DECnet Digital Network Architecture
Phase IV
Routing Layer Functional Specification
Order No. AA-X435A-TK

digital

software

DECnet Digital Network Architecture Phase IV

Routing Layer Functional Specification

Order No. AA-X435A-TK

December 1983

This document specifies the functions, interfaces, and protocols for implementing that part of the Digital Network Architecture that models the software controlling the routing of messages within DECnet communications networks.

SUPERSESSION/UPDATE INFORMATION: This is a new manual.

VERSION: 2.0.0

To order additional copies of this document, contact your local Digital Equipment Corporation Sales Office.

digital equipment corporation · maynard, massachusetts

First Printing, December 1983

The postage-prepaid READER'S COMMENTS form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

Distributed Systems Publications typeset this manual using DIGITAL's TMS–11 Text Management System.

MGTPEALL

CONTENTS

1.0 INTRODUCTION

This document describes the structure, functions, interfaces, protocols, and algorithms for implementing the Routing Layer. The Routing Layer is the part of the DIGITAL Network Architecture that models the software (or hardware) controlling the routing of messages, called packets, within DECnet communications networks.

A DECnet network is a family of software modules, data bases, and hardware components typically used to tie DIGITAL systems together for resource sharing, distributed computation, or remote system communication. DECnet network implementations follow the DIGITAL Network Architecture (DNA) model.

DNA is a layered structure. Modules in each layer perform distinct functions. Equivalent modules within the same layer in both the same and different nodes communicate using protocols. A node is an implementation of the DNA Session Control Layer. Usually a single computer is associated with one node. Protocols are the messages exchanged by modules and the rules governing the message exchanges. Modules in functionally different layers of DNA interface using either subroutine calls or a system-dependent method. This document describes these interfaces in the format of calls to subroutines.

The routing described in this document is Phase IV DECnet routing. It is the major function of the Routing Layer. DIGITAL's routing is intended for users with networks consisting of any combination of point-to-point links, X.25 links, multipoint links, and Ethernets.

Phase IV DECnet routing is hierarchical, in order to support large networks. A large network is partitioned by the network manager into "areas". Each node resides in exactly one area. Routing within an area is referred to as "level 1 routing". Routing between areas is referred to as "level 2 routing". Level 2 routers keep track of the paths to destination areas. Level 1 routers keep track only of the routing within their own area, and keep track of the nearest level 2 router within their area. When a level 1 router receives a packet for forwarding to a foreign area, it sends the packet to the nearest level 2 router. Then the packet travels via level 2 routing to the destination area, where it again travels via level 1 routing to the destination node.

Phase IV DECnet is upwards compatible with Phase III DECnet.

Thus there are the following types of nodes in Phase IV:

1. IV endnodes -- These nodes deliver packets to other nodes and receive packets from other nodes, but do not route packets through. They differ from Phase III endnodes in that they include an Initialization Sublayer for the Ethernet, and support hierarchical addressing in the layers above the Routing Layer.

2. IV level 1 routers -- These nodes deliver and receive packets from other nodes, and route packets from other source nodes through to other destination nodes. They route directly to nodes within their own area, and route towards a level 2 router when the destination node is in a different area.

3. IV level 2 routers -- These nodes act as IV level 1 routers in addition to acting as a node in the subnet consisting of level 2 routers. A node in the level 2 subnet routes towards a destination area.

4. III routers -- These nodes deliver and receive packets from other nodes within their own area, and route packets from other source nodes through to other destination nodes in their own area. These nodes do not include an Initialization Sublayer for the Ethernet, and thus cannot support the Ethernet.

5. III endnodes -- These nodes deliver packets to other nodes and receive packets from other nodes, but do not route packets through. These nodes do not include an Initialization Sublayer for the Ethernet, and thus cannot support the Ethernet.

Networks that include endnodes, and/or that consist of more than one area, are restricted in how the nodes can be interconnected.

In Phase IV a node's address is a 16 bit number, the top 6 bits of which define the area, and the bottom 10 bits of which give an address within an area.

A glossary at the end of this document defines many Routing Layer terms.

This document is intended for readers familiar with computer communications and with DECnet. The primary audience is those who are implementing DECnet systems. However, it may also be of interest to those who want to know the details of the Routing Layer design. The other current DNA functional specifications are:

<u>DNA Data Access Protocol (DAP) Functional Specification</u>, Version 5.6.0, Order No. AA-K177A-TK

<u>DNA Digital Data Communications Message Protocol (DDCMP) Functional Specification</u>, Version 4.1.0, Order No. AA-K175A-TK

<u>DNA Ethernet Data Link Functional Specification</u>, Version 1.0.0, Order No. AA-Y298A-TK

<u>DNA Ethernet Node Product Architecture Specification</u>, Version 1.0.0, Order No. AA-X440A-TK

<u>DNA Maintenance Operations Functional Specification</u>, Version 3.0.0, Order No. AA-X436A-TK

DNA Network Management Functional Specification, Version 4.0.0, Order No. AA-X437A-TK

DNA Network Services Protocol Functional Specification, Version 4.0.0, Order No. AA-X439A-TK

DNA Session Control Functional Specification, Version 1.0.0, Order No. AA-K182A-TK

The Ethernet - A Local Area Network - Data Link Layer and Physical Layer Specifications, Version 2.0, (Digital, Intel, and Xerox), Order No. AA-K759B-TK

The _DECnet DIGITAL Network Architecture (Phase IV) General Description_ (Order No. AA-N149A-TC) provides an overview of the network architecture and an introduction to each of the functional specifications.

2.0 FUNCTIONAL DESCRIPTION

The Routing Layer routes messages in DECnet networks and manages the message packet flow. A packet is a unit of data to be routed from a source node to a destination node. The Routing Layer consists of two sublayers:

1. Control. The Control Sublayer supplies full-duplex packet transmission between any pair of nodes. It is independent of the specific Data Link Layer below it, except for knowing about two generic types of links:

   1. non-broadcast links, which include DDCMP point-to-point, DDCMP multipoint, and X.25, and

   2. broadcast links, which include the Ethernet.

   The Routing Control Sublayer masks the physical and topological characteristics of the network from higher layers. It consists of the following components:

   o Routing

   o Congestion control

   o Packet lifetime control

2. Initialization. The Routing Initialization Sublayer masks the characteristics of the Data Link Layers from the Routing Control Sublayer. It consists of the following components:

   o Initialization

o Physical circuit monitor

The Routing Initialization Sublayer controls the Data Link Layer and is Data Link Layer dependent.

The Routing Layer components provide the following functions.

Routing. The routing function determines packet paths. A path is the sequence of connected nodes between a source node and a destination node. When the Routing Layer receives a packet, the routing component refers to a data base that is periodically updated by Routing Layer modules in adjacent nodes. The routing component uses information in this data base to determine if a path to a destination exists, and, if so, what the next hop in the path is. The routing component then forwards the packet to its destination. If more than one path exists to a destination, the routing component ascertains the best path.

The combined knowledge of all the Routing Layer modules of all the nodes in a network is used to ascertain the existence of a path, and route the packet to its destination. The routing component at a routing node has the following specific functions:

- o It extracts and interprets the route header in a packet.
- o It performs packet forwarding based on the destination.
- o It manages the characteristics of the path. If a node or link fails on a path, it finds an alternate route.
- o It interfaces with the Routing Initialization Sublayer to receive reports concerning a circuit or node that has failed or the subsequent recovery of a circuit or node.
- o It returns packets addressed to unreachable nodes to the End Communications Layer (ECL), if requested to do so by ECL. A node is unreachable if it is unknown, or the path to it exceeds the maximum hops of the network. A hop is the logical distance between two adjacent nodes. Maximum hops is a Routing Layer parameter that is equal to the maximum path length in the network.

Congestion control. Congestion control manages the buffers at each packet switching node (that is, at each node that permits route-through).

Packet lifetime control. Packet lifetime control bounds the number of nodes a packet can visit.

Initialization. The Initialization component supplies the following functions:

- o It identifies the adjacent node and the adjacent node's Routing Layer.

- o It performs node verification, if required.

Physical circuit monitor. This component monitors errors detected by the Data Link Layer.

2.1 Design Scope

The Routing Layer supports the following design requirements:

1. Deliverability. It accepts and delivers packets addressed to reachable destinations and rejects packets addressed to unreachable destinations.

2. Adaptability. It adapts to topological changes, but not to traffic changes. (Topological changes are changes in the configuration of active circuits and nodes in a network. Traffic changes are changes in the load on circuits in a network.)

3. Promptness. The period of adaptation to topological changes in the network is a reasonable function of the network diameter (that is, the maximum logical distance between network nodes) and circuit speeds.

4. Efficiency. The Routing Layer is both processing and memory efficient. It does not create excessive routing traffic overhead.

5. Robustness. The Routing Layer recovers from transient errors such as lost or temporarily incorrect routing messages. It tolerates imprecise parameter settings.

6. Stability. The Routing Layer stabilizes in finite time to "good routes," provided no continuous topological changes or continuous data base corruptions occur.

7. Operator control. An operator can control many routing functions via parameter changes, and inspect parameters, counters, and routes. Routing, however, will not depend on operator input for correct behavior.

8. Simplicity. The Routing Layer is sufficiently simple to permit performance tuning and failure isolation.

9. Maintainability. The Routing Layer provides mechanisms to detect, isolate, and repair most common errors that may affect the routing computation and data bases.

10. Verification of compatibility. The Routing Layer Initialization Sublayer prevents incompatible routing algorithms from coexisting in the network.

11. Heterogeneity. The Routing Layer operates over a mixture of network node types, communication circuits, and topologies. In particular, it supports point-to-point, multipoint, and multiaccess.

12. Support of subsets. The Routing Layer allows nodes to support a subset of the routing functions.

13. Extensibility. The Routing Layer accommodates increased routing functions, leaving earlier functions as a subset.

14. Evolution. The Routing Layer allows orderly transition from algorithm to algorithm without shutting down an entire network.

15. Deadlock Prevention. The congestion control component prevents deadlock, the condition in which the Routing Layer fails to deliver data.

16. Independence. Congestion control does not depend on routing for effective operation.

17. Duplicate message reduction. The packet lifetime control algorithm significantly reduces the risk of the user receiving duplicate messages.

18. Large networks. With hierarchical routing, the Routing Layer supports networks of several thousand nodes.

The following are not within the scope of Phase IV Routing Layer:

1. Traffic adaptation. The Routing Layer does not react to traffic flow automatically.

2. Traffic service classes. The Routing Layer does not distinguish among different classes of traffic in route determination.

3. Source-destination routing. The Routing Layer does not determine routes by source as well as destination.

4. Gross operator failure. The Routing Layer does not attempt to protect the network from operator actions, such as removing a circuit or a node, that may disconnect the network.

5. Guaranteed delivery. The Routing Layer does not guarantee delivery of all offered packets.

2.2 Relationship To DIGITAL Network Architecture

The DIGITAL Network Architecture (DNA) is a model that defines the functional requirements of all DECnet implementations. The model is a layered one. The Routing Layer lies between the End Communications Layer and the Data Link Layer, as shown in Figure 1.

```
       .----------------------------.
       !    User Modules            !
       `----------------------------'
          !         !         !
          !         !         V
       .- ! ------- ! -----------------------------------.
.------!  ! Network !  Management  Modules               !
!      `- ! ------- ! -----------------------------------'
!         !         !         !              !          !
!         !         V         V              !          !
!      .- ! -------------------------------- ! ------.  !
:----> !  ! Network Application Modules      !       !  !
!      `- ! -------------------------------- ! ------'  !
!         !                   !              !   !      !
!         V                   V              V   !      !
!      .---------------------------------------. !      !
:----> !    Session Control Modules            ! !      !
!      `---------------------------------------' !      !
!                             !                  !      !
!                             V                  !      !
!              .----------------------------.    !      !
:------------> ! End Communications Modules !    !      !
!              `----------------------------'    !      !
!                             !                  !      !
!                             V                  !      !
!              .--------------------------.      !      !
:------------> ! Routing Layer Modules    !      !      !
!              `--------------------------'      !      !
!                             !                  !      !
!                             V                  V      V
!              .-----------------------------------------.
:------------> ! Data Link Modules                       !
!              `-----------------------------------------'
!                             !
!                             V
!              .--------------------------.
`------------> ! Physical Link Modules    !
               `--------------------------'
                              !
                              `------------------------->
```

Figure 1. Routing Layer Relation to DNA

A brief description of each DNA layer follows:

1. User Layer. The highest layer, the User Layer supports user services and programs.

2. Network Management Layer. Modules in the Network Management Layer provide user control over and access to network parameters and counters. These modules also furnish up-line dumping, down-line loading, and testing functions. This layer is the only layer that has direct access to each lower layer for control purposes.

3. Network Application Layer. Modules in the Network Application Layer support network functions, such as remote file access and file transfer, used by the two higher layers.

4. Session Control Layer. The Session Control Layer defines the system-dependent aspects of logical link communication, which allows controlled data movement between network nodes.

5. End Communications Layer. The End Communications Layer defines the system-independent aspects of logical link communication.

6. Routing Layer. Modules in the Routing Layer route messages between source and destination nodes.

7. Data Link Layer. The Data Link Layer defines the protocol concerning data integrity and physical channel management.

8. Physical Link Layer. The Physical Link Layer encompasses a part of the device driver for each communications device plus the communications hardware itself. The hardware includes interface devices, modems, and the communication lines. This layer controls the end-to-end transmission of data.

Each DNA layer uses the services of the layer below it. In addition, Network Management provides a control interface to all the DNA layers below it. User and Network Application Layer modules can interface directly with Session Control.

2.3 Routing Layer Environment Requirements

The Routing Layer requires guarantees from the operating system, the End Communications Layer, and the Data Link Layer.

The required operating system guarantees are:

1. Priority scheduling such that the Routing Layer receives minimum processing guarantees

2. A quota of buffers to the Routing Layer sufficient to perform routing and packet lifetime control functions

3. Access to a timer or notification of specific timer expiration

The End Communications Layer must guarantee the return of a buffer within a short, bounded amount of time. Otherwise, the Routing Layer may discard packets received for ECL if ECL exceeds a quota.

The required Data Link Layer guarantees for point-to-point links are:

1. Provision that both source and destination nodes complete start-up before message exchange can occur

2. Detection of remote start-up

3. Provision that no old messages be received after start-up is complete

4. Masking of transient errors in order to prevent packet data corruption

5. Provision for not duplicating or corrupting packets

6. Packet sequentiality ensuring that, if a packet has been received, all previously sent packets have been received

7. Reporting of failures and degraded circuit conditions

The required Data Link Layer guarantees for Ethernets are:

1. Provision for not corrupting packets

2. Packet sequentiality ensuring that, if a packet has been received, no previously sent packet will be subsequently received

2.4 Routing Layer Characteristics

The Routing Layer possesses the following characteristics:

- Variable delay. There is a variable delay time. Delay is defined as the time between receipt of a packet from ECL at a source node and delivery of that packet to ECL at a destination node.

- o Nonsequential delivery. The Routing Layer does not guarantee delivery of packets to ECL at the destination node in the same sequence in which they were received from ECL at the source node.

- o Packet integrity. The Routing Layer will not modify or misdeliver a packet.

2.5 Routing Layer Control Functional Organization

The Routing Layer Control Sublayer components can be broken down into more specific functional components. These are described briefly here and in detail below.

2.5.1 Routing -

The Routing processes and data bases are:

- o Decision Process
- o Update Process
- o Forwarding Process
- o Receive Process
- o Routing data base
- o Forwarding data base

2.5.1.1 Decision Process - This process selects routes to each destination in the network. It consists of a connectivity algorithm that maintains path lengths and a traffic assignment algorithm that maintains path costs. Path length is the number of hops along a path between two nodes. Circuit cost is a nonzero, positive integer value associated with using a circuit, and path cost is the sum of the circuit costs along a path between two nodes.

When a routing node receives a Routing Message (a type of Routing Layer control message) from an adjacent node, the routing node executes the Decision Process. Execution of the Decision Process results in the determination of <circuit, neighbor> pairs (known as adjacencies) along which to forward packets and possibly the conclusion that one or more particular destination nodes are unreachable.

The system manager must set several of the parameters in the Routing data base that the Decision Process uses. These include maximum address, maximum number of areas (in level 2 nodes), maximum cost, maximum hops, maximum circuits, circuit costs, maximum total broadcast end-node adjacencies, maximum total broadcast router adjacencies, and maximum broadcast router adjacencies for each broadcast circuit. The values of the cost parameters are arbitrary. Appendix F suggests an algorithm for determining circuit costs. The values of the other parameters depend on the specific topology of the network. If these values are not set correctly, the decision algorithms will not work correctly.

The figure below shows a sample network, consisting of a single area, and depicts some of the Routing terms. The glossary contains definitions of these and other Routing Layer terms.

```
 .---.          2          .---.
 | D |---------------------| E |
 `---'                     `---'
   /   \                      \
3 /     \ 7                    \ 4
 /       \                      \
.---.  2  .---.        3        .---.
| C |-----| B |-----------------| F |
`---'     `---'                 `---'
            |
   1 hop    | 2
            |
          .---.
          | A |
          `---'
```

Legend:

```
 .---.
 | x |          = node
 `---'
 _____          = circuit

   n            = circuit cost
 .---.  .---.
 | x |--| x |   = hop
 `---'  `---'
```

Node A wants to send a packet to node D. There are three possible paths.

Path	Path Cost	Path Length
A-B, B-C, C-D	2 + 2 + 3 = 7 *	3 hops
A-B, B-D	2 + 7 = 9	2 hops
A-B, B-F, F-E, E-D	2 + 3 + 4 + 2 = 11	4 hops

* 7 is the lowest path cost; node A therefore routes the packet to node D via this path.

Figure 2. Routing Terms

2.5.1.2 Update Process - This process constructs and propagates Routing Messages. A Routing Message contains path cost and path length for all destinations. The Update Process sends Routing Messages to adjacent nodes after determining that certain conditions are met. General characteristics of the Update Process are:

- Level 1 Routing Messages are sent to adjacent routing nodes within the node's home area
- Level 2 Routing Messages are sent to adjacent level 2 routing nodes
- Level 1 Routing Messages contain information on all nodes within the home area
- Level 2 Routing Messages contain information about all areas
- Routing Message transmission is event-driven with periodic backup
- The routing update algorithm maintains an upper limit on routing traffic overhead

2.5.1.3 Forwarding Process - This process supplies and manages the buffers necessary to support packet route-through to all destinations.

It performs a table lookup to determine the output adjacency to use for forwarding to a given destination, reformats packets between short and long format if necessary, strips off the area fields when forwarding to a Phase III node, fills in the area fields when receiving from a Phase III node, and marks intra-Ethernet packets.

