297-2211-800

DMS-100 Family

TOPS ADAS Network Configuration

Reference Guide

ADAS07 and up Standard 01.04 October 1997

NORTEL
NORTHERN TELECOM

DMS-100 Family

TOPS ADAS Network Configuration

Reference Guide

Publication number:	297-2211-800
Product release:	ADAS07 and up
Document release:	Standard 01.04
Date:	October 1997

Printed in the United States of America.

NORTHERN TELECOM CONFIDENTIAL: The information contained in this document is the property of Northern Telecom. Except as specifically authorized in writing by Northern Telecom, the holder of this document shall keep the information contained herein confidential and shall protect same in whole or in part from disclosure and dissemination to third parties and use same for evaluation, operation, and maintenance purposes only.

Information subject to change without notice. Northern Telecom reserves the right to make changes in design or components as progress in engineering and manufacturing may warrant.

Publication history

October 1997

Version 01.04 Standard release for ADAS07.

September 1997

Version 01.03 Standard release for ADAS07.

August 1997

Version 01.02 Standard release for ADAS07.

June 1997

Initial version 01.01 Standard release for ADAS07.

Contents

1.0 Introduction

1.1 When to use this document

Use this document when setting up a network for use with the ADAS Workstation Remote Access feature. The ADAS Workstation Remote Access feature enables service administrators to configure, monitor, and support ADAS service from a remote location. Travel to each ADAS location (for the purpose of performing routing service management or data collection) is eliminated, allowing service data viewing and changing to be performed quickly and easily.

1.2 What this document contains

This document contains information, details, and specifications for configuring a network for use with the ADAS Workstation Remote Access feature. Various overviews, lists of system components, T1 access equipment, custom cables, typical configurations, sample files, guidelines, router/hub details are included in this document.

1.3 Compliance with local policies

This document is written for all Northern Telecom ADAS Workstation Remote Access feature customers with ADAS release 7 or higher. However, note that many telephone companies have company-specific and office-specific policies regarding networks. Review these policies, and resolve any differences between the policies and this document, before configuring the network for use with the ADAS Workstation Remote Access feature.

1.4 How to check the version and issue of this document

The version and issue of the document are indicated by numbers, such as 01.01. The first two numbers indicate the version, which increases each time the document is updated to support a new software release. The second two digits indicate the issue, which increases each time a document is re-issued within the same software release.

2.0 ADAS Workstation Remote Access

2.1 Feature Overview

The ADAS Workstation Remote Access feature enables service administrators to configure, monitor, and support ADAS service from a remote location. Travel to each ADAS location (for the purpose of performing routing service management or data collection) is eliminated, allowing service data viewing and changing to be performed quickly and easily.

All ADAS functions supported by the host workstation, such as ADAS Service Data Administration and Service Monitoring, Service Data uploads, DMS switch access, and Unix access, are available to a remotely connected user. Operations that require direct operator interaction with the ADAS workstation, such as installing or removing tapes from the DAT drive, cannot be performed remotely.

Remote access to an ADAS workstation is provided by an easy to use graphical interface, which allows a user to remotely connect to a site by simply selecting the site from a scrollable list and requesting a connection. The user can add and remove host ADAS sites from the site list as necessary.

For operating instructions and a complete description of the ADAS Workstation Remote Access feature, please refer to the AF6541 feature documentation.

2.2 Network Topology

The access vehicle for Remote Access is provided by industry-standard LAN/ WAN architectures. The local access media for both the ADAS OAM Workstation and a remote workstation is provided by a standard ethernet (10Base-T) network. The WAN connectivity media between the host ADAS Workstation and a remotely connected workstation is immaterial to the ADAS Remote Access feature; however, the only fully verified and supported configuration to date is based on 56 kb/s DS0 links over T1 facilities.

2.2.1 LAN Overview

The local network connectivity for both the host ADAS Workstation and a remotely connected workstation consists of a 10Base-T ethernet network. The ADAS Workstation supports two ethernet interfaces, one of which has been unused to this point.

The first ethernet interface provides connectivity between the ADAS Workstation and the DMS TOPS switch, and is not affected by the Remote Access feature. This interface provides a point-to-point connection between the ADAS Workstation and the EIU, and carries only ADAS traffic. For security reasons, this simple network must consist only of the ADAS Workstation and an EIU.

The second (previously unused) ethernet interface provides the local access point for the Remote Access feature. This permits essentially unlimited external connectivity to the ADAS Workstation without compromising the security and integrity of the existing ADAS network. The second LAN interface connects to an external, third-party router/hub access product, from which numerous WAN technologies may be used to interconnect remote ADAS sites.

Utilizing the second ethernet interface on the ADAS Workstation for Remote Access provides a number of advantages as compared to alternate approaches.

First, an ethernet interface provides an industry standard interface supported by the existing ADAS Workstation hardware and Unix software drivers, as well as innumerable third-party LAN network equipment suppliers.

Second, an ethernet interface permits the use of the networking capabilities inherent in Unix to provide Remote Access capabilities, eliminating the need to develop proprietary software drivers that would be required by other interfaces, such as a modem off of the workstation RS-232 serial port.

Third, the specifics of the routing and connectivity details are provided by the external networking equipment--the Remote Access feature does not have to know the intimate routing details or the specific datacom mechanics of connecting the remote workstation to a host ADAS Workstation.

And finally, using the second ethernet interface on the ADAS workstation permits the existing ADAS ethernet configuration and installed ethernet cabling to remain unchanged. This insures both the integrity of the link between the DSM switch and the ADAS Workstation, and the security[1] of the switch itself. This also eliminates the requirement to modify the existing node configurations for all of the ADAS elements (EIU, APUs, CM,...) that may have been required for alternate Remote Access connectivity options.

[1] Security in this sense implies no foreign traffic is placed directly on the ADAS LAN and no external element has direct access to a dedicated link to the CM of the switch. However, all precautions should be taken to insure the security of the external LAN connected to the ADAS Workstation. For although the ADAS Workstation will not directly route data between the two ethernet interfaces, if external access is gained to the ADAS Workstation the remotely connected user has all access privileges as a user seated directly at the workstation.

2.2.2 WAN Overview

As previously mentioned, the WAN transport used for interconnecting a remote workstation to a host ADAS site is essentially immaterial to the Remote Access feature, in the sense that the Remote Access feature only has visibility of the local 10Base-T ethernet network. For the most part, any WAN transport that can suitably interface between the two local ethernet LANs will suffice for successful operation of the Remote Access feature. However, the only fully verified and supported WAN transport media presently supported is based on T1 facilities. Namely, 56 kb/s sub-rate over a shared T1 span. Which is to say, a DS0 timeslot in a T1 link between the remote workstation and the ADAS sites to which remote access is desired, connected in a ring configuration. This is shown in ***Figure 1 on page 15***, which illustrates four ADAS installations remotely connected in a ring configuration to a two remote workstations.

As stated above, T1 facilities are the only currently supported and verified WAN technology for Remote Access connectivity. Two T1 connectivity options have been defined and lab tested: a high-speed option (1.544 Mb/s) and a low-speed option (56 kb/s or DS0). Both options, as well as any other WAN access/transport technology, function in an identical fashion as far as the Remote Access feature on the ADAS Workstation is concerned. The details of the two options are provided in the following sections.

Low Speed Remote Access Option

The low speed remote access option consists of a 56 kb/s link (one DS0 timeslot) over a T1 span. This bandwidth provides suitable performance for the Remote Access feature in an economical fashion, since the T1 facility can be shared with other applications. The network equipment configuration for the low speed Remote Access option is illustrated in the figure below.

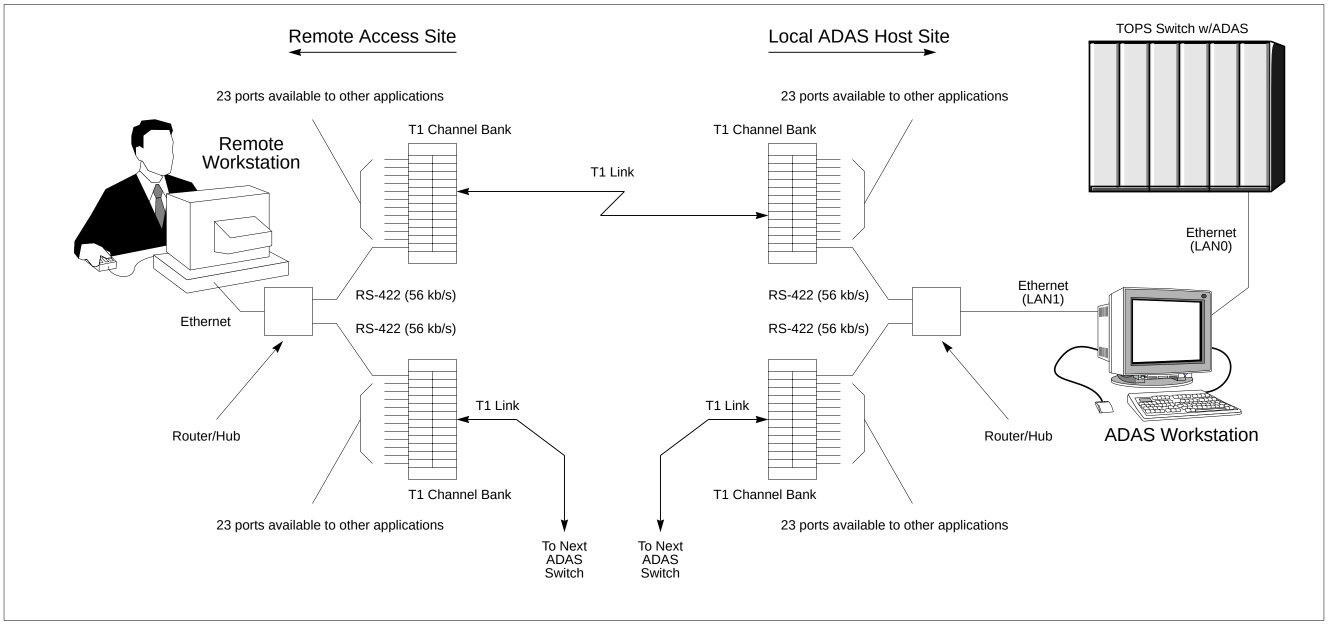
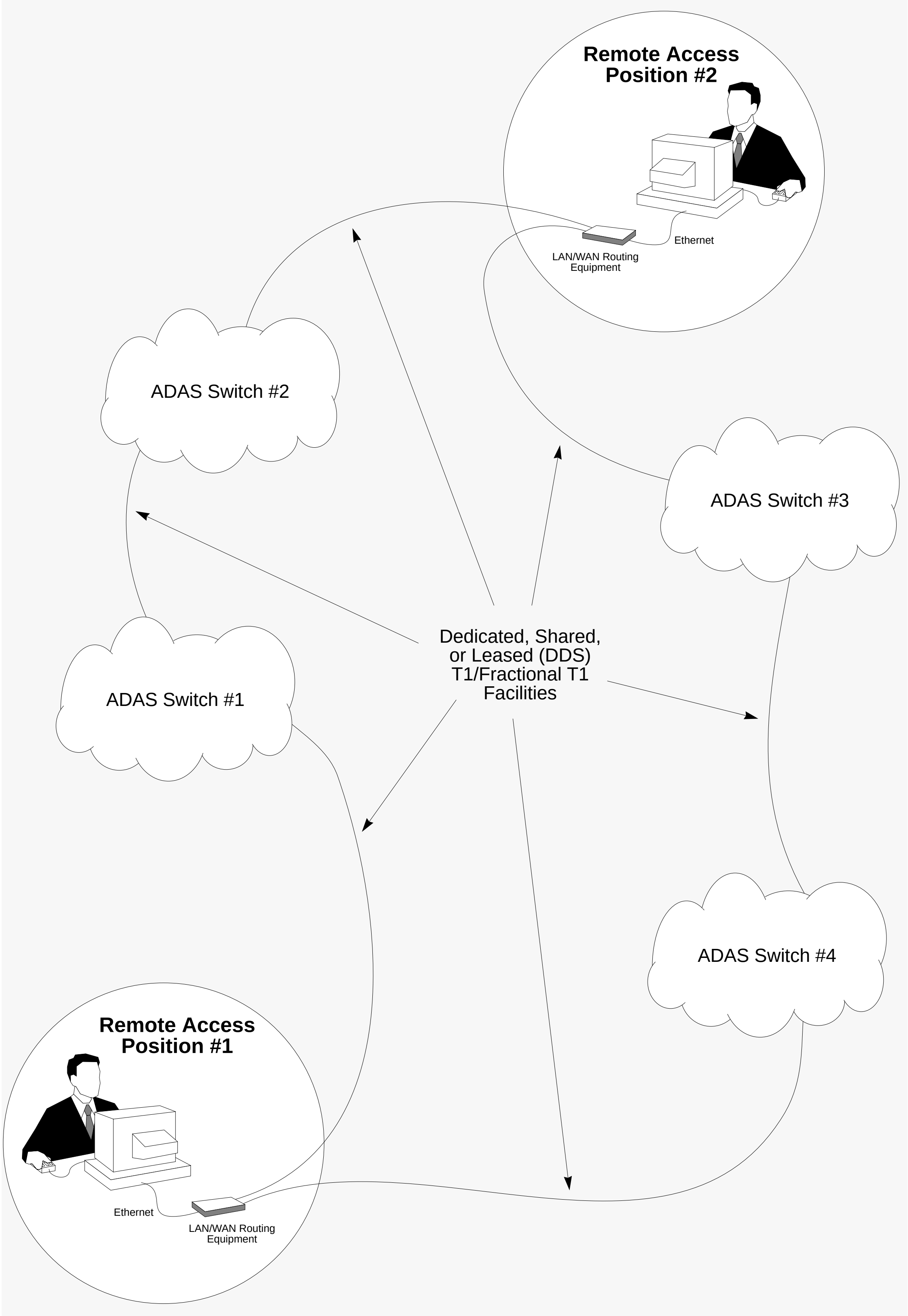
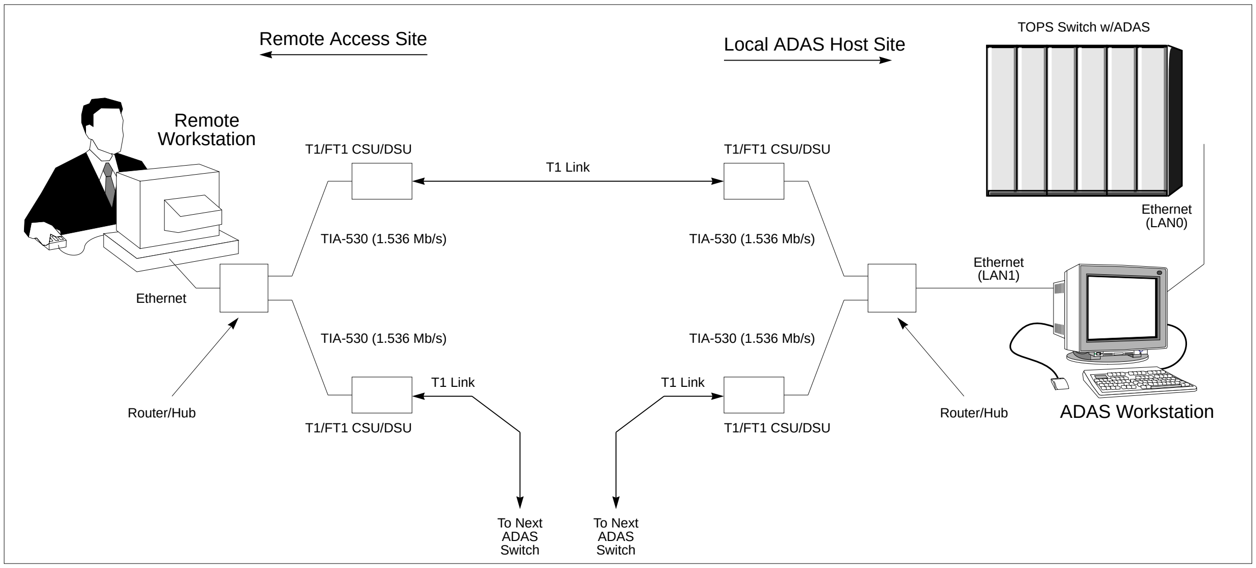

Remote Access Site
Local ADAS Host Site
TOPS Switch w/ADAS
23 ports available to other applications
23 ports available to other applications
Remote Workstation
T1 Channel Bank
T1 Channel Bank
T1 Link
Ethernet (LAN0)
Ethernet (LAN1)
Ethernet
RS-422 (56 kb/s)
RS-422 (56 kb/s)
RS-422 (56 kb/s)
RS-422 (56 kb/s)
T1 Link
T1 Link
Router/Hub
Router/Hub
ADAS Workstation
T1 Channel Bank
T1 Channel Bank
23 ports available to other applications
23 ports available to other applications
To Next ADAS Switch
To Next ADAS Switch

Figure 1 ADAS Remote Access WAN Connectivity



High Speed Remote Access Option

The high speed remote access option consists of an (n x 64) kb/s link up to 1.536 Mb/s over a T1 span. This bandwidth well exceeds the minimum requirements for suitable performance of the Remote Access feature, and would normally only be used in applications where channel banks are not available at all sites, or additional high-bandwidth applications are being performed, since the Remote Access feature itself does not justify the facilities expense of an entire T1. The network equipment configuration for the high speed Remote Access option is illustrated in the figure below.



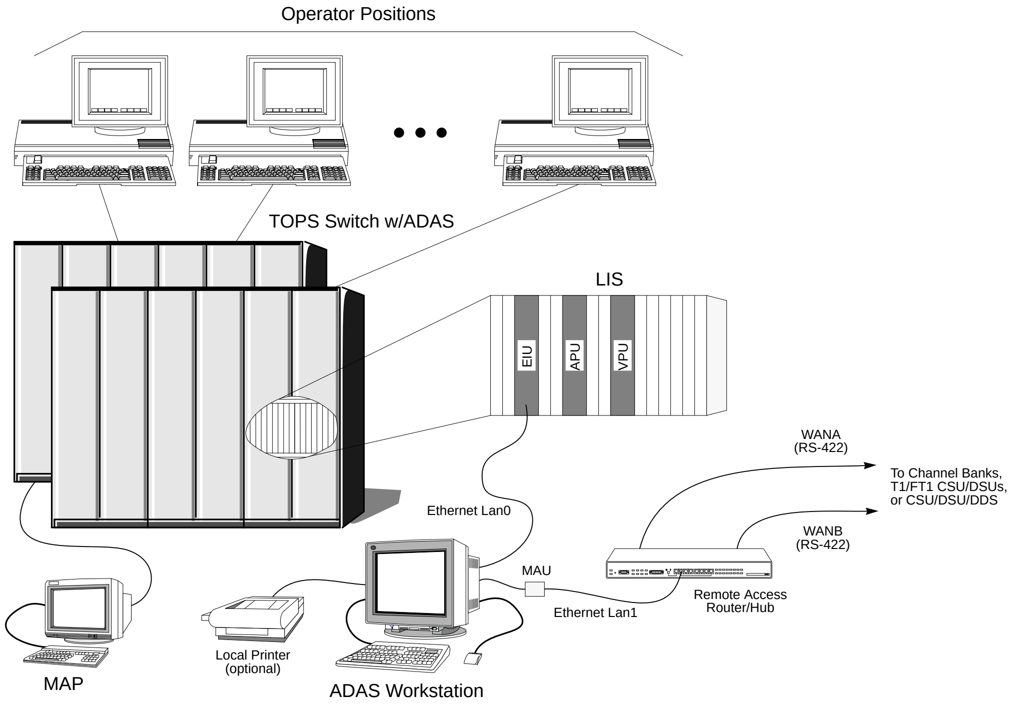
2.3 System Components

Minimal additional components beyond the standard ADAS hardware configuration are required for Remote Access. The necessary components are listed below:

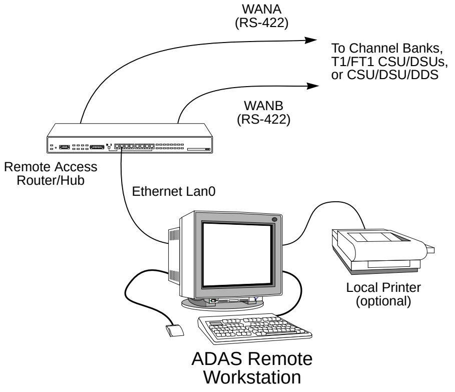
1. ADAS OAM Workstation ***(Available from Northern Telecom)***
2. Remote Access Workstation ***(Available from Northern Telecom)***
3. Router/Hub Network Equipment ***(Supplied by customer)***
4. T1 Access Equipment ***(Supplied by customer)***
   a. T1 Channel Bank
   b. T1/FT1 CSU/DSU
5. Custom Cabling ***(Supplied by customer)***

Figure 2 on page 17 Illustrates the basic equipment configuration for Remote Access for both the host ADAS site and the remote access location.

Figure 2 Basic Configuration for ADAS Remote Access



Basic Configuration of Host Switch for ADAS Remote Access



Basic Configuration of Remote Site for ADAS Remote Access

2.3.1 ADAS OAM Position (Host Workstation)

Northern Telecom CPC:	A0683224
Supplier:	Hewlett Packard Company
Supplier Part Number:	HP 712/60

The Host Workstation is a standard ADAS OAM Position that is connected to an ADAS system. The Host Workstation must be running the TOP07 or higher software load. The Remote Access software load does not need to be installed on the Host Workstation unless it will also be used as a Remote Workstation.

The ADAS OAM Workstation is supplied by Northern Telecom with all ADAS installations.

2.3.2 Remote Access Workstation

Northern Telecom CPC:	A0683224
Supplier:	Hewlett Packard Company
Supplier Part Number:	HP 712/60

The Remote Workstation is located at the remote location and is used to connect to one or more Host ADAS Workstations. The Remote workstation is not required to have the ADAS Workstation software load installed, but it must have the HP-UX 9.03 operating system[1] and the ADAS Remote Access software installed.

The Remote Workstation must be ordered from Northern Telecom as a separate item when the Remote Access feature is ordered.

2.3.3 Remote Access Router/Hub

Supplier:	Bay Networks, Incorporated
Model Number:	BayStack Access Node Hub (ANH)
Supplier Part Number:	AE1001010 (110/120 Vac) -48 Vdc powered units are also available

Supplier:	Bay Networks, Incorporated
Model Number:	AN/ANH Bay Networks Routing Services for IP Access
Supplier Part Number:	AE008032

[1] HP-UX 9.03 is the current operating system supplied with the ADAS Workstation. If this load is upgraded in the future, the operating system for the Remote Access Workstation would also need to be upgraded.

Both the above items must be purchased together for the Remote Access feature. The first item is the LAN/WAN network equipment, and the second is the routing software.

The Remote Access Router/Hub equipment provides the network access for both the local nodes (ADAS Workstations) and the remote node (Remote Workstations). One Router/Hub is required for each site (local or remote) that is to be connected via the Remote Access feature.

The Router/Hub is a cost effective access product that combines an eight port 10Base-T ethernet hub with a dual WAN interface router. The ADAS Workstation (or Remote Workstation) connects to the ethernet hub via standard category 3, twisted pair cable. Each of the WAN ports connects to a T1 access product: either a T1 CSU/DSU for the high speed remote access option, or a channel bank for the low speed option. From the T1 CSU/DSU or channel bank, T1 facilities are used to interconnect the ADAS sites with the remote site.

The Router/Hub network equipment is supplied by the customer. The equipment listed above is available from Bay Networks, Incorporated, and is the only LAN network equipment that has been validated for use with the Remote Access feature. While it is recommended that the customer implement the remote access network with this equipment, network components from alternate vendors may be used to provide equivalent functionality.

Regardless of the equipment used to implement the remote access network, the implementation, management, configuration, and support of the LAN/WAN infrastructure required by the Remote Access feature is solely the responsibility of the customer.

