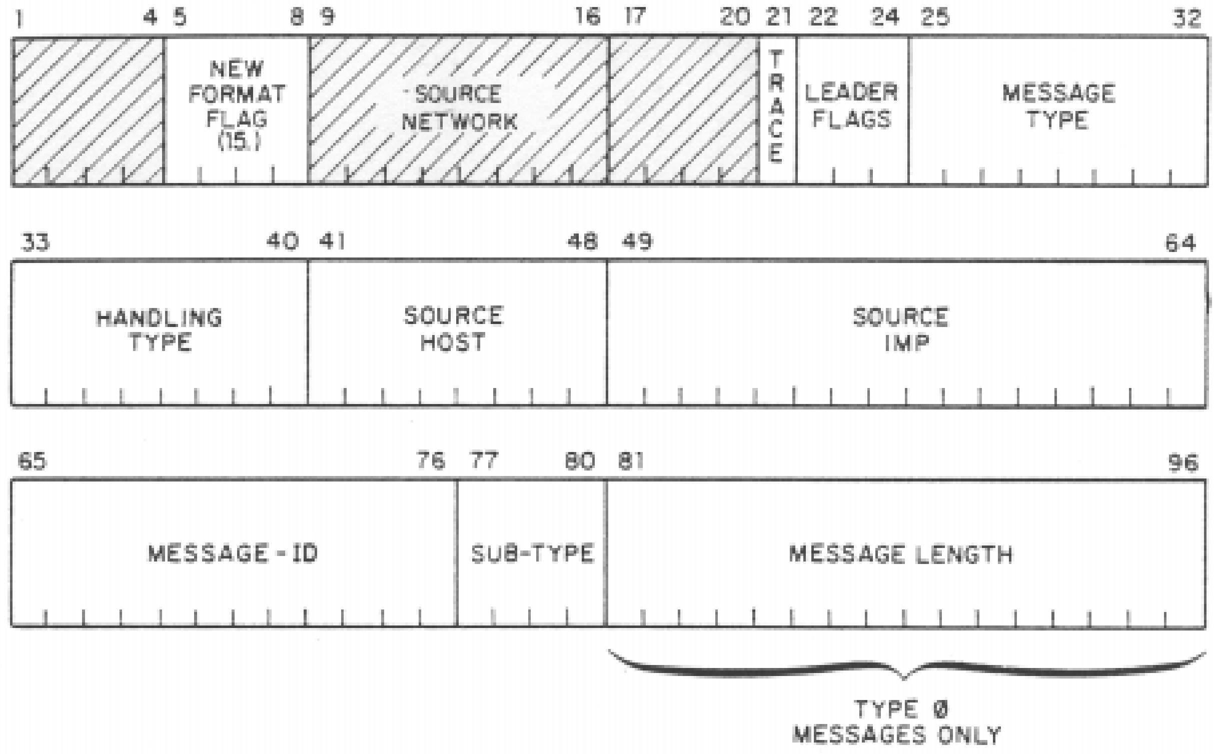

Figure 3-2 Host-to-IMP Leader Format

Bits 1–4 Unassigned—

Set to zero.

Bits 5–8 New Format Flag—

Set to 15.

Bits 9–16 Source Network—

Currently set to zero.

Bits 17–20 Unassigned—

Set to zero.

Bit 21 Trace—

If equal to one, source designated that message be traced (see Section 5.5). Used in type 0 messages only.

Bits 22–24 Leader Flags—

Bits 23 and 24 are currently unassigned and are set to zero. Bit 22 may be assigned a meaning by the destination Host, in which case it is used by the source Host to signal some special meaning, e.g. octal printing for the Teletype Fake Host. Used in type 0 messages only.

Bits 25–32 Message type—

0. *Regular Message*—All Host-to-Host communication occurs via regular messages. The subtype field is the same as sent in the Host-to-IMP message; in particular a sub-type of 3 indicates an *uncontrolled* message (see Section 3.6).

1. *Error in Leader*—the IMP detected an error in a previous Host-to-IMP message and had to assume that the leader was garbled.

   Sub-types:

   0. IMP's Error flip-flop set during the first 96 bits of a message (see Section 3.2).
   1. IMP received a message of less than 80 bits (32 if old format).
   2. IMP received a message of an illegal Type.
   3. IMP received a message of the opposite leader style than it was expecting.

2. *IMP Going Down*—The IMP will transmit this message to its Host before it voluntarily goes down. The Host should forward the information in the message to its users from the network (and to its own users of the network).

   Bits 65–80 of the message are coded as follows:

   Bits 65-66: Why;

   0. "last warning" or "panic restart": The IMP is going down in 30 seconds.
   1. Scheduled hardware PM
   2. Scheduled software reload
   3. Emergency restart

   Bits 67–70: How Soon; in 5 minute increments (zero implies immediately)

Bits 71–80: For How Long; in 5 minute increments (zero implies immediately)

3. Unused.

4. *NOP*—The Host should discard this message. It is used during initialization of IMP/Host communication. The Host and IMP fields will contain the local Host and IMP identification numbers, and the sub-type field will be zero. All other fields are unused.