2.5.1.4 Receive Process - The Receive Process inspects a packet's route header and dispatches the packet to an appropriate Routing Layer Control component or to the End Communications Layer (ECL).

2.5.2 Congestion Control - The Congestion Control component manages buffers by limiting the maximum number of packets on the transmit queue for a circuit. Congestion Control regulates the ratio of packets received directly from ECL to route-through packets. Congestion Control also checks the packet size for each packet to be sent.

2.5.3 Packet Lifetime Control - The packet lifetime control component requires three processes:

1. Loop Detector

2. Node Listener

3. Node Talker

2.5.3.1 Loop Detector - This process prevents excessive packet looping. It counts the number of nodes a packet has visited and removes a packet when it exceeds the visit limit.

2.5.3.2 Node Listener - This process determines that a minimum amount of activity has occurred between this node and an adjacent node. It also determines if the identity of the adjacent node has changed. Violations of the minimum activity audit result in the declaration that the adjacency between the nodes is down.

On non-broadcast circuits, Hello Messages need to be sent only in the absence of other traffic, to ensure that a minimum amount of activity occurs. On broadcast circuits, Hello Messages need to be sent periodically regardless of other forms of traffic.

2.5.3.3 Node Talker - This process provides the minimum activity for each adjacent Node Listener. It places an artificial load on the adjacency so failures can be detected. The Node Talker and Listener provide for detection of adjacent Routing Layer halt and adjacent node identity change.

3.0 INTERFACES

This section describes the three external Routing Layer interfaces:

1. Network Management Layer interface
2. Data Link Layer interface
3. End Communications Layer interface

In addition, this section describes the single internal Routing Layer interface, Initialization Sublayer.

The interfaces take the format of calls to subroutines, as follows:

FUNCTION (input ; output)

Each call represents a specific function. An implementation is not required to code the interface as calls to subroutines.

The following symbols are used throughout the document:

<>	not equal to
<=	less than or equal to
>=	greater than or equal to
>>	much greater than
SQRT(x)	the square root of
CEILING(x)	the least integer greater than or equal to x

3.1 Network Management Layer Interface

This interface allows Network Management to control and observe the Routing Layer interactively. Network Management can exert indirect control over the Routing Layer via parameter changes. The following Network Management functions form a set of primitive functions that can be used to construct more complex functions.

READ SELF PARAMETERS (;parameters)

This function gives the values of the node's parameters. Parameters are:

- Routing-State -- Routing Layer operating, or terminated, requiring problem correction and initialization

- Net-Management-State -- ON or OFF. Setting this parameter to "ON" from "OFF" forces Routing to initialize all its data bases.

- ID, top 6 bits of which is HOMEAREA, bottom 10 bits of which is Tid, node within area

- NN -- highest node number within the area

- NA (in level 2 routers only), highest area number in the network

- NC -- number of circuits supported by this node

- NBRA -- number of Broadcast Router Adjacencies (BRAs) supported by this node

- NBEA -- number of Broadcast Endnode Adjacencies (BEAs) supported by this node

- Maxh -- Maximum hops possible in a path to a reachable destination within an area

- Maxc -- Maximum cost possible in a path to a reachable destination within an area

- AMaxh -- (in level 2 routers only), maximum hops possible in a path to a reachable area

- AMaxc -- (in level 2 routers only), maximum cost possible in a path to a reachable area

- Maxv -- Maximum visits allowable for a packet

- T1 -- Background timer for routing updates on non-broadcast circuits

- BCT1 -- Background timer for routing updates on broadcast circuits

- Routing Type -- Phase IV area router, Phase IV router, Phase IV nonrouting

- Routing Version -- the current routing version, ECO, and user ECO

- BS -- Buffer size -- Six greater than the maximum buffer size for use by Routing, excluding Routing Layer Header and excluding Data Link Layer header. (Note -- the added 6 bytes are for historical compatibility with Phase III, when the Buffer size parameter included Routing Layer header, which was 6 bytes at the time.)

- SS -- Segment size -- Six greater than the maximum segment size to be used by ECL. SS must be less than or equal to BS. It will usually be equal to BS. It may be less than BS while the buffer sizes in the network are being increased or decreased.

- NB -- maximum number of buffers for forwarding

- Subaddresses -- (in nodes with X.25 Data Link Mapping only) -- the range of local DTE subaddresses that are acceptable on an X.25 circuit for an incoming call

SET SELF PARAMETER (PARAMETER, VALUE; status)

This function sets the parameter PARAMETER to VALUE. The parameters supported are those enumerated above in the READ SELF PARAMETERS call.

The returned status is:

- success

- unknown parameter

- illegal value

READ SELF COUNTER (; counters)

This function gives the values of the node's counters, maintained by the Routing Layer. Appendix E describes these counters in detail.

Counters are:

- node unreachable packet loss

- aged packet loss

- node out-of-range packet loss

- oversized packet loss
- packet format error
- partial routing update loss
- verification reject

READ AND CLEAR SELF COUNTERS (; counters)
This function returns the same values as READ SELF COUNTERS. It zeroes all counters upon completion

READ CIRCUIT PARAMETERS (CIRCUIT; status, parameters)
This function gives the values of the CIRCUIT's parameters. These parameters are described more fully in Section 4.2.

The returned status is one of:

- success
- unknown circuit

Parameters are:

- Type (Ethernet, X.25, DDCMP)
- Net-Management-State (ON or OFF)
- State (as determined by Initialization Sublayer)
- Cost
- Hello Timer
- Recall Timer
- Originating Packet Limiter (OPL)
- Type Specific Information
  - For Ethernet, the type specific information is:
    - NR -- Number of BRAs allowed on this Ethernet
    - Priority -- the router's priority to be Designated Router

- Designated Router -- the ID of the router currently chosen to be Designated Router on this Ethernet

o For X.25, the type specific information is:

- Port State -- supplied by the X.25 Data Link Layer

- Blocking -- indicates if blocking can be done by this node

- Negotiated Blocking -- indicates if neighbor also requested blocking

- Maximum Recalls -- the maximum number of call trials permitted from the Data Link Start state before halting

- Recall Count -- the number of call attempts that have been made to try to initialize a virtual circuit. This is reset by the operator turning the circuit on.

- VC Type -- type of virtual circuit:

  . PVC (permanent virtual circuit)

  . incoming switched virtual circuit

  . outgoing switched virtual circuit

- VC name. For an SVC, a network name and a DTE address. For a PVC, a PVC name.

SET CIRCUIT PARAMETER (CIRCUIT, PARAMETER, VALUE; status)
This function sets the parameter PARAMETER in circuit CIRCUIT to VALUE.

The returned status is:

. success

. unknown circuit

. unknown parameter

. illegal value

READ CIRCUIT COUNTERS (CIRCUIT; status, counters)

This function gives the values of the counters maintained by the Routing Layer for the specified circuit. Counters are described more fully in Appendix E.

The returned status is one of:

- success
- unknown circuit

Counters are:

- Transit ("Route-Through") Packets Received
- Transit Packets Sent
- Terminating Packets Received (packets received from other nodes addressed to this node)
- Originating Packets Sent (packets from this node addressed to other nodes)
- Transit Congestion Loss
- Terminating Congestion Loss
- Circuit Down
- Initialization Failure

In addition, for X.25 circuits, there is the type specific counter:

- Corruption Loss -- a count of the data errors detected for this circuit

READ AND CLEAR CIRCUIT COUNTERS (CIRCUIT; status, counters)

This function returns the same values as READ CIRCUIT COUNTERS. It zeroes the counters upon completion.

READ NODE PARAMETERS (NODE; parameters)

This function gives the values of parameters maintained by Routing for the specified node.

If the local node type is a level 1 router, and the specified node is in another area, the values of parameters maintained by Routing for the special destination #0, meaning "nearest level

2 router" are given.

Parameters are:

- Reachability Flag
- Output Circuit [and NextHop, for Ethernets only] to NODE
- Hops of minimum cost path to NODE
- Cost of minimum cost path to NODE

READ AREA PARAMETERS (AREA; parameters)

This function gives the values of the parameters maintained by Routing for the specified area. This function is implemented in level 2 routers only. Parameters are:

- Reachability Flag
- Output Circuit [and NextHop, for Ethernets only] to AREA
- Hops of minimum cost path to AREA
- Cost of minimum cost path to AREA

READ ADJACENCY PARAMETERS (ADJACENCY; parameters)

This function gives the values of the parameters maintained by Routing for the specified adjacency. An adjacency is a <circuit, node ID> pair. The argument "adjacency" is an adjacency # in the Routing Data Base.

Parameters are:

- In Use Flag
- Node ID
- Node Type, one of:
  - Level 1 router
  - Level 2 router
  - Phase IV endnode
  - Phase III router

  - Phase III endnode

- circuit
- Listen Timer
- neighbor's blocksize

READ EVENT (; event)

Return: the oldest event in the Routing Layer's internal event queue

The Routing Layer maintains an internal event queue into which it places events. Events are described in Appendix E. This function reads the oldest event in the queue.

CLEAR EVENTS

Returns: none

This function clears all events from the Routing Layer's internal event queue.

3.2 Data Link Layer Interface

This interface, between the Routing Layer's Initialization Sublayer and the Data Link Layer, consists of commands to and responses from the Data Link Layer. The interface supports the exchange of data, control, and error information. Data is information to be sent or received by the Data Link Layer protocol. Its description usually consists of a starting buffer address and a length or character count, or a chain of addresses and counts. The control information starts and stops the protocol. The error information reports circuit conditions. The functions of the interface, described as calls, are as follows:

TRANSMIT (circuit, buffer, [NextHop], [more data to follow])

Returns: none

This function gives a message to the Data Link Layer for transmission.

On a broadcast circuit, the parameter "NextHop" must be supplied. The parameter "NextHop" is not supplied on a non-broadcast circuit.

"More data to follow" is a flag specified only on X.25 circuits.

CHECK TRANSMIT BUFFER (buffer)

Returns: buffer is still queued

buffer is returned to the Routing Layer

This function returns information about the transmit buffer.

INITIALIZE CIRCUIT (circuit)

Returns: none

This function causes the Data Link protocol to initialize the circuit.

In the case of the Ethernet, Data Link padding must be enabled. Also, Routing must tell the Data Link Layer to enable the protocol type PROT-TYPE, and the multicast ID ALL-ROUTERS (if the local node is a router), or the multicast ID ALL-ENDNODES (if the local node is an endnode).

In the case of X.25, this tells the Data Link Layer that if the virtual circuit is in the UNSYNC state, send a "reset confirmation", and if the virtual circuit is in the RUNNING state, send a "reset" packet. If the virtual circuit is in the cleared state, a call must be initiated.

STOP circuit (circuit)

Returns: none

This function halts the Data Link operation on the specified circuit.

For X.25 circuits, this means send a "clear" packet on a Switched Virtual Circuit, and send a "reset" packet on a Permanent Virtual Circuit.

STATUS (circuit; status)

Returns: off

running

initializing

This function returns the Data Link state of the circuit. If the Data Link modules provide additional states (for example, a maintenance state), they are treated as the OFF state.

READ ERROR COUNTERS (circuit; error counters)

Returns: error counter values (value,status)

This function returns the values of the Data Link error counters, and notification if error thresholds have been exceeded.

SUPPLY RECEIVE BUFFER (buffer; status)

Returns: buffer accepted

buffer rejected

This function provides an empty buffer to the Data Link modules for receipt of the next sequential message.

CHECK RECEIVE BUFFER (buffer; circuit, [PrevHop], [more data to follow], status)

Returns: no packet received

packet received, buffer returned

This function returns the above information about the receive buffer.

The PrevHop value is returned when a packet is received on a broadcast circuit. The PrevHop value is not returned when a packet is received on a non-broadcast circuit.

The "more data to follow" flag is returned only on X.25 circuits.

3.3 End Communications Layer Interface

This interface, between the End Communications Layer and the Routing Layer, consists of commands to and responses from the Routing Layer. The commands and responses exchange data and control information.

Data is information that the Routing Layer sends or receives. The data description is a destination address, source address, buffer address and length of data. Destination and source addresses are two-byte integer numbers with the most significant 6 bits being the area number, and the least significant 10 bits being the address within the area. The Routing Layer uses node addresses only, not node names, which are resolved at a higher layer.

Control information starts or stops transmission and reception of data and regulates the data flow to a reachable destination.

The functions of the interface, described as calls, are as follows:

TRANSMIT (destination, return flag, [circuit,[NextHop]], Tryhard, buffer)

Returns: buffer is queued

buffer is not queued and is returned to ECL

This function sends a packet.

The return flag indicates whether or not ECL wants the packet returned if the destination is unreachable or becomes unreachable before the Routing Layer can deliver the packet. If the flag is set to "true" (Boolean), the Routing Layer attempts to return the contents of the buffer to ECL as a "received packet." If the flag is not set, the Routing Layer discards the packet. The Routing Layer returns the buffer after ECL issues the CHECK TRANSMIT BUFFER call (described next).

Circuit, selected by the End Communications Layer, is either unspecified or a valid circuit. If the circuit is a broadcast circuit, NextHop must also be specified. For circuit level loopback testing, the End Communications Layer must specify a circuit, and if required, a NextHop. Otherwise, the Routing Layer determines the adjacency.

TryHard, if set, tells the Routing Layer to flush any cache entries for the destination. Currently the only cache entries are in endnodes attached to broadcast circuits.

CHECK TRANSMIT BUFFER (buffer)

Returns: buffer still queued

buffer returned to ECL

This function checks the status of a previously queued transmit buffer. It returns the buffer to ECL after any of the following:

- o The buffer is copied into a Routing Layer buffer.
- o The packet is transmitted.
- o The packet is discarded because the destination is unreachable and the return flag is not set.
- o The packet contents are transferred to a receive buffer because the destination is unreachable and the return flag is set.

SUPPLY RECEIVE BUFFER (buffer)

Returns: buffer queued for receive by the Routing Layer

buffer not queued for receive by the Routing Layer

This function queues a receive buffer to the Routing Layer.

CHECK RECEIVE BUFFER (source, circuit, [PrevHop], buffer)

Returns: buffer remains queued by the Routing Layer

buffer returned to End Communications Layer with source node address (buffer contains a normal packet)

buffer returned to End Communications Layer (buffer contains a "return to sender" packet -- ECL Functional Specification)

This function checks the status of a previously queued receive buffer. It returns the buffer if the packet was received or if the node is unreachable and the return flag is set. The circuit variable returns a valid circuit number (or a value for an internal link) for each received packet. For packets received over a broadcast circuit, PrevHop is set to the node number of the node which forwarded the packet to this node.

READ BLOCKSIZE (;blocksize)

Returns: blocksize

This function informs ECL of the maximum blocksize the Routing Layer can handle, not including routing header. ECL should not transmit any packets larger than this size.

This value is equal to 6 less than the Segment Size (SS) SELF parameter set by network management.

3.4 Routing Layer Initialization Interface

This interface, between the Routing Layer Control Sublayer and the Routing Layer Initialization Sublayer, supports the routing events defined in Section 4.7.3. The interface consists of commands to and responses from the Routing Layer Initialization Sublayer.

TRANSMIT (adjacency, buffer)

Returns: none

This function transmits a buffer containing a packet.

CHECK TRANSMIT BUFFER (buffer; status)

Returns: buffer still queued

buffer returned to user

This function polls a buffer containing a packet that has been sent with the TRANSMIT function. If the packet has been transmitted, the buffer is returned to Routing Layer Control. If the packet has not yet been transmitted, a message is returned indicating that the buffer is queued.

STATUS (circuit; status)

Returns: off

initializing

circuit accepted by Routing Layer Initialization

running; current value of circuit cost

This function returns the status of the circuit. The off, initializing, and running states correspond to Data Link Layer states. For Ethernets, the only valid states are "off" and "running."

STATUS (adjacency; status)

Returns: unused entry

in use; node ID, node type, and corresponding circuit for that adjacency

This function returns the status of the adjacency.

REINITIALIZE (circuit)

Returns: none

This function turns the circuit off and initializes the circuit in such a manner that messages previously received in the circuit will be discarded.

SUPPLY RECEIVE BUFFER (buffer; status)

Returns: buffer accepted

buffer rejected

This function provides a receive buffer to Routing Layer Initialization so that it can receive a packet.

CHECK RECEIVE BUFFER (buffer; status, adjacency)

Returns: no packet received

packet received, buffer returned

This function polls the status of a buffer that the Routing Layer Control has just supplied with the SUPPLY RECEIVE BUFFER function. Upon receiving a packet into the buffer, the Routing Layer Initialization returns the buffer to the Routing Layer Control.

SUPPLY CIRCUIT UP COMPLETE (circuit)

Returns: none

This function informs the Routing Layer Initialization that the Decision Process recognizes that a circuit is up. (The process has completed its circuit up event algorithm.)

SUPPLY CIRCUIT DOWN COMPLETE (circuit)

Returns: none

This function informs the Routing Layer Initialization that the Decision Process recognizes that a circuit is down. (The process has completed its circuit down event algorithm.)

SUPPLY BROADCAST ADJACENCY UP COMPLETE (adjacency)

Returns: none

This function informs the Routing Layer Initialization that the Decision Process recognizes that an adjacency on a broadcast circuit is up. (The process has completed its adjacency up event algorithm.)

SUPPLY BROADCAST ADJACENCY DOWN COMPLETE (adjacency)

Returns: none

This function informs the Routing Layer Initialization that the Decision Process recognizes that an adjacency on a broadcast circuit is down. (The process has completed its adjacency down event algorithm.)

4.0 DETAILED ROUTING SPECIFICATION

The routing function consists of the following data bases and processes:

- o Routing data base
- o Forwarding data base
- o Decision Process
- o Update Process
- o Forwarding Process
- o Receive Process

4.1 Routing Parameters

The following parameters are settable via Network Management:

1. NN -- Maximum node number within the area

2. NA -- Maximum area number (For level 2 routers only)

3. NC -- Number of circuits supported by this node

4. NBRA -- Number of broadcast router adjacencies supported by this node

5. NBEA -- Number of broadcast endnode adjacencies supported by this node

6. NR -- Number of broadcast router adjacencies allowed on a given Ethernet (settable separately for each Ethernet). The sum of all the NR for circuits in Net-Management-State ON cannot exceed NBRA.

7. Maxh -- Maximum hops possible in a path to a reachable node within the area, suggested value twice the worst-case longest path length in hops.

8. Maxc -- Maximum cost possible in a path to a reachable node within the area, suggested value Maxh*Maxl.

9. AMaxh -- (level 2 routers only) -- Maximum hops possible in a path to a reachable area, suggested value twice the worst-case longest path length in hops.