2.3.4 T1 Access Equipment

T1 Channel Bank (Low Speed Access Option)

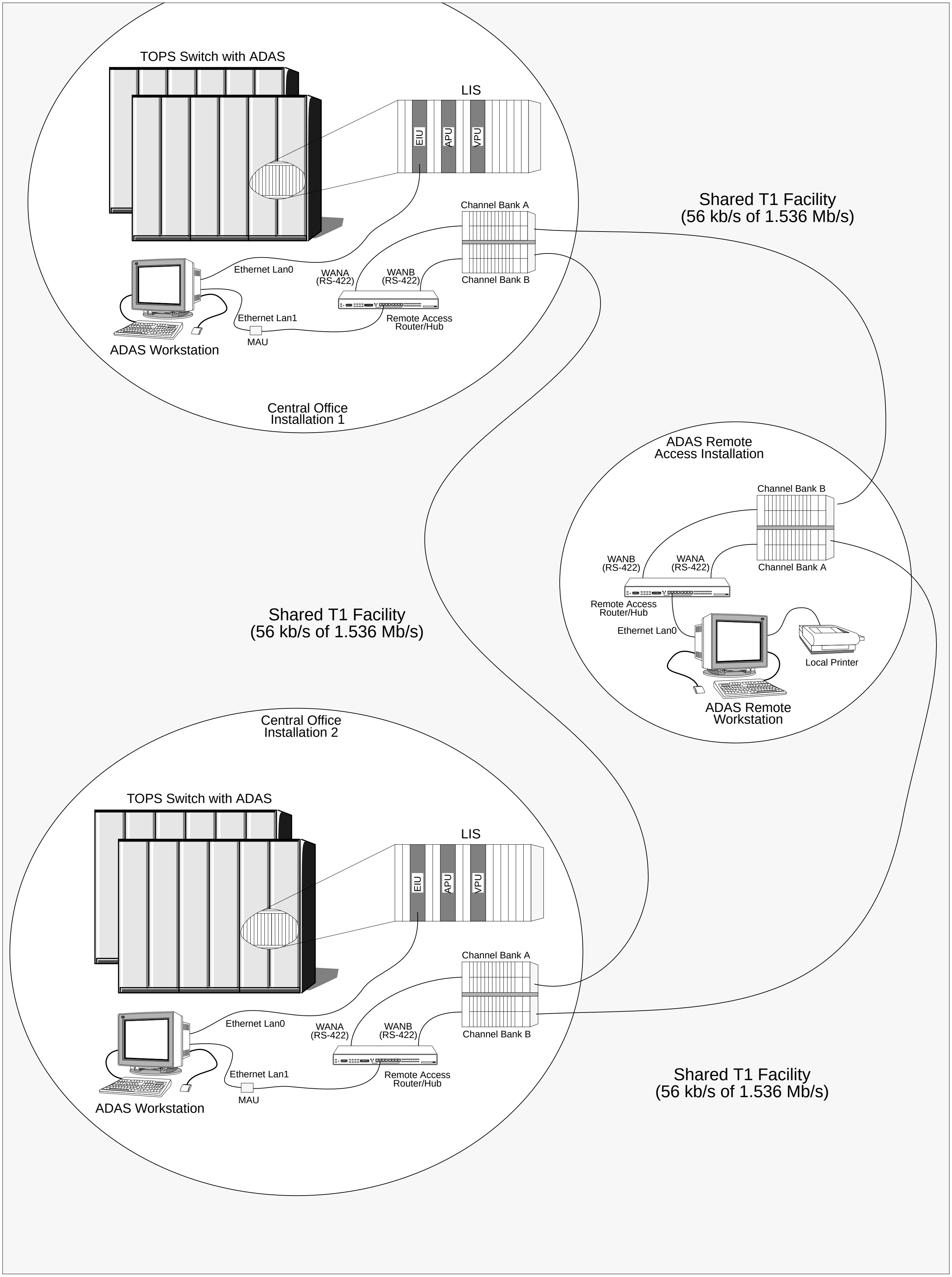
The T1 channel bank provides the T1 facilities interface for the remote access low speed option. It interfaces the RS-422/TIA-530 signals (56 kb/s) from the Router/ Hub and multiplexes the data onto one of the DS0 timeslots of the T1 link.

With the exception of the trivial case of a single ADAS site connected to one remote workstation, two channels banks are required at each site having remote access. This is necessary to support the ring network configuration--one channel bank interfaces the T1 link from the previous ADAS site in the ring, and the second channel bank interfaces the T1 link to the next ADAS site in the ring. ***Figure 3 on page 21*** provides an example of the low speed option for interconnecting ADAS sites to a remote workstation.

IMPORTANT! The T1 channel banks are supplied by the customer. No specific vendor’s product is recommended; however, the selected T1 channel banks must support RS-422 interface cards.

Regardless of the equipment used to provide the T1 facilities, the implementation, management, configuration, and support of the T1 WAN infrastructure required by the Remote Access feature is solely the responsibility of the customer.

Figure 3 ADAS Remote Access Example--Low Speed Option Over Shared T1 Facilities



T1 CSU/DSU (High Speed Access Option)

Supplier:	General DataComm, Incorporated
Model Number:	DT 554S DeskTop T1/FT1 CSU/DSU with TIA-530 I/F
Supplier Part Number:	048A102-003 + 048P042-001

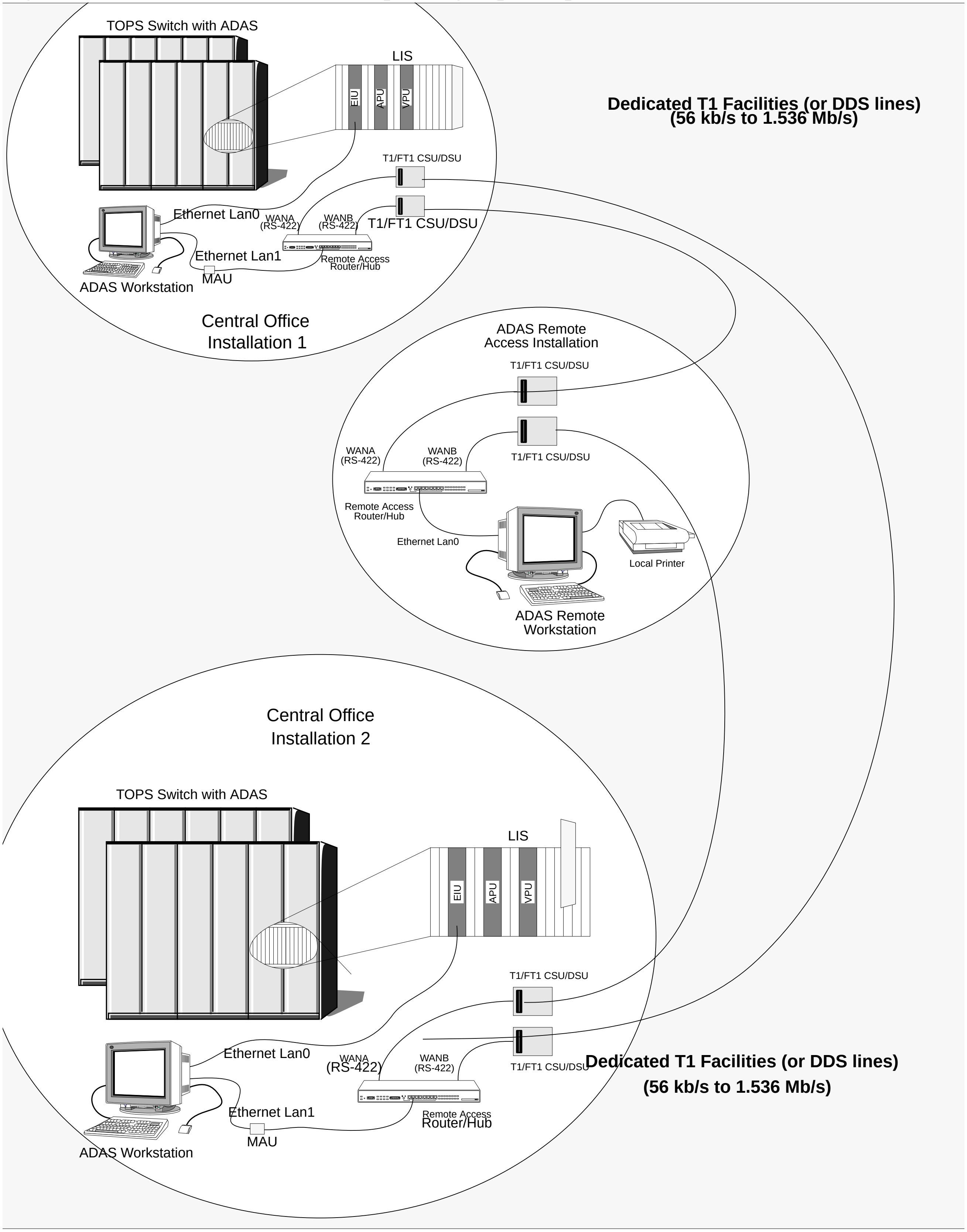
The T1 CSU/DSU provides the T1 facilities interface for the remote access high speed option. It interfaces the RS-422/TIA-530 signals (1.536 Mb/s) from the Router/Hub and formats the data onto the T1 link. Both the CSU functions (protection and signal conditioning) and DSU functions (signaling and data format translation) are provided by the T1 CSU/DSU.

With the exception of the trivial case of a single ADAS site connected to one remote workstation, two T1 CSU/DSUs are required at each site having remote access. This is necessary to support the ring network configuration--one CSU/DSU interfaces the T1 link from the previous ADAS site in the ring, and the second CSU/DSU interfaces the T1 link to the next ADAS site in the ring. ***Figure 4 on page 23*** provides an example of the high speed option for interconnecting ADAS sites to a remote workstation.

The T1 access equipment is supplied by the customer. The T1 CSU/DSU equipment listed above is available from General DataComm, Incorporated, and is the only T1 equipment that has been validated for use with the Remote Access feature. While it is recommended that the customer implement the remote access T1 network with this equipment (for the high speed option), network components from alternate vendors may be used to provided equivalent functionality.

Regardless of the equipment used to provide the T1 facilities, the implementation, management, configuration, and support of the T1 WAN infrastructure required by the Remote Access feature is solely the responsibility of the customer.

Figure 4 ADAS Remote Access Example--High Speed Option Over Dedicated T1 Facilities



2.3.5 Custom Cables

The IS9X85ZA Interconnect Schematic should be consulted for the full interconnect details of the ADAS System and the Remote Access installations and options. Reference cable assembly drawings for all the custom cables can be found in ***Appendix A***. However, for customer or third-party cable fabrication, the formal cable assembly drawings should be requested from Northern Telecom.

2.3.5.1WAN Cables: Low Speed Access

The low speed remote access option requires two WAN cables to complete the connection of the Router/Hub to the T1 channel bank; an NTNX36SG TIA-530 WAN cable and an NTNX36SK RS-422 WAN cable. Two cables are employed to ease cable routing and connectivity between the channel bank and the Router/Hub; one cable connects from the channel bank to an MDF panel, and the other cable completes the connection from the MDF panel to the Router/Hub.

The NTNX36SG cable connects from the Router/Hub to the MDF panel. It consists of a 44-pin, high-density D-sub connector on one end which mates with the Router/Hub, and unterminated, loose wires on the other end, which are punched down at the MDF.

The NTNX36SK cable connects from the MDF panel to the T1 channel bank. It consists of a 25-pin D-sub connector on one end which mates with the RS-422 card in the channel bank, and unterminated, loose wires on the other end which are punched down at the MDF to match up with the NTNX36SG cable, completing the connection from the channel bank to the Router/Hub.

The interconnection of the Router/Hub with the T1 channel bank via the NTNX36SG/NTNX36SK cable pair is illustrated in ***Figure 5 on page 26***.

Both the NTNX36SG and the NTNX36SK cables are engineered length cables. The end-to-end combined length of the cables must not exceed 1500 feet at 56 kb/s to insure reliable operation of the RS-422 interfaces.

IMPORTANT! The NTNX36SG and NTNX36SK cables are supplied by the customer, either via third-party cable vendors, or customer fabrication. Please refer to the formal cable drawings.

TIA-530 Low Speed WAN Cable (Low Speed Remote Access Option)

Design Authority:	Northern Telecom
Reference Drawing Number:	NTNX36SG

RS-422 Low Speed WAN Cable (Low Speed Remote Access Option)

Design Authority:	Northern Telecom
Reference Drawing Number:	NTNX36SK

2.3.5.2WAN Cables: High Speed Access

The high speed remote access option requires one WAN cable (NTNX36SE) to complete the connection between the Router/Hub and the T1 CSU/DSU, and one cable (NTGB0182) to connect the T1 CSU/DSU to the T1 network via an MDF panel for convenient cable routing.

The NTNX36SE cable connects from the Router/Hub to the T1 CSU/DSU. It consists of a 44-pin, high-density D-sub connector on one end which mates with the Router/Hub, and a 34-pin, Winchester connector on the other end to mate with the T1 CSU/DSU.

The NTGB0182 cable connects from the MDF panel to the T1 CSU/DSU. It consists of an 8-pin, RJ-48C modular jack on one end which mates with the network port of the CSU/DSU, and unterminated, loose wires on the other end which are punched down at the MDF.

The interconnection of the Router/Hub with the T1 CSU/DSU via the NTNX36SE cable, and the CSU/DSU with the T1 facilities via the NTGB0182 cable, is illustrated in ***Figure 6 on page 26***.

Both the NTNX36SE WAN cable is a fixed length of 2 feet, which permits convenient collocation of the Router/Hub and T1 CSU/DSU. The NTGB0182 T1 cable is length engineered per site requirements.

IMPORTANT! The NTNX36SE and NTGB0182 cables are supplied by the customer, either via third-party cable vendors, or customer fabrication. Please refer to the formal cable drawings.

TIA-530 High Speed WAN Cable (High Speed Remote Access Option)

Design Authority:	Northern Telecom
Reference Drawing Number:	NTNX36SE

T1 Network Cable (High Speed Remote Access Option)

Design Authority:	Northern Telecom
Reference Drawing Number:	NTGB0182

Figure 5   Low Speed Remote Access WAN Cabling Details

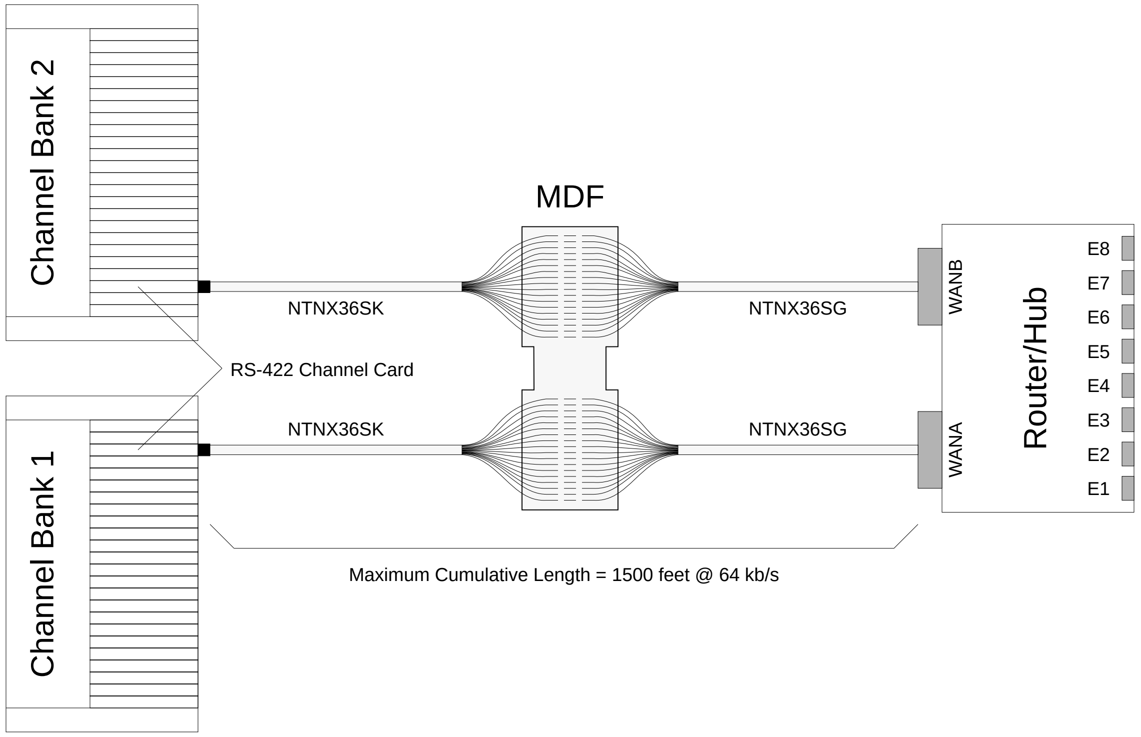
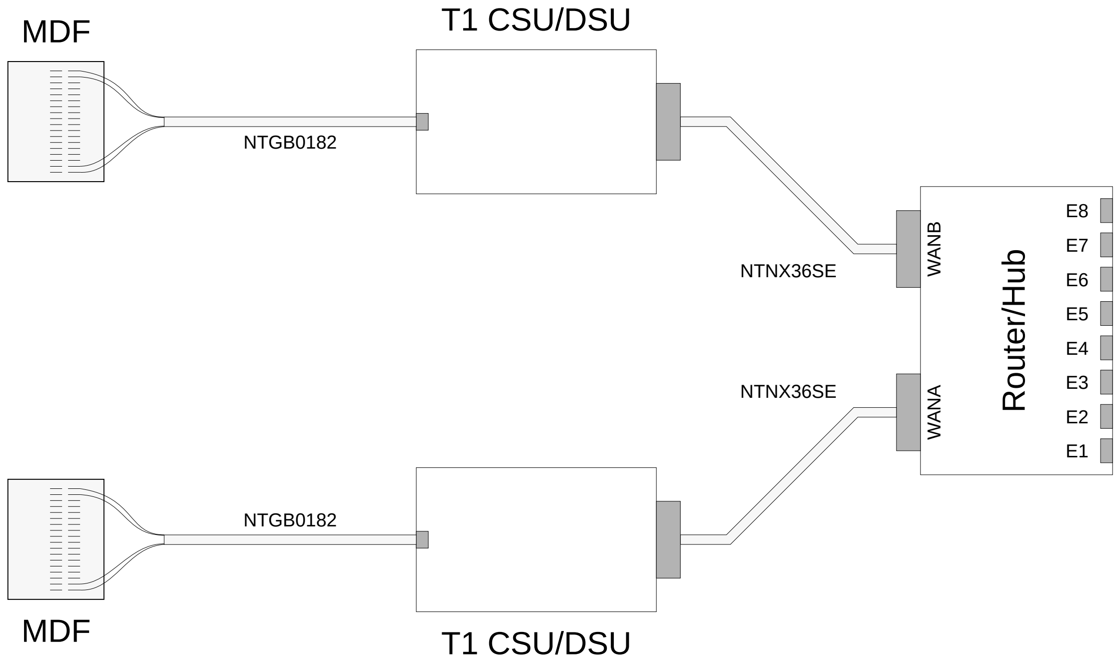


Figure 6   High Speed Remote Access WAN Cabling Details



2.3.5.3LAN Cables: Ethernet MAU-to-Hub

The NT9X8511 ethernet cable provides the 10Base-T LAN connectivity between the ethernet hub of the Hub/Router and the ADAS Workstation (or remote workstation). This cable consists of standard, category 3, twisted pair cable terminated with RJ-45 modular connectors on both ends. This cable is intended for MAU to ethernet hub connections, so the wiring is straight through, i.e., pin 1 to pin 1, pin 2 to pin 2,...

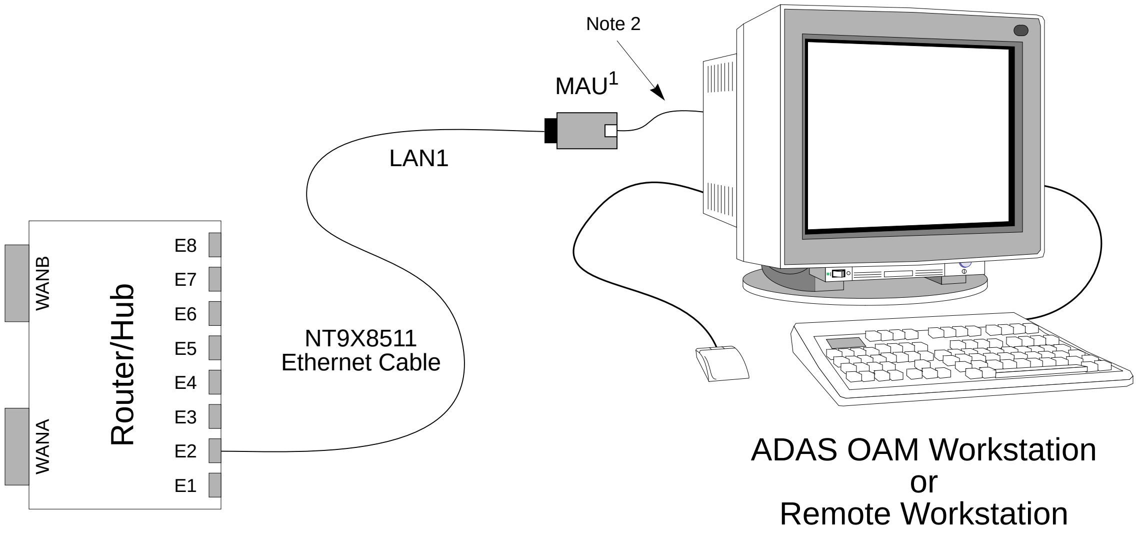
The NT9X8511 cable is length engineered per site requirements, with a maximum length of 100 meters to maintain compliance with the 802.3 ethernet LAN standards.

IMPORTANT! The NT9X8511 ethernet cable is supplied by the customer, either via third-party cable vendors, or customer fabrication. Please refer to the formal cable drawing.

MAU-To-Hub Ethernet LAN Cable

Design Authority:	Northern Telecom
Reference Drawing Number:	NT9X8511

Figure 7 MAU-to-Hub Ethernet LAN Cabling Details



Note 1: The MAU is not required for the remote workstation--the NT9X8511 cable should be connected directly to the 10Base-T ethernet port (LAN0) on the remote workstation for proper operation. Since the 10Base-T (LAN0) port on the ADAS Workstation is already dedicated to the EIU LAN, the LAN1 connection must be provided by a MAU connected to the second ethernet port.

Note 2: Combo serial port/AUI LAN cable; HP part # A2263-62045. (Provisioned as part of the ADAS Workstation configuration.)

2.4 Remote Access Network Configuration

Now that the various components required for ADAS Remote Access have been briefly introduced, the configuration of these elements can be discussed. But before the equipment configuration details are presented, the Remote Access LAN/WAN network must be considered. The following issues must be addressed before a customer implements the ADAS Remote Access feature.

1. Will the Remote Access network remain private, or will it (eventually) be integrated into the customer's business computing network?

   Northern Telecom recommends that the Remote Access network remain a private network interconnecting ADAS sites only, and Northern Telecom will only support Remote Access installations that conform to this configuration. This document presupposes that the Remote Access network is, and always will be, private.

   However, customers who choose to incorporate the ADAS Remote Access network into their existing business network must insure all IP addresses assigned to the Remote Access network elements conform with standard network addressing as assigned by the Internet Network Information Center, to insure compatibility with existing IP networks. Also, in order to maintain the security of the DMS switch environment, the customer must insure suitable security measures are in place, i.e., a firewall or equivalent device, to prevent unauthorized or malicious access to the ADAS network/DMS switch environment.

2. What does the overall Remote Access network look like?

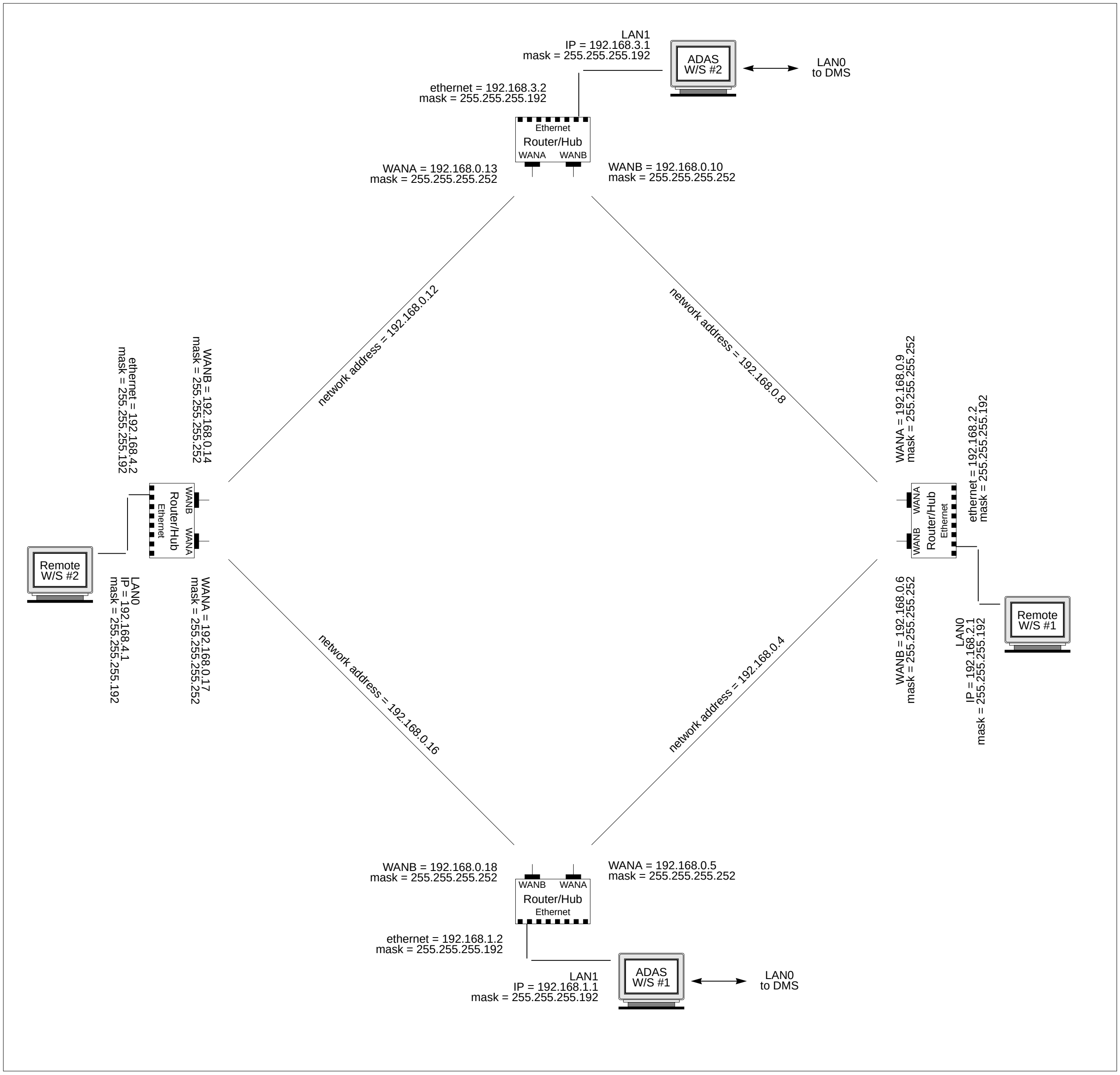
   Before installing the ADAS Remote Access feature, the customer should plan out the overall network, with consideration to such issues as the location of the remote access workstation(s), the number of ADAS sites, the availability of the T1 facilities, the routing of the T1 facilities between the ADAS installations,... While the Remote Access feature permits the addition and removal of ADAS sites as required, up front planning of the network will minimize future efforts if modifications to the network are required. Network planning is of particularly importance, since the Remote Access network is configured in a ring topology and the addition or removal of nodes to the ring requires rerouting the T1 facilities connecting the newly introduced node(s).