5. *RFNM*—“Ready for Next Message”. The named regular message was successfully delivered to the destination IMP, and the destination Host accepted it. In addition, if the named message is longer than one packet (about 1103 bits including leader) space may be reserved at the destination IMP for another transmission, but the space reservation will remain valid for only a short time (see Section 3.1). The subtype field will be 0 if the original message was non-refusable, and 1 if it was refusable.

6. *Dead Host Status*—Bits 65–76 (the message-id field) have the same meanings as bits 65–76 in the Host-to-IMP type 2 (Host-Going-Down) message described in Section 3.3. Bits 77–80 (the sub-type field) have the following meanings:

Value	***Meaning***
0	Currently Unused
1	The destination Host is not communicating with the network—it took its ready-line down without saying why.
2	The destination Host is not communicating with the network—the Host was tardy in taking traffic from the network without saying why.
3	The destination Host does not exist to the knowledge of the NCC.
4	The IMP software is preventing communication with this Host; this usually indicates IMP software re-initialization at the destination.
5	The destination Host is down for scheduled P.M.
6	The destination Host is down for scheduled hardware work.
7	The destination Host is down for scheduled software work.
8	The destination Host is down for emergency restart.
9	The destination Host is down because of power outage.
10	The destination Host is stopped at a software breakpoint.
11	The destination Host is down because of a hardware failure.
12	The destination Host is not scheduled to be up.
13–14	Currently Unused.
15	The destination Host is in the process of coming up.

When the value of the sub-type field is 1, 2, 3, 4, or l5, the message-id field will have the "unknown" indication.

Bit 33 in this message will always be set to zero and Hosts receiving this message should discard (without reporting an error) type 6 messages with bit 33 set to 1. This will allow the later addition of similar status information on dead destination IMPs.

The Dead Host status message will be returned to the source Host shortly (immediately, if possible) after each Destination Host Dead (type 7, subtype 1) message. The destination Host Dead message applies to a specific named message, although the information contained in the Destination Host Dead message should probably be reported to all users connected to the dead Host. The Dead Host Status message does not apply to a specific named message and all users connected to the dead Host should be notified of the information contained in the Dead Host Status message.

7. *Destination Host or IMP Dead (or unknown)*—This message is sent in response to a message for a destination which the IMP cannot reach. The message to the "dead" destination is discarded. Sub-types:

    0. The destination IMP cannot be reached.

    1. The destination Host is not up.

    2. Communication with the destination Host is not possible because it does not have the expanded (new) leader capability (see Appendix A).

    3. Communication with the destination Host is administratively prohibited.

    4-15. Currently unused.

8. *Error in Data*—The IMP's Error flip-flop was set after transmission of the leader of a message but before the end of the message.

9. *Incomplete Transmission*—The transmission of the named message was incomplete for some reason. An incomplete transmission message is similar to a RFNM, but is a failure indication rather than a success indication. Sub-types:

0. Destination Host did not accept the message quickly enough.

1. Message was too long (in excess of maximum number of packets specified for connection).

2. The Host took more than 15 sec. to transmit the message to the IMP. This time is measured from the last bit of the leader through the last bit of the message.

3. Message lost in the network due to IMP or circuit failures.

4. The IMP could not accept the entire message within 15 sec. because of unavailable resources (see Section 3.1).

5. Source IMP I/O failure during receipt of this message.

6–15. Currently unused.

10. *Interface Reset*—The IMP's ready line has been dropped and pending output to the Host has been discarded (see Section 3.2). This probably indicates that the Host did not accept data from the IMP fast enough. Since dropping the ready line also sets the IMP's error flip-flop, the next message from the Host will be discarded and answered with a type 1 (sub-type 0) message. The sub-type field is unused.

11. *Refused, Try Again*[*]—A type 0, subtype 1 or 2 message was received from the Host but a certain "non-markable" resource needed for sending the message was not available. The message was discarded, and the Host should try to send it again when best able to do so. Sub-type:

    0. IMP buffer was not available.
    1. Transmit block for connection was not available.
    2–15. Currently unused.

12. *Refused, Will Notify*[*]—A type 0, subtype 1 or 2 message was received from the Host but a certain "markable" resource needed for sending the message was not available. The message was discarded, and the Host will be notified via a type 14 (*Ready*) message when the resource becomes available. Sub-types:

[*] The non-blocking Host interface (see Section 3.7) is not yet implemented.

0–1. Currently unused.
2. Connection not available.
3. Reassembly space (for multi-packet message only) not available at destination.
4. Message number not available.
5. Transaction block for message not available.
6–15. Currently unused.