10. AMaxc -- (level 2 routers only) -- Maximum cost possible in a path to a reachable area, suggested value AMaxh*Maxl.

11. Maxv -- Maximum visits for a packet before Routing assumes the packet is looping, suggested value is Maxh + k, where 1 < k <=Maxh

12. T1 -- Background frequency timer for non-broadcast circuits; maximum time period for exchanging Routing Messages with adjacent node on a non-broadcast circuit. The purpose of this timer is to recover from database corruption. Suggested value is 10 minutes.

13. BCT1 -- Background frequency timer for broadcast circuits; maximum time period between broadcasted Routing Messages on the Ethernet. The purpose of this timer is to recover from lost packets on the Ethernet. Suggested value is 10 seconds.

14. T3 -- Hello timer. Settable separately for each circuit.

The following are implementation parameters that are not settable via Network Management:

1. T2 -- Rate control frequency timer: minimum time period before another Routing Message can be sent. Suggested value is 1 second.

2. CACHETIMEOUT -- amount of time Ethernet endnodes leave a cache entry in the On-Ethernet Cache without traffic which confirms the entry's correctness before erasing the cache entry (See Ethernet Initialization Sublayer). Suggested value is 1 minute.

The following parameters are architectural constants:

1. Infh = 31.

2. Infc = 1023.

3. Maxl -- Maximum cost assignable to a circuit, 25.

4. T3MULT -- The multiple of the neighbor's Hello Timer (on a non-broadcast circuit) at which your Listen Timer should be set (i.e., T4 <-- neighbor's T3*T3MULT).

   T3MULT = 2.

5. BCT3MULT -- The multiple of the neighbor's Hello Timer (on a broadcast circuit) at which your Listen Timer should be set (i.e., T4 <-- neighbor's T3*BCT3MULT).

   BCT3MULT = 3.

6. HIORD -- The 32 bit quantity to be prefixed to the 16-bit node address to form the 48 bit Ethernet physical address.

   HIORD = AA-00-04-00.

   Therefore, if the node address is A1-A2, with A1 the least significant byte, then the formed 48 bit Ethernet physical address will be AA-00-04-00-A1-A2.

7. ALL-ROUTERS -- the multicast ID "All Routers".

   ALL-ROUTERS = AB-00-00-03-00-00.

8. ALL-ENDNODES -- the multicast ID "All Endnodes".

   ALL-ENDNODES = AB-00-00-04-00-00.

9. PROT-TYPE -- the protocol type used by Routing on the Ethernet.

   PROT-TYPE = 60-03.

10. DRDELAY -- the number of seconds an Ethernet router waits before declaring itself Designated Router

    DRDELAY = 5.

The following relationships exist between the above parameters:

1. NN >= actual maximum address within area
   NN < 1024

2. NA >= actual maximum area number (for level 2 routers only)
   NA < 64

3. Maxh >= actual maximum path length in an area
   Maxh <= 30

4. Maxc >= (actual maximum network path length) X (Maxl)
   Maxc <= 1022

5. Maxv >= actual maximum path length in the entire network
   Maxv <= 63

6. AMaxh >= actual maximum path length to any area
   AMaxh <= 30

7. AMaxc >= actual maximum path cost to any area
   AMaxc <= 1022

8. T1 >> T2

9. T3 <= 8191 (to ensure that it fits into a 16-bit word when multiplied by BCT3MULT or T3MULT)

4.2 Routing Data Base (in Level 1 And Level 2 Routers)

The routing data base contains routing data summarized in Table 1. The Network Management action of setting the Net-Management-State SELF parameter to ON from OFF initializes the data base. This is a prerequisite to normal node operation. Table 1 also shows the initial values of the data.

Table 1
Level 1 Routing Data Base

Symbol	Description	Initial value
Adj	Adjacency Data Base	empty
Circuits	Circuits Data Base	empty
Hop	Network connectivity matrix	*
Cost	Cost matrix	*
Minhop	Network minimum connectivity vector	Infh
Mincost	Minimum traffic assignment vector	Infc
Srm	Send Routing Message flags	0
Tid	Routing Layer identification	**
HomeArea	Home Area	**

* All entries in Hop are initialized to Infh, and all entries in Cost are initialized to Infc, except Hop(Tid,0) and Cost(Tid,0) are initialized to 0.

** This value is supplied as a SELF parameter by Network Management.

A description of each element of the data base follows.

Adjacency Vector. This contains information about each adjacency. An entry consists of Adj(i),

where:

Adj(i) contains the information:

1. state (unused entry, currently undergoing initialization, up)

2. nodeID -- neighbor ID

3. type -- neighbor node type -- Types are:

   - Level 1 router
   - Level 2 router
   - Phase IV endnode
   - Phase III router
   - Phase III endnode
   - No neighbor -- Adjacency is one of the broadcast circuits

4. circuit -- circuit corresponding to this adjacency in Circuits vector.

5. blocksize requested by neighbor

6. neighbor's Hello Timer (T3, if neighbor is Phase III)

7. time of last Hello heard from neighbor

8. priority -- (BRAs only), neighbor broadcast router's priority on that Ethernet

i is an integer in the range 1-<NC+NBRA+NBEA> representing an adjacency.

Note: NC counts all broadcast circuits plus neighbors on non-broadcast circuits, NBRA counts all router adjacencies on Ethernet, and NBEA counts all endnode neighbors on Ethernet.

Adjacencies 1 through NC are in one-to-one correspondence with the circuits. Adjacencies [NC+1] through [NC+NBRA] are reserved for Broadcast Router Adjacencies. (So that columns in the Hop and Cost matrices can correspond exactly with the first NC+NBRA adjacencies.)

Circuits Vector. This contains information about each circuit. An entry consists of Circuit(i),

where:

Circuit(i) Contains the information:

1. type of circuit (Ethernet, X.25, DDCMP)

2. state (e.g. off, running, initializing, etc.) as determined by Initialization Sublayer

3. cost

4. datalink blocksize (up to and including Routing envelope)

   Note that on Ethernet circuits, the datalink blocksize must be greater than or equal to BS-6 (the buffer size SELF parameter set by network management minus 6) plus the overhead in the routing envelope in long data packet format.

   On DDCMP circuits, the datalink blocksize must be greater than or equal to BS-6 plus the overhead in the routing envelope in short data packet format.

5. Hello Timer (T3)

6. last time Hello issued on this circuit

7. Recall Timer -- Amount of time Routing should wait before reinitializing the Data Link Layer

8. OPL -- Originating Packet Limit

9. counters

10. type specific information

    - For Ethernet, the type specific information is:

      1. NR -- Number of BRAs allowed on this Ethernet

      2. Priority -- the router's priority to be Designated Router

      3. Designated Router -- the ID of the router currently chosen to be Designated Router on this Ethernet

    - For X.25, the type specific information is:

      1. Port State -- supplied by the X.25 Data Link Layer

      2. Blocking -- indicates if blocking will be requested by this node

3. Negotiated Blocking -- indicates if neighbor also requested blocking

4. Maximum Recalls -- the maximum number of call attempts permitted from the Data Link start state before halting

5. Recall Count -- the number of call attempts that have been made to try to initialize a virtual circuit. This is reset by the operator turning the circuit on.

6. VC Type -- type of virtual circuit:

   . PVC (permanent virtual circuit)

   . incoming switched virtual circuit

   . outgoing switched virtual circuit

7. VC name. For an SVC, a network name and a DTE address. For a PVC, a PVC name.

i is an integer in the range 1-NC representing a circuit.

Network connectivity matrix (Hop). This contains information on the path length to each destination over each circuit or BRA. An entry consists of Hop(i,j),

where:

Hop(i,j) represents the path length from this Routing Layer to the destination, with the following values:

Value	Meaning
0	Self
1	Adjacent node
2-Maxh	Other reachable nodes
Infh	Unreachable node

i is an integer in the range 0-NN representing a destination address. Destination 0 is the special destination "nearest level 2 router".

j is an integer in the range 0-<NC+NBRA> representing self, a circuit, or an Ethernet router adjacency.

Value	Meaning
0	Self
1-NC	Circuit in Circuit(j)
[NC+1]-[NC+NBRA]	Broadcast Router Adjacency in Adj(j)

Traffic assignment matrix (Cost). This contains information on the path cost to each destination over each adjacency. This information determines which adjacency to use for traffic to a destination. An entry consists of Cost(i,j),

where:

Cost(i,j) represents the path cost from this Routing Layer to the destination, with the following values:

Value	Meaning
0	Self
1-Maxc	Path cost to other reachable nodes
Infc	Unreachable nodes

i is an integer in the range 0-NN representing a destination address.

j is an integer in the range 0-<NC+NBRA> representing self, a circuit, or a Broadcast Router Adjacency as in the Hop matrix.

Minimum network connectivity vector (Minhop). This summarizes the path length information contained in the Hop table. An entry consists of Minhop(i),

where:

Minhop(i) represents the path length via the adjacency yielding the least cost path from this Routing Layer to the destination node, with the following values:

Value	Meaning
0	Self
1	Path length to adjacent node
2-Maxh	Path length to other reachable nodes
Infh	Node unreachable

i is an integer in the range 0-NN representing a destination address.

Minimum traffic assignment vector (Mincost). This summarizes the path cost information contained in the Cost table. An entry consists of Mincost(i),

where:

Mincost(i) Represents the smallest cost from this Routing Layer to the destination, with the following values:

Value	Meaning
0	Self
1-Maxc	Smallest cost to other reachable nodes
Infc	Unreachable node

i is an integer in the range 0-NN representing a destination address.

Send Routing Message flags (Srm). The Srm flags extend permission to the Update Process to send a Routing Message about a given destination on a given circuit. The Update Process determines the actual propagation rules. An entry consists of Srm(i,j),

where:

Srm(i,j) indicates whether or not a Routing Message should be sent about destination i to circuit j.

i is an integer in the range 0-NN representing a destination

j is an integer in the range 1-NC representing the circuit on which to send the message

Routing Layer Identification (Tid). This contains the bottom ten bits of the value set by Network Management in the ID parameter of the SELF parameters.

Home Area (HomeArea). This contains the top 6 bits of the value set by Network Management in the ID parameter of the SELF parameters.

4.3 Area Routing Data Base In Level 2 Routers

The routing data base in level 2 routers contains routing data on destination areas as summarized in Table 2. This database, like the routing database described in section 4.1, is initialized by the Network Management SET NODE STATE ON function.

Table 2
Area Routing Data Base

Symbol	Description	Initial value
AHop	Area Hop Information	*
AMinhop	Area Minimum Hops	Infh
ACost	Area Cost Information	*
AMincost	Area Minimum Cost	Infc
ASrm	Area Send Routing Msg Flags	0
AttachedFlg	Other Areas Reachable Flag	False

* All entries in AHop are set to Infh, and all entries in ACost are set to Infc upon initialization, except that AHop(HomeArea,0) and ACost(HomeArea,0) are initialized to 0.

A description of each element of the area database follows.

Area Hop Information (AHop). This contains information on the number of hops to each destination area via each adjacency. An entry consists of AHop(i,j), where:

AHop(i,j) represents the number of hops to destination area i via adjacency j

i is an integer in the range 1-NA representing a destination area

j is an integer in the range 0-<NC+NBRA> representing the attached area (0), a circuit (1-NC), or a broadcast router adjacency ([NC+1]-[NC+NBRA])

Area Minimum Hops (AMinhop). This summarizes the information in AHop. An entry consists of AMinhop(i), where:

AMinhop(i) represents the path length in hops via the adjacency along the path of least cost from this node to the destination area

i is an integer in the range 1-NA representing a destination area.

Area Cost Information (ACost). This contains information on the cost to each destination area via each adjacency. An entry consists of ACost(i,j), where:

ACost(i,j) represents the path cost to destination area i via circuit or broadcast router adjacency j.

i is an integer in the range 1-NA representing a destination area

j is an integer in the range 0-<NC+NBRA> representing the attached area (0), a circuit (1-NC), or a broadcast router adjacency ([NC+1]-[NC+NBRA])

Area Minimum Cost (AMincost). This summarizes the information in ACost. An entry consists of AMincost(i), where

AMincost(i) represents the smallest cost via any adjacency from this node to the destination area

i is an integer in the range 1-NA representing a destination area.

Area Send Routing Message flags (ASrm). The ASrm flags extend permission to the Update Process to send a Level 2 Routing Message about a given destination area on a given circuit. The Update Process determines the actual propagation rules. An entry consists of ASrm(i,j), where:

ASrm(i,j) -- Indicates whether or not a Level 2 Routing Message should be sent about destination area i to circuit j.

i is an integer in the range 1-NA representing an area

j is an integer in the range 1-NC representing the circuit on which to send the message

Other Areas Reachable Flag (AttachedFlg). Indicates other areas are reachable if true.

4.4 Forwarding Data Base In Level 1 And Level 2 Routers

The information in this data base indicates whether or not a destination is reachable and, if reachable, what adjacency to use to get there. This data base consists of two vectors, as follows:

Reachability vector (Reach). This indicates whether or not the destination is reachable. An entry consists of Reach(i), where:

Reach(i) is either True (reachable) or False (unreachable)

i is an integer in the range 0-NN representing the destination node address. Destination #0 is the special destination "nearest level 2 router".

Output adjacencies (OA). This identifies the adjacency on which to forward a packet to a destination. An entry consists of OA(i), where:

OA(i) represents the adjacency to be used when forwarding a packet to destination i. OA contains one of the following values:

Value	Meaning
0	Deliver to a Routing Layer user at this node
1-<NC+NBRA+NBEA>	An adjacency on which to forward a packet

i is an integer in the range 0-NN representing a destination node address. Destination #0 is the special destination "nearest level 2 router".

4.5 Area Forwarding Data Base (level 2 Routers Only)

The information in this data base indicates whether or not a destination area is reachable, and if reachable, what adjacency to use to get there. This data base consists of two vectors, as follows:

Area Reachability Vector (AReach). This indicates whether or not the destination area is reachable. An entry consists of AReach(i), where:

AReach(i) is either True (reachable) or False (unreachable)

i is an integer in the range 1-NA representing a destination area

Area Output Adjacencies (AOA). This indentifies the adjacency on which to forward a packet to a destination area. An entry consists of AOA(i), where

AOA(i) represents the adjacency to be used when forwarding a packet to destination area i.

Value	Meaning
0	Home Area--Use level 1 Forwarding Information
1-<NC+NBRA+NBEA>	An adjacency on which to forward a packet

i is an integer in the range 1-NA representing a destination area.

4.6 Data Base Protection

A memory failure, a corrupted Routing Message, or a software error can corrupt a routing data base. Such corruptions cause a transient disruption of packet delivery. If the corruption is transient, the routing data bases stabilize to correct routes. If the corruption is continuous, the routing data bases remain in a transient incorrect state. Two conditions are necessary for self-stabilization:

1. The Routing Layer must periodically propagate Routing Messages.

2. Column 0 of the Cost and Hop matrices (in other words, the values relating to self) cannot be corrupted. For level 2 routers, column 0 of the AHop and ACost matrices similarly cannot be corrupted.

4.7 Decision Process

The Decision Process selects paths and maintains the routing and forwarding data bases. The following events serve as input to the Decision Process:

- o Operator command to set Net-Management-State to ON from OFF.
- o Adjacency down
- o Adjacency up
- o Circuit down
- o Circuit up
- o Routing Message received
- o Operator command to change circuit cost
- o Operator command to change parameters Maxh or Maxc
- o Timer expiration

The Decision Process produces the following output:

- o Modifications to the routing data base

- o Modifications to the forwarding data base

4.7.1 Decision Controller - The controller performs the following functions:

- o Performs buffer management necessary to receive Routing Messages.

- o Supports Routing Layer initialization (Section 7).

- o Checks for valid Routing Message. A valid Routing Message has:

  1. A valid checksum

  2. A valid Routing Layer control header

  If the Routing Message is not valid, then an adjacency down event on an Ethernet, or a circuit down event on a point-to-point link is generated, and the Routing Message is discarded and recorded. A valid Routing Message is processed through end of message or end of table. If the Routing Message is too long (that is, length beyond end of table), then the controller examines the overrun. The data in the overrun portion of the message must contain values corresponding to Infh in the hop fields and to Infc in the cost fields. If not, then the controller finds the highest node number for which the values are not infc/infh, and logs an event.

- o Updates the forwarding data base

4.7.2 Decision Algorithms -

The Decision Process contains an algorithm for computing the minimum cost path. It then computes the path length of that path, and sets the reachability vector, Reach, according to whether the cost and hops of the path found are within the set limits. It also sets the output adjacency vector, OA, to the adjacency which is the next hop of the minimum cost path for each destination.

The following subroutines represent the decision algorithms. They are followed by a description of the action that the decision module takes for each event received.