3. Network IP addressing scheme.

   Whether the Remote Access network is private or integrated into the customer's business network, consideration should be given to the IP network addressing of the ADAS nodes to insure network compatibility of the ADAS nodes. If the recommendations of this document are followed, there should be no network compatibility issues. Customer specific installations are not supported by Northern Telecom.

Figure 8 on page 29 illustrates a hypothetical Remote Access network that conforms to the configuration recommendations found later in this document. This figure, illustrating two Remote Access workstations connected to two different host ADAS sites, serves to illustrate the WAN network configuration, as well as the ethernet configuration of the ADAS and remote workstations and router/hubs.

Figure 8 Hypothetical Remote Access Network



A number of points should be noted regarding the network and equipment configuration illustrated in the figure above.

- Every ethernet node connected to the remote access network has a unique IP address.
- Each WAN segment of the network, represented by a line connecting the WANA port of one router/hub to the WANB port of another router/hub, physically consists of a T1 facility (or a single DS0 timeslot in a T1 facility).

  Note: For clarity, the necessary T1 CSU/DSUs and/or T1 channel banks are not shown in the above figure, as these components only provide the access/transport facilities for Remote Access and do not affect the ethernet configuration of the workstations or network equipment.
- A single class C network (192.168.0.x) provides the WAN connectivity for the ADAS local and remote sites.

  Note: The LAN standards define a class C network with the address of 192.168.x.x as reserved for private networks. That is, if a customer's network is ***never*** connected to the public network (the internet, for example) IP addresses composed of 192.168.x.x may be freely used. However, these IP addresses ***are not*** guaranteed to be unique outside the customer's private network, and could cause problems if the customer's network is connected to a public network. Since it is recommended that the ADAS Remote Access network remain a private network, the 192.168.x.x IP addresses are used for all nodes connected to this network. If the customer chooses to connect the ADAS Remote Access network to their corporate business network, formal, i.e., "real" IP addresses should be assigned.
- Using the appropriate subnet mask (255.255.255.252), the class C network providing the Remote Access WAN connectivity is partitioned into a number of smaller networks (subnetworks), each supporting four IP addresses, as follows:

  1. the address of the network
  2. the address of WAN port A of the first router/hub
  3. the address of WAN port B of the second router/hub
  4. the broadcast address.
- Each ADAS Workstation LAN1 interface and each Remote Access workstation LAN0 interface is connected to a unique class C network, with each network supporting up to 62 hosts (using the 255.255.255.192 subnet mask). Two host IP addresses are assigned for each local ethernet LAN:

  1. the address of the ethernet hub
  2. the address of the workstation.
- The subnet masks for each element of the network are consistent among all elements connected to that network, i.e., all of the WAN segments have the

identical 255.255.255.252 subnet mask, and all of the workstations and ethernet hubs have the same 255.255.255.192 subnet masks. Connecting two or more nodes having different subnet masks to the same ethernet network may result in unexpected and probably undesired results.

The specific details regarding the mechanics of how the workstation and network equipment is configured are provided in the appropriate sections later in this document.

3.0 Workstation Configuration

3.1 ADAS OAM Workstation

The ADAS Workstation requires no additional hardware in order to support the Remote Access feature. The workstation has always been provisioned with a second ethernet interface, that to this point, has been unused.

The second ethernet interface is provided by an add-on RS-232 serial port/ethernet interface card. A combo serial port/ethernet AIU cable (HP part number A2263-62045) provides standard connectorization for each port, i.e., a 9-pin D-sub connector for the serial port and a 15-pin D-sub connector for connection to a 10Base-T MAU. Both the ethernet MAU (HP part number 28685B) and the serial port/ethernet AUI cable have always been provisioned with the ADAS Workstation. If either of these components are unavailable, they must be procured (from Hewlett Packard) and installed before the Remote Access feature can function.

No additional software beyond the TOP07/ADAS07 Workstation load is required for the Remote Access feature to function on the ADAS Workstation[1]. However, a number of changes are required to one of the workstation configuration files in order for remote access to properly function. These changes are limited to the **/etc/netlinkrc** file and are summarized below:

1. The LAN0 ethernet interface is subnetted with an appropriate class C network subnet mask.
2. The default route/gateway is removed from LAN0.
3. A static route is added to LAN0 for the CM network. This permits all existing DMS-based ethernet nodes (EIUs, APUs, CM) to be accessed from the ADAS Workstation without requiring changes to IP addresses currently assigned to these nodes.
4. The LAN1 ethernet interface (remote access port) is enabled with a new, unique IP address.
5. The default route/gateway is added to the LAN1 network. This permits access to/from any remote site without having to include the IP address of the remote site in the routing tables of the ADAS Workstation. (All necessary routing details for intersite access are automatically collected by the Router/Hub equipment once it has been configured and connected to the network.)

[1] If the ADAS Workstation is also going to function as a remote access position, the Remote Access software must also be loaded onto the ADAS Workstation.

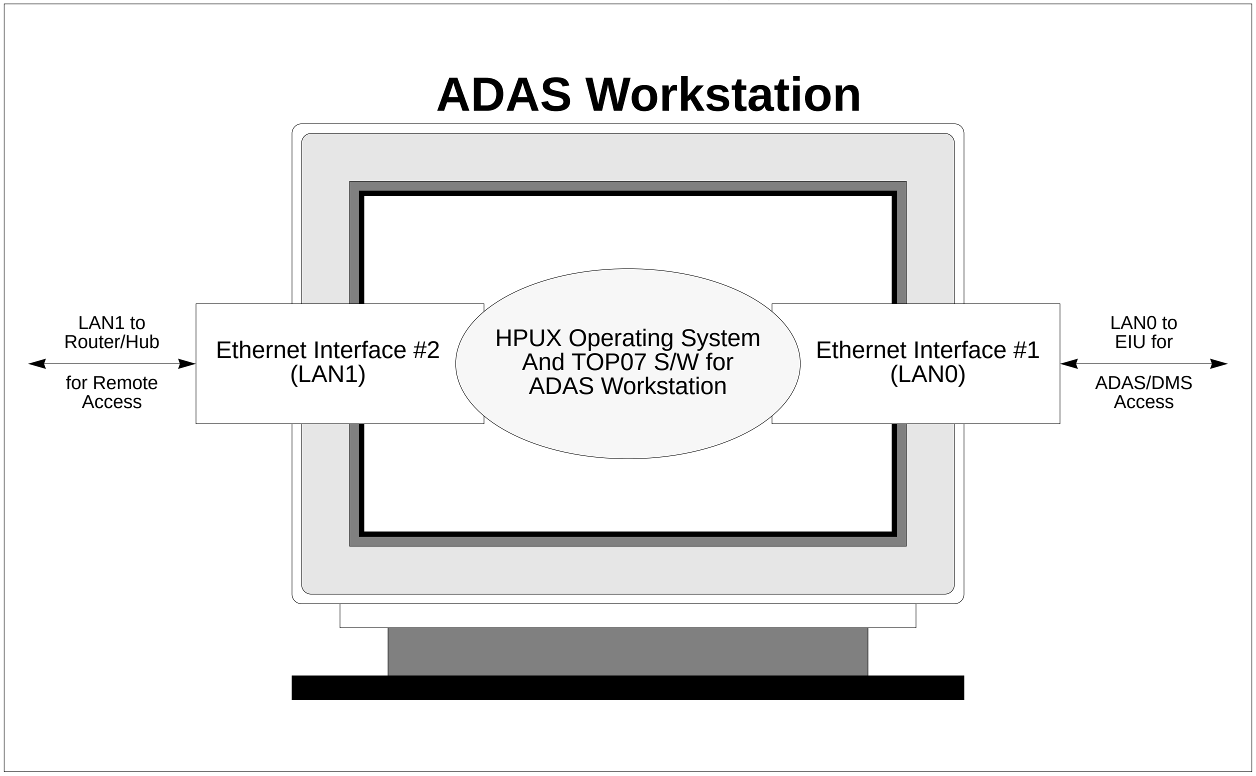
For an example of the **/etc/netlinkrc** file for the ADAS OAM Workstation with the changes above, please see ***Section 3.1.2 on page 37***.

Please note again that no changes are required to any of the existing IP addresses of any of the nodes already connected at an ADAS installation. These nodes include the ADAS Workstation, the EIUs, APUs, and CMs.

Finally, the ADAS Workstation comes provisioned with all of the necessary software drivers installed and bound into the Unix kernel. Activation of the Remote Access feature does not require the Unix kernel on the workstation to be rebuilt.

A pictorial representation of the software and hardware interaction of the Remote Access feature on the ADAS Workstation is provided in ***Figure 9*** below.

Figure 9 ADAS Workstation Ethernet Connectivity for Remote Access


ADAS Workstation
LAN1 to Router/Hub
for Remote Access
Ethernet Interface #2 (LAN1)
HPUX Operating System And TOP07 S/W for ADAS Workstation
Ethernet Interface #1 (LAN0)
LAN0 to EIU for
ADAS/DMS Access

3.1.1 ADAS Workstation Remote Access Configuration Guidelines

The customer is required to perform very few actions with regards to actually making changes to the ADAS Workstation to support the Remote Access feature, as nearly all of the configuration changes are performed automatically by the ADAS installation scripts. However, customers wishing to perform a customized installation of the Remote Access feature to incorporate it into their corporate network should adhere to the following guidelines.

Note: Northern Telecom strongly advises against, and will not support, customized installations of the Remote Access feature.

The guidelines below should be followed when configuring the ADAS OAM Workstation for use with the Remote Access feature. It may be helpful to reference ***Figure 8 on page 29*** to understand the relationship of the IP address assignments to the various nodes in the Remote Access network. To understand the syntax of the commands used to perform specific configuration actions, refer to the sample **netlinkrc** file in ***Section 3.1.2 on page 37***.

IMPORTANT! The following configuration details presume the ADAS installation has been completed following the directions of ***Installation Method 20-2020 "Initial Installation for ADAS"***. ADAS installations following the directions in **IM 20-2020** will be identifiable by ADAS elements having IP addresses of the form ***192.1.1.x***. If upon interrogation of the network it is determined that the ADAS installation is not in conformance with **IM 20-2020**, the customer should contact Nortel Customer Support for assistance with the installation of the Remote Access feature.

The first step is to configure the **LAN0** ethernet interface in the ADAS Workstation. Since this interface already has an IP address assigned (that will not be changed), the **LAN0** changes only involve ***subnet mask*** and ***route*** changes.

Note: The changes listed below are made in the **/etc/netlinkrc** file. All of the changes could be made via appropriate **ifconfig** and **route** commands at the Unix prompt, but changes made in this fashion are not permanent and will not survive a reboot of the workstation.

1. Change the **ifconfig** entry for **LAN0** to modify the subnet mask to ***255.255.255.192***. The **ifconfig** entry for **LAN0** will now appear as

   /etc/ifconfig lan0 inet 192.1.1.66 netmask 255.255.255.192 up

2. Remove all **route add** entries pertaining to **LAN0**.

3. Add a single **route add** entry associated with **LAN0** to provision the EIU as the gateway for DMS-bound ethernet messages. If the ADAS installation is consistent with **IM 20-2020**, the **route add** entry will appear as

   /etc/route add net 192.1.1.128 192.1.1.65 1

   If a new install is being performed, or an upgrade is being performed on an ADAS installation that is not consistent with **IM 20-2020**, follow the directions provided in ***Section 3.1.3 on page 46*** to ascertain the necessary information, i.e. the IP address of the CM network and the ethernet address of the EIU, required by the **route add** command. Using the information collected above, the **route add** command would have the following form:

   /etc/route add net <*CM Network IP Address*> <*EIU Ethernet Address*> 1

The next step is to configure the **LAN1** ethernet interface for the ADAS Workstation.

4. Select a unique IP address for the **LAN1** ethernet interface in the ADAS Workstation. The IP address should be selected from the private, class C network ***192.168.x.x***.

   Note: Northern Telecom has adopted the following address assignment convention for Remote Access. The ***first*** workstation (ADAS OAM Position **or** Remote Access workstation) added to the Remote Access network is assigned IP address ***192.168.1.1***. The ***second*** workstation added to the network is assigned IP address 192.168.***2***.1, the ***third*** workstation is assigned IP address 192.168.***3***.1., the ***fourth*** workstation is assigned IP address 192.168.***4***.1,...

5. The subnet mask for the **LAN1** ethernet interface in the ADAS Workstation is defined to be ***255.255.255.192*** for all workstations added to the network. Therefore, the **ifconfig** entry for **LAN1** for the ***first*** workstation added to the Remote Access network would appear as

   /etc/ifconfig lan1 inet 192.168.1.1 netmask 255.255.255.192 up

6. The associated **lanconfig** command must be added for **LAN1** as follows

   /etc/lanconfig lan1 ether

7. Finally, a single route is added to the routing tables for **LAN1**, setting the ***default*** route to be a **gateway** with the IP address of ***192.168.x.2***, where x = 1,2,3,4,... depending on the workstation number. For example, the ***first*** workstation would have the default route set to 192.168.***1***.2, the ***second*** workstation would use IP address 192.168.***2***.2, the ***third*** workstation would use 192.168.***3***.2, the ***fourth*** workstation would use 192.168.***4***.2,... Therefore, the **route add** entry associated with **LAN1** of the ***first*** workstation added to the Remote Access network would appear as

   /etc/route add default 192.168.1.2 1

3.1.2 Sample netlinkrc File for the ADAS Workstation

The **netlinkrc** file resides in the **/etc** directory of the ADAS Workstation and provides, among other things, the configuration details of the LAN interfaces installed in the workstation. The **netlinkrc** file is read and executed each time the ADAS Workstation is rebooted or powered up. Changes to the LAN interfaces that are intended to be permanent are placed in the **netlinkrc** file. Temporary changes to the ethernet interfaces can be made at any time through the execution of the appropriate Unix commands, or permanent changes can be made via sam (System Administration Manager).

Note: The installation of the ADAS Workstation software for TOP07/ADAS07 will automatically configure the appropriate files for remote access as part of the installation procedure.

The file below illustrates the **netlinkrc** file for the ADAS Workstation, and includes the modifications required to support the Remote Access feature.

Note: The **netlinkrc** file shown below is consistent with ADAS installations that conform with the instructions in **IM 20-2020** for IP address assignments of ADAS nodes. Installations not conforming with **IM 20-2020** will require different **ifconfig** and **route add** entries for the **LAN0** interface. Refer to ***Section 3.1.3 on page 46*** for information as to how to determine the **ifconfig** and **route add** entries.

Sample netlinkrc file for the /*first*/ workstation added to the Remote Access network

Note: Modifications required for the Remote Access feature are indicated by ***bold italic*** type. These modifications include changes to existing configuration commands, as well as, entirely new commands added to the file.

Note: This sample netlinkrc file represents the configuration for the **first** (ADAS) workstation added to the Remote Access network. Subsequent workstations added to the network will have different entries for the **ifconfig** and **route** commands to reflect the appropriate IP addresses and networks.

```
#! /bin/sh
## Configured using SAM by root on Thu Oct 26 16:33:06 1995

# @(#)netlinkrc: $Revision: 1.6.109.7 $ $Date: 92/07/13 08:21:12 $
# $Locker: $

#
# Shell script for initialization of link networking product.
#
# net_init flag is used for Instant Ignition. If net_init is set,
# then netlinkrc return “exit 1”. In order for Instant Ignition
# to work correctly, netlinkrc needs to check the STATUS variable
# after each program or scripts it calls.
#
net_init=0
```

```
if [ -f /etc/clusterconf ]
then
      ROOTSERVER=`/bin/cnodes -r`
      NODENAME=`/bin/cnodes -m`
      DOMAIN=`/bin/cnodes -r`
      ORGANIZATION=diskless
else
      ROOTSERVER=`hostname`
      NODENAME=$ROOTSERVER
      DOMAIN=`/bin/uname -n`
      ORGANIZATION=standalone
fi

#
# Start logging daemon *before* any other networking initialization.
# See nettl(1m) for more information.
#
/etc/nettl -start
STATUS=$?
if [ ! $STATUS -eq 0 ]
then
      net_init=1
fi
#
# Remove the existing /etc/netstat_data file.  The first time
# netstat is executed, a new /etc/netstat_data file will be
# created.
#
/bin/rm -f /etc/netstat_data

#
# Initialize networking interfaces.
#
# (STEP 1)
#
# The ifconfig(1m) command assigns an IP address to a LAN interface and
# configures network interface parameters.  The lanconfig(1m) command
defines
# the packet encapsulation method for the LAN interface.
#
# The “case $NODENAME” construct below allows each node in a diskless
cluster
# to execute node specific calls if necessary.  Add entries to
# the case construct for specific nodes in the diskless cluster only if
# needed. For example, if a specific node has more than one LAN interface,
# the node must execute separate commands for each of the interfaces.
#
# For example:
#
#     case $NODENAME in
#     $ROOTSERVER)/etc/ifconfig lan0 inet 192.6.1.3 up
#                 /etc/lanconfig lan0 ether
#                 /etc/ifconfig lan1 inet 15.4.64.1 netmask
255.255.248.0 up
#                 /etc/lanconfig lan1 ether
#                 ;;
#           *)    /etc/ifconfig lan0 inet `hostname` up
#                 /etc/lanconfig lan0 ether ieee
#                 ;;
#     esac
#                 /etc/ifconfig lo0 inet 127.0.0.1 up
#
# assigns to the two interfaces lan0 and lan1 on a rootserver the DARPA
# Internet addresses 192.6.1.3 and 15.4.64.1 respectively; the lan0
# interfaces on all other nodes (* is the wildcard) are assigned their
# respective internet addresses as found in /etc/hosts.
#
# The ifconfig command line below is sufficient to initialize the network
# interface for any node that has one LAN interface card and whose
# hostname and Internet address are present in the hosts(4) file.
#
# NOTE:If the ifconfig command line does not specify a subnet mask,
#     the subnet mask defaults to the network mask.
#     It is not necessary for both encapsulation methods to be turned on
#     for the LAN Interface. For further explanation see lanconfig(1m)
#
# The loopback interface must be explicitly configured for each address
# family of interest. The following command assumes that the hostname
# has already been set and is mapped to an IP Address in /etc/hosts.
#
# SEE ALSO: ifconfig(1m), lanconfig(1m)

case $NODENAME in
      $ROOTSERVER)
      /etc/ifconfig lan0 inet 192.1.1.66 netmask 255.255.255.192 up
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      /etc/lanconfig lan0 ether
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi

      /etc/ifconfig lan1 inet 192.168.1.1 netmask 255.255.255.192 up
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      /etc/lanconfig lan1 ether
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      ;;
      *) /etc/ifconfig lan0 inet `hostname` netmask 255.255.240.0 up
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      /etc/lanconfig lan0 ether
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      ;;
esac
      /etc/ifconfig lo0 inet 127.0.0.1 up
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi

# The x25init(1m) command configures X.25 network interface parameters.
The
# “case $NODENAME” construct below allows each node in a diskless cluster
# to execute node specific x25init calls if necessary. Add entries to
# the case construct for specific nodes in the diskless cluster only if
# the nodes have X.25 interfaces. The nodes must execute separate x25init
# commands for each of the interfaces. The STATUS checking is for Instant
# Ignition.
#
# For example:
#
#     case $NODENAME in
#           NODEA ) /etc/x25init -c /etc/x25/config_filename1
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           /etc/x25init -c /etc/x25/config_filename2
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           /etc/x25init -a /etc/x25/ip_x25_mapfile
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           ;;
#           NODEB ) /etc/x25init -c /etc/x25/config_file_nodea
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           ;;
#     esac
#
# initializes two X.25 interfaces on NODEA and one interface on NODEB.
# For nodes which have IP configured over X.25, the x25init -a command
# provides the mapping of IP Addresses to X.121 addresses. It is
recommended
# to put the configuration and ipmap files in the /etc/x25 directory.
#
# In the above example, at least one of NODEA’s X.25 Cards supports IP
# since IP-to-X.25 Map table is initialized on NODEA.
#
# SEE ALSO: x25init(1m)

#
# Initialize network routing.
#
# (STEP 2) (OPTIONAL, FOR NETWORKS WITH GATEWAYS ONLY)
#
# The route(1m) command manipulates the network routing tables.
# The “case $NODENAME” construct below allows each node in a diskless
# cluster to execute node specific route calls if necessary. Add entries
# to the case construct for specific nodes in the diskless cluster if
needed.
# The STATUS checking is for Instant Ignition.
#
# For example,
#
#     case $NODENAME in
#           $ROOTSERVER) /etc/route add 192.0.2 gatenode 1
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           ;;
#           *) /etc/route add default 15.2.104.69 1
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           ;;
#     esac
#
# adds network destination “192.0.2” to the rootserver’s routing tables,
# indicating a correspondence between that destination and the gateway
# “gatenode”, and specifying the number of hops to the gateway as 1. For
# all other nodes (* is the wildcard), the default gateway is set to
# 15.2.104.69.
#
# The route command should be invoked once per gateway.
#
# SEE ALSO: route(1m), routing(7)

case $NODENAME in
      $ROOTSERVER)
      /etc/route add net 192.1.1.128 192.1.1.65 1
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi

      /etc/route add default 192.168.1.2 1
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      ;;
      *) /etc/route add default `hostname` 0
      ;;
esac

#
# Initialize the network node name.
#
# (STEP 3)
#
# The nodename(1m) command assigns an NS node name to the node.
# Nodename takes an option of the form “nodename.domainname.orgname”
where,
#
# nodename        is the name of the local node
# domainname      is the name of the domain
# orgname         is the name of the organization
#
# Each name must start with an alphabetic character.
#
# It is strongly recommended that the string used for “nodename” above be
# identical to the string used as an argument to the hostname(1) command,
# which is typically invoked from the system initialization shell script
# file “/etc/rc”. The NS nodename used on each node in your network needs
# to be unique within that network. The “case $NODENAME” construct below
# allows each node in a diskless cluster to execute a node specific
# nodename(1) call if necessary. Add entries to the case construct for
# specific nodes in the diskless cluster only if needed.
#
# For example,
#
#     case $NODENAME in
#           * ) /bin/nodename `/bin/uname -n`.mydomain.myorg
#     ;;
#     esac
#
# sets the NS nodename for all nodes (* is the wildcard) in domain
# “mydomain” and organization “myorg”.
#
# The nodename command line below sets the nodename field to the system
# hostname, the domainname field to the rootserver’s name, and the orgname
# field to “diskless”.
#
# SEE ALSO: nodename(1)

if [ -x /bin/nodename ]
then
      case $NODENAME in
            *) /bin/nodename `/bin/uname -n`.$DOMAIN.$ORGANIZATION
            STATUS=$?
            if [ ! $STATUS -eq 0 ]
            then
                  net_init=1
            fi
            ;;
      esac
fi

#
# Start remote loop back daemon
#
if [ -f /usr/adm/rld.log ]
then
      /bin/mv /usr/adm/rld.log /usr/adm/OLDrld.log
fi
if [ -x /etc/rlbdaemon ]
then
      (/etc/rlbdaemon 2>&1 ) > /usr/adm/rld.log
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

/bin/echo “Network Link started”

#
# Start NFS. This requires installation of the NFS product.
#
if [ -x /etc/netnfsrc ]
then
      /etc/netnfsrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

/bin/echo “ARPA/Berkeley daemons started: \c”

#
# Start the Internet daemon.
#

[ -x /etc/inetd ] && /etc/inetd && /bin/echo “inetd \c”
STATUS=$?
if [ ! $STATUS -eq 0 ]
then
      net_init=1
fi

#
# Start ARPA/BSD networking services.
#
if [ -x /etc/netbsdsrc ]
then
      /etc/netbsdsrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

/bin/echo

#
# Do nfs mounts after inetd is running
#
if [ -x /etc/netnfsrc2 -a -f /etc/nfs.up ]
then
      /etc/netnfsrc2
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

#
# Start NS networking services.
#
if [ -x /etc/netnssrc ]
then
      /etc/netnssrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

#
# Start HP Network Management Agent
#
if [ -x /etc/netnmrc ]
then
      /etc/netnmrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

#
# Start HP LAN Manager/X.
#
if [ -x /etc/netlmrc ]
then
      /etc/netlmrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

#
# Start NCS. This requires installation of the NCS product.
# NCS must be started before any other NCS products are started.
#
if [ -x /etc/netncsrc ]
then
      /etc/netncsrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

#
# Start NetLS. This requires installation of the NetLS product.
# NCS must be started before NetLS is started.
#
if [ -x /etc/netlsrc ]
then
      /etc/netlsrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

# return exit code for Instant Ignition
if [ $net_init -eq 0 ]
then
      exit 0
else
      exit 1
fi
```

3.1.3 Querying ADAS Workstation Ethernet Configuration Parameters

Planning a network and configuring the ethernet interface devices requires specialized knowledge of both network topologies and routing mechanics. Modifying an installed ethernet network with minimal changes to the existing IP addressing is especially challenging. Every effort has been expended to insure the recommendations documented herein result in a properly configured network with minimal effort and detailed ethernet knowledge from the customer. However, some installations may require custom configuration parameters different from the standard values presented in earlier sections. This section provides recommendations as to how a customer may determine some of the necessary parameters required to install and configure the Remote Access feature.