13. *Refused, Still Trying*[*]—A type 12 response is indicated, but a type 14 message has already been queued for some previous type 12 response. The message was discarded and no other response will be given. The subtype field is unused.

14. *Ready*[*]—The needed resource has become available for some previous type 0, subtype 1 or 2 message. The actual message is “named” by the message-id field.

15–255. *Unassigned*. Messages of other than type 0 are sent to the Host prior to messages of type 0.

Bits 33–40 Handling Type—

The value assigned by the source Host, this field is used only in message types 0, 5, 7–9, and 11–14.

Bits 41–48 Source Host—

See Source IMP, below.

Bits 49–64 Source IMP—

For type 0 messages, these fields identify the particular Host and IMP site that originated the message. For type 4 messages, these fields identify the local Host and IMP, and for message types 5–9 and 11–14, these fields identify the particular Host and IMP site to which a type 0 message was sent or will be sent. The fields are unused in all other message types.

Bits 65–76 Message-id—

For message types 0, 5, 7–9, and 11–14, this is the value assigned by the source Host to “name” the message. The field is also used by message types 2 and 6, and unused by all other message types.

Bits 77–80 Sub-type—

[*] The non-blocking Host interface (see Section 3.7) is not yet implemented.

This field is used by message types 0–2, 4–7, 9, and 11–12.

Bits 81–96 Message Length—

This field is contained in type 0 messages only, and is the actual length in bits of the message (exclusive of leader, leader padding, and hardware padding) as computed by the destination IMP using the end of message padding conventions. It should be noted that the IMP will *not* verify the length of the message if it is specified by the Host.

The following table shows which non-constant fields are used by each valid message type.

	Message Type													
Fields	0	1	2	4	5	6	7	8	9	10	11	12	13	14
Trace	x													
Leader Flags	x													
Message Type	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Handling Type	x				x		x	x	x		x	x	x	x
Source Host	x			x	x	x	x	x	x		x	x	x	x
Message-id	x		x		x	x	x	x	x		x	x	x	x
Sub-type	x	x	x	x	x	x	x	x	x		x	x		
Message Length	x													

3.5 Word Length Mismatch and Message Boundaries

There are two related aspects of word length mismatch: first, the obvious need for message formatting in order for Host computers having different word lengths to communicate; and, second, the need for locating the end of a message, since mismatched word lengths may lead to messages that end in the middle of words. The IMP design guarantees that between Hosts of identical word length, the natural word boundaries are preserved. Generally, however, reformatting is left to the Hosts. The problem of recognizing the end of a message at the receiving Host is solved in the following manner. As a message passes from the transmitting Host to its IMP, the standard Host/IMP interface appends a one to the bit string when it receives the end-of-message signal. This bit may fall in any position of an IMP word. The hardware then fills any remaining bits of this IMP word with trailing zeros. This process is called *IMP padding*. The transmitting Host may also specify the message length (in bits), which need not be the same as the physical length of the message.

As the message is serially shifted to the receiving Host, the last bit from the IMP will generally fall somewhere in the middle of the receiving Host's word. The remaining bits in this word are to be filled in with additional trailing zeroes from the Host's special interface hardware. (Note that a one is purposely omitted here.) Thus, the message appears in the receiving Host with a one immediately following the last data bit in the message and a string of zero or more trailing zeroes, that terminates at a Host word boundary, following the one. The last Host word in the received bit stream does not necessarily contain the last data bit in the message; it may contain nothing but padding.

The maximum message that is shipped across the interface from the IMP to the destination Host contains 8160 bits (i.e., it includes the source IMP's padding). The destination Host's special interface unit will generally add padding of its own to round out the total number of bits going into the Host's memory to a multiple of the destination Host's word length. The destination Host should, therefore, be prepared to accept messages of at least 8160 bits. Not counting the destination Host's padding, messages of greater than 8160 bits in length should be discarded by the receiving Host.

It should be noted that Hosts may specify *leader padding* (see Section 3.3, NOP message). This padding is some integral number of 16-bit words which are transmitted and received immediately following the 96-bit leader of type 0 messages. This facility is designed to assist the Host in aligning some portion of the transmitted or received data with its own word boundaries. In particular, the Host may wish to make the sum of leader, leader padding, and other elements of Host-to-Host Leader equal to an integral number of Host words. This leader padding is *not* counted in the message length and exists only across the Host/IMP interface (i.e., not in the network).

3.6 Uncontrolled Packets

For certain limited experiments which are being carried on using the network, it may be desirable for specified Hosts to be able to communicate without using the normal ordering and error-control mechanisms in the IMP. Communication of this type is possible using the Host-to-IMP and IMP-to-Host message type 0, sub-type 3. The rules governing IMP handling of these messages are:

1. Messages of type 0, subtype 3 are limited to the Host-to-IMP leader (96 bits) and not more than 991 additional data bits. Messages which exceed this length will be discarded without error notification.