Subroutine: Rowmin(M, I, minimum, VECT)

```
; This routine determines the minimum for row I of
; Matrix M and stores the column number in VECT(I).

       Matrix M
       Integer I,minimum
       Vector VECT
            minimum = "big number"
            FOR j = 0 to NC+NBRA DO
            BEGIN
                 IF (M(I,j) < minimum ),
                   OR ((M(I,j) = minimum) AND
                      (ADJ(j).nodeID > ADJ(VECT(I)).nodeID))
                 THEN
                      minimum = M(I,j)
                      VECT(I) = j
                 ENDIF
            END
```

Subroutine: Minimize(I,M,V,Pl,P2,VECT)

```
; This routine determines entries for vector V,
; containing the minimum of each row of matrix M,
; and passes to Rowmin the vector VECT in which to store the
; resulting output adjacency number.

     Integer I
     Matrix M
     Vector V
     Parameters Pl, P2
     Vector VECT
          Rowmin(M,I,minimum,VECT)
          IF (minimum > Pl) THEN minimum = P2
          V(I) = minimum
          ENDIF
```

Subroutine: Routes(FirstDest, LastDest)

```
; This routine determines the reachability and output adjacency
; for each destination in the range FirstDest to LastDest
; within the area, with destination #0 the nearest level 2 router.

  INTEGER OLD_HOP, OLD_COST
     FOR i = FirstDest to LastDest
     DO
        OLD_HOP = MINHOP (i)
        OLD_COST = MINCOST (i)
        Minimize (row i, Cost, Mincost, Maxc, Infc, OA)
        Col = OA(i)
        Minhop(i) = Hop(i,Col)
        IF Minhop(i) > Maxh THEN Set Minhop(i) = Infh
      ;now set OA to adjacency rather than column
        IF Col <= NC AND Circuit(Col).Type=Ethernet THEN ;need
                                                 ;to convert to BEA
           Search for BEA with node ID i
           set OA(i) to that BEA
        IF (Minhop(i) = Infh OR Mincost(i) = Infc)
          THEN
            BEGIN
            Reach(i) = False
            Minhop(i) = Infh
            Mincost(i) = Infc
            END
          ELSE Reach(i) = True
        ENDIF
        IF(MINHOP(i) < > OLDHOP OR MINCOST(i) < > OLDCOST
              THEN for each k from 1 to NC
                Set Srm(i,k)
              ENDIF
   ENDDO
```

```
Subroutine:ARoutes(FirstArea, LastArea)

; This routine determines the reachability and output adjacency
; for each area in the range FirstArea to lastArea.

  Integer OLD_HOP, OLD_COST

   FOR i = FirstArea to LastArea
   DO
        OLD_HOP = AMinhop (i)
        OLD_COST = AMincost (i)
        Minimize (row i, ACost, AMincost, AMaxc, Infc, AOA)
        Col = AOA(i)
        AMinhop(i) = AHop(i,Col)
        IF AMinhop(i) > AMaxh THEN
          Set AMinhop(i) = Infh
   ;note in this case Col is the Adjacency, since a BEA cannot
   ;be a path to a different area
        IF (AMinhop(i) = Infh OR AMincost (i) = Infc)
          THEN
            BEGIN
            AReach(i) = False
            AMinhop(i) = Infh
            AMincost(i) = Infc
            END
          ELSE AReach(i) = True
        ENDIF
        IF (AMinhop(i) <> OLD_HOP OR AMincost(i) <> OLD_COST
             THEN FOR j = 1 to NC
                   IF Adj(j).Type=level 2 router
                     OR Circuit(j).Type=Ethernet
                       THEN Set ASrm(i,j)
                   ENDIF
   ENDDO
   ;set value of AttachedFlg and Hop(0,0) and Cost(0,0)
   AttachedFlg=False
   Hop(0,0)=Infh
   Cost(0,0)=Infc
   FOR i = 1 to NA
     IF AReach(i)=True AND i < > HomeArea THEN
        Hop(0,0)=0
        Cost(0,0)=0
        AttachedFlg=True
     ENDIF
   ENDFOR
   CALL Routes(0,0)   ;to calculate for nearest attached
                      ;level 2 router
```

Subroutine: Check

```
; This routine detects any corruption of column 0 in the
; Hop, Cost, AHop and ACost matrices.

     BEGIN
        ;first check column 0 of Hop and Cost
        Check that Hop (Tid,0) and Cost (Tid,0) = 0
        IF a level 2 router AND AttachedFlg = True
          Check that Hop(0,0) and Cost (0,0) = 0
        IF a level 2 router AND AttachedFlg = False
          Check that Hop(0,0) = Infh and Cost (0,0) = Infc
        FOR i = 0 to NN
          UNLESS (i = Tid OR (i = 0 AND node is level 2 router))
             Check that Hop (i,0) = Infh
             AND Cost (i,0) = Infc

        ;now check column 0 of AHop and ACost
        IF a level 2 router
            FOR I = 1 to NA
              IF HomeArea = i
                 Check that AHop (i,0) = 0 and ACost (i,0) = 0
              ELSE
                 Check that AHop(i,0) = Infh and ACost(i,0) = Infc
        ENDIF
        IF any check fails, then
               Terminate the Routing Layer
        ENDIF
        IF both checks are successful, then
               EXIT
        ENDIF
     END
```

4.7.3 Decision Process Events And Actions -

A. Operator command to set Net-Management-State to ON from OFF.

1. Tell Ethernet Data Link the physical address consisting of B1-B2-B3-B4-B5-B6, where B1-B2-B3-B4 are the four bytes of HIORD, with B1 the most significant byte, and B5-B6 is ID, with B5 the least significant byte of ID.

2. Initialize Routing databases

3. Set Hop(Tid,0)=0 and Cost(Tid,0)=0

4. IF a level 2 router, set AHop(HomeArea)=0, ACost(HomeArea)=0

5. Call Routes (0,NN)

6. IF a level 2 router, call ARoutes(1,NA)

B. Broadcast adjacency j down (note that if j <= NC, the event is a circuit down event)

1. IF (NC+1) <= j <= (NC+NBRA), then ;adjacency j is a BRA

a. Set each entry in column "j" of Hop matrix to Infh. IF a level 2 router, set each entry in column "j" of AHop matrix to Infh.

b. Set each entry in column "j" of Cost matrix to Infc. IF a level 2 router, set each entry in column "j" of ACost matrix to Infc.

c. IF local node type is level 2, and node type of adjacency j is level 2, call ARoutes(1,NA)

d. Call Routes(0,NN)

2. IF NC+NBRA < j, THEN ;adjacency j is a BEA

```
    NODEID <-- Adj(j).NodeID
    k <-- circuit pointed to in Adj(j)
    Set Hop(NODEID,k) = Infh
    Set Cost(NODEID,k) = Infc
    Call Routes(NODEID,NODEID)
```

3. Supply "adjacency down complete" to Initialization Sublayer

C. Broadcast adjacency j up

1. IF (NC+1) <= j <= (NC+NBRA) ;j is BRA

```
    CIRC <= Adj(j).Circuit
    FOR i = 0 to NN, Set Srm(i,CIRC)
    IF local node is a level 2 router, AND
    Adj(j).Type=level 2
        FOR i = 1 to NA, Set ASrm(i,CIRC)
```

2. IF NC+NBRA < j, THEN ;j is BEA

```
    NODEID <-- Adj(j).NodeID
    k <-- Adj(j).Circuit
    Set Hop(NODEID,k) = 1
    Set Cost(NODEID,k) = cost of Circuit(k)
    Call Routes(NODEID,NODEID)
```

3. Supply "adjacency up complete" to Initialization Sublayer

D. Circuit j down

1. Call Check.

2. Set each entry in column "j" of Hop matrix to Infh. IF a level 2 router, set each entry in column "j" of AHop matrix to Infh.

3. Set each entry in column "j" of Cost matrix to Infc. IF a level 2 router, set each entry in column "j" of ACost matrix to Infc.

4. FOR each adjacency k pointing to circuit j
    Declare broadcast adjacency k down

5. IF local node type is level 2 router Call ARoutes(1,NA).

6. Call Routes(0,NN).

7. Supply "circuit down complete" to Initialization Sublayer.

E. Circuit j up

Case 1: Circuit j is a non-broadcast circuit with k=node number as determined by Initialization Sublayer

1. Call Check.

2. IF Adj(j).Type=endnode, THEN

    a. Hop(k,j)=1

    b. IF Circuit(j).Cost is not > 0 THEN the Routing Layer terminates.

    c. Cost(k,j)=Circuit(j).Cost

    d. Call Routes(k,k)

3. Set Srm(i,j) FOR i = 0 to NN

4. IF local node type is level 2 router, AND Adj(j).Type=level 2, set ASrm(i,j) FOR i = 1 to NA

5. Supply "circuit up complete" to Initialization Sublayer.

Case 2: Circuit j is a broadcast circuit

1. Call Check

2. IF Circuit(j).Cost is not > 0 THEN the Routing Layer terminates.

3. FOR i = 0 to NN, set Srm (i,j)

4. IF local node type is level 2 router, set ASrm(i,j) FOR i = 1 to NA

5. Supply "circuit up complete" to Initialization Sublayer.

F. Level 1 Routing Message received on adjacency j, 1 <= j <= (NC+NBRA). The Routing Message contains destinations in set S.

1. Call Check.

2. Copy hop subfield of Routing Message for each i in set S, onto row i, column "j" of Hop matrix.

3. Add 1 to Hop (i,j) for each i in set S.

4. Copy cost subfield of Routing Message for each i in set S, onto row i, column "j" of Cost matrix.

5. IF Cost of Circuit pointed to by adjacency j is not > 0, THEN the Routing Layer terminates. Otherwise, add that cost to Cost (i,j) for each i in set S.

6. FOR each i in set S, Call Routes(i,i).

G. Level 2 Routing Message received on adjacency j (level 2 routers only). The Routing Message contains areas in set S.

1. Call Check.

2. Copy hop subfield of Routing Message, for each i in set S, onto row i, column "j" of AHop matrix.

3. Add 1 to AHop (i,j) for each i in set S.

4. Copy cost subfield of Routing Message, for each i in set S, onto row i, column "j" of ACost matrix.

5. IF Cost of Circuit pointed to by adjacency j is not greater than 0, then the Routing Layer terminates. Otherwise, add that cost to Cost (i,j) for each i in set S.

6. FOR each i in set S, Call ARoutes(i,i).

H. Circuit j cost change

1. Call Check.

2. Calculate the difference between the new cost and the old cost for circuit j. Note that the new cost and the old cost must both be greater than 0, otherwise the Routing Layer terminates.

3. Add this difference to each entry in column "j" of Cost matrix.

4. IF circuit type is broadcast, add this difference to each entry in each column k of the Cost matrix, where k is a broadcast router adjacency with circuit=j

5. Call Routes(0,NN).

6. IF local node is level 2 router, add this difference to each entry in column "j" of the ACost matrix.

7. IF local node is level 2 router and the circuit type is broadcast, add this difference to each entry of the ACost matrix for each column k, where k is a broadcast router adjacency with circuit=j

8. IF local node is level 2 router, call ARoutes(1,NA).

I. Maxh, Maxc, AMaxh, or AMaxc change

1. Call Check.

2. Call Routes(0,NN)

3. IF local node type is level 2 router, Call ARoutes(1,NA)

J. T1 Timer expires

1. Call Check.

2. FOR j = 1 to NC

```
a.  IF Circuit(j).Type=nonbroadcast, AND
    Adj(j).Type=router (level 1, level 2, or  Phase  III)
    THEN
        FOR i = 0 to NN, Set Srm(i,j)

b.  IF local node is level 2 router AND
    Adj(j).Type=level 2, AND
    Adj(j).Circuit.Type=nonbroadcast THEN
        FOR i = 1 to NA, Set ASrm(i,j)
```

3. Call Routes(0,NN).

4. IF local node is level 2, Call ARoutes(1,NA)

K. BCT1 Timer expires

1. Call Check.

2. FOR j = 1 to NC

a. IF Circuit(j).Type=broadcast, THEN
FOR i = 0 to NN, Set Srm(i,j)

b. IF local node is level 2 router AND
IF Circuit(j).Type=broadcast, THEN
FOR i = 1 to NA, Set ASrm(i,j)

4.8 Update Process

The Update Process propagates Routing Messages and determines their content. It consists of an algorithm and a format module. The Update Process accepts the following as input:

- The minimum hop vector, Minhop
- The minimum cost vector, Mincost
- The area minimum hop vector, AMinhop
- The area minimum cost vector, AMincost
- The Send Routing Message flags, Srm
- The Area Send Routing Message flags, ASrm

The Update Process produces a Routing Message for an adjacent node as output.

4.8.1 Update Algorithm - A Level 1 Routing Message containing a given set of destinations is sent on a circuit j when a buffer is available from the quota given to the Routing Process for Update use, AND at least T2 has elapsed since the last transmission of a Routing Message containing the same destinations on circuit j, AND some Srm(i,j) is set, for at least one i in that set of destinations.

A Level 2 Routing Message is sent on a circuit j when a buffer is available from the quota given to the Routing Process for Update use, AND at least T2 has elapsed since the last transmission of a Routing Message containing those areas on circuit j, AND some ASrm(i,j) is set, for at least one i in that set of areas.

Any algorithm that sends, as a minimum, all information flagged by the Srm and ASrm flags can be used. (An algorithm may send more data than this to simplify the algorithm or reduce the data storage for Srm flags.)

When the circuit is of type Ethernet, the Routing Message is sent to the multicast ID "all routers".

Care must be taken that no nodes, areas, or circuits are starved. For example, an algorithm that does not scan through all circuits and destination (nodes or areas) before restarting, might never reach some circuit or destination.

Care must also be taken that systematic ordering of the transmitted segments does not cause the same segments to always be lost. For example, on an Ethernet, if all segments are sent, one after another, in rapid succession, the last segments might be lost with high probability. Thus, on Ethernets, each time a periodic complete Routing Message is sent, the segments should be cycled through, so that each segment gets a chance to be the first segment.

The T2 timer is intended to be restarted on a circuit after all destinations with Srm or ASrm flags set during one pass through are transmitted on that circuit.

On a point-to-point circuit, a Routing Message may be as long as the neighbor's blocksize. On an Ethernet circuit, a Routing Message may be as long as the minimum blocksize of all BRAs on that circuit.

4.9 Forwarding Process

The Forwarding Process supplies and manages the buffers necessary for route-through. Packets are discarded if buffer thresholds are exceeded.

Packet formats, and names of fields, are described in Section 10.

Data packets can be in one of three formats:

1. long format

2. short format

3. Phase III format

Packets in long format have bit 2 of the FLAGS byte=1. Packets in short format or Phase III format have bit 2 of the FLAGS byte=0. Phase IV nodes are required to be able to receive short format on point-to-point circuits, and long format on Ethernet circuits. They should transmit short format on point-to-point circuits, and long format on Ethernet circuits. They must receive and transmit Phase III format to Phase III adjacencies.

Packets in all three formats have bit 6 ("Future version")=0. Discard but do not log packets received with bit6 = 1.

Discard and log (as "message format error") short packets (in other words, packets with less than a packet route header) and packets with an invalid route header (including packets in short format received on an Ethernet circuit, if this node cannot handle such packets, or packets received in long format on a point-to-point circuit, if this node cannot handle such packets).

If the incoming adjacency is a Phase III endnode or router, if the top 6 bits of Destination ID=0, fill in "HomeArea" in the top 6 bits of Destination ID. If the top 6 bits of Source ID are 0, fill in "HomeArea" in the Source ID field.

If the source and destination are attached Ethernet endnodes, on the same Ethernet, set the intra-Ethernet bit.

If the destination is in a foreign area, and the local node type is a level 1 router, or the local node type is a level 2 router and AttachedFlg=False, then forward based on OA(0).

If the destination is in a foreign area, and the local node type is a level 2 router and AttachedFlg=True, then forward based on AOA(foreign area).

If the adjacency the packet is to be forwarded on has node type Phase III router or Phase III endnode, clear the area field in the Destination ID. If the area in Source ID is "HomeArea", clear that as well. If the area in Source ID is not "HomeArea", and the Destination is the next node, and the Destination's node type is "Phase III router" or "Phase III endnode", do not forward the packet. Instead drop the packet or return it to sender, as with an unreachable destination.

If the destination is unreachable, and "return to sender requested" is set, set "return to sender", clear "return to sender requested", switch the destination and source ID fields, and forward the packet towards the new destination (the previous source). Otherwise, drop the packet.

If the circuit the packet is to be forwarded on is an Ethernet, and the packet is in short format, then reformat the packet to long format. If the "intra-Ethernet bit" is set, clear the bit unless the packet is to be forwarded onto the same circuit from which it was received.

When originating a packet on an Ethernet, always set the intra-Ethernet bit.

If the circuit the packet is to be forwarded on is point-to-point, and the packet is in long format, check that the first 4 bytes of D-ID and S-ID are set to HIORD. Drop the packet and log (as "message format error") if they are not. Reformat the packet to short format.

4.10 Receive Process

The Receive Process receives packets from the Data Link Layer. It then passes the packet to the appropriate process, as follows:

Packet Type	Process
Routing Message	Decision Process
Hello Message	Node Listener Process
Packet for Self	End Communications Layer
Packet for other destination	Forwarding Process

4.11 Loop Detector Process

The loop detector limits the number of nodes that a packet can visit. It increments the node visit field in the packet route header by one. The loop detector discards the packet if this number exceeds the maximum node visit limit, Maxv. Note that the parameter Maxv must always be greater than or equal to the parameter Maxh.

The algorithm the loop detector executes when it receives a packet is the following:
Add 1 to node visit field in packet route header.

```
IF ((node visit > Maxv) and ("return to sender" is not set))
   If "return to sender requested" is set,
        set "return to sender"
        clear "return to sender requested"
        reverse source and destination
        forward the packet towards the (new) destination.
   Else drop the packet and record
ENDIF

IF ((node visit > 2 * Maxv) and ("return to sender" is set))
   Discard packet and record
ENDIF
```

5.0 DETAILED CONGESTION CONTROL SPECIFICATION

The transmit management subroutine handles congestion control. Transmit management consists of the following components:

- o Square root limiter. Reduces buffer occupancy time per packet by using a square root limiter algorithm (Appendix F). The square root limiter also queues packets for an output circuit, and prevents buffer deadlock by discarding packets when the buffer pool is exhausted. Section 5.1 specifies the Square Root Limiter Process.

- o Originating packet limiter. Limits originating packet traffic when necessary to ensure that transit packets are not rejected. An originating packet is a packet from the ECL at this node. A transit packet is a packet from another node to be routed through to another destination node.

- o Flusher. Flushes packets queued for an adjacency that has gone down.

- o Packet size checker. Resolves differences between packet size and Data Link receive blocksize.

5.1 Square Root Limiter

The square root limiter discards a transit packet or rejects an originating packet when the output circuit queue exceeds the discard threshold, Ud. Ud is given as follows:

Ud = CEILING (NB/ SQRT(NC))

where:

NB = Number of Routing Layer buffers for all output circuits.

NC = Number of active output circuits.

5.2 Originating Packet Limiter

The originating packet limiter first distinguishes between originating packets and transit packets. It then imposes a limit on the number of buffers that originating packets can occupy on a per circuit basis. In times of heavy load, originating packets may be rejected while transit packets continue to be routed. This is done because originating packets have a relatively short wait, whereas transit packets, if rejected, have a long wait -- a retransmission period.

The originating packet limiter accepts as input:

- o A packet received from End Communications Layer
- o A transmit complete from the Data Link Layer for an End Communications Layer packet

The originating packet limiter produces the following as output:

- o Packet accepted
- o Packet rejected
- o Modifications to originating packet counter

There is a counter, N, and an originating packet limit, OPL, for each active output circuit. Each N is initialized to 0. Each OPL is initialized to the number of buffers necessary to prevent the circuit from idling.

5.3 Flusher

The flusher ensures that no packet is queued on a circuit whose state is not RUN, or on a nonexistent adjacency.

5.4 Packet Size Checker

The packet size checker checks the size of each packet that it is about to queue on an output adjacency. This includes packets from both the End Communications Layer at this node and transit packets. When the packet size exceeds the Data Link receive blocksize (established during Routing Layer initialization), the packet size checker discards the packet and records the event. Endnodes on a broadcast circuit are not required to have a packet size checker.

6.0 ROUTING LAYER INITIALIZATION SUBLAYER

The Initialization Sublayer masks the characteristics of the different kinds of Data Link Layers from the Control Sublayer. The only two types of circuits the Control Sublayer sees is broadcast or non-broadcast.