3.1.3.1Determine Hostname of the ADAS Workstation

The hostname of the workstation can be determined by entering the following command from the Unix prompt from an xterm window:

hostname *<Return>*

which will return the following, if the workstation installation conforms with **IM 20-2020**:

oam_ws

3.1.3.2Determine IP Address of the Ethernet Interface

The IP address of the first ethernet interface can be determined by entering the following command at the Unix prompt from an xterm window:

ifconfig lan0 *<Return>*

which will provide a response similar to the following:

lan0: flags=63<UP,BROADCAST,NOTRAILERS,RUNNING>

```
inet 192.1.1.66 netmask ffffffc0 broadcast 192.1.1.127
```

The above response indicates that the IP address for the **LAN0** interface in this workstation is ***192.1.1.66***. Also, note that the subnet mask is provided in the response, which in this instance happens to be ***ffffffc0*** (or ***255.255.255.192*** in decimal-dot notation).

Alternatively, the **/etc/netlinkrc** file can be examined for an entry similar to the following:

```
/etc/ifconfig lan0 inet 192.1.1.66 netmask 255.255.255.192 up
```

which indicates the IP address for **LAN0** is ***192.1.1.66***. Again, also note that the subnet mask is provided in the command (***255.255.255.192***).

3.1.3.3Determine IP Address of Gateway for LAN0 (the EIU Ethernet IP Address)

If a software upgrade of an existing ADAS installation is being performed, the IP address of the gateway for the first ethernet interface (LAN0) can be determined by entering the following command at the Unix prompt from an xterm window:

```
arp -a <Return>
```

which will provide a response similar to the following:

```
? (192.1.1.65) at 0:0:75:f0:36:10 ether
```

Since there is only one entry in the address translation table from the above response, we can assume that all ethernet accesses are directed to a single external gateway for routing to the DMS nodes. The above response indicates that the ethernet IP address for the EIU connected to the **LAN0** interface in this workstation is ***192.1.1.65***.

However, if this is a new installation, the arp information may not be available. In this case, the DMS switch datafill must be queried to determine the correct IP address. From a MAP terminal, table **iprouter** can be examined for the ethernet address of the EIU, as shown below:

```
>table iprouter
MACHINES NOT IN SYNC - DMOS NOT ALLOWED
JOURNAL FILE UNAVAILABLE - DMOS NOT ALLOWED
TABLE: IPROUTER
>list all
TOP
RKEYROUTERSNIPADRETHIPADRETHARPETHPARP

0EIU 0192 168 8 3192 168 19 68YESYES
```

```
1EIU 1192 168 8 5192 1 1 65YESYES
2EIU 2192 168 8 17192 168 162 84YESYES
3EIU 3192 168 8 18192 168 162 104YESYES
BOTTOM
>
```

The user must know which EIU the ADAS Workstation is connected to. As an example, if the ADAS Workstation is connected to EIU 1, it is readily observed from the **ETHIPADR** field in table **iprouter** that the EIU’s ethernet IP address is ***192.1.1.65***.

3.1.3.4Determine IP Address of CM Network

The address of the CM network is determined via a two-step process. For an upgrade of an existing ADAS installation, the IP address of the CM is first determined by examination of the **/etc/hosts** table. This is accomplished by entering the following command at the Unix prompt from an xterm window:

cat /etc/hosts *<Return>*

which will provide a response similar to the following:

```
## Configured using SAM by root on Thu Oct 26 16:33:07 1995
# @(#)$Header: hosts,v 1.8.109 91/11/21 12:01:46 kcs Exp $
#
# The form for each entry is:
# <internet address> <official hostname> <aliases>
#
# For example:
# 192.1.2.34      hpfcrm            loghost
#
# See the hosts(4) manual page for more information.
# Note:           The entries cannot be preceded by a space.
#                 The format described in this file is the correct
format.
#                 The original Berkeley manual page contains an error
in
#                 the format description.
#

127.0.0.1       localhost         loopback
192.1.1.129     CM0000
192.1.1.132     APU000X
192.1.1.134     APU001X
192.1.1.136     APU002X
192.1.1.138     APU003X
192.1.1.139     APU004X
```

```
192.1.1.140          APU005X
192.1.1.142          APU006X
```

Examination of the **/etc/hosts** table above indicates that the IP address for the CM is ***192.1.1.129*** (from the CM000 entry).

If a new install is being performed, the **/etc/hosts** table may not be populated on the ADAS Workstation. In this case, the DMS switch datafill must be queried for the IP address of the CM. From a MAP terminal, examine table **ipnetwrk** as shown below:

```
>table ipnetwrk
MACHINES NOT IN SYNC - DMOS NOT ALLOWED
JOURNAL FILE UNAVAILABLE - DMOS NOT ALLOWED
TABLE: IPNETWRK
>list all
TOPOPTION
KEYREFCMIPADDRSUBNETPARMAREA

0 192 1 1 12918( EIU 0)$
( SCRNFLAG N)$
BOTTOM
>
```

From the **CMIPADDR** field in table **ipnetwrk** above, it can be seen that the IP address for the CM is ***192.1.1.129***.

Next, the subnet mask for **LAN0** (found in a previous step) is bit-wise **ANDed** with the CM IP address, as shown below:

	IP Address			
CM IP Address	192.	1.	1.	129
	1 1 0 0 0 0 0 0.	0 0 0 0 0 0 0 1.	0 0 0 0 0 0 0 1.	1 0 0 0 0 0 0 1
Subnet Mask	255.	255.	255.	192
	1 1 1 1 1 1 1 1.	1 1 1 1 1 1 1 1.	1 1 1 1 1 1 1 1.	1 1 0 0 0 0 0 0
CM Network Address	192.	1.	1.	128
	1 1 0 0 0 0 0 0.	0 0 0 0 0 0 0 1.	0 0 0 0 0 0 0 1.	1 0 0 0 0 0 0 0

As the above figure illustrates, the bit-wise **AND** of the CM IP address of ***192.1.1.129*** with the subnet mask of ***255.255.255.192*** gives the network address of ***192.1.1.128***, which is the **CM network address** for this example.

3.2 Remote Access Position

The Remote Access workstation consists of an ADAS OAM workstation that is loaded with the Remote Access software. While a workstation that is performing as an ADAS OAM position may also serve as a Remote Access position, this section discusses the application of a workstation dedicated solely to remote access purposes.

The Remote Access workstation is a new ADAS component introduced in the TOP07 software stream. As such, there are no requirements for field upgrades or any field compatibility issues. Other than the 9.03 HP-UX operating system, the only additional software required on the Remote Access workstation is the remote access software load.

Like the ADAS OAM position, the Remote Access workstation is provisioned with two ethernet interfaces. However, only the first ethernet interface (LAN0) is enabled on the Remote Access workstation. All necessary configuration of the workstation for remote access is accomplished during the installation of the Remote Access software.

The ethernet interface configuration for remote access, contained in the **/etc/netlinkrc** file, are summarized below:

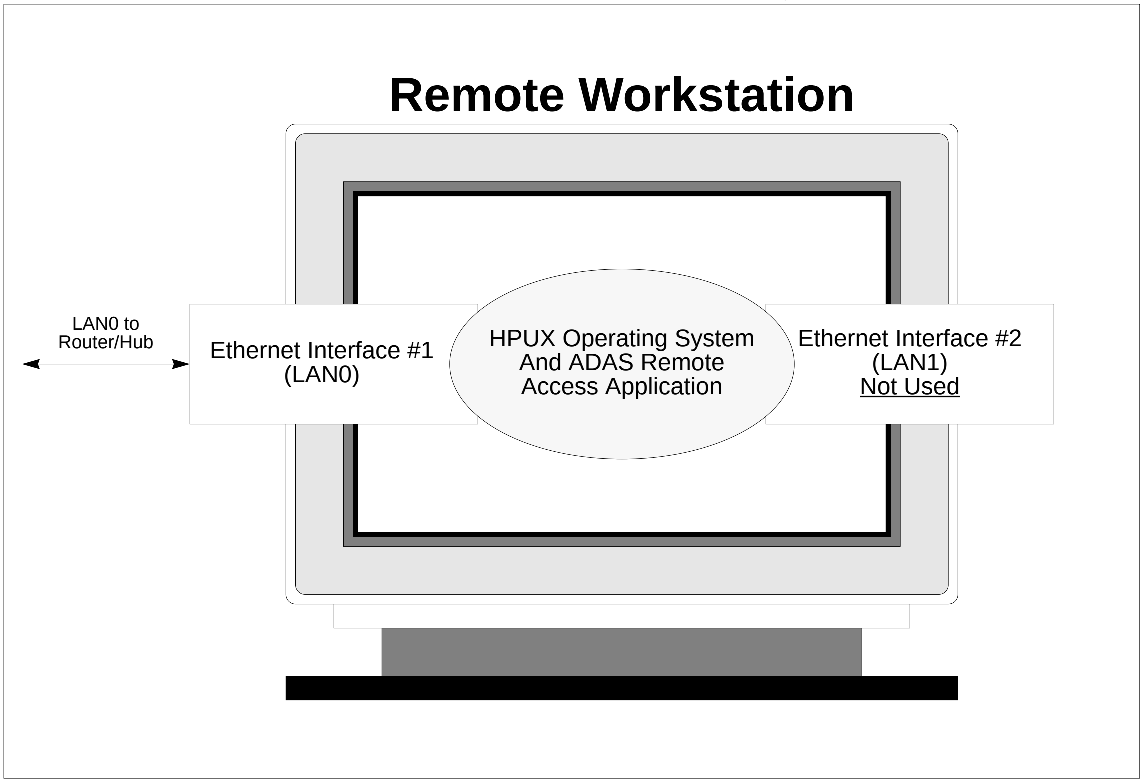
1. A unique IP address is defined for LAN0.
2. The LAN0 ethernet interface is subnetted with the appropriate class C subnet mask.
3. A default route/gateway is defined for LAN0.

For an example of the **/etc/netlinkrc** file for the Remote Access workstation, please see ***Section 3.2.2 on page 52***.

The Remote Access workstation comes provisioned with all of the necessary software drivers installed and bound into the Unix kernel. Activation of the Remote Access feature does not require the Unix kernel on the workstation to be rebuilt.

A pictorial representation of the software and hardware interaction of the Remote Access feature on the Remote Access workstation is provided in ***Figure 10 on page 51***.

Figure 10 Remote Access Workstation Ethernet Connectivity


Remote Workstation
LAN0 to Router/Hub
Ethernet Interface #1 (LAN0)
HPUX Operating System And ADAS Remote Access Application
Ethernet Interface #2 (LAN1) Not Used

3.2.1 Remote Access Workstation Configuration Guidelines

The customer is required to perform very few actions with regards to actually making changes to the Remote Access workstation to support the Remote Access feature, as nearly all of the configuration changes are performed automatically by the installation scripts. However, customers wishing to perform a customized installation of the Remote Access feature to incorporate it into their corporate network should adhere to the following guidelines.

Note: Northern Telecom strongly advises against, and will not support, customized installations of the Remote Access feature.

The guidelines below should be followed when configuring the Remote Access workstation. It may be helpful to reference ***Figure 8 on page 29*** to understand the relationship of the IP address assignments to the various nodes in the Remote Access network. To understand the syntax of the commands used to perform specific configuration actions, refer to the sample **netlinkrc** file in ***Section 3.2.2 on page 52***.

The first step is to configure the **LAN0** ethernet interface using the appropriate ***subnet mask*** and ***route*** changes.

Note: The changes listed below are made in the **/etc/netlinkrc** file. All of the changes could be made via appropriate **ifconfig** and **route** commands at the Unix prompt, but changes made in this fashion are not permanent and will not survive a reboot of the workstation.

1. Select a unique IP address for the **LAN0** ethernet interface in the Remote Access workstation. The IP address should be selected from the private, class C network ***192.168.x.x***.

   Note: Northern Telecom has adopted the following address assignment convention for Remote Access. The ***first*** workstation (ADAS OAM Position **or** Remote Access workstation) added to the Remote Access network is assigned IP address ***192.168.1.1***. The ***second*** workstation added to the network is assigned IP address 192.168.***2***.1, the ***third*** workstation is assigned IP address 192.168.***3***.1., the ***fourth*** workstation is assigned IP address 192.168.***4***.1,...

2. The subnet mask for the **LAN0** ethernet interface in the Remote Access workstation is defined to be ***255.255.255.192*** for all workstations added to the network. Therefore, the **ifconfig** entry for **LAN0** for the ***first*** workstation added to the Remote Access network would appear as

   /etc/ifconfig lan0 inet 192.168.1.1 netmask 255.255.255.192 up

3. The associated **lanconfig** command must be added for **LAN0** as follows

   /etc/lanconfig lan0 ether

4. A single route is added to the routing tables for **LAN0**, setting the ***default*** route to be a **gateway** with the IP address of ***192.168.x.2***, where x = 1,2,3,4,... depending on the workstation number. For example, the ***first*** workstation would have the default route set to 192.168.***1***.2, the ***second*** workstation would use IP address 192.168.***2***.2, the ***third*** workstation would use 192.168.***3***.2, the ***fourth*** workstation would use 192.168.***4***.2,... Therefore, the **route add** entry associated with **LAN0** of the ***first*** workstation added to the Remote Access network would appear as

   /etc/route add default 192.168.1.2 1

5. Finally, enable the loopback for the **LAN0** interface with the following **ifconfig** entry

   /etc/ifconfig lo0 inet 127.0.0.1 up

3.2.2 Sample netlinkrc File for the Remote Access Workstation

The **netlinkrc** file resides in the **/etc** directory of the Remote Access workstation and provides, among other things, the configuration details of the LAN interfaces installed in the workstation. The **netlinkrc** file is read and executed each time the Remote Access workstation is rebooted or powered up. Changes to the LAN interfaces that are intended to be permanent are placed in the **netlinkrc** file. Temporary changes to the ethernet interfaces can be made at any time through the execution of the appropriate Unix commands, or permanent changes can be made via sam (System Administration Manager).

Note: The installation of the Remote Access software will automatically configure the appropriate files as part of the installation procedure.

The file below illustrates the **netlinkrc** file for the Remote Access workstation.

Sample netlinkrc file for the /*first*/ workstation added to the Remote Access network

Note: Modifications required for the Remote Access feature are indicated by ***bold italic*** type.

Note: This sample netlinkrc file represents the configuration for the **first** workstation added to the Remote Access network. Subsequent workstations added to the network will have different entries for the **ifconfig** and **route** commands to reflect the appropriate IP addresses and networks.

```
#! /bin/sh
## Configured using SAM by root on Thu Oct 26 16:33:06 1995

# @(#)netlinkrc: $Revision: 1.6.109.7 $ $Date: 92/07/13 08:21:12 $
# $Locker: $

#
# Shell script for initialization of link networking product.
#
# net_init flag is used for Instant Ignition. If net_init is set,
# then netlinkrc return “exit 1”. In order for Instant Ignition
# to work correctly, netlinkrc needs to check the STATUS variable
# after each program or scripts it calls.
#
net_init=0

if [ -f /etc/clusterconf ]
then
      ROOTSERVER=`/bin/cnodes -r`
      NODENAME=`/bin/cnodes -m`
      DOMAIN=`/bin/cnodes -r`
      ORGANIZATION=diskless
else
      ROOTSERVER=`hostname`
      NODENAME=$ROOTSERVER
      DOMAIN=`/bin/uname -n`
      ORGANIZATION=standalone
fi

#
# Start logging daemon *before* any other networking initialization.
# See nettl(1m) for more information.
#
/etc/nettl -start
STATUS=$?
if [ ! $STATUS -eq 0 ]
then
      net_init=1
fi
#
```

```
# Remove the existing /etc/netstat_data file.  The first time
# netstat is executed, a new /etc/netstat_data file will be
# created.
#
/bin/rm -f /etc/netstat_data

#
# Initialize networking interfaces.
#
# (STEP 1)
#
# The ifconfig(1m) command assigns an IP address to a LAN interface and
# configures network interface parameters.  The lanconfig(1m) command
defines
# the packet encapsulation method for the LAN interface.
#
# The “case $NODENAME” construct below allows each node in a diskless
cluster
# to execute node specific calls if necessary.  Add entries to
# the case construct for specific nodes in the diskless cluster only if
# needed. For example, if a specific node has more than one LAN interface,
# the node must execute separate commands for each of the interfaces.
#
# For example:
#
#     case $NODENAME in
#     $ROOTSERVER)/etc/ifconfig lan0 inet 192.6.1.3 up
#                 /etc/lanconfig lan0 ether
#                 /etc/ifconfig lan1 inet 15.4.64.1 netmask
255.255.248.0 up
#                 /etc/lanconfig lan1 ether
#                 ;;
#           *)    /etc/ifconfig lan0 inet `hostname` up
#                 /etc/lanconfig lan0 ether ieee
#                 ;;
#     esac
#                 /etc/ifconfig lo0 inet 127.0.0.1 up
#
# assigns to the two interfaces lan0 and lan1 on a rootserver the DARPA
# Internet addresses 192.6.1.3 and 15.4.64.1 respectively; the lan0
# interfaces on all other nodes (* is the wildcard) are assigned their
# respective internet addresses as found in /etc/hosts.
#
# The ifconfig command line below is sufficient to initialize the network
# interface for any node that has one LAN interface card and whose
# hostname and Internet address are present in the hosts(4) file.
#
# NOTE:If the ifconfig command line does not specify a subnet mask,
#     the subnet mask defaults to the network mask.
#     It is not necessary for both encapsulation methods to be turned on
#     for the LAN Interface.  For further explanation see lanconfig(1m)
#
# The loopback interface must be explicitly configured for each address
# family of interest.  The following command assumes that the hostname
# has already been set and is mapped to an IP Address in /etc/hosts.
#
# SEE ALSO: ifconfig(1m), lanconfig(1m)

case $NODENAME in
      $ROOTSERVER)
      /etc/ifconfig lan0 inet 192.168.1.1 netmask 255.255.255.192 up
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      /etc/lanconfig lan0 ether
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      ;;
      *) /etc/ifconfig lan0 inet `hostname` netmask 255.255.240.0 up
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      /etc/lanconfig lan0 ether
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      ;;
esac
      /etc/ifconfig lo0 inet 127.0.0.1 up
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi

# The x25init(1m) command configures X.25 network interface parameters.
The
# “case $NODENAME” construct below allows each node in a diskless cluster
# to execute node specific x25init calls if necessary.  Add entries to
# the case construct for specific nodes in the diskless cluster only if
# the nodes have X.25 interfaces. The nodes must execute separate x25init
# commands for each of the interfaces. The STATUS checking is for Instant
# Ignition.
#
# For example:
#
#     case $NODENAME in
#           NODEA ) /etc/x25init -c /etc/x25/config_filename1
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           /etc/x25init -c /etc/x25/config_filename2
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           /etc/x25init -a /etc/x25/ip_x25_mapfile
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           ;;
#           NODEB ) /etc/x25init -c /etc/x25/config_file_nodea
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           ;;
#     esac
#
# initializes two X.25 interfaces on NODEA and one interface on NODEB.
# For nodes which have IP configured over X.25, the x25init -a command
# provides the mapping of IP Addresses to X.121 addresses. It is
recommended
# to put the configuration and ipmap files in the /etc/x25 directory.
#
# In the above example, at least one of NODEA’s X.25 Cards supports IP
# since IP-to-X.25 Map table is initialized on NODEA.
#
# SEE ALSO: x25init(1m)

#
# Initialize network routing.
#
# (STEP 2) (OPTIONAL, FOR NETWORKS WITH GATEWAYS ONLY)
#
# The route(1m) command manipulates the network routing tables.
# The “case $NODENAME” construct below allows each node in a diskless
# cluster to execute node specific route calls if necessary.  Add entries
# to the case construct for specific nodes in the diskless cluster if
needed.
# The STATUS checking is for Instant Ignition.
#
# For example,
#
#     case $NODENAME in
#           $ROOTSERVER) /etc/route add 192.0.2 gatenode 1
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           ;;
#           *) /etc/route add default 15.2.104.69 1
#           STATUS=$?
#           if [ ! $STATUS -eq 0 ]
#           then
#                 net_init=1
#           fi
#           ;;
#     esac
#
# adds network destination “192.0.2” to the rootserver’s routing tables,
# indicating a correspondence between that destination and the gateway
# “gatenode”, and specifying the number of hops to the gateway as 1. For
# all other nodes (* is the wildcard), the default gateway is set to
# 15.2.104.69.
#
# The route command should be invoked once per gateway.
#
# SEE ALSO: route(1m), routing(7)

case $NODENAME in
      $ROOTSERVER)
      /etc/route add default 192.168.1.2 1
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
      ;;
      *) /etc/route add default `hostname` 0
      ;;
esac

#
# Initialize the network node name.
#
# (STEP 3)
#
# The nodename(1m) command assigns an NS node name to the node.
# Nodename takes an option of the form “nodename.domainname.orgname”
where,
#
# nodename          is the name of the local node
# domainname        is the name of the domain
# orgname           is the name of the organization
#
# Each name must start with an alphabetic character.
#
# It is strongly recommended that the string used for “nodename” above be
# identical to the string used as an argument to the hostname(1) command,
# which is typically invoked from the system initialization shell script
# file “/etc/rc”.  The NS nodename used on each node in your network needs
# to be unique within that network. The “case $NODENAME” construct below
# allows each node in a diskless cluster to execute a node specific
# nodename(1) call if necessary.  Add entries to the case construct for
# specific nodes in the diskless cluster only if needed.
#
# For example,
#
#     case $NODENAME in
#           * ) /bin/nodename `/bin/uname -n`.mydomain.myorg
#     ;;
#     esac
#
# sets the NS nodename for all nodes (* is the wildcard) in domain
# “mydomain” and organization “myorg”.
#
# The nodename command line below sets the nodename field to the system
# hostname, the domainname field to the rootserver’s name, and the orgname
# field to “diskless”.
#
# SEE ALSO: nodename(1)

if [ -x /bin/nodename ]
then
      case $NODENAME in
            *) /bin/nodename `/bin/uname -n`.$DOMAIN.$ORGANIZATION
            STATUS=$?
            if [ ! $STATUS -eq 0 ]
            then
                  net_init=1
            fi
            ;;
      esac
fi

#
# Start remote loop back daemon
#
if [ -f /usr/adm/rld.log ]
then
      /bin/mv /usr/adm/rld.log /usr/adm/OLDrld.log
fi
if [ -x /etc/rlbdaemon ]
then
      (/etc/rlbdaemon 2>&1 ) > /usr/adm/rld.log
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

/bin/echo “Network Link started”
```

```
#
# Start NFS. This requires installation of the NFS product.
#
if [ -x /etc/netnfsrc ]
then
      /etc/netnfsrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

/bin/echo “ARPA/Berkeley daemons started: \c”

#
# Start the Internet daemon.
#

[ -x /etc/inetd ] && /etc/inetd && /bin/echo “inetd \c”
STATUS=$?
if [ ! $STATUS -eq 0 ]
then
      net_init=1
fi

#
# Start ARPA/BSD networking services.
#
if [ -x /etc/netbsdsrc ]
then
      /etc/netbsdsrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

/bin/echo

#
# Do nfs mounts after inetd is running
#
if [ -x /etc/netnfsrc2 -a -f /etc/nfs.up ]
then
      /etc/netnfsrc2
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi
```

```
#
# Start NS networking services.
#
if [ -x /etc/netnssrc ]
then
      /etc/netnssrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

#
# Start HP Network Management Agent
#
if [ -x /etc/netnmrc ]
then
      /etc/netnmrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

#
# Start HP LAN Manager/X.
#
if [ -x /etc/netlmrc ]
then
      /etc/netlmrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

#
# Start NCS. This requires installation of the NCS product.
# NCS must be started before any other NCS products are started.
#
if [ -x /etc/netncsrc ]
then
      /etc/netncsrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi
```

```
#
# Start NetLS. This requires installation of the NetLS product.
# NCS must be started before NetLS is started.
#
if [ -x /etc/netlsrc ]
then
      /etc/netlsrc
      STATUS=$?
      if [ ! $STATUS -eq 0 ]
      then
            net_init=1
      fi
fi

# return exit code for Instant Ignition
if [ $net_init -eq 0 ]
then
      exit 0
else
      exit 1
fi
```

4.0 LAN/WAN Equipment Configuration

Before presenting the equipment configuration details, it may be helpful to present some general, background information regarding ethernet networks. This information will assist the user in understanding some of the concepts presented in subsequent sections.