2. At the destination IMP, these messages are put on the output queue for the destination Host in the order in which they are received; the messages are likely to be delivered in a different order from the order in which they were sent. Duplicate copies of some messages may be delivered.

3. There is no source-to-destination control of these messages. Lost messages will not be retransmitted. No RFNM, Incomplete Transmission, Destination Dead, etc., will be returned to the source.

4. The same bit-level error control applied to Regular messages will be applied to these messages passing between IMPs; i.e., type 0 subtype 3 messages are delivered with a very low probability of bit error.

5. If at any time there are insufficient resources in the network to handle one of these messages, it will be immediately discarded.

6. Use of these messages between two Hosts will not affect use of regular messages between these Hosts. Regular messages and subtype 3 messages may be intermixed over the Host/IMP interface.

7. Uncontrolled use of these messages will degrade the performance of the network for *all* users. Therefore, ability to use these messages will be regulated by the Network Control Center and will require prior arrangement for each experiment.

3.7 Non-Blocking Host Interface[*]

As mentioned in Section 3.1, it is sometimes necessary for the source IMP to block the transmission of a message from the source Host. When this blocking occurs, *all* messages from that Host are held back, even though some of them might well be transmitted unimpeded if allowed into the IMP. Such might be the case, for example, if Host *A* is sending to Hosts *B*, *C*, and *D*, and the connection to Host *B* has eight messages in transit, the first (oldest) of which has become lost in the net. If a ninth message is sent to *B*, the interface will be blocked for the duration of the “incomplete” timeout (30–45 seconds), waiting for a message slot to become available on that connection. During this time, however, it would have been possible for *A* to send messages to *C* and *D*, had the interface not been blocked.

[*] This section is a preliminary specification and is still subject to modification. The extensions required to the Host/IMP protocol have not yet been implemented.

The non-blocking Host interface is a software mechanism which provides the source Host with the capability of keeping the interface unblocked for the vast majority of situations under which it might otherwise have become blocked. There will still be a few circumstances, associated with bandwidth and storage limitations of the source IMP, under which the interface may be blocked regardless of the mechanism used by the Host.

The non-blocking mechanism works by allowing the Host to flag some or all of its type 0 messages as “refusable”, thus allowing the IMP to discard them if they would otherwise block the interface. In such a case, not only is the Host notified that the message was discarded, but it is also given guidance as to when the message should be retransmitted. In most cases, the particular resource that was missing is “markable”, and the Host can be notified when the resource becomes available. In some cases, the resource is not “markable”, and the Host must simply retransmit in accordance with its own requirements. The specific protocol for this mechanism is now described.

Host-to-IMP type 0 messages have four subtypes: *Non-refusable*, *Refusable*, *Get Ready*, and *Uncontrolled*. The *uncontrolled* subtype, described in Section 3.6, is never refused, and because it does not require most of the resources of “controlled” messages, is seldom blocked. The *Non-refusable* subtype is the standard mode of operation, which can cause interface blocking under the various circumstances described in Section 3.1. The *Refusable* subtype is treated identically to the *Non-refusable* subtype if blocking is not necessary. Under most circumstances where blocking would have been necessary, however, this message subtype is discarded, and one of three types of IMP-to-Host messages sent back to the Host. A *Refused, Try Again* (type 11) message indicates a “non-markable” resource was required, and the Host should merely retransmit at its convenience. A *Refused, Will Notify* (type 12) message indicates a “markable” resource was required. The Host should wait for a fourth IMP-to-Host message type, *Ready* (type 14), before retransmitting. The IMP will send the *Ready* when the resource becomes available. A *Refused, Still Trying* (type 13) message indicates that the IMP has already given a *Refused, Will Notify* on that connection, but has not yet sent the *Ready* (it will only queue one such response at a time for any connection). There is no additional response after the *Refused, Still Trying*, and the Host should queue the message to be retransmitted after the one for which the *Ready* is expected.

The *Get Ready* subtype of the type 0 Host-to-IMP message is not a real message in the sense that it contains only the leader of an intended (future) message. It is provided so that the Host can determine whether or not a message could get through without blocking, without actually sending the data in the message through the interface. The possible responses to this subtype are identical to those of the *Refusable* subtype, except that in the normal case, when the *Refusable* message would have been transmitted to the destination without any interface blocking followed eventually by a RFNM, the IMP’s response to the *Get Ready* is to send a *Ready* back to the Host.

Finally, it should be noted that a *Ready* does not guarantee that a retransmission will not be blocked, since no resources are actually reserved for some particular *message-id*, and in fact many are shared by all connections. The best strategy for the Host willing to use the non-blocking feature is to make all messages *Refusable*, even when responding to a *Ready*.