In Phase IV, the only type of supported broadcast circuit is the Ethernet. The only types of supported non-broadcast circuits are DDCMP point-to-point, DDCMP multipoint, and X.25.

6.1 Version Skew

In initialization with an adjacent node, a node must drop and ignore Initialization Messages with higher version numbers. The version number check must be done before looking at any of the other fields in the Initialization Message, since higher version Initialization Messages might not be compatible with lower version messages.

It is the responsibility of the node with the higher version number to note that it has a neighbor with a lower version number, and if the node with the higher version number knows about the earlier protocol, that node must start sending Initialization Messages with the lower version number. If the node with the higher version number does not know about the earlier protocol, an initialization failure, version skew event is logged.

In comparing version numbers, the version number byte is the only byte compared. The ECO number byte and the User ECO number byte are not accessed during the comparison.

7.0 INITIALIZATION SUBLAYER: DDCMP OR X.25

The Routing Layer initialization is a start-up procedure between two adjacent nodes. The procedure involves exchanging Routing Layer Initialization Messages and possibly Routing Layer Verification Messages. This exchange identifies the nodes to each other and provides additional node information. This section describes:

- o The Routing Layer Initialization circuit states
- o The Routing Layer Initialization circuit events
- o The Routing Layer Initialization operation and message requirements
- o The Routing Layer Initialization state table and diagram

7.1 Node Listener Process

The Node Listener Process detects node failures. The process consists of a controller and an algorithm module. The Node Listener accepts the following as input:

- o Hello Message received
- o Any message received on a non-broadcast circuit
- o Timer pulse received

The Node Listener Process produces the following as output:

- o Modifications to the adjacency data base

7.1.1 Node Listener Controller - The Node Listener Controller manages the buffers necessary for receiving Hello Messages. It also checks for valid Hello Messages. A valid Hello Message contains a valid Routing Layer control header and valid data. If the Hello Message is not valid, then a circuit down event is generated, and the Hello Message is recorded and discarded.

7.1.2 Node Listener Algorithm - The Node Listener algorithm determines when the adjacent node is no longer talking. Such a node is considered down. Consequently, the circuit is reinitialized. The following describes the algorithm for handling each Node Listener event.

A. Hello Message or any message received on adjacency j

   1. Reset T4 by:

```
Set TEMP = T3
If Adj(j).Type=Phase IV router or endnode
    Set TEMP = neighbor's Hello Timer
Endif

Set T4 = TEMP*T3MULT
```

   2. If received message is Hello, check that TEST DATA is all octal 252

B. Timer pulse

   1. Decrement T4

   2.

```
IF (T4 <= 0), then
    Reset T4
    Set circuit state to CR ("circuit rejected.")
ENDIF
```

7.2 Node Talker Process

The Node Talker propagates Hello Messages. It sends a Hello Message to an adjacent node when:

- o T3 has elapsed since the last transmission of any message.
- o A circuit has come up.

The Node Talker can be interlocked with the Decision and Update Processes. When either Decision or Update fails, then the Node Talker ceases.

7.3 Routing Layer Initialization Circuit States

The Routing Layer Initialization circuit states are:

(Symbol) State	Description
(RU) RUN	The Routing Layer can use the circuit to transmit packets between two nodes.
(CR) CIRCUIT REJECTED	The circuit is degraded. To avoid excessive packet delay the circuit will be declared down. The Routing Decision Process has not yet processed a circuit down event.
(DS) DATA LINK START	The circuit is undergoing Data Link Layer initialization.
(RI) ROUTING LAYER INITIALIZE	The circuit has successfully undergone Data Link initialization and is waiting to receive a Routing Layer Initialization Message.
(RV) ROUTING LAYER VERIFY	A valid Routing Layer Initialization Message has been received for this circuit and the circuit requires verification.
(RC) ROUTING LAYER COMPLETE	The Routing Layer has completed a valid exchange of Routing Layer Initialization and possibly Routing Layer Verification Messages.
(OF) OFF	The Routing Layer cannot use the circuit. The Routing Decision Process has not yet processed a circuit down event.
(HA) HALT	The Routing Layer cannot use the circuit. A circuit down event is required.

7.4 Routing Layer Initialization Circuit Events

The Routing Layer Initialization circuit events are as follows:

(Symbol)	Description
(nri)	The Routing Layer received a valid new Routing Layer Initialization Message.
(nrv)	The Routing Layer received a valid new Routing Layer Verification Message.
(rt)	The Routing Layer timed out.
(sc)	The Routing Layer received a start complete notification (in other words, a transition from the initializing state to the running state) from the Data Link Layer.
(ste)	The Routing Layer received a start notification (in other words, a transition from any state to the stop state) or threshold error notification from the Data Link Layer. In the case of X.25, a start notification is given by the Data Link Layer upon receipt of a "Clear Indication" or "Reset" packet, or when a data error is observed.
(opo)	Operator turned circuit on.
(opf)	Operator turned circuit off.
(im)	The Routing Layer received an invalid Routing Layer Initialization Message or an unexpected message.
(rc)	The Routing Layer received a reject complete from the circuit rejection component of the circuit monitor.
(cdc)	The Routing Layer Initialization received a circuit down complete event from the Decision Process in the Routing Layer Control Sublayer.
(cuc)	The Routing Layer Initialization received a circuit up complete event from the Decision Process in the Routing Layer Control Sublayer.

When the Data Link Layer has initialized, a timer starts. If the timer expires before the circuit accepted state is reached, then the circuit is reinitialized. If the timer expires after the circuit accepted state is reached, then the timer is ignored.

7.5 Routing Layer Initialization Operation And Message Requirements

The actions required of the Routing Layer Initialization in the "Routing Layer Initialization State Table" following are:

1. Issue reinitialize command to the Data Link Layer and start Recall Timer.

2. Issue stop to the Data Link Layer.

3. Send a valid Routing Layer Initialization Message.

4. Send a valid Routing Layer Verification Message.

A valid Routing Layer Initialization Message has the following characteristics:

- o A valid Routing Layer control header with node address less than or equal to this node's NN

- o If this node is a level 1 router, then the neighbor's home area must match HomeArea, unless the neighbor is Phase III

- o If this node is a level 2 router, then the neighbor's home area must match HomeArea unless the neighbor node type is level 2 router or Phase III

- o A received Data Link blocksize greater than or equal to the maximum (Routing Message size of Routing Message containing NN nodes if neighbor is Phase III router, Hello Message size, 246)

- o An acceptable routing version (see section on Version Skew)

A valid Routing Layer Verification Message has a value that agrees with the function value.

7.6 Routing Layer Initialization State Table And Diagram

The following table shows all the possible state transitions from the Routing Layer's viewpoint at a single node. It also shows the events that cause the state changes and the actions the Routing Layer Initialization takes, if any, upon the occurrence of an event. The numbers in the "actions" column correspond to those in the list of actions above.

Routing Layer Initialization State Table

This table shows each possible new state and action relating to the occurrence of each event in each state. The actions are shown by a slash (/) followed by the number of the action. A dash (-) signifies no action. The actions numbers are defined above.

	Old State							
Event	RU	CR	DS	RI	RV	RC	OF	HA
nri	CR/-	CR/-	DS/-	*	DS/1	DS/1	OF/-	HA/-
nrv	CR/-	CR/-	DS/-	DS/1	RC/-	DS/1	OF/-	HA/-
rt	RU/-	CR/-	DS/1	DS/1	DS/1	RC/-	OF/-	HA/-
sc	CR/-	CR/-	RI/3	DS/1	DS/1	DS/1	OF/-	HA/-
ste	CR/-	CR/-	DS/1	DS/1	DS/1	DS/1	OF/-	HA/-
opo	RU/-	CR/-	DS/-	RI/-	RV/-	RC/-	CR/-	DS/1
opf	OF/2	OF/-	HA/2	HA/2	HA/2	HA/2	OF/-	HA/-
im	CR/-	CR/-	DS/-	DS/1	DS/1	DS/1	OF/-	HA/-
rc	CR/-	CR/-	DS/-	RI/-	RV/-	DS/1	OF/-	HA/-
cdc	RU/-	DS/1	DS/-	RI/-	RV/-	RC/-	HA/-	HA/-
cuc	RU/-	CR/-	DS/-	RI/-	RV/-	RU/-	OF/-	HA/-

* NOTE

There are four possible new state/action sets for this transition, as follows:

1. Action: 4; New state: RV; Verification requested in received message; verification required by this node.

2. Action: 4; New state: RC; Verification requested in received message; verification not required by this node.

3. Action: -; New state: RV; Verification not requested in received message; verification required by this node.

4. Action: -; New state: RC; Verification not requested in received message; verification not required by this node.

The Routing Decision Process generates circuit down events in the states CR and OF. It generates a circuit up event in the state RC.

Once the Recall Timer is set, it must expire before another reinitialize command is given to the Data Link Layer.

The following figure shows the Routing Layer state transitions.

```
           .----------------------------------------------------------.
           |                                                          |
           v                                                          |
      .---------.          .----------.            .-----------.      |
      |         |--------->|          |----------->|           |      |
   .->|   RU    |    .---->|    CR    |   .------->|    DS     |<--.  |
   |  |         |    |     |          |   |   .--->|           |   |  |
   |  `---------'    |     `----------'   |   |    `-----------'   |  |
   |       |         |                    |   |          A         |  |
   |       |         |                    |   |          |         |  |
   |       |  .------'            .-------'   |          |         |  |
   |       |  |                   |           |          |         |  |
   |       v  v                   v           |          v         |  |
   |  .---------.          .----------.       |    .-----------.   |  |
   |  |         |          |          |       |    |           |   |  |
   |  |   OF    |--------->|    HA    |<------|----|    RI     |---|-'
   |  |         |          |          |       | .-|            |   |
   |  `---------'          `----------'       | |  `-----------'   |
   |       ^                     A            | |        |         |
   |       |                     |            | |        |         |
   |       |                     |            | |        |         |
   |       |                     |            | |        v         |
   |       |               .----------.       | |  .-----------.   |
   |       ----------------|          |-------' |  |           |   |
   `-----------------------|    RC    |<--------'  |    RV     |___'
                           |          |<-----------|           |
                           `----------'            `-----------'
```

Legend:

```
   .----.
   |    |   contains symbol representing Routing Initialization
   |    |   state
   `----'
```

Note: These state transitions are not guaranteed.

7.7 Closing Down

If at all possible, before a router is brought down, it should reinitialize the Data Link Layer, causing its neighbor to bring down the circuit without unnecessary delay. In the case of DDCMP, the Routing Layer should command DDCMP to put out a DDCMP Start. In the case of an X.25 SVC, the Routing Layer should command X.25 to do a Clear of the circuit. In the case of an X.25 PVC, the Routing Layer should command X.25 to do a Reset of the circuit.

8.0 ADDITIONAL INITIALIZATION SUBLAYER FOR X.25

8.1 Incoming Call Control

If the circuit parameter VC Name is set by network management for a circuit of type incoming SVC, then Routing should reject an incoming call unless the DTE address and network name match those in VC Name.

8.2 Error Control

Error Control is introduced to detect possible corruption of data within the public data net. A 16 bit cyclic redundancy check word based on the CRC-16 polynomial is prefixed to each Routing Layer datagram. The check character is generated on the message as a string of bits beginning with the LSB of the message ending with the MSB of the message.

(Note- A Routing Layer datagram may traverse the X.25 net as a number of X.25 packets as described below.) Received datagrams are checked for data errors. If an error is found, it is accounted for, by incrementing the error counter in the Circuit database, the datagram is discarded, and the event is treated as an "ste" event by the Initialization Layer. This will cause the circuit to be reinitialized.

8.3 Fragmentation, Assembly And Blocking

Fragmentation of Routing Layer datagrams may be required if the Public Data network requires a smaller packet size than that of DECnet. This poses a twofold problem.

1. The Routing Layer datagram must be fragmented into multiple packets before transmission and reassembled on receipt.

2. Transmitting the Routing Layer datagram in multiple packets may require additional buffer capacity depending on the characteristics of the lower level implementation. This increase in the buffer requirement is due to the 5 byte X.25 packet header.

The buffer problem above is not solved here as it is implementation dependent. A particular implementation may solve this, for example, by dedicating a fixed number of fragmentation/assembly buffers per virtual circuit or via hardware capable of supplying packet headers from separate buffers.

The maximum packet size supported on a particular virtual circuit will be set via Network Management. The actual packet size will be negotiated when the circuit is established and will be stored in the circuit data base. Fragments of datagrams will be transmitted across the data link interface sequentially. All but the last packet will specify "more data to follow" in the TRANSMIT call. Similarly, receive packets will be checked for "more data to follow" and assembled into Routing Layer datagrams before being forwarded to Error Control.

In some instances, blocking of multiple Routing datagrams within a single X.25 packet may be motivated by vendor facilities and tariffs. When X.25 vendors support large packet sizes, improvements in cost and responsiveness may be obtained by blocking. (Blocking is also somewhat motivated by the presence of ECL control messages.)

Blocking is enabled or disabled via the Network Management interface. It may be desirable to disable blocking when operating at small packet sizes. Negotiation is performed at Routing Layer initialization to determine whether blocking will be used on a given virtual circuit. When blocking has been enabled by network management, the "blocking requested" bit in the "TIINFO" field of the Routing Layer Initialization Message is set.

Blocking is used if both sides have requested it.

When blocking is in use, Routing Layer datagrams are viewed as a serial stream of data (fragmentation procedures discussed above are not used). A header is prefixed to the datagram (including the CRC generated by the error control function) and it is placed in an X.25 packet assembly buffer. Datagrams are loaded into packet assembly buffers until full. In general filling assembly buffers will require fragmenting datagrams. The figure below shows the mapping of the Routing Layer data stream onto the X.25 data stream.

```
Routing Layer datagrams                                    data stream
                                                            ---------->
      7          6    5                    4    3          2    1
------X-----I-----X-----X-----I----------I----X-----X-----I-----X----X
            5                 4          3                2          1
X.25
                    X  Frames Routing Layer datagrams
                    I  Frames X.25 packets
```

The header that is prefixed to the datagram is a two byte field which is a count of the number of bytes used to frame the datagram.

The message formats for datagrams are illustrated below for the two cases of blocking not enabled and blocking enabled.

```
+-----------------------------------------+
|    CRC    | DG                          |
+-----------------------------------------+
No Blocking
```

```
+-------------------------------------------------------------+
| BC |CRC |      DG      | BC |CRC |      DG      | BC |CRC | DG |
+-------------------------------------------------------------+
Blocking multiple datagrams into X.25 packets
```

BC - 16 bit byte count of Routing Layer datagram and CRC

DG - Routing Layer datagram

In both cases fragmentation may occur at any point in the message.

Efficient implementation of a blocking facility requires a buffer management policy sensitive to the blocking operation. A specific policy is not required by the architecture as it will not impact correct operation. Under light load conditions, a decision must be made as to whether to pass a partially filled packet across the X.25 packet level interface or to wait in anticipation of additional data to fill the packet. Policies may require a timer to perform this function. However, the policy recommended here is independent of a timer. The policy is basically to buffer as few X.25 packets as possible in X.25 level 3 (one or two). The Routing Layer supplies packets to level 3 only as the short queue permits, thereby retaining the data as long as possible without sacrificing efficiency. This policy will have the property of being cost efficient under moderate to heavy loading and always being responsive.

8.4 Closing Down

It is costly to keep attempting to bring an X.25 circuit up if it is not usable. Thus a retry counter is maintained for Routing Layer time-outs (rt Initialization circuit event) and X.25 Data Link Layer Start Notifications (ste Initialization circuit event). When in state DS, if actions rt or ste occur MAXIMUM RECALLS in a row, the transition is made to the OF state, requiring operator intervention to bring the circuit back up.

9.0 INITIALIZATION SUBLAYER: ETHERNET

9.1 Routers

9.1.1 Ethernet Router Hello Messages -

Routers broadcast Ethernet Router Hello Messages to the multicast ID "all routers". These contain the transmitting router's ID, T3, and PRIORITY, plus a list containing information about the other routers the transmitting router has heard from "recently enough" (described below), and provided that the number of routers does not exceed NR. Each entry in the list contains a router's ID, that router's PRIORITY, and a bit indicating whether that router includes the transmitting router in its Ethernet Router Hello Messages.

When a new router NEWROUTER is heard from, and NR and NBRA are not exceeded, NEWROUTER is included in future Ethernet Router Hello Messages by this node. An adjacency slot is set aside for NEWROUTER, on an adjacency numbered between [NC+1] and [NC+NBRA]. The state of the adjacency is set to "initializing", until it is known that the communication between this node and NEWROUTER is two-way.

When NEWROUTER reports this node's ID in its Ethernet Router Hello Messages, the adjacency's state is changed to "up", and an adjacency up event is generated.

A separate Listen Timer is kept for each Ethernet neighbor. The value of the Listen Timer is BCT3MULT * neighbor's T3 as reported in its Hello Messages in the TIMER field. If a neighbor is not heard from in that time, it is purged from the database and an adjacency down event is generated.

If a Hello is received from neighbor OLDROUTER, and this node's ID is no longer in OLDROUTER's Hello Message, the adjacency's state is changed to "initializing", and an adjacency down event is generated.

If a router already has heard from NR routers on an Ethernet, and a new router issues an Ethernet Router Hello Message, the router with lowest priority is purged from the database (or the new router is ignored, if its priority is lowest). If more than one router has the lowest priority, the router with the lowest ID is purged. (An Ethernet address is treated in numerical comparisons as an unsigned 48 bit integer, with the first byte transmitted, and the leftmost byte as written, treated as the least significant byte.) If an old router must be purged, an adjacency down event is generated for the adjacency occupied by the old router.

After the Control Sublayer issues an "adjacency down complete", then an adjacency up event is generated for that adjacency number with the new neighbor.

9.1.2 Designated Router -

The Designated Router is the highest priority router, with numerically highest ID breaking ties. The set of all routers to be considered includes all routers in the area on that circuit in the "up" or "initializing" states. (See Section 9.1.1 for definition of numerical comparison of Ethernet addresses.)

A new router must not declare itself to be the Designated Router until DRDELAY*T2 has transpired.

The Designated Router periodically broadcasts its Ethernet Router Hello Message additionally to the multicast ID "all endnodes".

9.1.3 When To Transmit Router Hellos -

An Ethernet Router Hello Message is sent immediately when the circuit has been turned on.