Ethernet was selected for the data communications infrastructure of the Remote Access feature due to its well documented, universal support, both from networking equipment suppliers and third-party software vendors. Moreover, the base Unix operating system on the ADAS Workstation provides inherent ethernet support that is easily extended from the existing ADAS LAN connecting the OAM Position to the DMS switch, to include remote WAN access from distant locations. Industry standard routing protocols are employed throughout the network, simplifying both the installation and maintenance of the Remote Access feature.

Additionally, the ADAS Workstation is already provisioned with a second ethernet interface that can be exploited by the Remote Access feature to provide a secure interface to the OAM Position. This is particularly important, as the integrity and security of the link to the DMS switch are fundamental requirements of any remote access solution.

4.1 Ethernet Communication Overview

As the ADAS Workstation currently employs a very simple ethernet LAN to communicate with the DMS switch, the customer should already by familiar with many of the essential ethernet network concepts. The same technology employed by the existing ADAS LAN is used by the Remote Access feature to provide access to the ADAS Workstation over a WAN network composed of T1/Fractional T1 facilities.

Both the existing ADAS LAN, and the new Remote Access LAN, are based on ethernet 10Base-T (10 Mbps, twisted-pair, copper) technology due to its robustness, flexibility, and cost effectiveness. Ethernet provides a simple, yet flexible and very reliable communications infrastructure. Additionally, ethernet communications concepts are very straightforward and can be easily understood by referencing the Open Systems Interconnection (OSI) model of data communications.

The OSI model was developed in 1984 by the International Standards Organization (ISO) and is used as a frame of reference when describing protocol architectures and functional characteristics.

4.1.1 Open Systems Interconnection Model

The OSI model of data communications consists of a seven-layer model as shown below:

Application	Layer 7
Presentation	Layer 6
Session	Layer 5
Transport	Layer 4
Network	Layer 3
Data Link	Layer 2
Physical	Layer 1

The OSI model of layering is based on a *service provider/service user* concept. In the OSI model, a layer is a *service provider* to the layer above it, and a *service user* of the layer below it. For example, the Data Link layer is a *service provider* to the Network layer, and a *service user* of the Physical Layer. A guiding concept of layering is that a *service provider* must provide its services while hiding the details of how it is doing it from the *service user.* A brief description of each of the seven layers of the OSI model are provided below.

4.1.1.1OSI Physical Layer

The *Physical Layer* provides the physical transmission service. It accepts data from the *Data Link* layer for transmission over the physical medium. The mechanical (e.g., connector type), electrical (e.g., voltage levels), functional (e.g., pin assignments), and procedural (e.g., handshake) characteristics are defined at this layer.

4.1.1.2OSI Data Link Layer

The *Data Link Layer* is responsible for the reliable transfer of data across the physical link. This layer is responsible for functions such as flow control, data frame formatting, error detection, and link management.

4.1.1.3OSI Network Layer

The *Network Layer* provides routing services across the internetwork. This layer also shields the higher layers from the details about the underlying network (e.i., the network topology and roadmap) and routing technology that may have been employed to connect different networks together.

4.1.1.4OSI Transport Layer

The *Transport Layer* guarantees reliable and orderly end-to-end delivery of data between systems. OSI defines five different protocols at this layer, each with various levels of reliability. The *Transport Layer* also performs functions such as data multiplexing and demultiplexing.

4.1.1.5OSI Session Layer

The *Session Layer* is responsible for establishing, maintaining, and arbitrating the dialogues between communicating applications. Additionally, this layer is also responsible for orderly failure recovery.

The *Sessions Layer*, as well as the *Presentation* and *Application Layers*, are strictly application-oriented layers. These layers are only concerned with services pertinent to applications--no attention is paid to the details of the data exchange and routing service mechanisms provided by the lower layers.

4.1.1.6OSI Presentation Layer

The *Presentation Layer* is responsible for resolving the differences between the data syntax used by communicating applications.

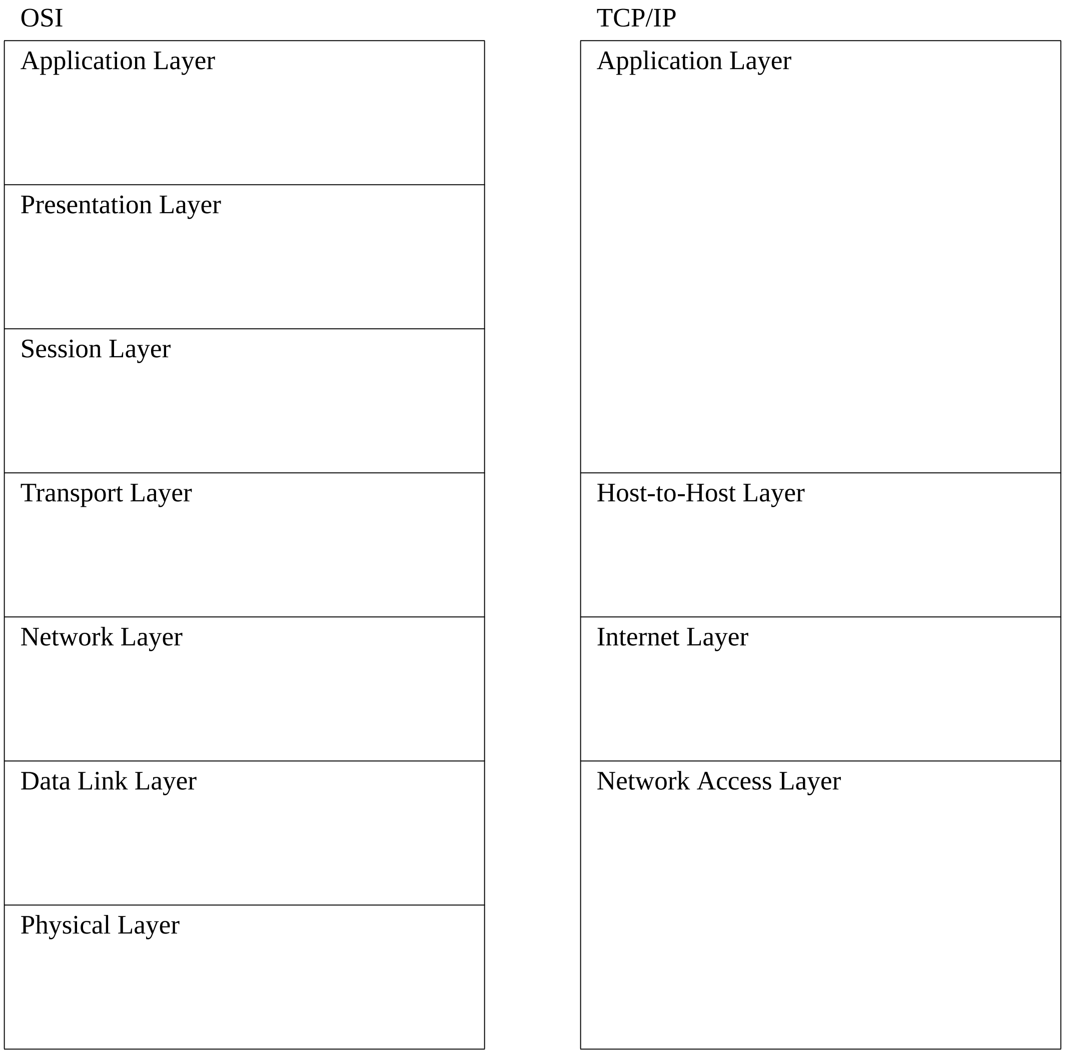
4.1.1.7OSI Application Layer

The *Application Layer* provides the engines and interfaces employed by end-user applications, and should not be confused with the actual applications.

4.1.2 TCP/IP Communications Architecture

The ethernet communications protocol used in the ADAS Remote Access Network is TCP/IP (Transmission Control Protocol/Internet Protocol). TCP/IP is a fully reliable, connection-oriented, data-streaming service, which insures the integrity of the ethernet LAN/WAN connections established between a Remote Access Workstation and a host ADAS Workstation. The comparison of the TCP/IP communications architecture to the OSI model is illustrated in the figure on the following page.

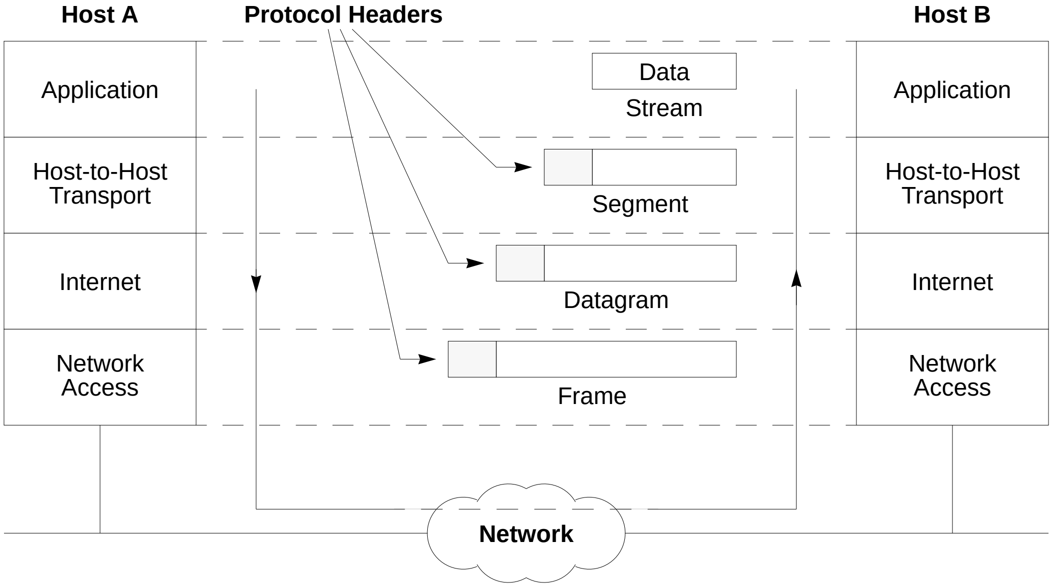
Figure 11 TCP/IP Architecture



Following the OSI model, two hosts using TCP/IP to communicate with each other across a network perform data encapsulation at each layer of the protocol as the data as it is passed down across protocol boundaries. In this fashion, each layer of the TCP/IP protocol stack engages in peer communications with its counterpart over the network by adding header information to the data before submitting it to the layer below. This header information is intended only for its peer in each host.

The data encapsulation that occurs as two hosts communicate via TCP/IP over a network is illustrated in ***Figure 12 on page 67***.

Figure 12 Data Encapsulation Under TCP/IP



At each layer of the protocol, the headers contain various information, such as:

- At the *Transport Layer*, the header contents include destination and source port numbers. These are treated as process identification numbers, which help in the exchange of encapsulated data between designated processes without confusing these processes with others that may be running simultaneously on the same involved hosts. The data and header at this layer form a unit referred to as a *data segment*.
- At the *Internet Layer*, the header also contains the IP address of the communicating (ultimate) end systems. The data and header at this layer form a unit known as an *IP datagram*.
- At the *Network Access Layer*, the header includes the media access control (MAC) addresses of the source and destination devices communicating on the same physical network. The data and header at this layer from a unit known as a *data frame*.

On the transmitting host, each layer of the protocol stack processes data from the next higher layer and adds the appropriate header information for the peer process on the receiving host, before passing the data packet down to the next layer.

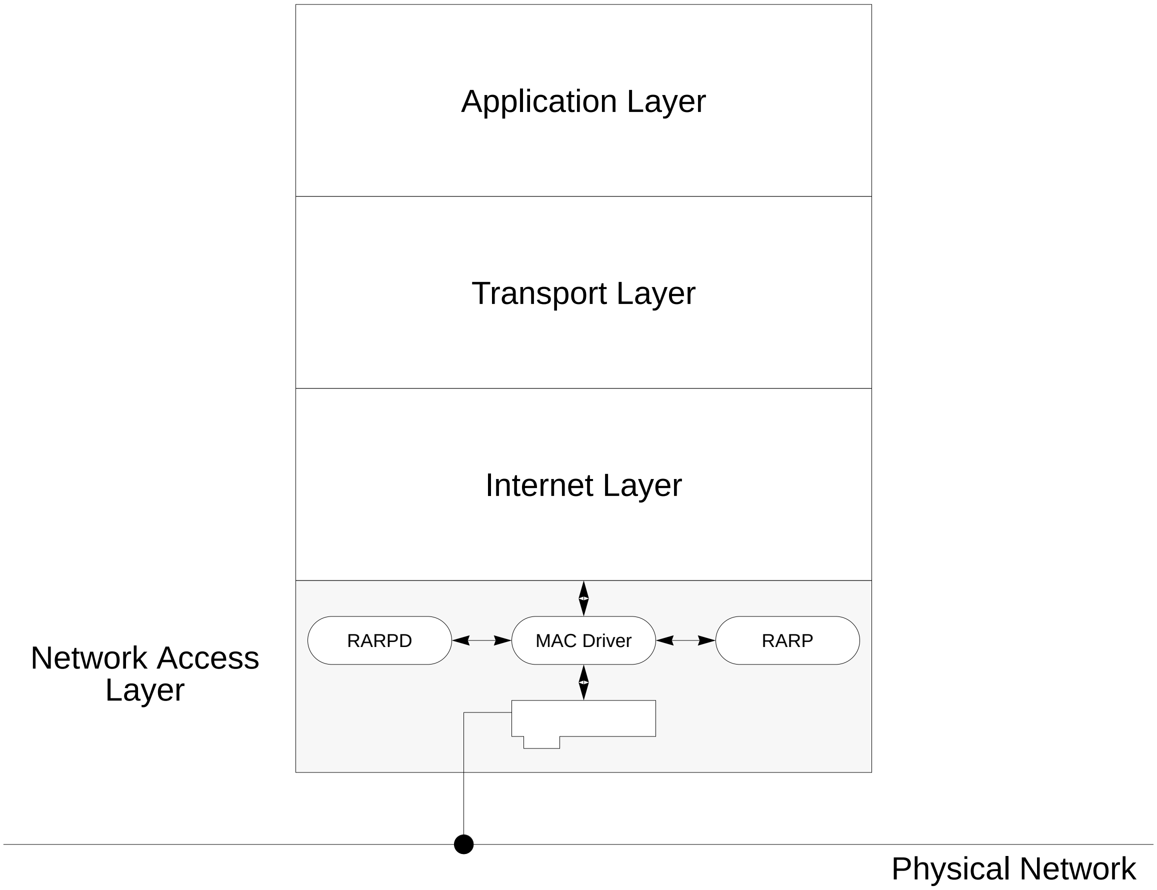
On the receiving host, at each layer in the protocol stack, the peer process interprets the header information for that layer and strips off the header before the data is presented to the next higher layer in the stack.

4.1.2.1TCP/IP Network Access Layer

As illustrated in ***Figure 11 on page 66*** the TCP/IP *Network Access Layer* provides the same functions as the *Physical* and *Data Link Layers* in the OSI model. Protocols implemented at this layer are responsible for the delivery of data to devices connected to the *same* physical network. The *Network Access Layer* is the only layer that is aware of the details of the underlying physical/electrical network.

The *Network Access Layer* implementation includes the network interface card, i.e., the hardware components, that is compatible with the communications media and protocols being employed in the network. ***Figure 13 on page 68*** illustrates the *Network Access Layer* and some of the protocols implemented at this layer. One of the primary protocols implemented at this layer include the Address Resolution Protocol (ARP), which maps the (symbolic) IP address to the corresponding (real) hardware (MAC, or Media Access Control) address. Notice that not all of the data received by the network interface card from the network is passed to higher layers. Some data may be passed to adjacent protocols existing at the same layer, such as RARP (Reverse Address Resolution Protocol) and RARPD (Reverse Address Resolution Protocol Daemon).

Figure 13 TCP/IP Network Access Layer Components



Among other functions, the *Network Access Layer* encapsulates data from the *Internet Layer* into frames for subsequent delivery to the network. The frame format is a function of the media access technology employed in the network. While TCP/IP supports many different MAC technologies at the D*ata Link Layer*, the technology pertinent to Remote Access is ethernet.

Ethernet Technology

Ethernet data link technology was originally developed by Xerox in the early 1970s, and was later jointly standardized by Xerox Corporation, Intel Corporation, and Digital Corporation in 1978. A few years later, the IEEE established the 802 committee, which formally standardized the 802.3 data link technology as an adaptation (with slight differences) of the ethernet technology.

Ethernet Physical Layer

Ethernet supports various physical and electrical interfaces, as shown in the table below:

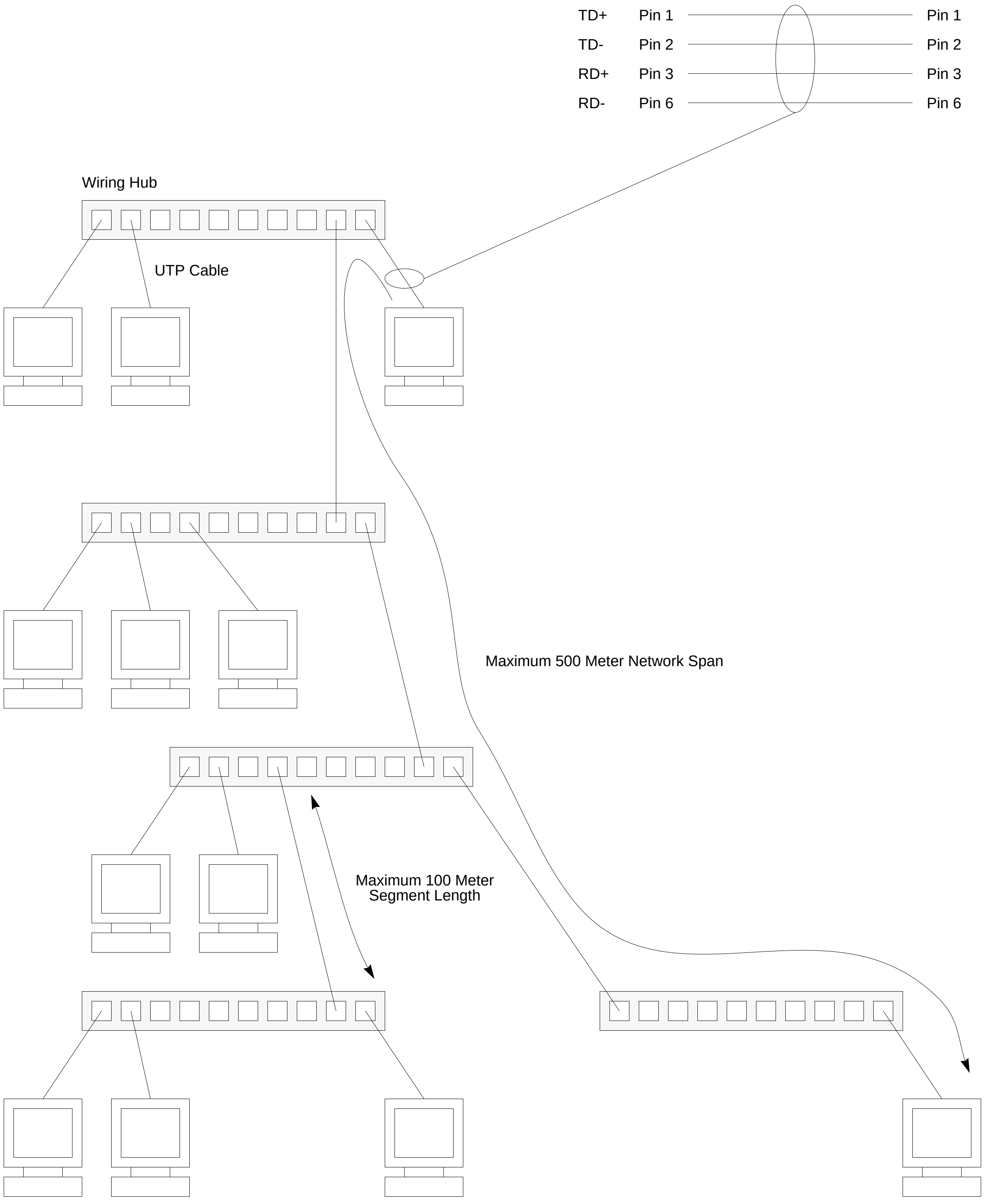
	10Base-2	**10Base-5**	**10Base-T**	**10BROAD-36**
Physical Medium	Coaxial Cable (50 ohm)	Coaxial Cable (50 ohm)	Unshielded Twisted Pair	Coaxial Cable (75 ohm)
Data Rate (Mbps)	10	10	10	10
Signaling	Baseband	Baseband	Baseband	Broadband
Maximum Segment Length	185 meter	500 meter	100 meter	1800 meter
Network Span	925 meter	2500 meter	500 meter	3600 meter
Maximum Nodes per Segment	30	100	n/a	n/a

The physical interface of concern to Remote Access is 10Base-T, as it provides the least expensive and most common network wiring topology without compromising network performance or integrity. The 10Base- designation indicates a fixed 10 megabits per second data rate using baseband signaling, while the T character implies a twisted pair wiring topology.

In a 10Base-T network, the nodes are interconnected via wiring hubs (or concentrators or media attachment units) using four 22 to 26 AWG unshielded wires (two twisted pairs). These wires attach to pins 1, 2, 3, and 6 of an RJ-45 modular jack. Pins 1 and 2 are used for transmitting data, while pins 3 and 6 are used for receiving data. The maximum distance between the nodes in a network is 100 meters. The wiring hubs can also be interconnected to increase the size of a network. Up to four hubs can be daisy-chained together to created a maximum network span of 500 meters.

Wiring hubs provide more than just a connectorization point and wiring concentrator. The hubs also regenerate received signals before transmitting them to the network, reject (do not propagate) severely deformed received signals, and also isolate network segments that appear to be faulty, i.e., generating excessive collisions. The following figure illustrates the ethernet 10Base-T components discussed above.

Figure 14 Ethernet 10Base-T Wiring Components



Ethernet Link Arbitration Access and Control

Baseband ethernet dictates that only one node can transmit data at any given time. Therefore, access to the network must be arbitrated among the nodes sharing the medium. For 10Base-T ethernet, *carrier sense, multiple access/collision detect* (CSMA/CD) is the arbitration scheme governing access to the network medium.

With CSMA/CD, there is no central arbitration authority. Instead, each individual node determines when it is appropriate to transmit data. As the transmitting station has no means to determine if another station is attempting to transmit data at the same time, each station must listen to the wires and wait for the link to become idle before attempting to transmit. Since another station may also be attempting to transmit data at the same time, each station must continue to monitor the wire during transmission and determine if a collision has occurred (two or more stations transmitting at the same time). If a collision is detected, the station will attempt to retransmit the data later. But first the station transmits a jamming signal (causing further collisions) to alert and insure all stations detect the collision and cease transmission. After a random period of time, known as the *backoff period*, the station attempts to transmit the data again. The station keeps count of the number of times an attempt is made to transmit the data. If the number of collisions is deemed excessive, the attempt to transmit the data is aborted and an error is reported.

Figure 15 on page 74 illustrates the simplified CSMA/CD transmission algorithm.

Ethernet Data Encapsulation

As previously stated, each layer of the TCP/IP protocol encapsulates the data from the previous layer, adding header (and perhaps trailer) information, before passing the data to the next layer. Headers contain control information that permit the orderly and reliable exchange of data. By the time the data reaches the Network Access layer, it will have at least two headers added to it--the transport header pertaining to either TCP or UDP, and the IP header to become an IP datagram. At the Network Access layer, ethernet encapsulates the IP datagram using both a header and a trailer, as shown below.