It is also sent when at least T2 has transpired since the last transmission of a Router Hello Message on this circuit by this node and either:

1. BCT3 has expired

2. the contents of the next Hello Message to be transmitted would differ from the contents of the previous Hello Message transmitted by this node, or

3. this node has decided to become Designated Router

Sending of a Hello Message should restart T2 and T3. For maximum performance in the case of a clock with fine granularity, these should be restarted only after successful transmission of the Hello Message by the Data Link Layer.

9.1.4 Closing Down -

If at all possible, before a router is brought down, it should issue an empty Ethernet Router Hello Message, which will cause all routers receiving this message to bring down their adjacencies to the departing router, without unnecessary delay.

9.1.5 Database Of Endnodes -

An endnode is entered into the adjacency database, on an adjacency between [NC+NBRA+1] and [NC+NBRA+NBEA] when a hello is received from the endnode, provided there is room (not more than NBEA endnodes have been heard from). An adjacency up event is generated on that adjacency.

A timer is set for the value BCT3MULT times the timer reported in the endnode's Ethernet Endnode Hello Message. If another Hello is not received from the endnode before that timer expires, the endnode is purged from the database, provided that the Control Sublayer has issued an "adjacency up complete" on this adjacency (otherwise the adjacency will be cleared as soon as the "adjacency up complete is received from the Control Sublayer).

When the adjacency is cleared, the Control Sublayer is informed of an "adjacency down" event, and the adjacency can be reused after an "adjacency down complete" is received from the Control Sublayer.

9.1.6 Multiple Areas On An Ethernet -

If there are multiple areas on an Ethernet, routers must filter out control traffic from other areas.

When a level 1 router receives an endnode or router hello on an Ethernet, it checks that the area field in the ID equals "HomeArea". If not, the packet is dropped without being logged. Thus a level 1 router will not keep any adjacencies from other areas.

A level 2 router must keep adjacencies to other level 2 routers, besides the adjacencies in its own area. A level 2 router drops without logging any Ethernet Endnode Hello Messages it receives from other areas, and any Ethernet Router Hello Messages from level 1 routers (node type in IINFO field indicates level 1 router) in other areas.

When a level 2 router R receives an Ethernet Router Hello Message from a level 2 router A in another area, it does not drop the packet. R does include A in its adjacency database. R also includes A in E-LIST in its Ethernet Router Hello Messages. However, R does not include A in R's calculation of Designated Router for the Ethernet.

9.2 Endnodes

9.2.1 Ethernet Endnode Hello Messages -

Endnodes broadcast Ethernet Endnode Hello Messages to the multicast ID "all routers" when they first come up, and periodically thereafter, with period T3.

9.2.2 Designated Router's Ethernet Router Hello Message -

When an Ethernet Router Hello Message is received by an endnode, ID is checked to ensure that it is in the endnode's area. If the area field in ID does not equal "HomeArea", the packet is dropped without being logged. Otherwise, ID is copied into the endnode's ROUTERID variable, T4 is set to BCT3MULT times the value in HELLOTIME, and an adjacency up event is generated.

If ROUTERID had a value different from ID, an adjacency down event is generated, followed by an adjacency up event. If T4 expires without receipt of a Designated Router's Ethernet Router Hello Message, ROUTERID is erased and an adjacency down event is generated.

9.2.3 On-Ethernet Cache -

The endnode maintains a cache of destinations with which it is in contact and which are on the Ethernet. An entry for A is made if a packet is received with the "intra-Ethernet bit" set, from source A.

An entry for A is erased if:

1. the Routing Layer user gives the Routing Layer a packet with destination A and the directive "Tryhard"

2. no traffic is received from A validating the cache entry for CACHETIMEOUT (a parameter), (i.e. no packet is received that would have created an entry for A in the cache as per the rules above)

3. A is the least recently used cache entry, and the room in the cache is needed for a new entry

9.2.4 Filling In "next Hop" In Packet Headers -

On transmission of a packet for destination A, fill in the next hop in the Ethernet header according to the rules:

1. if A is in the cache, send to A

2. else if ROUTERID is filled in, send to ROUTERID

3. else send to A

10.0 MESSAGES

This section describes the message formats of the Routing Layer protocol. There are two types of Routing Layer messages:

- Packet route header -- This is used for ECL segments, which may require forwarding. There are two possible formats for data packet route headers:

  1. short format (identical to Phase III format)

  2. long format

- Routing Layer control -- These control Routing Layer routing and initialization functions. On non-broadcast circuits the types of Routing Layer control messages are:

  1. Initialization Message

  2. Verification Message

  3. Hello and Test Message

  4. Level 1 Routing Message

  5. Level 2 Routing Message

  On broadcast circuits the types of Routing Layer control messages are:

  1. Ethernet Router Hello Message

  2. Ethernet Endnode Hello Message

  3. Level 1 Routing Message

  4. Level 2 Routing Message

10.1 Message Format Notation

The following notation is used to describe the messages:

FIELD (LENGTH) : CODING Description of field

where:

FIELD is the name of the field.

(LENGTH) is the length of the field, one of:

1. A number meaning the number of 8-bit bytes.
2. A number followed by a "B" meaning the number of bits.
3. The letters "I-n" meaning an image field, with n being a number that specifies the maximum length of 8-bit bytes in the image. The image is preceded by a 1-byte count of the length of the remainder of the field. Image fields are variable length and may be null (count = 0). All eight bits of each byte are information bits.

CODING represents the type of coding used, one of:

1. B = Binary.
2. BM = Bit map. Each bit has independent meaning.
3. C = Constant.
4. NULL = Interpretation is data dependent.

Fields in separate messages with identical names are the same field and have identical meanings. All numeric values are decimal unless otherwise noted. All header fields and data bytes are transmitted low order or least significant bit first on the data line unless otherwise noted. Multiple byte fields are transmitted low order or least significant byte first.

10.2 Reserved Fields

Reserved fields in all received packets are ignored and transmitted as 0, except that reserved bits set in received data packets to be forwarded are passed along unchanged. If translating between long and short format, a reserved bit which was set in a field to be dropped is dropped along with the field to be dropped.

10.3 Optional Padding

All Routing Layer messages except Initialization can be padded. Padding can be used when communicating with Phase IV nodes, but must not be used when communicating with adjacent Phase III nodes.

If the top bit of the first byte is set, the remainder of the first byte is a count of the number of pad bytes, including the first byte. The total length of a message, including the padding, must not exceed the neighbor's blocksize. If the neighbor's blocksize is unknown (as in the case of Ethernet endnodes), then the maximum total pad sequence length is 7.

Thus the format of the optional padding is as follows:

```
+---------+-----+
| PLENGTH | PAD |
+---------+-----+
PLENGTH (1) : BM     the total length of the pad sequence
                              +---+---+---+---+---+---+---+---+
                     Bit:     | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
                              +---+---+---+---+---+---+---+---+
                     Set to:  | 1 | TOTAL-PAD-SEQUENCE-LENGTH |
                              +---+---+---+-----+-----+---+---+

PAD ((TOTAL-PAD-SEQUENCE-LENGTH) - 1) : No Meaning
```

10.4 Short Data Packet Format

The packet route header in short format is as follows:

```
+-------+---------+--------+---------+
| FLAGS | DSTNODE | SRCNODE | FORWARD |
+-------+---------+--------+---------+
```

FLAGS (1) : BM the set of flags used by the routing nodes. The format of this field is as follows:

```
         +----+---+---+-----+-----+---+---+---+
Bit:     | 7  | 6 | 5 | 4   |  3  | 2 | 1 | 0 |
         +----+---+---+-----+-----+---+---+---+
Set to:  | PF | V | R | RTS | RQR |    SFDP   |
         +----+---+---+-----+-----+---+---+---+
```

Bit	Definition
0-2	SFPD = 2, meaning short format
3	RQR (Return to Sender Request) 1 indicates try to return 0 indicates discard
4	RTS Return to Sender 1 => packet is on return trip
5	Reserved
6	Version, set to 0
7	PF pad field = 0 indicating no padding follows

DSTNODE (2) : B the destination node address

SRCNODE (2) : B the source node address

FORWARD (1) : BM information useful in the forwarding of the message. The format of this field is as follows:

```
         +---+---+-------+
Bit:     | 7 | 6 | 5 - 0 |
         +---+---+-------+
Set to:  | 0 | 0 | VISIT |
         +---+---+-------+
```

VISIT (6B) : BM the count of the number of nodes visited by this packet

10.5 Long Data Packet Format

```
+-------+--------+----------+-----+-------+----------+-----+
| FLAGS | D-AREA | D-SUBAREA | D-ID | S-AREA | S-SUBAREA | S-ID |
+-------+--------+----------+-----+-------+----------+-----+

            +-----+----------+--------+----+
            | NL2 | VISIT-CT | S-CLASS | PT |
            +-----+----------+--------+----+
```

FLAGS (1) : BM the set of flags used by the routing nodes. The format of this field is as follows:

```
         +----+---+-----+-----+-----+---+---+---+
Bit:     | 7  | 6 |  5  | 4   |  3  | 2 | 1 | 0 |
         +----+---+-----+-----+-----+---+---+---+
Set to:  | PF | V | I-E | RTS | RQR |    LFDP   |
         +----+---+-----+-----+-----+---+---+---+
```

```
Bit      Definition
---      ----------
0-2      LFDP = 6, meaning long format
 3       RQR (Return to Sender Request)
         1 indicates try to return
         0 indicates discard
 4       RTS    Return to Sender
         1 => packet is on return trip
 5       Intra-Ethernet packet
 6       Version, set to 0
         discarded if 1
 7       PF  pad field = 0 indicating
         no padding follows
```

D-AREA (1) : B	Reserved
D-SUBAREA (1) : B	Reserved
D-ID (6) : B	Destination ID (first 4 bytes must be set to HIORD)
S-AREA (1) : B	Reserved
S-SUBAREA (1) : B	Reserved
S-ID (6) : B	Source ID (first 4 bytes must be set to HIORD)
NL2 (1) : B	Next level 2 router, reserved
VISIT-CT (1) : B	Visit Count (0)
S-CLASS (1) : BM	Service Class, reserved
PT (1) : B	Protocol Type, reserved

10.6 Initialization Message

```
+-------+---------+--------+---------+-------+-------+----------+
| FLAGS | SRCNODE | TIINFO | BLKSIZE | TIVER | TIMER | RESERVED |
+-------+---------+--------+---------+-------+-------+----------+
```

FLAGS (1) : BM the Routing Layer control flag, with the following format:

```
Bit:      | 7  | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
          +----+---+---+---+---+---+---+---+
Set to:   | PF |    RES    |    TYPE   | 1 |
          +----+---+---+---+-----------+---+
```

Bit	Definition
0	1 indicates Control Packet
1-3	Type = 0
4-6	Reserved
7	PF pad field = 0 indicating no padding follows

SRCNODE (2) : B the identification of the source node's Routing Layer, containing the value ID as set by Network Management in the SELF parameters.

TIINFO (1) : BM Routing Layer information on node type and service requests, as follows:

```
Bit:     | 7 | 6 | 5 | 4 |  3  |   2   | 1 | 0 |
         +---+---+---+---+-----+-------+---+---+
Set to:  | 0 | 0 | 0 | 0 | BLO | VERIF | NTYPE |
         +---+---+---+---+-----+-------+-------+
```

NTYPE (2B) : B the Routing Layer node type:

- 0 Reserved
- 1 level 2 router
- 2 level 1 router
- 3 endnode

VERIF (1B) : BM Routing Layer Verification Message required if this bit is set.

BLO (1B) : BM Blocking Requested if this bit is set.

BLKSIZE (2) : B the maximum Data Link Layer receive block size this node will accept. (includes Routing header, excludes Data Link header)

TIVER (3) : B	the Routing Layer version, with the following format: Byte 1 -- version number (2 (00000010 binary)) Byte 2 -- ECO number (0 (00000000 binary)) Byte 3 -- user ECO number (0 (00000000 binary))
TIMER (2) : B	Hello Timer, in seconds
RESERVED (I-64)	A reserved field containing a count of 0.

10.7 Verification Message

```
+--------+---------+--------+
| FLAGS  | SRCNODE | FCNVAL |
+--------+---------+--------+
```

FLAGS (1) : BM the Routing Layer control flag, with the following format:

```
            +----+---+---+---+---+---+---+---+
Bit:        | 7  | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
            +----+---+---+---+---+---+---+---+
Set to:     | PF |     RES     |    TYPE   | 1 |
            +----+---+---+---+-----------+---+
```

Bit	Definition
0	1 indicates Control Packet
1-3	Type = 1
4-6	Reserved
7	PF pad field = 0 indicating no padding follows

SRCNODE (2) : B the identification of the source node's Routing Layer, containing the value ID as set by Network Management in the SELF parameters.

FCNVAL (I-64) : B the function value.

10.8 Hello And Test Message

```
+-------+---------+-----------+
| FLAGS | SRCNODE | TEST DATA |
+-------+---------+-----------+
```

FLAGS (1) : BM the Routing Layer control flag, with the following format:

```
             +----+---+---+---+---+---+---+---+
Bit:         | 7  | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
             +----+---+---+---+---+---+---+---+
Set to:      | PF |     RES   |    TYPE   | 1 |
             +----+---+---+---+-----------+---+
```

Bit	Definition
0	1 indicates Control Packet
1-3	Type = 2
4-6	Reserved
7	PF pad field = 0 indicating no padding follows

SRCNODE (2) : B the identification of the source node's Routing Layer, containing the value ID as set by Network Management in the SELF parameters.

TEST DATA (I-128) : B a sequence of 0 to 128 bytes of data used to test the circuit. Each byte is 252 octal.

10.9 Level 1 Routing Message

A Level 1 Routing Message contains one or more segments, each segment referring to COUNT destinations starting with STARTID.

Destination #0 is understood to mean "nearest level 2 router".

```
+-------+---------+-----+---------+---------+-----+----------+
| FLAGS | SRCNODE | RES | SEGMENT | SEGMENT | ... | CHECKSUM |
+-------+---------+-----+---------+---------+-----+----------+
```

FLAGS (1) : BM the Routing Layer control flag, with the following format:

```
                 +----+---+---+---+---+---+---+---+
      Bit:       | 7  | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
                 +----+---+---+---+---+---+---+---+
      Set to:    | PF |    RES    |    TYPE   | 1 |
                 +----+---+---+---+-----------+---+
```

Bit	Definition
0	1 indicates Control Packet
1-3	Type = 3
4-6	Reserved
7	PF pad field = 0 indicating no padding follows

SRCNODE (2) : B the identification of the source node's Routing Layer, containing the value ID as set by Network Management in the SELF parameters.

RES (1) : B A reserved field of 1 byte

SEGMENT is of the FORM:

```
+-------+---------+---------+
| COUNT | STARTID | RTGINFO |
+-------+---------+---------+
```

COUNT (2) : B the number of IDs in the RTGINFO segment

STARTID (2) : B the first ID reported in the Routing Message, with the top 6 bits (the area bits) set to 0

RTGINFO : BM the hops and cost to a destination, in the format:

```
          +----+---------+-------+
  Bit:    | 15 | 14 - 10 | 9 - 0 |
          +----+---------+-------+
          | 0  |  HOPS   | COST  |
          +----+---------+-------+
```

CHECKSUM (2) : B a check on the routing data base, and a check to ensure Phase IV Routing Messages not be mistaken for Phase III Routing Messages, as well as a check of the message. It is a one's complement add starting with the first SEGMENT and continuing until the CHECKSUM. To ensure that a Phase IV Routing Message will be distinguished from a Phase III Routing Message, the sum on the Phase IV Routing Message is initialized to 1.

10.10 Level 2 Routing Message

A Level 2 Routing Message contains one or more segments, each segment referring to COUNT areas starting with STARTAREA.

```
+-------+---------+-----+---------+---------+-----+----------+
| FLAGS | SRCNODE | RES | SEGMENT | SEGMENT | ... | CHECKSUM |
+-------+---------+-----+---------+---------+-----+----------+
```

FLAGS (1) : BM the Routing Layer control flag, with the following format:

```
            +----+---+---+---+---+---+---+---+
Bit:        | 7  | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
            +----+---+---+---+---+---+---+---+
Set to:     | PF |    RES    |    TYPE   | 1 |
            +----+---+---+---+-----------+---+
```

Bit	Definition
0	1 indicates Control Packet
1-3	Type = 4
4-6	Reserved
7	PF pad field = 0 indicating no padding follows

SRCNODE (2) : B the identification of the source node's Routing Layer, containing the value ID as set by Network Management in the SELF parameters.

RES (1) : B A reserved field of 1 byte

SEGMENT is of the FORM:

```
+-------+-----------+---------+
| COUNT | STARTAREA | RTGINFO |
+-------+-----------+---------+
```

COUNT (2) : B the number of areas in the RTGINFO segment

STARTAREA (2) : B The first area reported in the Routing Message, with the top 10 bits 0 (area is a 6 bit quantity)

RTGINFO : BM Hops and cost to a destination area, in the format:

```
       +----+---------+--------+
Bit:   | 15 | 14 - 10 | 9 - 0  |
       +----+---------+--------+
       | 0  |  HOPS   | COST   |
       +----+---------+--------+
```

CHECKSUM (2) : B a check on the routing data base, and a check to ensure Phase IV Routing Messages not be mistaken for Phase III Routing Messages, as well as a check of the message. It is a one's complement add starting with the first SEGMENT and continuing until the CHECKSUM.

To ensure that a Phase IV Routing Message will be distinguished from a Phase III Routing Message, the sum on the Phase IV Routing Message is initialized to 1.

10.11 Ethernet Router Hello Message

```
+-------+-------+----+-------+---------+----------+
| FLAGS | TIVER | ID | IINFO | BLKSIZE | PRIORITY |
+-------+-------+----+-------+---------+----------+
     +------+-------+-----+--------+
     | AREA | TIMER | MPD | E-LIST |
     +------+-------+-----+--------+
```

FLAGS (1) : BM the Routing Layer control flag, with the following format:

```
            +----+---+---+---+---+---+---+---+
 Bit:       | 7  | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
            +----+---+---+---+---+---+---+---+
 Set to:    | PF |    RES    |    TYPE   | 1 |
            +----+---+---+---+-----------+---+
```

Bit	Definition
0	1 indicates Control Packet
1-3	Type = 5
4-6	Reserved
7	PF pad field = 0 indicating no padding follows

TIVER (3) : B the Routing Layer version, with the following format:

Byte 1 -- version number (2)
Byte 2 -- ECO number (0)
Byte 3 -- user ECO number (0)

ID (6) : B the system ID of the transmitter

IINFO (1) : BM flags

Bit	Definition
0,1	2=level 1 router 1=level 2 router
2	Verification Required flag (0 on the Ethernet)
3	Reject Flag, Reserved
4	Verification Failed, Reserved
5	No Multicast Traffic Accepted 0=accepts multicast
6	Blocking Requested Flag (0 on Ethernet)
7	Reserved

BLKSIZE (2) : B maximum Data Link Layer receive block size -- (Includes Routing header, excludes Data Link header)

PRIORITY (1) : B	router's priority
AREA (1) : B	reserved
TIMER (2) : B	Hello Timer in seconds
MPD (1) : B	reserved
E-LIST (I-244)	list of router states for logical Ethernets on this physical Ethernet The format of each list item is:
NAME (7) : B	logical Ethernet name, reserved
R/S-LIST (I-236)	list of router/state pairs. The format of each list item is:

ROUTER (6) : B router ID
PRISTATE (1) : BM priority and state
Bit 7: State: 1 means known 2-way, 0 otherwise
Bits 0-6: Priority: This router's priority

Note: E-LIST will always contain a single entry of the format:

1. NAME = 0

2. R/S-LIST = list of router/state pairs

10.12 Ethernet Endnode Hello Message

```
+-------+-------+----+-------+---------+------+
| FLAGS | TIVER | ID | IINFO | BLKSIZE | AREA |
+-------+-------+----+-------+---------+------+
     +------+----------+-------+-----+--------+
     | SEED | NEIGHBOR | TIMER | MPD | [DATA] |
     +------+----------+-------+-----+--------+
```

FLAGS (1) : BM the Routing Layer control flag, with the following format:

```
         +----+---+---+---+---+---+---+---+
Bit:     | 7  | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
         +----+---+---+---+---+---+---+---+
Set to:  | PF |    RES    |    TYPE   | 1 |
         +----+---+---+---+-----------+---+
```

Bit	Definition
0	1 indicates Control Packet
1-3	Type = 6
4-6	Reserved
7	PF pad field = 0 indicating no padding follows

TIVER (3) : B the version, with the following format:

Byte 1 -- version number (2)
Byte 2 -- ECO number (0)
Byte 3 -- user ECO number (0)

ID (6) : B ID of the transmitting node. The first 4 bytes must be set to HIORD. The bottom 2 bytes are the Phase IV address (ID) assigned to the node.

IINFO (1) : BM flags

Bit	Definition
0,1	the node type (value 3, for endnode)
2	Verification Required flag (0 on the Ethernet)
3	Reject Flag, Reserved
4	Verification Failed, Reserved
5	No Multicast Traffic Accepted 0=accepts multicast
6	Blocking Requested Flag (0 on Ethernet)
7	Reserved

BLKSIZE (2) : B	maximum Data Link Layer receive block size -- (Includes Routing header, excludes Data Link header)
AREA (1) : B	reserved
SEED (8) : B	the verification seed (0)
NEIGHBOR (6) : B	neighbor's system ID, ID of Designated Router on Ethernets (0 if no Designated Router)
TIMER (2) : B	Hello Timer, in seconds
MPD (1) : B	reserved
DATA (I-128) : B	A sequence of 0 to 128 bytes of data used to test the circuit. Each byte is 252 octal.

APPENDIX A

ROUTES, ADDRESSES, AND NAMES

This appendix explains the relationship between addresses and names in a network.

The Routing Layer identifies nodes in a network by unique numbers (addresses). However, it is often more convenient for users to identify nodes by an alphabetic or alphanumeric name. In addition, several users at one node may each wish to identify network nodes by different names. Moreover, users may not want to use only names that are unique within the network. Thus a problem arises as to how to bind node names to node addresses in a network.

The solution is to use names that are unique within a node (locally unique) and addresses that are unique within the network (globally unique). The following rules apply:

- o The Routing Layer knows nodes only by their addresses.
- o Names are assigned individually on a node-by-node basis (local) and are unique within a node.
- o The local naming function (name to address directory) is not necessarily a one-to-one function. In other words, alias node names may be assigned to a node name as long as overall local uniqueness is preserved.

This solution has the following advantages:

- o Aliases are not global. When networks merge, there is no need to change local aliases.
- o It preserves the correspondence between names and addresses.
- o It avoids the complex problem of maintaining duplicate copies of a distributed data base in an automatic function.
- o It avoids network maintenance problems related to name and address directories. Incorrect directories affect only local

users.

This solution imposes some responsibilities on network managers. The local network manager must ensure the local directories preserve uniqueness of names. The central network manager must ensure the directories preserve uniqueness of addresses.

APPENDIX B

ROUTING SUBSETS AND TOPOLOGIES

This appendix defines routing and nonrouting nodes in terms of their composition, and outlines topological considerations that must be made when planning network configurations.

B.1 NODE TYPES

In Phase IV there are the following types of nodes:

- Level 2 routers
- Level 1 routers
- Phase III routers
- Phase IV endnodes
- Phase III endnodes

B.2 TOPOLOGICAL CONCEPTS

Network topology involves two concepts: physical connectivity and logical connectivity.

Physical connectivity defines rules for connecting network nodes by physical circuits.

A node's position in the network may be restricted.

Two nodes are physically connected if they are connected by a sequence of active circuits.

Two nodes are logically connected if they can communicate.

B.3 DECNET TOPOLOGICAL PRINCIPLES

The following are goals:

- Logical connectivity should equal physical connectivity.
- Any subset of a legal topology should be a legal topology.

B.4 ENDNODE RESTRICTIONS

An endnode must be attached to the network by at most one circuit.

If an endnode is attached to another endnode via a non-broadcast circuit, the entire network consists of those two endnodes, and no other nodes can be added to the network.

When there are Ethernets in the network, the sum of the number of endnodes on Ethernets attached to common routers cannot exceed the number of Broadcast Endnode Adjacencies that each of the common routers can handle.
For example:

```
                                  Ethernet #1
    --------------+-----------------------------
                  |              Ethernet #2
-------------+----|-----+----------------
             |    |     |
             R1---+     R3
             |                     Ethernet #3
-------------+---------+-----------------------
                       |
                       R2
                       |           Ethernet #4
   --------------------+---------------------
```

Suppose in this network the number of BEAs R1 can handle is R1.NBEA, R2 can handle R2.NBEA, and R3 can handle R3.NBEA.

Suppose there are E1 endnodes on Ethernet #1, E2 endnodes on Ethernet #2, E3 endnodes on Ethernet #3, and E4 endnodes on Ethernet #4.

Then

- E1+E2+E3 must be less than or equal to R1.NBEA.
- E3+E4 must be less than or equal to R2.NBEA
- E2 must be less than or equal to R3.NBEA

B.5 ETHERNET ROUTER RESTRICTIONS

The sum of the number of routers on Ethernets attached to router R cannot exceed R's parameter NBRA. The number of routers on an Ethernet to which router R is connected cannot exceed R's parameter NR for that Ethernet.

There is a large amount of control traffic overhead associated with routers on Ethernets. This is due to two causes:

1. Each router on the Ethernet frequently issues Routing Messages, as well as Ethernet Router Hello Messages.
2. The control traffic associated with the temporary looping property of the routing algorithm, as nodes count to realize the unreachability of unreachable nodes, is proportional to the square of the number of routers on the Ethernet.

Thus there is some practical limit to the number of routers that can coexist on an Ethernet.

B.6 LARGER NETWORKS

In Phase III, a legal network was restricted in size to the number of nodes that fit in a Routing Message. With segmented Routing Messages, this node limit is no longer necessary. However, depending on the application, performance will no longer be acceptable above some network size. Also, memory restrictions also impose a maximum network size.

Phase III nodes cannot deal with segmented Routing Messages. Thus the number of nodes a Phase III node can handle is limited by the number of nodes that fit in a Routing Message of the size supported by the blocksize of that node's circuits. In a mixed Phase III/IV network, a Phase III node cannot communicate with nor be included on a path to a node with node number higher than the Phase III node can handle.

If the rule that "any subset of a legal topology is a legal topology" is to hold, any network containing Phase III nodes must not contain node numbers higher than any router's capacity.

B.7 HIERARCHICAL NETWORKS

When a Phase IV network is hierarchical, additional topological restrictions apply and it is not possible to satisfy the rule that "any subset of a legal topology is a legal topology." Thus logical connectivity may not equal physical connectivity.

The rules for a hierarchical network are:

1. Each node belongs to exactly one of the areas. Note that this applies to Phase III nodes as well. Although Phase III nodes are not told an area number by network management, they must be logically associated with a single area, and must not have any circuits outside that area.

2. Only level 2 routers are allowed to have neighbor nodes in other areas.

3. Each area must be physically intact, i.e. there must exist a path totally internal to the area between each pair of nodes in the area.

4. The subnet consisting of level 2 routers must be physically intact.

A Phase III routing node cannot communicate with nodes in other areas, nor can it be on a path between two Phase IV nodes in different areas. Thus, for each area, the subnet consisting of the Phase IV nodes in that area must be physically intact.

It is strongly recommended for performance reasons that all level 1 routers and endnodes on an Ethernet be in the same area.

APPENDIX C

NONROUTING OPERATION

This section describes the routing operation for nonrouting nodes on non-broadcast circuits. The operation of routing nodes, and the operation of nonrouting nodes on broadcast circuits, are covered in the body of the specification.

C.0.1 Receive Module

The Node Listener processes a packet upon receipt. The Node Listener then passes the packet or discards it, depending on whether it is a packet for self or any other message. If the packet is a Routing Message, the Node Listener discards it.

When transmitting to a node other than self, the node sends the packet out over the only circuit available, unless the adjacent node is an endnode, not the destination. Otherwise, the node returns or discards the packet, depending on the packet route header option.

C.0.2 Interfaces

The ECL and Data Link interfaces are as described for routers. Nonrouting nodes support the entire Network Management interface except:

READ NODE PARAMETERS
READ AREA PARAMETERS
READ ADJACENCY PARAMETERS

Inapplicable SELF parameters are:

- NN
- NA
- NBRA
- NBEA
- Maxh
- Maxc
- Maxv
- T1
- BCT1

Inapplicable SELF counters are:

- node unreachable packet loss
- aged packet loss
- node out-of-range packet loss
- partial routing update loss

Inapplicable CIRCUIT counters are:

- transit packet received
- transit packet sent
- transit congestion loss

APPENDIX D

PHASE III COMPATIBILITY

Phase III nodes cannot appear on broadcast circuits. Thus there is no Phase III compatibility problem with broadcast adjacencies.

On non-broadcast circuits, if an Initialization Message is received with TIVER 1.3.0, the adjacency is marked as Phase III router or endnode, and an Initialization Message with TIVER 1.3.0, and in Phase III format, is sent to the Phase III neighbor. Since Phase III routers cannot receive segmented Routing Messages, the Data Link blocksize in the received Initialization Message must be large enough to accommodate an entire Phase III Routing Message. Otherwise, initialization failure is logged, and the circuit is not brought up. The Hello Timer in the adjacency is set to T3, since a Phase III neighbor does not report the value of its Hello Timer in control messages.

When Routing Messages are received from the Phase III neighbor, they are treated as Phase III format. When Routing Messages are sent to a Phase III neighbor, they are sent in Phase III format. The field for "nearest level 2 router" is not sent, and the Routing Message cannot be segmented. To ensure that nodes do not confuse Phase III and Phase IV Routing Messages, the checksum is computed differently.

Phase III and Phase IV data packet formats are the same, except that Phase III nodes will not know how to interpret the area field in the source and destination addresses. Thus a Phase IV node must strip off the destination area field (which will of course be equal to the home area if a packet is routed through a Phase III node) before forwarding the packet to the Phase III neighbor. In addition, for "return to sender" to work with a Phase III neighbor, the source area field is stripped off as well, provided that the source area is equal to home area.

When a data packet is received from a Phase III neighbor, home area is written into the destination area field, and home area is written into the source area field, provided that it is 0.

Note that the Phase III functional specification specifies that Phase III nodes must check the SRCNODE field in the data packet header and drop the packet if SRCNODE is out of range. This would cause a Phase III node to drop a packet that arrived from a different area. Thus any Phase III nodes that perform that check must be patched or removed from the network before a multiple area Phase IV network can be implemented.

If a Phase III router, or chain of connected Phase III routers, connects to two different areas, these areas become merged into one area, disrupting operation of nodes in the different areas which have the same low order node IDs. To prevent this problem, which might occur due to inadvertent connection of two nodes, node verification is used. For example, if each node is assigned a password which is area specific, this area leakage can be prevented.

To enforce password usage, Phase IV routers not in area 1 must require verification from Phase III routers. Specifically, matching of non-null passwords is required before bringing up such a circuit. Individual passwords on each circuit, or the ability to enable verification on a per circuit or per node basis is required for level 2 routers, so that passwords can be assigned according to the above recommendation.

APPENDIX E

ROUTING LAYER COUNTERS AND EVENTS

This appendix specifies the Routing Layer counters and events. These provide Network Management with a means to detect and isolate certain failures. The events and counters described in this Appendix relate generally to Routing Layer activities, congestion, faults, topological changes, and verification (security) violations. Counters and events related to packet modification, misdelivery, or duplication, which ECL can detect, are not specified here.

In the following discussion, the term fault refers to the cause of a problem. The term error refers to a manifestation of a fault.

Routing Layer counters and events capture sufficient information to detect and isolate single faults. Multiple faults are detected, but only isolated when it is cost effective to do so.

E.1 SOURCE EVENTS

A source event is any event that may result in incrementing a counter or logging an event. Several source events may cause a single counter to increment. Similarly, several source events may produce a single generic event for logging.

Source events are classified in two ways: by event manifestation and by cause. The primary classification is by event manifestation. Within each event manifestation category are subcategories of specific source events. Each of these specific source events has a particular cause. The following three tables describe source event manifestations, causes, and source events. Table 4 defines each event manifestation category. Table 5 defines each source event cause. Table 6 defines each source event along with its event manifestation category and probable cause. The abbreviations in Table 6 are defined in Table 5.

Table 4
Source Event Manifestations

Event Manifestation	Definition
Data Movement	This group of source events reflects the movement of data through the Routing Layer.
Congestion	This group of source events reflects the discarding of data packets due to Routing Layer congestion.
Data Packet Discarded	This group of source events reflects the discarding of data packets due to a fault.
Message Format Error	The format of the message is in error.
Partial Routing Update Loss	This source event represents the receipt of a Routing Message that is too long to process so that some information from the message is discarded.
Circuit Down	This group of source events represents the detection of a circuit failure.
Adjacency Down	This group of source events represents a loss of connection to an adjacent node.
Initialization Failure	This group of source events represents a failure to initialize with an adjacent node.
Verification Reject	This source event represents the receipt of an invalid Verification Message.
Circuit Up	This source event represents the successful initialization of a circuit.
Adjacency Up	This source event represents the successful initialization with an adjacent node.
Node Reachability Change	This group of source events represents a change in node reachability (between reachable and unreachable.
Adjacency Reject	This group of source events represents this node being forced to reject an adjacency.

Table 5
Source Event Causes

(Symbol)	Cause	Explanation
(A)	Activity	The normal activity of the Routing Layer in moving data.
(C)	Congestion	The resource limit condition, detected by Routing that causes Routing to discard normal data.
(S)	System Fault	Failures of either hardware or software and undetected circuit failures, excluding circuit faults.
(O)	Operator Initiated	Events directly caused by the action of an operator, including failure of an operator to set parameters correctly for the harmonious operation of multiple nodes. Typical operator faults include a node's maximum address that is too small or an adjacent node's ID that is too large.
(T)	Topological Change	Modifications in topology that result in a node changing its reachability status.
(V)	Verification Violation	The detected attempt of a node to initialize without providing the expected verification information.
(L)	Circuit Fault	The detected failure of a circuit. A circuit fault does not necessarily result in a topological change.

Table 6
Source Events

Event Manifestation	Source Event Number	Probable Cause	Definition
Data Movement	1	A	A data packet is received from an adjacency for another node in the network.
	2	A	A data packet from another node in the network is sent over an adjacency.
	3	A	A data packet is received from an adjacency for this node's ECL.
	4	A	A data packet from this node's ECL is sent over an adjacency.
Congestion	5	C	A transit data packet is discarded for congestion reasons.
	6	C	A terminating packet is discarded due to the inability of ECL to process packets fast enough.
Data Packet Discarded	7	T	Destination node is unreachable.
	8	T	Packet is too old.
	9	S	Destination node is out of range.
	10	O	Received data packet is too large to forward due to the blocksize of the data link that would be used.

(continued on next page)

Table 6 (Cont.)
Source Events

Event Manifestation	Source Event Number	Probable Cause	Definition
Message Format Error	11	S	Message format is in error.
Partial Routing Update Loss	12	O	Received long Routing Message with reachable node number greater than this node's NN.
Circuit Down	13	L	Data link synchronization lost.
	14	L	Data link threshold error detected.
	14.1	L	Data Corruption detected in X.25 net.
	15	L	Circuit rejection algorithm rejected circuit (for reason other than threshold error).
Adjacency Down	15.1	L	Node Listener timeout.
	15.2	L	Node Listener received invalid data.
	16	S	Unexpected control (Initialization or Verification) message received.
	17	S	Routing Message received with checksum error.
	18	O	Node id from Routing or Hello Message not the expected one.
	19	O	Hello received indicating connectivity became 1-way

(continued on next page)

Table 6 (Cont.)
Source Events

Event Manifestation	Source Event Number	Probable Cause	Definition
Initialization Failure	20	L	Verification Message not received in timeout period.
	21	L	Data link synchronization lost.
	22	L	Data link threshold error detected.
	23	O	Version skew detected.
	24	O	Node id in received Initialization Message too large.
	25	O	Block size in received Initialization Message too small.
	26	O	Invalid verification seed value in received Initialization Message.
	26.1	O	Password required from Phase III node
	27	S	Unexpected message received.
	27.1	O	Area mismatch.
Verification Reject	28	V	Invalid verification received.
Circuit Up	29.1	L	Circuit useable
Adjacency Up	29.2	L	Initialization with neighbor complete

(continued on next page)

Table 6 (Cont.)
Source Events

Event Manifestation	Source Event Number	Probable Cause	Definition
Node Reachability Change	30	T	Node reachable.
	31	T	Node unreachable.
Adjacency Reject	32	O	Too many BRAs (NBRA or NR on an Ethernet exceeded).
	33	O	Too many BEAs
Area Reachability Change (or nearest level 2 router, for level 1 nodes)	34	T	Became reachable
	35	T	Became unreachable

E.2 COUNTERS

There are two types of counters -- node and circuit. The Routing Layer maintains one node counter for each of the defined node counters. The Routing Layer maintains one circuit counter per circuit for each of the defined circuit counters. Node counters count source events attributed to topological changes, faults, and verification violation. Circuit counters count source events attributed to activity, congestion, and faults.