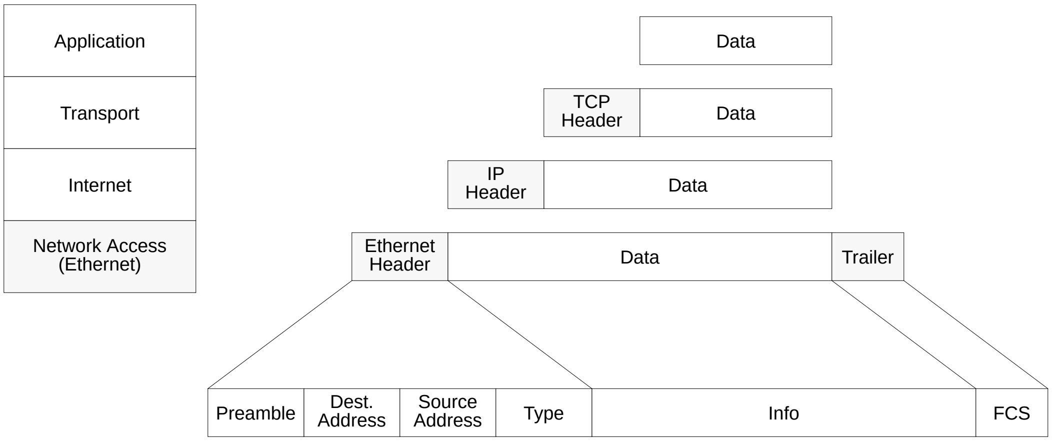
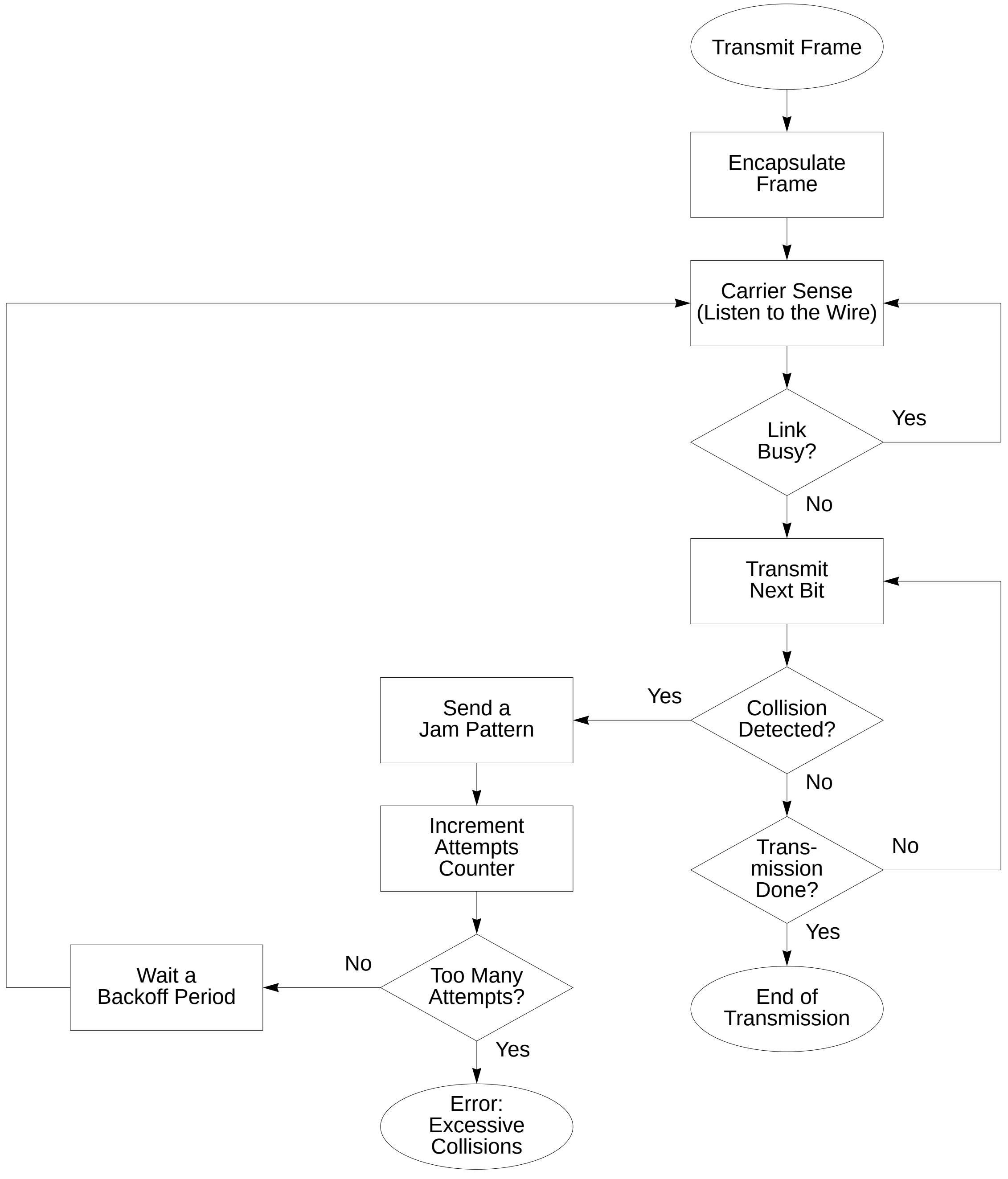


Figure 15 Simplified Ethernet Transmit Process (CSMA/CD)



An example of the ethernet frame format is illustrated ***Figure 16 on page 75***. When an ethernet frame is transmitted, an eight-byte *Preamble* field, consisting of alternating 1’s and 0’s (starting with a 1 and ending with a 0) is sent first. Since it is very improbable that this pattern would be generated by noise on the wire, the preamble serves as an attention getting signal to alert the nodes on the network to an impending transmission.

The next field is a six-byte *Destination Address*, containing the MAC address of the node to which the data is being sent. The destination address is followed by a six-byte *Source Address*, which contains the MAC address of the node from which the data is being sent.

Next comes a two-byte *Type* field, which identifies the protocol (or service user) to which the data will be delivered. Following the type field is the *Information* field, which contains the actual data being transmitted. This field can be from 46 to 1,500 bytes, and can be an IP datagram, an ARP request, or other data.

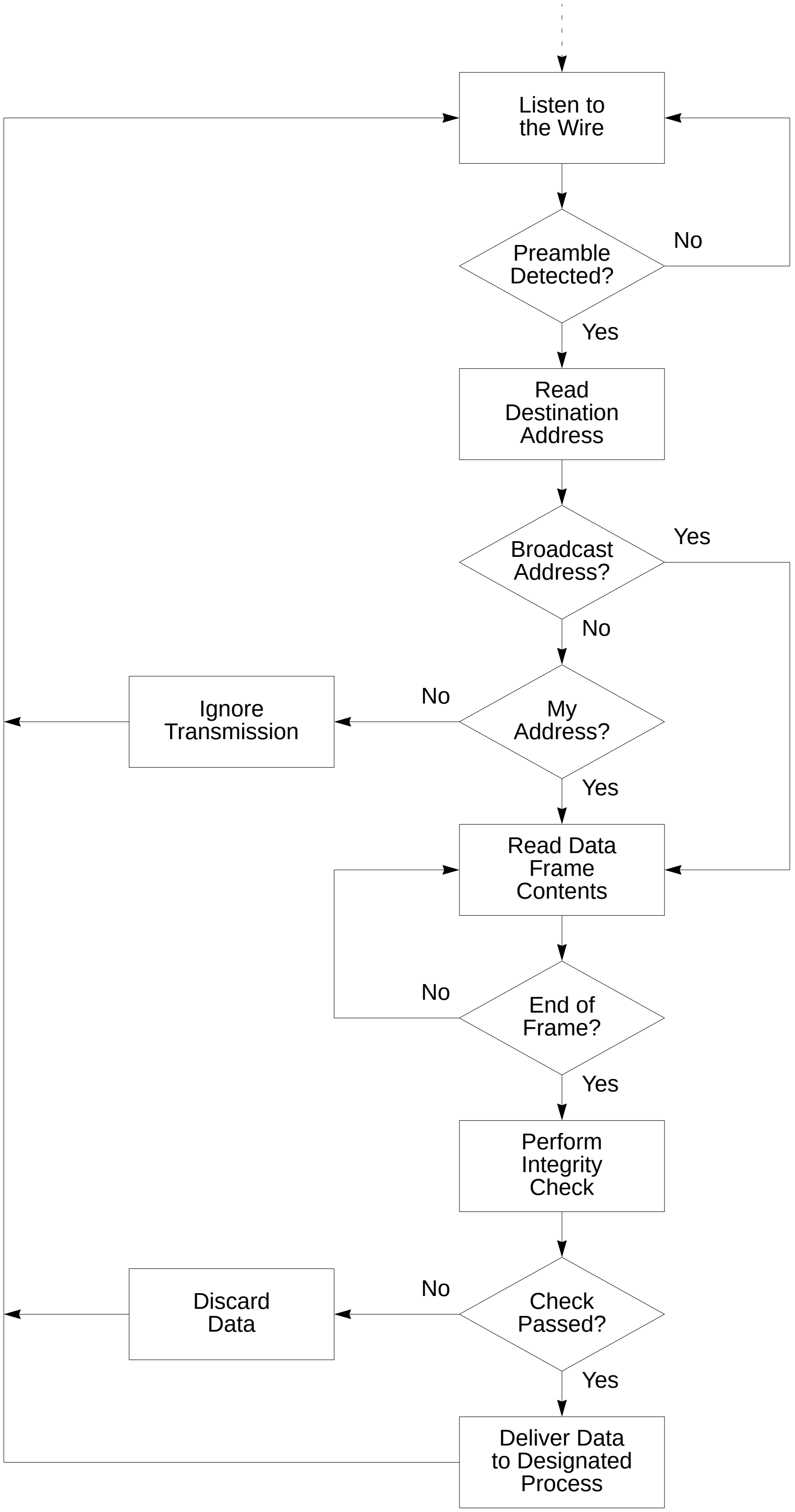
The final field consists of a four-byte *FCS*, or *Frame Check Sequence*, which is used by the receiving end to verify the integrity of the transmission.

Figure 16 Ethernet Frame Format

Ethernet Frame Format					
Preamble	Destination Address	Source Address	Type	Information	FCS
8 Bytes	6 Bytes	6 Bytes	2 Bytes	46 Bytes - 1,500 Bytes	4 Bytes

The simplified receive algorithm for handling an ethernet frame like the above is illustrated in ***Figure 17 on page 76***.

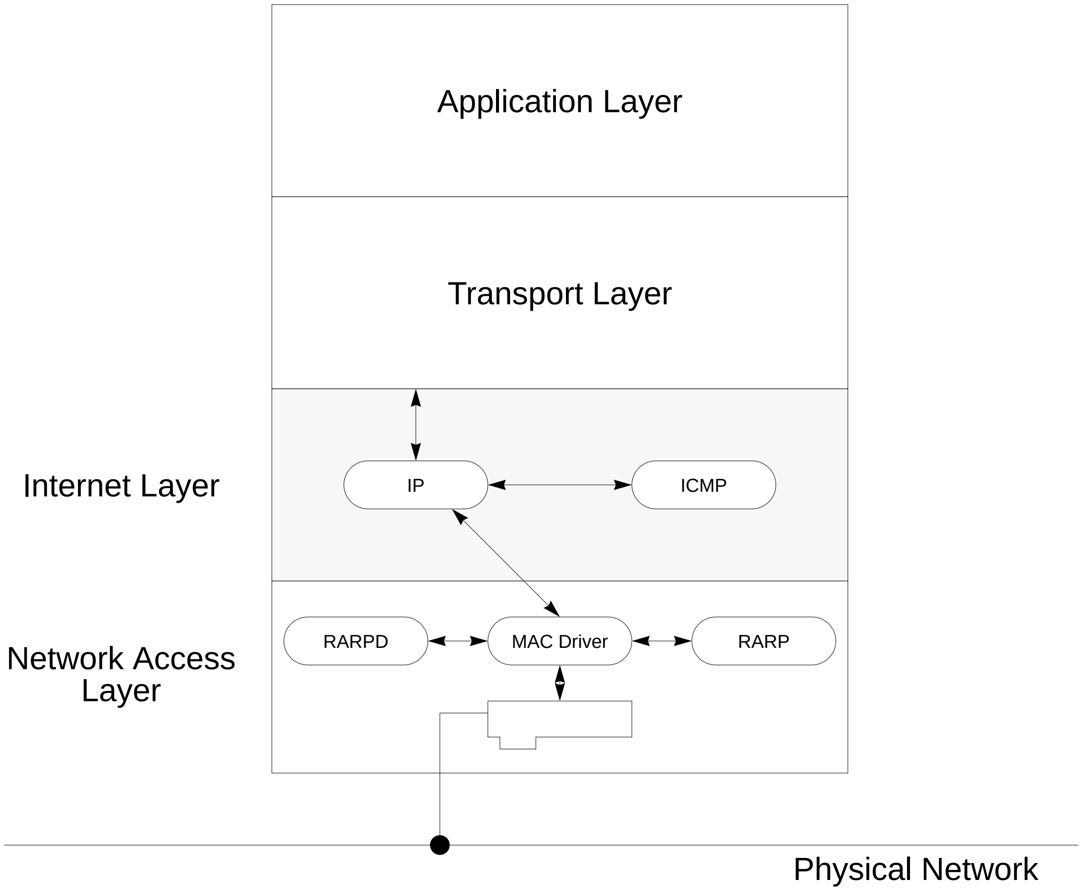
Figure 17 Simplified Ethernet Receive Process



4.1.2.2TCP/IP Internet Layer

The TCP/IP *Internet* layer, which is responsible for routing information across the physical network, roughly corresponds to the *network* layer in the OSI model. At this layer, two primary protocols are defined to complete the routing task. The first protocol, called the *Internet Protocol* (IP), is responsible for routing packets around the internetwork. The second protocol, the *Internet Control Message Protocol* (ICMP), is responsible for routing error detection and recovery. The relationship of the *Internet* layer to the other layers of the TCP/IP architecture is illustrated in ***Figure 18 on page 77***. IP is the cornerstone of the TCP/IP protocol suite. All TCP/IP protocols communicate with their peers on the network via IP datagrams.

Figure 18 TCP/IP Internet Layer Components



Characteristics of Internet Protocol (IP)

IP is a *connectionless* protocol, meaning that it does not attempt to establish a connection with its peer before sending data to it. Additionally, *IP* is also provides an *unreliable* service. That is, IP does not provide error detection or recovery. These functions are delegated to the higher-layer service users of *IP*. All *IP* is concerned with is the delivery of data to the designated destination.

The primary functions of *IP* include:

- Data encapsulation and header formatting
- Data routing across the internetwork
- Data transmission between other protocols
- Data fragmentation and reassembly

IP Data Encapsulation

IP data encapsulation involves accepting data from the *Transport* layer and adding *IP*'s header control information to it. As shown in ***Figure 19*** below, the IP header is five or six 32-bit words in length, depending on whether the optional field is included or not.

Figure 19 IP Datagram Structure

<table>
<tr><th>0</th><th>4</th><th>8</th><th>16</th><th></th><th>31</th></tr>
<tr><td>Version</td><td>IHL</td><td>Type of Service</td><td colspan="3">Total Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flags</td><td colspan="2">Fragment Offset</td></tr>
<tr><td colspan="2">Time to Live</td><td>Protocol</td><td colspan="3">Header Checksum</td></tr>
<tr><td colspan="6">Source IP Address</td></tr>
<tr><td colspan="6">Destination IP Address</td></tr>
<tr><td colspan="5">IP Options</td><td>Padding</td></tr>
<tr><td colspan="6">Data</td></tr>
<tr><td colspan="6">--------------</td></tr>
</table>

The first field in the *IP* header is the *version*, which indicates the version of *IP* in use, with the current version being 4. The second field is the *IHL*, or *Internet Header Length*, which indicates the total length of the *IP* header. The *Type of Service* (*TOS*) field specifies the *class of service* requested by the application. Examples of *class of service* include *minimum delay,* which is used by applications such as *RLOGIN* and *TELNET*, and *maximum throughput*, which is used by applications such as *FTP* and *SMTP*.

The *Total Length* field minus the *IHL* field provide the length of the data field. The *Identification* and *Fragment Offset* fields are used by *IP* during fragmentation and recovery. The *Time to Live* field is initialized by *IP* during transmission and indicates the maximum number of routers the datagram can cross before it reaches its destination. This field is decremented by one by each router it passes through. If the *Time to Live* field reaches zero, the datagram is removed from the network by the next router to detect the anomaly. This prevents a lost datagram from endlessly looping around the network. The *Protocol* field indicates to which protocol the datagram is to be delivered.

Although *IP* is an *unreliable* protocol, it is concerned with the integrity of its own control information header. The *Header Checksum* field is used by *IP* to verify the integrity of the data in the header fields. If the integrity check fails, *IP* simply discards the datagram, with no failure notification is provided to the sending host.

The *Source IP Address* and *Destination IP Address* fields identify the ultimate communicating hosts. The *IP Options* field may provide optional control information, such as a *route record*, which includes the address of every router traversed by the datagram during its trip through the network. The *Padding* field is used as a fill to properly align the beginning of the data field.

IP Routing

IP routing is one of the simplest, yet most efficient methods for routing data on a complex internetwork. With respect to routing, the two primary components for *IP* are a *host* and a *gateway.* A *gateway* in TCP/IP connects two or more networks for the purposes of providing forwarding services between them. A *host* is the end system where user applications execute. By default, routing on a *host* is limited to the delivery of the datagram directly to the remote system if both *hosts* are attached to the same network. If the *hosts* are not on the same network, *IP* delivers the datagram to a *default gateway,* which is a router attached to the same network as the *host*. The *host* “trusts” the *default gateway* to assist with the delivery of the datagram to other *hosts* on remote networks. The simplified IP routing algorithms executed on a *host* is provided in ***Figure 20 on page 81***. Two additional topics are required regarding routing: the MAC and IP address structures.

- Medium Access Control (MAC) Addresses

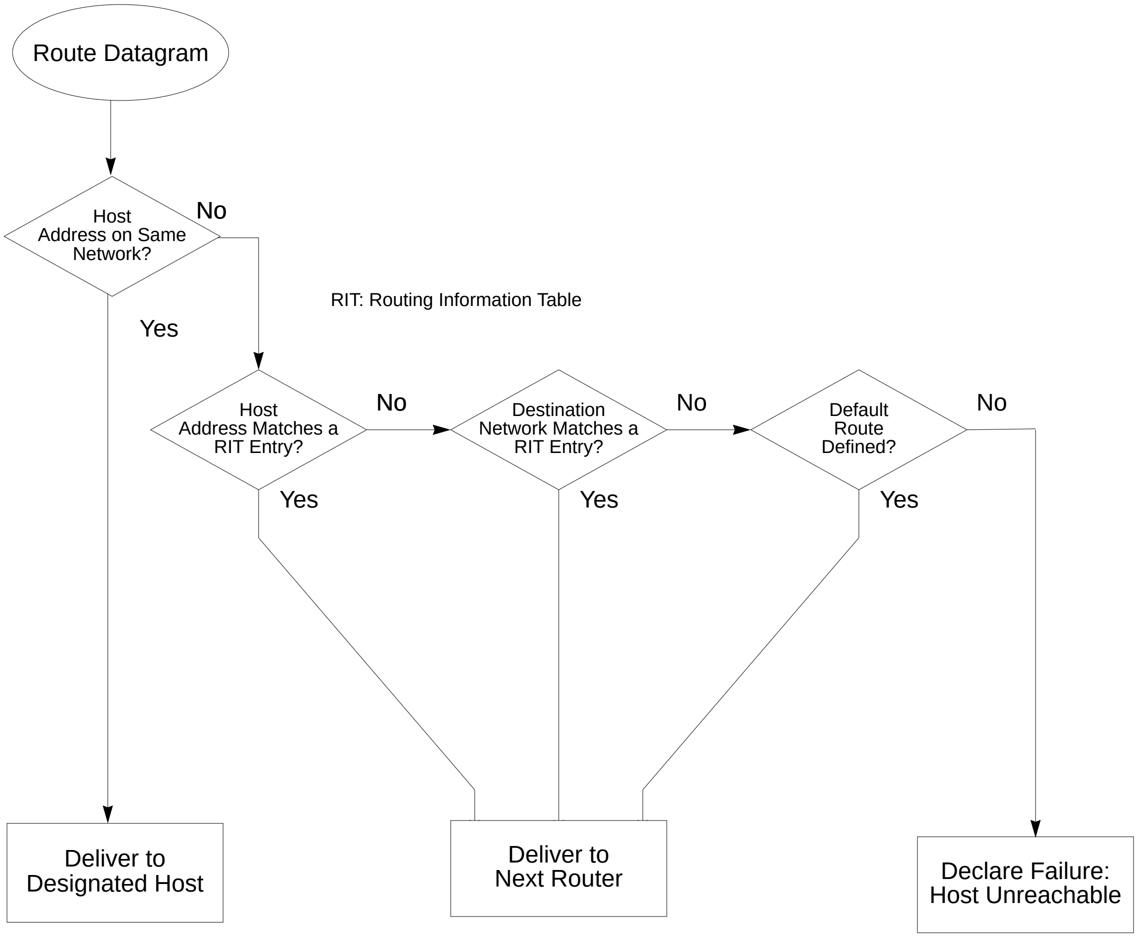
  A MAC address is a 48-bit number that provides a unique identity for devices that are connected to an ethernet network. The MAC address is hardcoded into the network interface itself and is used by the lowest level of the communications protocol to establish the source and destination addresses for all ethernet messages. The IEEE 802 standardization committee has adopted an address format that is identical for all the MAC standards it defines. Three types of addresses are supported by the IEEE standard, as listed below:

  an ***individual address***, which is used to uniquely identify an individual station on a network

  a ***broadcast address***, which has all its address bits set to 1 and is used to broadcast to all active devices on a network

  a ***multicast address***, which is assigned to a logical group of workstations.

Figure 20 Simplified IP Routing Algorithm


Route Datagram
Host Address on Same Network?
No
Yes
RIT: Routing Information Table
Host Address Matches a RIT Entry?
No
Yes
Destination Network Matches a RIT Entry?
No
Yes
Default Route Defined?
No
Yes
Deliver to Designated Host
Deliver to Next Router
Declare Failure: Host Unreachable

According to the IEEE, 48-bit individual addresses can be either locally or universally administered. Locally administered addresses are set up by the LAN user, in which case it is the user’s responsibility to ensure the uniqueness of each assigned address. Universally administered addresses are assigned and administered by the IEEE committee itself. A universally assigned MAC address is guaranteed to be unique worldwide, implying that a workstation with a universal address can be connected to any network with similarly assigned addresses without having to reconfigure the workstation MAC address.

The format of an individual address conforming to the universally administered standard is illustrated below. The two most significant bits are set to 0, with the following 22 bits equal to a unique organizational identifier assigned by the IEEE to communications vendors. The remaining 24 bits are assigned by the equipment vendor and are guaranteed to be unique for all products manufactured by that vendor.

Format of the IEEE Universally Administered MAC Address			
0	0	22-Bit Unique Organizational ID	24-Bit Organizationally Assigned Address

- IP Address Structure

In TCP/IP, every device on the network must have a unique complete network address. This assigned address, known as a *symbolic IP address*, is composed of two parts: the *network address*, which is common to all devices on the same physical network, and the *node address*, which is unique to each host on that network. Note, however, that neither of these addresses has anything to do with the hardcoded MAC address on the network interface card.

The IP address is used by TCP/IP to route data between nodes on a network. The IP address is a 32-bit number (four bytes) that occupies the source and destination fields of the IP header. How many bits of the IP address belong to the network address and how many bits belong to the node address depends on the IP address class into which the address falls. IP defines five different classes of networks, as shown below. Note that only three classes (Class A, Class B, and Class C) are of interest to Remote Access.

Network Class	**First Byte of IP Address**	**Decimal Range of First Byte**	**Binary Representation**			
Class A	0	0-127	0xxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
			Network Byte	Three Node Bytes		
Class B	10	128-191	10xxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
			Two Network Bytes		Two Node Bytes	
Class C	110	192-223	110xxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
			Three Network Bytes			Node Byte
Class D	1110	224-239	1110xxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
			Three Network Bytes			Node Byte
Class E	1111	240-255	11110xxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
			Three Network Bytes			Node Byte

The following observations can be made from the above:

1. There are 126 Class A networks, with each Class A network supporting up to 16,777,214 hosts. Also note 0.0.0.0 is reserved as the default network address and 127.0.0.0 is reserved as the loopback network.
2. There are 16,384 Class B networks, with each Class B network supporting up to 65,534 hosts.
3. There are 2,097,152 Class C networks, with each Class C network supporting up to 254 hosts.
4. The lowest address in a network (x.0.0.0) is reserved for the address of the network itself, and the highest address in the network (x.255.255.255) is reserved for the broadcast address.
5. Class D networks are reserved for multicasting applications and Class E networks are reserved for experimental use.

IP Data Transmission Between Protocols

Since all TCP/IP protocols send their data in *IP datagrams*, some mechanism must be provided to assist *IP* with submitting received datagrams to the intended protocol. The *protocol* field in the *IP* header serves this purpose. As per TCP/IP standards, each protocol that uses *IP* routing services is assigned a protocol number. *IP* uses this identifier to direct received datagrams to the appropriate protocols. Example protocol identifiers include six for *TCP*, one for *ICMP*, and zero for *IP* itself.

IP Fragmentation and Reassembly

Referencing ***Figure 19 on page 78***, it is observed that the length field in the *IP* header is 16-bits wide, indicating that *IP* can manage datagrams up to 65,535 bytes in size. However, some underlying networks may not tolerate data frames that large. (An ethernet frame, for example, cannot exceed 1,514 bytes.) Therefore, *IP* provides the ability to fragment large datagrams to a size compatible with the underlying network capabilities. This is referred to as *data fragmentation*.

While in most cases, all data fragments will follow the same route from sending to receiving host, it is possible that some fragments may follow a different route (due to network congestion, for example). In this case, it is highly likely that the fragments will be received out of order from how they were transmitted. In such instances, *IP* uses the *Fragmentation Offset* field of the *IP* header to properly sequence the fragments to insure the data is restored to the original order. *IP* does not pass data to the higher layer protocols until all fragments are correctly received and *reassembled*. In practical systems, *IP* will be concurrently managing multiple datagrams from numerous protocols. *IP* uses the *Identification* field of the *IP* header to distinguish between fragments belonging to different datagrams. The same value is placed in the *Identification* field to uniquely associate fragments belonging to the same datagram.

Characteristics of Internet Control Message Protocol (ICMP)

The *Internet Control Message Protocol* is an integral component of the *IP* protocol that transfers messages between hosts. These messages include control, informational, and error recovery data. Examples of such messages include:

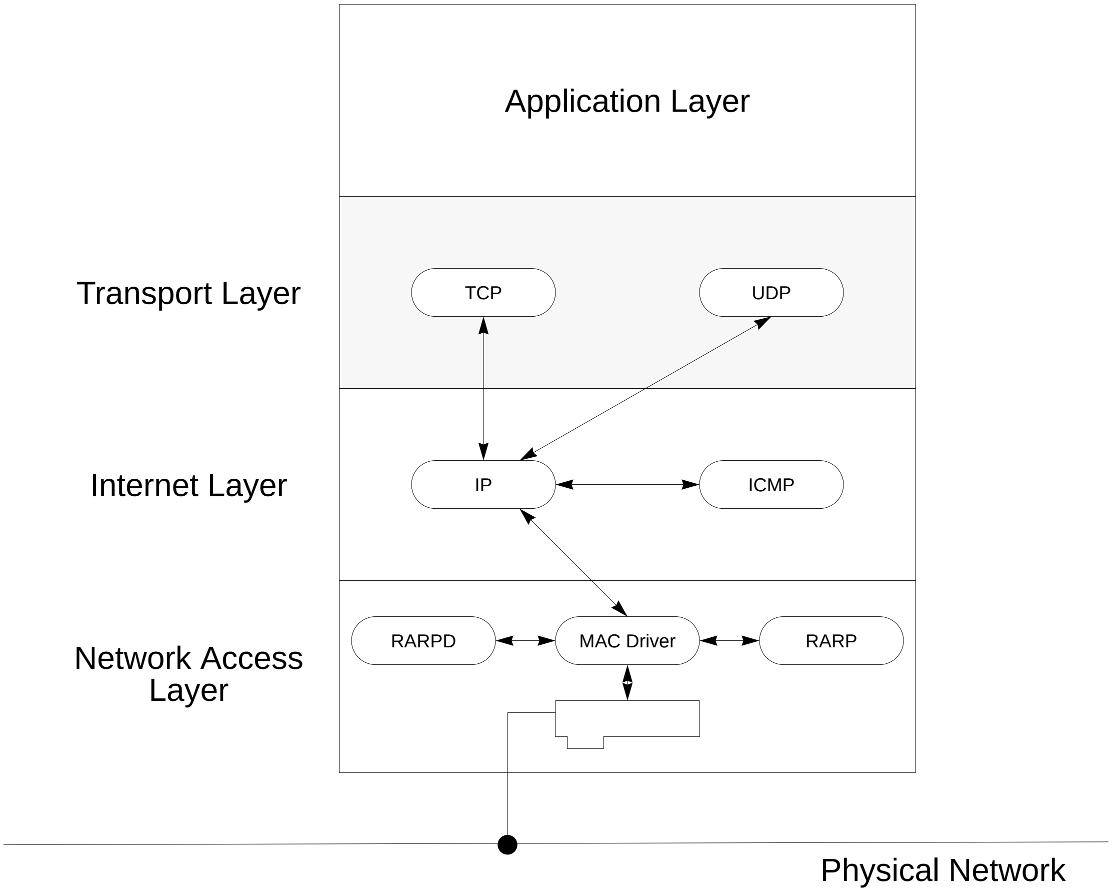
- ***Source Quench:*** This is a flow control message that a receiving host sends to the source, requesting that it stop transmitting data. This normally happens when the receiving host’s input buffers are close to full.
- ***Route Redirect:*** This is an informational message that a gateway sends to a host seeking its routing services. A gateway sends this message to inform the sending host about another gateway on the network that it trusts to be closer to the destination.

- ***Host Unreachable:*** A gateway or host that encounters a problem during the delivery of a datagram (due to link failures, network congestion, end host failures,...) sends this error message to the sending host. Normally, the *ICMP* packet includes information detailing the reason why the host cannot be reached.
- ***Echo Request/Echo Reply:*** These *ICMP* messages are invoked by the execution of the ***ping*** command on a host. When executed, ***ping*** sends a *echo request* packet from the host executing the command to the remote system entered with the ***ping*** command. The remote system, if operational, responds with an *echo reply* message. The successful reception of the *echo reply* message may be interpreted as proof of network connectivity between the two nodes.

4.1.2.3TCP/IP Host-to-Host Transport Layer

The *Host-to-Host Transport* layer, or simply the *Transport* layer, is the only layer aware of the identity of the ultimate user representative processes at the *Application* layer. As such, this layer is responsible for the delivery of information between applications executing on one host and applications executing on another host. At the *Transport* layer, *Application* layer protocols are assigned *port numbers*, which are used in the source and destination port fields in the *Transport* protocol header. The *Transport* layer uses the *port numbers* much like *IP* uses the *protocol* field: to distinguish between *Application* layer protocols utilizing the services. Two communicating hosts must have transport protocols that use the port numbers in the same fashion for the data to be correctly delivered to the intended application protocols. The Transport Layer and its relation to the other layers in the TCP/IP architecture is illustrated in ***Figure 21 on page 85***.

Figure 21 TCP/IP Transport Layer Components



TCP/IP defines two protocols at the *Host-to-Host Transport* layer: *UDP* (*User Datagram Protocol*) and *TCP* (*Transmission Control Protocol*). UDP is a *connectionless, unreliable* protocol, while TCP is *connection-oriented, fully-reliable* protocol.

UDP Datagram Protocol (UDP)

UDP provides a connectionless, unreliable service to the application protocols. Being connectionless, *UDP* undergoes no handshaking mechanism, nor does it negotiate any control parameters with its peer before exchanging application protocol data. The applications using the service of *UDP* must perform any necessary flow control, data resequencing, and error control. The structure of the *UDP* datagram is illustrated in ***Figure 22 on page 86***.

Figure 22 UDP Datagram Format

0 16 31

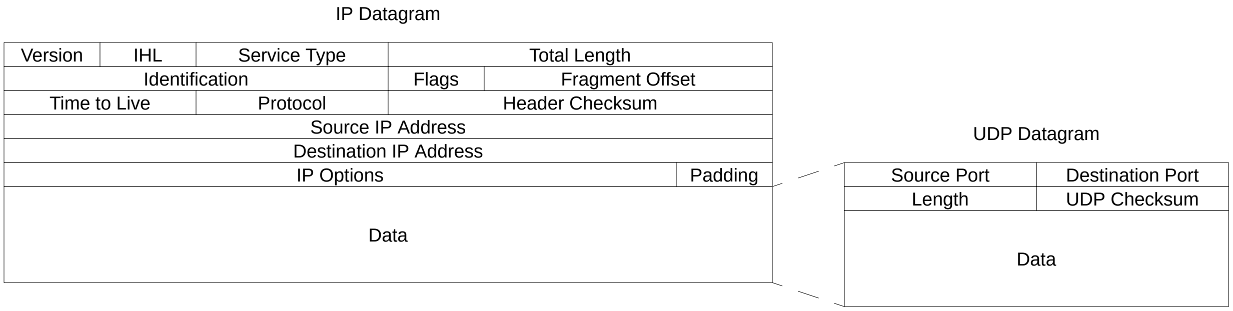
Source Port	Destination Port
Length	UDP Checksum
Data	

The *UDP* datagram is composed of two parts: the *UDP* header, which contains information pertinent to the data exchange process, and data field, which contains the application data.

The *UDP* header is composed of four 16-bit fields. The *Source Port* and *Destination Port* fields designate the application protocols which are using the service of *UDP*. The *Length* field indicates the total length of the *UDP* datagram, including the header information, while the *UDP Checksum* field is used to validate the integrity of the transmitted data.

UDP users *IP* services to send and receive data from the network. The relationship between the *IP* and UDP *datagrams* is illustrated in ***Figure 23 on page 86***. Notice that the *UDP* datagram becomes the occupant of *IP*'s data field. As a final note, *UDP* does not support *data fragmentation*; this function would have to be performed by the application using the UDP service.

Figure 23 Relationship Between UDP and IP Datagrams



IP Datagram

Version	IHL	Service Type	Total Length	
Identification		Flags	Fragment Offset	
Time to Live	Protocol	Header Checksum		
Source IP Address				
Destination IP Address				
IP Options				Padding
Data				

UDP Datagram

Source Port	Destination Port
Length	UDP Checksum
Data	

Transmission Control Protocol (TCP)

TCP is a fully-reliable, connection-oriented, acknowledged, data stream-oriented service. *TCP* supports data fragmentation and reassembly, as well as data multiplexing and demultiplexing using source and destination port numbers (similar to *UDP*). The *TCP* data segment format is illustrated in ***Figure 24 on page 87***.

Figure 24 TCP Data Segment Format

<table>
<tr><th colspan="3">0</th><th colspan="2">16   31</th></tr>
<tr><td colspan="3">Source Port</td><td colspan="2">Destination Port</td></tr>
<tr><td colspan="5">Sequence Number</td></tr>
<tr><td colspan="5">Acknowledgment Number</td></tr>
<tr><td>Header Length</td><td>Reserved</td><td>Code Bits</td><td colspan="2">Window</td></tr>
<tr><td colspan="3">Checksum</td><td colspan="2">Urgent Pointer</td></tr>
<tr><td colspan="4">Options (if any)</td><td>Padding</td></tr>
<tr><td colspan="5">Data</td></tr>
<tr><td colspan="5">----------------</td></tr>
</table>

Being a fully-reliable service, *TCP* guarantees the error-free delivery of data between systems and safeguards data against loss, duplication, and corruption from noise, physical failures of the wires, or other network anomalies. The primary features of TCP are:

Connection-Oriented: *TCP* is a connection-oriented protocol. Before any user data is exchanged by applications, *TCP* performs a handshake process, during which connection control parameters are negotiated and a *virtual circuit* connection established.

Data Fragmentation and Reassembly: *TCP* is capable of breaking up large data segments into smaller pieces that can be better accommodated by the underlying network.

Reliable Transfer: *TCP* provides reliable data transfer by using sequence numbers, integrity checksums, positive acknowledgments, and retransmissions. Retransmissions are determined based on the sequence number, acknowledgment number, and checksum fields.

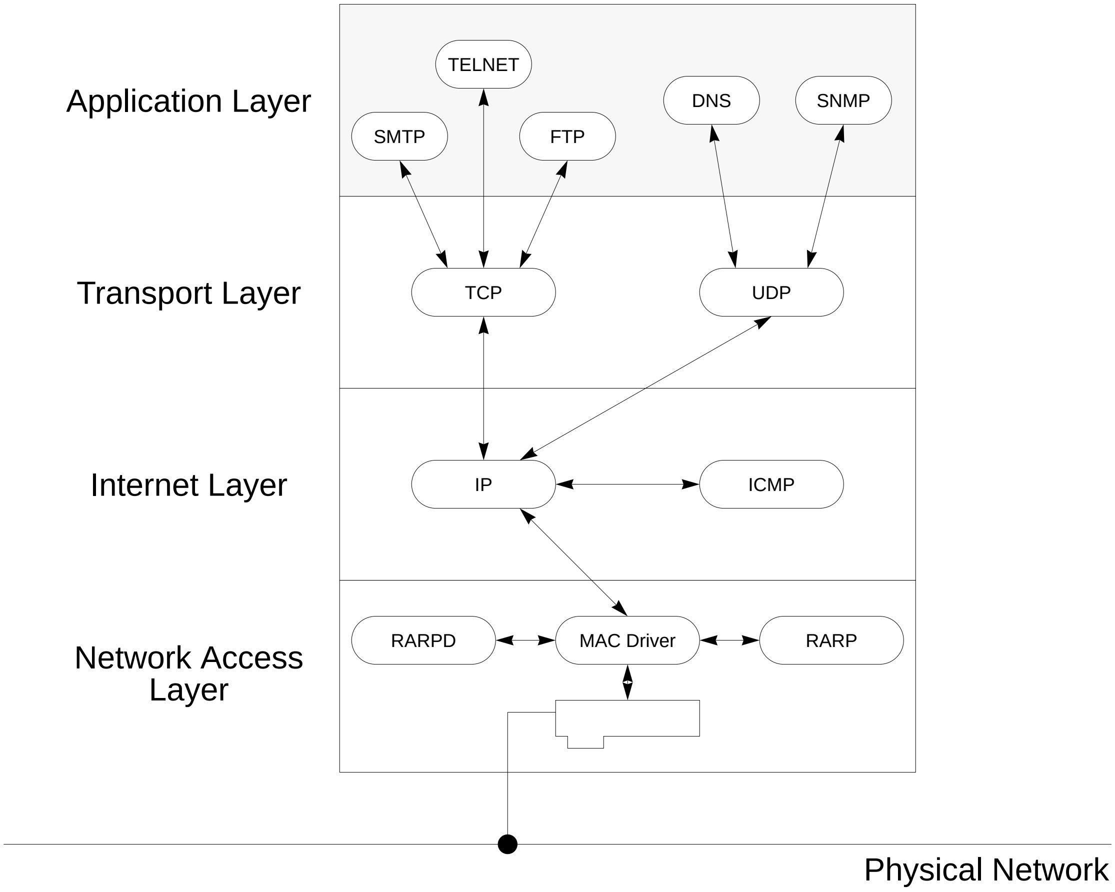
Data Stream Orientation: *TCP* transmits application data as unstructured streams of bits aligned around boundaries of 8-bit bytes. *TCP* does not recognize any form of data structures such as records or fields, and its services are application independent. Any data structure is provided at the application layer.

Sliding Window Technique: *TCP* uses a sliding window technique for data transmission, which improves transmission efficiency by reducing the connection idle time. This technique also permits effective/efficient data flow control.

4.1.2.4TCP/IP Application Layer

The application protocols are resident at the top layer of the TCP/IP architecture, which is known as the *Application* layer. This layer provides the user-oriented functions and features not provided by the lower layer protocols. For example, at this layer, users would be able to use the *ftp* application protocol to transfer files across a network. The two hosts involved in the file transfer need not be similar, because ftp uses the supplied protocols to circumvent/resolve any differences that may exist between the communicating systems. The *Application* layer and its relation to the other elements of the TCP/IP architecture is illustrated in ***Figure 25 on page 88***.

Figure 25 Example TCP/IP Application Layer Components



Some of the application layer protocols supported by TCP/IP include:

File Transfer Protocol (FTP): *FTP* is a file transfer protocol that handles file transfers between like or dissimilar systems across a TCP/IP network. For example, users on a DOS system would use the DOS implementation of TCP/IP to transfer files between a unix host.

Telnet: Telnet provides users with terminal emulation and login services. Using telnet, a users can establish a connection to a remote host and use its resources.

Simple Mail Transfer Protocol (SMTP): *SMTP* is the primary engine for internet electronic mail applications.

Network File System (NFS): *NFS* provides files sharing services on various hosts sharing a network. A host with *NFS* running in the background allows other hosts on the network to have access to its file system resources by connecting those resources to the remote hosts' local file systems. The end user is able to treat the *NFS* accessed file system as if it were local to that host.

Simple Network Management Protocol (SNMP): *SNMP* provides network managers and administrators with the capability to gather and analyze performance statistics pertaining to the hosts on the network.

4.1.3 TCP/IP WAN Protocols: PPP

In addition to local network media such as ethernet and token-ring, TCP/IP also supports a number of wide area network serial link protocols for communications between geographically separate hosts. The links between these widely separated nodes would typically supports data rates from 56 kbps (DS0 rate) to 1.536 Mbps (T1 rate) and higher. The WAN protocol of importance to the Remote Access feature is known as Point-to-Point Protocol, or simply PPP.

PPP is a TCP/IP, media-independent standard supporting both bit-oriented synchronous, and byte-oriented asynchronous, data transmission, at speeds ranging from 1200 bps through 1.536 Mbps. PPP supports RS-232, RS-422, and V.35 interfaces, as well as 56 kbps or 1.536 Mbps dedicated facilities, and dial-up links. PPP can also be used to connect dissimilar networks such as ethernet, token-ring, and FDDI.

The PPP data frame format, which is an adaptation of the HDLC (High-level Data Link Control) protocol format, is illustrated below. PPP is characterized by reduced protocol overhead and high throughput. As shown below, each PPP data frame starts and ends with a single byte *Flag* character set to *0x7E*. A one byte

Address field, set to *0xFF* follows the start flag. Following the *Address* field is a *Control* byte set to *0x03*.

PPP Frame Format							
Flag	Address	Control	Protocol Info		Information (IP datagram or other data)	FCS	Flag
7E	FF	03	Byte 1	Byte 2		2 or 4 Bytes	7E

A two byte *Protocol* field comes next. This field contains a universally assigned protocol identification number which describes the nature of the data contained in the *Information* field. And finally, the *FCS*, or *Frame Check Sequence*, field provides a means to verify the end-to end integrity of the transmission. All but the flag fields are included in the FCS generation and verification.

4.2 Router/Hub

The third party routing equipment is the central enabling technology for the Remote Access feature. Since all of the routing details necessary to access a remote location are maintained in the networking equipment, the ADAS OAM Workstation and the Remote Access positions can have very generic LAN configurations. In other words, the workstations do not have to maintain routing information for every node on the remote access network. Furthermore, changes or additions to the remote access network are automatically adjusted for by the routing protocols in the networking equipment and do not require changes to the workstation LAN configurations.

For remote access purposes, the workstations only need to know two IP addresses. The first is that of the router, and the second is that of the destination node. The routing equipment maintains all the information required to connect the remote node with the host workstation. Thus, details of WAN network are effectively hidden from the individual workstations, simplifying both the configuration of the workstation LAN interface and the maintenance of the network.

4.2.1 Information Required to Configure Remote Access Router

Regardless of whether the customer chooses to use the routing equipment recommended herein or equipment from another vendor, numerous details regarding the remote access network must be established prior to configuring the routing equipment. Many of these details are specific to each node in the remote access network, such as the IP address of the workstation, while numerous others are common to the entire remote access network, such as the protocol for the WAN links. The following list provides a run-down of the principal parameters required to configure the routing equipment for remote access.

1. Ethernet Hub Port IP Address: ***host specific***
2. Ethernet Hub Port Subnet Mask: ***255.255.255.192***
3. WAN Port #1 IP Address: ***node specific***
4. WAN Port #1 Subnet Mask: ***255.255.255.252***
5. WAN Port #2 IP Address: ***node specific***
6. WAN Port #2 Subnet Mask: ***255.255.255.252***
7. IP Routing Protocol: ***OSPF***
8. WAN Routing Protocol: ***PPP***
9. Routing Specific Parameters

4.2.2 Sample Access Node Hub (ANH) Configuration Session

The following section illustrates a sample configuration session using the Technician Interface to configure the Access Node Hub (ANH) for Remote Access. Responses entered by the user are shown in **bold** type. In many instances, the ANH provides a default or anticipated response for a parameter--these defaults are enclosed by square brackets [ ]. The user may simply press the enter key to accept the default values.

For clarity, all requested parameters are explicitly entered in the following file, even when the entered values are identical to the default values.

Note: Comments are added throughout the file for clarification purposes. These comments are indicated by *italic* type and are enclosed in brackets < >. These comments are not part of the screen interaction for the ANH configuration session.

<First, login to the ANH. From the login prompt, enter the following user ID.>

```
Login: Manager
Mounting new volume...
Device label:
Directory: 1:
New Present Working Directory: 1:

      Welcome to the Backbone Technician Interface
```

<Confirm the configuration options by entering the following command.>

```
[1:1]$ getcfg

Boot Options:
      boot image=local <The boot image is stored on the Flash Card>
      boot config=local <The configuration file is stored on the Flash Card>

NetBoot Parameters:
      XCVR1...None <Local boot only>
      COM1....EZ-Install <Local boot only>
      COM2....EZ-Install <Local boot only>
```

<Query existing router configuration>

```
[1:1]$ show ip base
IP not configured

[1:1]$ show ip circuits
No Circuits found

[1:1]$ show ip routes
No routes found
```

<Configure the router via the local Technician Interface by entering the following command>

```
[1:1]$ run install.bat
      More Mode: OFF
      Lines per screen: 24
-----------------------------------------------------------------

 ####    #   #   #      #   # #### ##### #     #  ##  ###  #  #  ###
 #   #  # #   # #       ##   # #      #   #     # #  # #  # #  # #
 ####  #   #   #        # # # ###    #  #  #  # #  # ###  ##   ###
 #   # #####   #       #  ## #      #   # # # # #  # #  # # #     #
 ####  #   #   #        #   # ####   #   #   #   ##  #  # #  # ###

         ###  #  # ###  ## #  #      ### #####  #   ###  #####
        #   # #  #  #  #   # #      #      #   # #  #  #   #
        #   # #  #  #  #   ##   ### ###    #  #   # ###    #
        # # # #  #  #  #   # #         #   #  ##### #  #   #
         ###  #### ###  ## #  #     ###    #  #   # #  #   #
            #

                          Version 1.166
                       Copyright 1993-1996
```

```
-----------------------------------------------------------------

                         Introduction
                         ------------

This part of the Quick-Start procedure configures the initial IP
network interface on the router. You perform this procedure so
that
the router can communicate with the network management station.

Each step of this procedure is further described in the Quick-
Start Guide.
As you perform the procedure, refer to the Quick-Start Guide for
additional helpful information and examples.

When you are finished with this procedure, the router will be able
to
communicate with the network management station over the IP
network.  You
are then ready to install the network management software, as
described in the Quick-Start Guide.

Each procedure step requires you to do one of the following things:
      1. Enter a number that corresponds to a selection.
      2. Enter ‘y’ for Yes;  ‘n’ for No; ‘q’ for Quit.
      3. Enter a word or phrase referred to as a “text string”
      4. Enter <Return> to accept default displayed in [].

You must press the <Return> key after entering one of the above
responses.

Press <Return> to Continue, q<Return> to Quit: <Press enter or return key>
-----------------------------------------------------------------

             Preliminary Information You Need to Know
             ----------------------------------------

Before you begin this procedure, you should gather the network
information listed below:

You Need to Know This Information:              For Example:
----------------------------------              ------------
Type of Link Module connecting the router’s     DSDE
IP network interface to the Site Manager.

Slot number where the Link Module resides.      2

Communication type and connector number         Ethernet XCVR1

IP address of initial IP network interface      192.32.10.189

Subnet mask of initial IP network interface     255.255.255.0

IP address of Site Manager workstation          192.32.10.100
```

```
Do you wish to continue? (y/n)[y]: y <Continue with configuration>
-----------------------------------------------------------------

Step 1. Specify the slot number where the Link Module resides.

                    Slot Menu for Link Module
                    -------------------------

Slot     Link Module        Processor Module
----     -----------        ----------------
1        ANSEDSG            Access Node

Note: AN system, default is slot 1.
Slot 1 selected.<The ANH provides this information - no user input required>

-----------------------------------------------------------------
Step 2. Specify the Link Module and network interface information
for
       the initial IP connection to the Site Manager.

Link Module: ANSEDSG

Driver Type Menu
----------------
1. Ethernet
2. Synchronous

Enter driver type number [1]: 2 <Configure the WAN ports first>

Connector Menu
--------------
1. COM1
2. COM2

Enter connector number [1]: 1 <The first WAN port is selected>

Clock Source
------------
1. Internal
2. External

Enter clock source number [2]: 2 <Always select external clock source>

Recommended Circuit Name: S11

Enter circuit name [S11]: S11 <Select default value>
-----------------------------------------------------------------

Step 3.  Specify the IP configuration information for the network
      interface.

                      IP Configuration Menu
                      ---------------------
```

```
IP address format:###.###.###.###

IP subnetwork mask format: ###.###.###.###
      Example:             255.255.255.0
```

<Enter the appropriate IP address for the specific router. This address will be different for each router in the Remote Access network. IP addresses should be consistent with the recommendations in this document.>

```
Enter IP address in dotted decimal notation: 192.168.0.5
```

<If the recommendations of this document are followed, the following netmask should be used for all router WAN ports in the Remote Access network.>

```
Enter IP subnetwork mask in dotted decimal notation:
255.255.255.252
```

<The Remote Access network does not employ a Site Manager workstation.>

```
Is the router connected to the same local area network as
the Site Manager workstation? (y/n)[n]: n

Since the router is not on the same network as the Site
Manager workstation an IP Routing Protocol must be
configured in order to manage the box remotely

              IP Routing Protocol Configuration Menu
              --------------------------------------
      1. RIP
      2. OSPF
      3. Static Route to Site Manager.

Enter Routing Protocol Number [1]: 2 <Select OSPF as the IP routing protocol>

-----------------------------------------------------------------

                     OSPF Configuration Menu
                     ------------------------

OSPF Router ID
--------------

The Router ID uniquely identifies this router in the OSPF
domain. By convention, and to ensure uniqueness, one of the
router's IP interface addresses should be used as the Router
ID.

The Router ID will determine the Designated Router on a
broadcast link if the priority values of the routers being
considered are equal. The higher the Router ID, the greater
priority.
```

<Accept the default parameter provided by the ANH. It must be the same as the IP address entered for the WAN port in Step 3 above.>

```
Enter OSPF router ID in dotted decimal notation [192.168.0.5]:
192.168.0.5

OSPF Area ID
------------

Next configure the OSPF Area ID. Remember that it must match the
Area ID of this router’s neighbor.

Note: The backbone area ID is always 0.0.0.0.
```