The node and circuit counters are defined below. Each counter relates to a source event number. Refer to the glossary for definitions of terminating, originating, and transit packets. A packet is received when it passes from the Data Link Layer to the Routing Layer. A packet is sent when it passes from the Routing Layer to the Data Link Layer.

Node Counters

Counter Name	Counter Width	Source Events Included
node unreachable packet loss	16 bits	7
aged packet loss	8 bits	8
node out-of-range packet loss	8 bits	9
oversized packet loss	8 bits	10
packet format error	8 bits	11
partial routing update loss	8 bits	12
verification reject	8 bits	28

Circuit Counters

Counter Name	Counter Width	Source Events Included
transit packet received	32 bits	1
transit packet sent	32 bits	2
terminating packet received	32 bits	3
originating packet sent	32 bits	4
transit congestion loss	16 bits	5
terminating congestion loss*	16 bits	6
circuit down	8 bits	13 - 15
initialization failure	8 bits	20 - 27.1
corruption loss (X.25 only)	8 bits	14.1

* Only required in the implementations in which ECL does not guarantee Routing that it will process a terminating packet (thereby freeing the buffer holding the packet) in a short, bounded period of time.

E.3 EVENTS

Network Management groups some of the source events (Section E.1) together for logging. The DNA Network Management Functional Specification specifies this logging operation. When a source event to be logged occurs, the Routing Layer identifies it by type, time-stamps it, and places it in an internal Routing event queue. If the event queue is full, Routing discards the newest event in the queue and replaces it with an "event(s) lost" event.

Routing Layer Events

Event Type	Source Events	Logged Information
Node unreachable packet loss	7	adj, packet header
Aged packet loss	8	packet header
Node out-of-range packet loss	9	adj, packet header
Oversized packet loss	10	adj, packet header
Message format error	11	adj, packet header
Partial routing update loss	12	adj, packet header highest node address
Circuit down - circuit fault	13-15	adjacency
Circuit down	15.1-17	adj, packet header
Adjacency down	15.1-17	adj, packet header
Circuit down - operator initiated	18	adj, packet header
Adjacency down - operator initiated	19	adj, packet header
Initialization failure - circuit fault	20-22	circuit
Initialization failure - operator initiated	23-26.1	adjacency, pkt hdr, received version (23 only)
Initialization failure - software	27	adj, packet header
Verification reject	28	circuit, node ID from message
Circuit up	29.1	adjacency
Adjacency up	29.2	adjacency
Node reachability change	30,31	node address, state
Area reachability change	34-35	area (level 2)
Adjacency reject	32.1-33	adj, exceeded param.

Notes:

1. A logged event of a single type that can result from more than one source event also contains a reason code to specify the source event.

2. "Packet header" denotes the first 6 bytes (48 bits) of a Routing Layer message in short format, and the first 21 bytes for packets in long format.

3. "Adjacency" is logged as <circuit, node ID>

4. "Adjacency up" is logged as "Circuit Up", and "Adjacency Down" is logged as "Circuit Down" on point-to-point circuits.

APPENDIX F

ALGORITHMS AND MODELS

This appendix describes algorithms and models pertaining to:

- Circuit cost
- Buffer management

F.1 CIRCUIT COST ASSIGNMENT ALGORITHM

The assignment of cost to circuits can reflect both delay and throughput data. Delay data can include transmission delay, propagation delay, processing delay, and retransmission delay. Delay data does not include queuing delay. Throughput data can include circuit bandwidth, circuit overhead, and processor bandwidth. Throughput data does not include actual traffic overhead. Basically, it is desirable to avoid a circuit cost assignment algorithm with high sensitivity to traffic fluctuations, thereby producing a condition where routes change to accommodate traffic changes and the new traffic flow causes new route changes, and so on.

A circuit cost assignment occurs as a result of a node generation or an Initialization module. An operator can always override any assignment.

One such assignment is based on circuit bandwidth and is as follows:

```
            1 where bandwidth x >= 100,000 bits/second

 F(x) = [100,00/x] for 4,000 bits/second < x < 100,000 bits/sec
            25 where x <= 4,000 bits/second
```

where x is circuit bandwidth (bits/sec)

F.2 BUFFER MANAGEMENT

When no buffers are available for receiving packets from a circuit, store and forward deadlock can occur. Deadlock can be avoided by insuring that at least one buffer is available per circuit, or a buffer can be made available without requiring additional resources. Such deadlock avoidance can require discarding packets.

When receive buffers are not available quickly enough, a circuit can go down unnecessarily at the Data Link Layer. It is much better for the Routing Layer to discard a packet than for a circuit to go down.

The Routing Layer should not initialize unless it can obtain at least the minimum number of receive buffers for each circuit. If an implementation obtains these buffers from a shared system buffer pool, then the minimum number must be permanently allocated from the pool by the Routing Layer when it initializes. They can, of course, be returned when Routing halts.

The only time a circuit may be allowed to go below its minimum number of buffers is when the system can guarantee that a receive buffer can be allocated to the circuit soon enough in the future to prevent the circuit from going down. This means that if the system has run out of free buffers and is down to the minimum number of receive buffers for each circuit, then:

- A received data packet that would normally be forwarded on another circuit must be discarded.
- A received Routing control message can and should be processed.
- A received data packet for this node should be given to ECL only if ECL is known to be able to return the buffer in a short, bounded period of time. Otherwise, discard the packet.

Compute the minimum number of receive buffers required for a given circuit for the circuit speed and an estimate of the maximum time that Routing (or possibly ECL) can take to process a received message.

F.3 POSSIBLE BUFFER MANAGEMENT MODEL

Routing has a common pool of buffers that can transmit or receive. If the implementation of ECL in a system is known to be unable to make the guarantee of short, bounded processing of received data packets, then Routing must limit the number of outstanding, received packets that ECL can hold onto simultaneously (provided that ECL and Routing are sharing a common buffer pool). This is best done by a fixed quota.

Setting this quota to the square root limit used by the congestion control algorithm is acceptable, but other values may be used as well. If a quota is used, then any packets discarded due to the ECL quota being filled must be counted.

The Routing Layer buffer quota provided by the system is divided into the following buffer quotas:

1. Decision (0)

2. Update (1 sufficient; 1 per circuit recommended)

3. Node Listener (0)

4. Node Talker (1)

5. Forwarding (at least 1 per circuit; 12-15 per terrestrial circuit and 30 - 35 per satellite circuit recommended).

6. A separate receive quota for each circuit (depends on speed of circuit -- at least 1, 2, or 3 recommended).

   If an implementation is constructed using a single buffer pool that the Routing Layer shares with other system processes, and if the Routing Layer does not do any actual data moving from one buffer to another, then all buffers containing data to be transmitted are either obtained from ECL or are receive buffers that contain data packets that are being forwarded.

   The rules above and the congestion control algorithms adequately define the use of these buffers. However, note the following:

   First, the square root limit is defined to be the number of buffers available for forwarding divided by the square root of the number of circuits. The number of buffers available for forwarding should not include the minimum number of receive buffers, nor should it include ECL's quota, if such a quota exists.

   Second, in such a model, a single buffer beyond the minimum number of receive buffers and a single ECL transmit buffer are sufficient to allow the Routing Layer to run correctly without starving a circuit for receive buffers. In general, for adequate performance additional buffers will also be required.

F.4 DETAILS OF CHARGING AND CREDITING AGAINST QUOTAS

All buffers not free will be charged against a specific quota. The quota will never be exceeded except possibly for a brief instant while a Routing process frees the buffer by consuming the information or by discarding a packet.

A quota is charged for a buffer upon the following events:

1. A free buffer is assigned to the Data Link Layer for reception on a specific circuit.

2. A buffer is moved from one Routing module to another. The receiving quota is charged.

3. A buffer is supplied by ECL that contains input data.

4. A free buffer is seized by a process to send a Routing Layer control message.

A quota is credited for a buffer upon the following events:

1. Transmission of a buffer is completed by the Data Link Layer. (The quota is credited whether or not the transmission was successful.)

2. A buffer is moved from one Routing Layer module to another. The sending quota is credited.

3. A process consumes the contents of a Routing Layer control message and returns the empty buffer.

4. A process discards a packet and returns the buffer.

5. ECL issues a successful CHECK RECEIVE command.

APPENDIX G

BUFFER SIZES

There are two SELF parameters set by network management:

1. Buffer Size (BS)

2. ECL Segment Size (SS)

The buffer size is 6 greater than the size of buffers Routing uses for forwarding, not including routing header or datalink header.

The ECL segment size is reported by Routing to ECL. It equals SS-6. SS-6 is the maximum size segment the ECL is allowed to pass to Routing. NSP requires SS-6 to be at least of some minimal size. (The size of the maximum length CI Message, with maximum length connect data required by the Session Layer.)

Usually, BS=SS. However, when the network buffer size is being changed, BS may be greater than SS. BS can never be less than SS.

Thus to expand the buffer sizes in the network, each node's BS must be increased, one at a time. When all nodes' BS parameters are increased, then each node's SS can be increased, one at a time. Similarly, to shrink the buffer sizes in the network, each node's SS must be decreased, one at a time. When all nodes' SS parameters are decreased, then each node's BS can be decreased, one at a time.

Note that BS and SS do not include route header overhead. They do, however, include an extra 6 bytes for compatibility with Phase III. There are two formats of route header in Phase IV -- long format and short format. The overhead in long format is greater than the overhead in short format.

For each circuit, there is a datalink blocksize, which includes route header overhead. Thus for Ethernet circuits, the datalink blocksize must be greater than or equal to BS minus 6 plus the route header overhead in long data packet format. For DDCMP circuits, the datalink blocksize must be greater than or equal to BS minus 6 plus the route header overhead in short format.

For each adjacency, there is a negotiated datalink blocksize, which is the smaller blocksize requested by either end. For adjacencies on non-X.25 circuits, the negotiated blocksize must be greater than or equal to BS minus 6 plus the route overhead (long format for Ethernets, short format for point-to-point circuits). Also, if the node type of the adjacency is Phase III router, the negotiated blocksize must be large enough to fit an entire Routing Message, since Phase III routers cannot accept a segmented Routing Message.

Since the X.25 Initialization Sublayer performs fragmentation and reassembly, the datalink blocksize on X.25 circuits does not need to meet the above constraints.

APPENDIX H

GLOSSARY

1. adjacency -- a [circuit, nodeID] pair. An Ethernet with n attached nodes represents n-1 adjacencies to a router on that Ethernet. A router attached to another router via n circuits has n adjacencies to the second router.

2. aged packet -- a packet that has exceeded the maximum number of visits.

3. BEA (broadcast endnode adjacency) -- an endnode connected to the same Ethernet as this node. The term also applies to the state information a router associates with the adjacency.

4. BRA (broadcast router adjacency) -- a router connected to the same Ethernet as this node. The term also applies to the state information a router associates with the adjacency. If the same router is attached to more than one common Ethernet as this node, that router appears as a BRA on each common Ethernet.

5. broadcast circuit -- a circuit on which multiple nodes are connected, and there exists a method for transmitting a packet which will be received by multiple receivers.

6. circuit -- one of the following:

   . an Ethernet

   . a DDCMP link

   . an attachment to one of the nodes in a DDCMP multipoint link. In other words, in a DDCMP multipoint link with n nodes, the router which is the control station on that link has n-1 circuits for that multipoint link.

   . an X.25 circuit -- if there are n nodes reachable to a router via an X.25 network, that X.25 network represents n circuits for that router.

7. congestion -- The condition that arises when there are too many packets to be queued.

8. datagram -- A unit of data passed between the Routing Layer and the End Communications Layer. When a route header is added, it becomes a packet.

9. Designated Router -- the router on the Ethernet chosen to perform additional duties, such as informing the endnodes on the Ethernet of the existence and identity of the Ethernet routers. The router chosen is the one with highest priority, with highest ID breaking ties.

10. end node -- A nonrouting node.

11. error -- The manifestation of a fault.

12. event -- Occurrences that are logged for recording by Network Management. Events result from occurrences of source events.

13. fault -- The cause of a problem in the operation of the Routing Layer modules.

14. hop -- The logical distance between two adjacent nodes in a network.

15. initialization -- A start-up procedure between two adjacent nodes.

16. logical connectivity -- The ability of nodes to communicate.

17. multiaccess channel -- A special type of broadcast circuit on which the channel is shared on a contention basis.

18. nonbroadcast circuit -- Any circuit other than a broadcast circuit. For example, a multipoint DDCMP circuit is not a broadcast circuit because a packet cannot be received by more than one node.

19. originating packet -- A packet from this node's End Communications Layer.

20. packet -- A unit of data to be routed from a source node to a destination node. When stripped of its route header and passed to the End Communications Layer, it becomes a datagram.

21. packet looping -- A condition where a packet revisits a node.

22. path -- The route a packet takes from source node to destination node.

23. path cost -- The sum of the circuit costs along a path between two nodes.

24. path length -- the number of hops along a path between two nodes.

25. physical connectivity -- The result of nodes being attached to each other via active circuits and nodes.

26. reachable node -- a node to which a routing node believes it can direct a packet.

27. received packet -- a packet received by this node's Routing Layer from the Data Link Layer

28. route-through -- packet switching.

29. routing -- Directing data message packets from source nodes to destination nodes.

30. sent packet -- a packet passed from this node's Routing Layer to the Data Link Layer.

31. source event -- a specified occurrence in this node's Routing Layer that may cause a counter to be incremented or an event to be logged.

32. terminating packet -- A packet whose destination is this node.

33. topology -- The physical arrangement and relationships of interconnected nodes and circuits in a network. A legal topology satisfies the requirements of this specification.

34. transit packet -- a packet arriving at this node from a source node and destined for another node.

35. unreachable node -- a node to which a routing node has determined that the path exceeds the maximum hops of the network.

APPENDIX I

REVISION HISTORY

I.1 CHANGES FROM PHASE III

I.1.1 Ethernet Support

1. The concept of lines became two concepts--adjacencies and circuits.

2. An NI Initialization Sublayer was added.

3. Extra routing timers and Hello Timer parameters were added due to the different characteristics of the NI.

I.1.2 Hierarchical Routing

1. Code and parameters for level 2 routers was added.

2. Code and parameters in level 1 routers was added for dealing with interarea packets and "nearest level 2 router".

I.1.3 Segmented Routing Messages

1. Srm flags made into a matrix.

2. Code added for dealing with segmented Routing Messages.

I.1.4 Terminology Changes

1. Transport => Routing (state names and state actions involving word Transport changed also)
2. Phase IV => Phase V
3. 3E => Phase IV
4. NSP => ECL (End Communications Layer)

I.1.5 Clean-ups

1. Self vector removed, Open and Close calls removed
2. Select Process removed
3. MinHops algorithm changed to reflect hops of chosen path
4. Hello Timer communication, and setting Listen Timer to a function of neighbor's Hello Timer
5. Buffer size bug fixed, enabling buffer sizes to be migrated
6. Line Rejection algorithm removed
7. Useless or Wrong diagrams removed
8. Extensible fields changed to single byte fields
9. Formats of packets complete (rather than kept in two places as before)
10. Lots of editing to get it in machinable shape.

I.1.6 Phase II Support Removed

I.1.7 X.25

1. Data Link mapping spec merged in
2. Verification as done removed
3. X.25 version number removed

I.1.8 Miscellaneous

1. Appendix on buffers added

2. Parameter ECL segment size added

3. Redefined arithmetic comparison of NI addresses to have leftmost byte be least significant

4. Changed word "top" to "first" when discussing NI address bytes

5. Fixed bug--level 2 routers initialize HOP(0,0) and COST(0,0) to 0 and also check it in CHECK

6. Reserved field of a byte added to Routing Messages for word alignment

7. Check for S-ID done only when necessary in Forwarding Process.

8. Priority of BRA neighbor added into Adjacency database

9. All mention of NI was changed to Ethernet

10. Optional padding added to all Routing Messages

11. Speed up and clarify Ethernet initialization

12. Graceful going down messages for Ethernet, X.25, and DDCMP

13. Hello Timer-2 bytes

14. BCT3MULT changed to 3 from 8

15. Upper limits placed on values of T3, Maxc, Maxh, AMaxc, AMaxh, Maxv, NN, NA

16. Packet formats renamed "long" and "short" -- Forwarding Process rewritten for when to receive and transmit in each format

17. Circuit parameter "Recall Timer" added

18. In Initialization Sublayer, action "start timer" clarified to be "start Recall Timer". Also, note added that Recall Timer must expire before Data Link Layer reinitialized.

19. OPL (originating packet limiter), put into the circuits database. Text of section 5.2 states that it is set by network management

20. Link Dependent Sublayer of X.25 to specify that if VC name set on incoming switched virtual circuit, number must match or call rejected

21. Changes for detached level 2 router to act as level 1 router

22. Forwarding Size renamed Buffer Size, and increased by 6 to be compatible with Phase III. ECL Segment Size decreased by 6 when being reported to ECL, so that its value can equal BS, yet be a sensible value for ECL.

23. Circuit down, corruption loss, and initialization failure counters changed from 16 to 8 bits, to be compatible with network management. Adjacency down counter removed.

24. Numerical comparison of Ethernet addresses clarified

25. Packet formats reorganized. 1st byte named FLAGS instead of sometimes FLAGS, RTFLG, or CTLFLG.

26. Limit of size of Routing Message specified in Update Algorithm

27. Enabling of multiple areas on Ethernets

28. Adjacency up/down for Designated Router by endnodes

29. Look only at 1 byte of version number when comparing version numbers

30. Note that adjacency events logged as circuit events on point-to-point links

31. Routing Type added to SELF parameters set by Network Management

32. ECL interface--source removed from TRANSMIT, destination removed from RECEIVE

33. Rowmin changed to make routing deterministic by using node ID for breaking ties

34. Rules for intra-Ethernet bit changed to be originating nodes always set, and routers keep it on when forwarding onto same circuit

35. Endnodes on Ethernets not required to have a packet size checker (no adjacency database necessarily)

36. Verification required of Phase III routers to prevent area leakage

37. No checking of reserved bits

38. Inter-area packet not forwarded to destination Phase III node

READER'S COMMENTS

NOTE: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement.

Did you find errors in this manual? If so, specify the error and the page number.

Please indicate the type of user/reader that you most nearly represent.

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Other (please specify) ______

Name ______ Date ______

Organization ______

Street ______

City ______ State ______ Zip Code or Country ______

—Do Not Tear - Fold Here and Tape —

No Postage Necessary if Mailed in the United States

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO.33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

SOFTWARE DOCUMENTATION
1925 ANDOVER STREET TW/E07
TEWKSBURY, MASSACHUSETTS 01876

Do Not Tear - Fold Here and Tape

Cut Along Dotted Line

Printed in U.S.A.