<Accept the default parameter provided by the ANH. Do not change this value.>

```
Enter the OSPF area ID in dotted decimal notation [0.0.0.0]:
0.0.0.0

OSPF Authentication Type
------------------------

Enable or disable password authentication for the area.  If
you select Simple Password (enabling password authentication),
only those routers sharing the correct password will be able to
communicate with each other. If you accept the default None,
password authentication is disabled for this area.
```

<The Remote Access network does not use password authentication.>

```
Enable Simple Password authentication? (y/n)[n]: n
Default Route For Unknown Subnets
---------------------------------

The default route will not apply for subnets unless
default route for unknown subnets is enabled.
```

<Enable default routes for unknown subnets.>

```
Follow default paths for unknown subnets? (y/n)[n]: y


OSPF MTU Configuration
----------------------

Select the MTU size for OSPF packets sent out this interface

      1. Default
      2. Ethernet-size (Bay Networks 5-series compatible)
      3. User Defined MTU

Enter the OSPF MTU size selection [1]: 2
```

<Select ethernet size MTU>

```
OSPF Interface Type Menu
------------------------

Select the interface’s type (the type of network to
which it is attached). Set this parameter to Broadcast if
this network is a broadcast LAN, such as Ethernet. Set it
to NBMA for an X.25 or similar type of interface. Set it
to point-to-point for a synchronous, point-to-point
interface.  OR, set it to point-to-multipoint for a star
Frame Relay topology

      1. Broadcast
      2. NBMA
      3. Point to Point
      4. Point to MultiPoint (Proprietary)
      5. Point to MultiPoint (Per OSPF Standard)

Enter OSPF interface type selection [1]: 3 <The WAN port is a Point to Point
network>

OSPF Hello Interval
-------------------

The Hello Interval Indicates the number of seconds between the
Hello Packets that the router sends on the interface.  Set this
to the value that the other router(s) on the network are using.

Enter decimal value in seconds for Hello Interval [10]: 10 <Accept
default>

OSPF Router Dead Interval
-------------------------

The Dead Interval Indicates the number of seconds that a
router’s Hello packets have not been seen before it’s neighbors
declare the router down. Set this to the value that the other
router(s) on the network are using.

Enter decimal value in seconds for Router Dead Interval [40]: 40
<Accept default>

OSPF Router Priority
--------------------

Select the priority of this interface. The Router Priority value
is
used in multi-access networks (Broadcast, NBMA, or Point-to-
multipoint),
for the election of the designated router. If this parameter is set
to 0, this router is not eligible to become the designated router
on
this particular network.

In the case of equal Router Priority values, the router ID will
```

```
determine which router will become designated router. However, if
there already is a designated router on the network when you boot
up, it will remain the designated router no matter what your
priority or router ID.

Enter decimal value for Router Priority [1]: 1 <Accept default>

-----------------------------------------------------------------

                    OSPF Configuration Summary
                   ---------------------------

1. Router ID                                192.168.0.5
2. Area ID                                  0.0.0.0
3. Stub Area                                No
4. Authentication                           No
5. Configured Password                      ""
6. OSPF MTU Size                            Ethernet-size
7. Interface Type                           Point-to-Point
8. Hello Interval                           10
9. Router Dead Interval                     40
10. Router Priority                         1
11. Poll Interval                           ------
12. Configured Neighbors                    ------

Are the values specified correct? (y/n)[y]: y <Accept if information is
correct>
OSPF Configuration Complete

-----------------------------------------------------------------

                     Wide Area Protocol Menu
                     -----------------------

1. Bay Networks Point-to-Point Protocol (Proprietary).
2. Frame Relay
3. Point-to-Point Protocol Standard (PPP)
4. Switched Multimegabit Data Service (SMDS)

Enter wide area protocol number [1]: 3 <Select PPP as WAN protocol>

-----------------------------------------------------------------

                      PPP Line Configuration
                      ----------------------


PPP Echo Configuration
----------------------

The PPP Echo configuration sets the parameters governing
Echo-Request packet transmission rate and Echo-Reply packet
acceptable loss threshold.
```

```
Do you wish to turn on the PPP echo function? (y/n)[n]: y <Enable
echo>

Echo Request Timer
------------------

Select the number of seconds that the router waits between the
transmission of Echo-Request packets.

Number of seconds (1-100) [3]: 10 <Provide enough time for slow networks>

Echo Reply Loss Threshold
-------------------------

Select the number (1-100) for Echo-Reply acceptable loss. This is
the
number of unacknowledged Echo-Reply packets counted before
declaring
the link down.

Enter echo loss threshold (1-100) [3]: 10 <Provide enough tolerance for slow
networks>

Local Authentication Protocol
-----------------------------

Bay Networks supports PAP (Password Authentication Protocol)
or CHAP (Challenge Handshake Authentication Protocol).
This feature is disabled if neither PAP or CHAP is chosen.

      <Passwords are not used in the Remote Access network.>

Enable PAP (Password Authentication Protocol) (y/n)[n]: n
Enable CHAP (Challenge Handshake Authentication Protocol) (y/
n)[n]: n


Remote Peer Authentication Protocol
-----------------------------------

Does the Remote Peer have PAP authentication enabled? (y/n)[n]: n

Link Quality Reporting Protocol
-------------------------------

Specify whether or not to enable the Link Quality
Reporting (LQR) Protocol.

NOTE:  Link Quality Monitoring on a Bay Networks 5-series router is
not
       compatible with this feature since the 5-series LQ functions
       are based on older, non-compatible RFCs.
```

```
Enable the LQR Protocol? (y/n)[n]: n <LQR is not used>

-----------------------------------------------------------------

                    PPP Configuration Summary
                    -------------------------

1. Echo Protocol Enabled:                Yes
Time between Xmit of Echo-Requests:      10
Echo-Reply Acceptable Loss:              10
2. Local Authentication:                 Disabled
3. Remote Authentication:                Disabled
4. Link Quality Monitoring:              No

Are these PPP parameters correct? (y/n)[y]: y <Accept if information is correct>
PPP Configuration Complete
-----------------------------------------------------------------

                  SNMP Community Management Menu
                  ------------------------------

Setting up SNMP community management is optional.

It allows you to limit control of this router to a single
Site Manager workstation at a given IP address.  The default
is to allow any Site Manager from any workstation to manage
and to configure the router.

Note: You can later configure this using Site Manager.

      <SNMP is not currently used in the Remote Access network.>

Do you wish to set SNMP community management? (y/n)[n]: n
-----------------------------------------------------------------
Step 4.  Select TFTP default volume.

                     TFTP Default Volume Menu
                     ------------------------

NVFS File System:

VOL    STATE        TOTAL SIZE   FREE SPACE      CONTIG   FREE
SPACE
-----------------------------------------------------------------
 1:    FORMATTED    4194304      1785565         1785565

Enter volume number [1]: 1 <Accept default value>

TFTP default volume is 1:

-----------------------------------------------------------------
Step 5.  Select FTP default volume.
```

```
                          FTP Menu
                          --------

Do you want to enable FTP? (y/n)[n]: y <Enable FTP>

NVFS File System:

VOL    STATE        TOTAL SIZE   FREE SPACE      CONTIG   FREE
SPACE
-----------------------------------------------------------------
 1:    FORMATTED    4194304      1785565         1785565

Enter volume number [1]: 1 <Accept default>


FTP default volume is 1:

-----------------------------------------------------------------
Step 6.  Enable TELNET

            Enable the Technician Interface via TELNET
            ------------------------------------------

Do you want to enable TI TELNET? (y/n)[n]: y <Enable telnet to Technician
Interface>

TI TELNET enabled.

-----------------------------------------------------------------

                      Configuration Summary
                      ---------------------

Link Module:                 ANSEDSG
Connector:                   1
Slot:                        1
Circuit Name:                S11
Encapsulation:               PPP
IP address:                  192.168.0.5
IP subnetwork mask:          255.255.255.252
Routing Protocol:            OSPF
TFTP Default Volume:         1:
FTP Default Volume:          1:
TI TELNET:                   Yes

Press [RETURN] to continue: <Press enter or return>

-----------------------------------------------------------------
Step 7. Specify a name for the configuration file.

                  Save configuration to a file.
                  -----------------------------
```

```
The Quick-Start configuration of the router is now complete and
active.
```

<Save the configuration to a temporary file.>

```
Do you wish to save this configuration to a file? (y/n)[y]: y

Default file name is startup.cfg on the current volume.

NOTE: Do *NOT* name this file ‘config’. Later, you may wish to
rename
      this file ‘config’ after you perform a named boot and verify
its
      operation.
```

<Enter a name for the temporary configuration file. It will be renamed later.>

```
Enter file name [startup.cfg]: temp.cfg
-----------------------------------------------------------------
Step 8.  Test this initial IP interface configuration.

                         TEST IP Interface
                         -----------------

IP Interface 192.168.0.5 is up.
```

<The ANH will automatically attempt to test the interface just configured.>

```
Testing local IP interface.

ping -IP 192.168.0.5 -r5
IP ping: 192.168.0.5 is alive (size = 16 bytes)
IP ping: 192.168.0.5 is alive (size = 16 bytes)
IP ping: 192.168.0.5 is alive (size = 16 bytes)
IP ping: 192.168.0.5 is alive (size = 16 bytes)
IP ping: 192.168.0.5 is alive (size = 16 bytes)
```

<After the interface is successfully tested, the ANH will attempt to ping the Site Manager workstation. The Remote Access network does not employ a Site Manager workstation, so do not perform the following test.>

```
This test attempts to ping the Site Manager workstation.

NOTE: If routing has not yet converged, an attempt
      to ping the Site Manager workstation may fail.  If
      this happens, you may either enter a new IP address or
      quit and wait a short period of time and try again from
      the TI command line.
```

```
Type q<return> to cancel this test.

Enter IP address of Site Manager workstation: q
```

<Do not perform this test>

```
Exiting...
      More Mode: ON
      Lines per screen: 24
```

<This completes the configuration of the first WAN port. The install.bat command must now be run again to configure the second WAN port.>

```
[1:1]$ run install.bat
      More Mode: OFF
      Lines per screen: 24
-----------------------------------------------------------------

 ####    #   #   #      #   # #### ##### #     #  ##  ###  #  #  ###
  #   #  # #   # #       ##  # #      #   #     # #  # #  # #  # #
 ####  #   #   #        # # # ###    #   #  #  # #  # ###  ##   ###
 #   # #####   #        #  ## #      #   # # # # #  # #  # # #     #
 ####  #   #   #        #   # ####   #    #   #   ##  #  # #  # ###

         ###  #  # ###  ## #  #      ### #####  #   ###  #####
        #   # #  #  #  #   # #      #      #   # #  #  #   #
        #   # #  #  #  #   ##   ### ###    #  #   # ###    #
        # # # #  #  #  #   # #        #    #  ##### #  #   #
         ###  #### ###  ## #  #     ###    #  #   # #  #   #
            #

                        Version 1.166
                     Copyright 1993-1996
-----------------------------------------------------------------

                         Introduction
                         ------------

This part of the Quick-Start procedure configures the initial IP
network interface on the router. You perform this procedure so
that
the router can communicate with the network management station.

Each step of this procedure is further described in the Quick-
Start Guide.
As you perform the procedure, refer to the Quick-Start Guide for
additional helpful information and examples.

When you are finished with this procedure, the router will be able
to
communicate with the network management station over the IP
network.  You
are then ready to install the network management software, as
described in the Quick-Start Guide.

Each procedure step requires you to do one of the following things:
      1. Enter a number that corresponds to a selection.
      2. Enter ‘y’ for Yes;  ‘n’ for No; ‘q’ for Quit.
      3. Enter a word or phrase referred to as a “text string”
      4. Enter <Return> to accept default displayed in [].

You must press the <Return> key after entering one of the above
responses.

Press <Return> to Continue, q<Return> to Quit: <Press enter or return key>
-----------------------------------------------------------------

             Preliminary Information You Need to Know
             ----------------------------------------

Before you begin this procedure, you should gather the network
information listed below:

You Need to Know This Information:               For Example:
----------------------------------               ------------
Type of Link Module connecting the router’s      DSDE
IP network interface to the Site Manager.

Slot number where the Link Module resides.       2

Communication type and connector number          Ethernet XCVR1

IP address of initial IP network interface       192.32.10.189

Subnet mask of initial IP network interface      255.255.255.0

IP address of Site Manager workstation           192.32.10.100

Do you wish to continue? (y/n)[y]: y <Continue with configuration>
-----------------------------------------------------------------

Step 1. Specify the slot number where the Link Module resides.

                    Slot Menu for Link Module
                    -------------------------

Slot     Link Module        Processor Module
----     -----------        ----------------
1        ANSEDSG            Access Node

Note: AN system, default is slot 1.
Slot 1 selected.<The ANH provides this information - no user input required>

-----------------------------------------------------------------
Step 2. Specify the Link Module and network interface information
for
       the initial IP connection to the Site Manager.

Link Module: ANSEDSG

Driver Type Menu
----------------
```

```
1. Ethernet
2. Synchronous

Enter driver type number [1]: 2 <Configure the WAN ports first>

Connector Menu
--------------
1. COM1
2. COM2

Enter connector number [1]: 2 <The second WAN port is selected>

Clock Source
------------
1. Internal
2. External

Enter clock source number [2]: 2 <Always select external clock source>

Recommended Circuit Name: S12

Enter circuit name [S12]: S12 <Select default value>
-----------------------------------------------------------------

Step 3.  Specify the IP configuration information for the network
      interface.

                      IP Configuration Menu
                      ---------------------

IP address format:###.###.###.###

IP subnetwork mask format: ###.###.###.###
      Example:             255.255.255.0
```

<Enter the appropriate IP address for the specific router. This address will be different for each router in the Remote Access network. IP addresses should be consistent with the recommendations in this document.>

```
Enter IP address in dotted decimal notation: 192.168.0.18
```

<If the recommendations of this document are followed, the following netmask should be used for all router WAN ports in the Remote Access network.>

```
Enter IP subnetwork mask in dotted decimal notation:
255.255.255.252
```

<The Remote Access network does not employ a Site Manager workstation.>

```
Is the router connected to the same local area network as
the Site Manager workstation? (y/n)[n]: n

Since the router is not on the same network as the Site
```

```
Manager workstation an IP Routing Protocol must be
configured in order to manage the box remotely

              IP Routing Protocol Configuration Menu
              --------------------------------------
      1. RIP
      2. OSPF
      3. Static Route to Site Manager.

Enter Routing Protocol Number [1]: 2 <Select OSPF as the IP routing protocol>

-----------------------------------------------------------------

                     OSPF Configuration Menu
                     ------------------------

OSPF Router ID
--------------

The Router ID uniquely identifies this router in the OSPF
domain. By convention, and to ensure uniqueness, one of the
router’s IP interface addresses should be used as the Router
ID.

The Router ID will determine the Designated Router on a
broadcast link if the priority values of the routers being
considered are equal. The higher the Router ID, the greater
priority.
```

<Accept the default parameter provided by the ANH. It must be the same as the IP address entered for the WAN port in Step 3 above.>

```
Enter OSPF router ID in dotted decimal notation [192.168.0.18]:
192.168.0.18

OSPF Area ID
------------

Next configure the OSPF Area ID. Remember that it must match the
Area ID of this router’s neighbor.

Note: The backbone area ID is always 0.0.0.0.
```

<Accept the default parameter provided by the ANH. Do not change this value.>

```
Enter the OSPF area ID in dotted decimal notation [0.0.0.0]:
0.0.0.0

OSPF Authentication Type
------------------------

Enable or disable password authentication for the area.  If
you select Simple Password (enabling password authentication),
only those routers sharing the correct password will be able to
```

```
communicate with each other. If you accept the default None,
password authentication is disabled for this area.
```

<The Remote Access network does not use password authentication.>

```
Enable Simple Password authentication? (y/n)[n]: n
Default Route For Unknown Subnets
---------------------------------

The default route will not apply for subnets unless
default route for unknown subnets is enabled.
```

<Enable default routes for unknown subnets.>

```
Follow default paths for unknown subnets? (y/n)[n]: y


OSPF MTU Configuration
----------------------

Select the MTU size for OSPF packets sent out this interface

      1. Default
      2. Ethernet-size (Bay Networks 5-series compatible)
      3. User Defined MTU

Enter the OSPF MTU size selection [1]: 2 <Select ethernet size MTU>


OSPF Interface Type Menu
------------------------

Select the interface’s type (the type of network to
which it is attached). Set this parameter to Broadcast if
this network is a broadcast LAN, such as Ethernet. Set it
to NBMA for an X.25 or similar type of interface. Set it
to point-to-point for a synchronous, point-to-point
interface.  OR, set it to point-to-multipoint for a star
Frame Relay topology

      1. Broadcast
      2. NBMA
      3. Point to Point
      4. Point to MultiPoint (Proprietary)
      5. Point to MultiPoint (Per OSPF Standard)

Enter OSPF interface type selection [1]: 3 <The WAN port is a Point to Point
Network>

OSPF Hello Interval
-------------------

The Hello Interval Indicates the number of seconds between the
Hello Packets that the router sends on the interface.  Set this
```

```
to the value that the other router(s) on the network are using.

Enter decimal value in seconds for Hello Interval [10]: 10 <Accept
default>

OSPF Router Dead Interval
-------------------------

The Dead Interval Indicates the number of seconds that a
router’s Hello packets have not been seen before it’s neighbors
declare the router down. Set this to the value that the other
router(s) on the network are using.

Enter decimal value in seconds for Router Dead Interval [40]: 40
<Accept default>

OSPF Router Priority
--------------------

Select the priority of this interface. The Router Priority value
is
used in multi-access networks (Broadcast, NBMA, or Point-to-
multipoint),
for the election of the designated router. If this parameter is set
to 0, this router is not eligible to become the designated router
on
this particular network.

In the case of equal Router Priority values, the router ID will
determine which router will become designated router. However, if
there already is a designated router on the network when you boot
up, it will remain the designated router no matter what your
priority or router ID.

Enter decimal value for Router Priority [1]: 1 <Accept default>

-----------------------------------------------------------------

                    OSPF Configuration Summary
                   ---------------------------

1. Router ID                              192.168.0.18
2. Area ID                                0.0.0.0
3. Stub Area                              No
4. Authentication                         No
5. Configured Password                    “”
6. OSPF MTU Size                          Ethernet-size
7. Interface Type                         Point-to-Point
8. Hello Interval                         10
9. Router Dead Interval                   40
10. Router Priority                       1
11. Poll Interval                         ------
12. Configured Neighbors                  ------
```

```
Are the values specified correct? (y/n)[y]: y <Accept if information is
correct>
OSPF Configuration Complete

-----------------------------------------------------------------

                      Wide Area Protocol Menu
                      -----------------------

1. Bay Networks Point-to-Point Protocol (Proprietary).
2. Frame Relay
3. Point-to-Point Protocol Standard (PPP)
4. Switched Multimegabit Data Service (SMDS)

Enter wide area protocol number [1]: 3 <Select PPP as WAN protocol>

-----------------------------------------------------------------

                      PPP Line Configuration
                      ----------------------

PPP Echo Configuration
----------------------

The PPP Echo configuration sets the parameters governing
Echo-Request packet transmission rate and Echo-Reply packet
acceptable loss threshold.

Do you wish to turn on the PPP echo function? (y/n)[n]: y <Enable
echo>

Echo Request Timer
------------------

Select the number of seconds that the router waits between the
transmission of Echo-Request packets.

Number of seconds (1-100) [3]: 10 <Provide enough time for slow networks>

Echo Reply Loss Threshold
-------------------------

Select the number (1-100) for Echo-Reply acceptable loss. This is
the
number of unacknowledged Echo-Reply packets counted before
declaring
the link down.

Enter echo loss threshold (1-100) [3]: 10 <Provide enough tolerance for slow
networks>

Local Authentication Protocol
-----------------------------
```

```
Bay Networks supports PAP (Password Authentication Protocol)
or CHAP (Challenge Handshake Authentication Protocol).
This feature is disabled if neither PAP or CHAP is chosen.

      <Passwords are not used in the Remote Access network.>

Enable PAP (Password Authentication Protocol) (y/n)[n]: n
Enable CHAP (Challenge Handshake Authentication Protocol) (y/
n)[n]: n

Remote Peer Authentication Protocol
-----------------------------------

Does the Remote Peer have PAP authentication enabled? (y/n)[n]: n

Link Quality Reporting Protocol
-------------------------------

Specify whether or not to enable the Link Quality
Reporting (LQR) Protocol.

NOTE:  Link Quality Monitoring on a Bay Networks 5-series router is
not
      compatible with this feature since the 5-series LQ functions
       are based on older, non-compatible RFCs.


Enable the LQR Protocol? (y/n)[n]: n <LQR is not used>

-----------------------------------------------------------------

                    PPP Configuration Summary
                    -------------------------

1. Echo Protocol Enabled:                Yes
Time between Xmit of Echo-Requests:      10
Echo-Reply Acceptable Loss:              10
2. Local Authentication:                 Disabled
3. Remote Authentication:                Disabled
4. Link Quality Monitoring:              No

Are these PPP parameters correct? (y/n)[y]: y <Accept if information is
correct>
PPP Configuration Complete
-----------------------------------------------------------------

                  SNMP Community Management Menu
                  ------------------------------

Setting up SNMP community management is optional.

It allows you to limit control of this router to a single
Site Manager workstation at a given IP address.  The default
is to allow any Site Manager from any workstation to manage
```

```
and to configure the router.

Note: You can later configure this using Site Manager.

      <SNMP is not currently used in the Remote Access network.>

Do you wish to set SNMP community management? (y/n)[n]: n
-----------------------------------------------------------------
Step 4.  Select TFTP default volume.

                     TFTP Default Volume Menu
                     ------------------------

NVFS File System:

VOL    STATE         TOTAL SIZE   FREE SPACE      CONTIG    FREE
SPACE
-----------------------------------------------------------------
 1:    FORMATTED    4194304      1785565         1785565

Enter volume number [1]: 1 <Accept default value>


TFTP default volume is 1:

-----------------------------------------------------------------
Step 5.  Select FTP default volume.

                             FTP Menu
                             --------

Do you want to enable FTP? (y/n)[n]: y <Enable FTP>

NVFS File System:

VOL    STATE         TOTAL SIZE   FREE SPACE      CONTIG    FREE
SPACE
-----------------------------------------------------------------
 1:    FORMATTED    4194304      1785565         1785565

Enter volume number [1]: 1 <Accept default>


FTP default volume is 1:

-----------------------------------------------------------------
Step 6.  Enable TELNET

            Enable the Technician Interface via TELNET
            ------------------------------------------

Do you want to enable TI TELNET? (y/n)[n]: y <Enable telnet to Technician
Interface>
```

```
TI TELNET enabled.

-----------------------------------------------------------------

                      Configuration Summary
                      ---------------------

Link Module:                  ANSEDSG
Connector:                    2
Slot:                         1
Circuit Name:                 S12
Encapsulation:                PPP
IP address:                   192.168.0.18
IP subnetwork mask:           255.255.255.252
Routing Protocol:             OSPF
TFTP Default Volume:          1:
FTP Default Volume:           1:
TI TELNET:                    Yes

Press [RETURN] to continue: <Press enter or return>

-----------------------------------------------------------------
Step 7. Specify a name for the configuration file.

                  Save configuration to a file.
                  -----------------------------

The Quick-Start configuration of the router is now complete and
active.

      <Save the configuration to a temporary file.>

Do you wish to save this configuration to a file? (y/n)[y]: y

Default file name is startup.cfg on the current volume.

NOTE: Do *NOT* name this file ‘config’. Later, you may wish to
rename
      this file ‘config’ after you perform a named boot and verify
its
      operation.

      <Enter a name for the temporary configuration file. It will be renamed later.>

Enter file name [startup.cfg]: temp.cfg
-----------------------------------------------------------------
Step 8.  Test this initial IP interface configuration.

                        TEST IP Interface
                        -----------------

IP Interface 192.168.0.18 is up.
```

<The ANH will automatically attempt to test the interface just configured.>

```
Testing local IP interface.

ping -IP 192.168.0.18 -r5
IP ping: 192.168.0.18 is alive (size = 16 bytes)
IP ping: 192.168.0.18 is alive (size = 16 bytes)
IP ping: 192.168.0.18 is alive (size = 16 bytes)
IP ping: 192.168.0.18 is alive (size = 16 bytes)
IP ping: 192.168.0.18 is alive (size = 16 bytes)
```

<After the interface is successfully tested, the ANH will attempt to ping the Site Manager workstation. The Remote Access network does not employ a Site Manager workstation, so do not perform the following test.>

```
This test attempts to ping the Site Manager workstation.

NOTE: If routing has not yet converged, an attempt
      to ping the Site Manager workstation may fail.  If
      this happens, you may either enter a new IP address or
      quit and wait a short period of time and try again from
      the TI command line.

Type q<return> to cancel this test.

Enter IP address of Site Manager workstation: q <Do not perform this test>

Exiting...
      More Mode: ON
      Lines per screen: 24
```

<This completes the configuration of the second WAN port. Both WAN ports should now be properly configured. The install.bat command must now be run for a third time to configure the ethernet hub port.>

```
[1:1]$ run install.bat
      More Mode: OFF
      Lines per screen: 24
-----------------------------------------------------------------

 ####    #   #   #      #   # #### ##### #     #  ##  ###  #  #  ###
 #   #  # #   # #       ##  # #      #   #     # #  # #  # # #  #
 ####  #   #   #        # # # ###    #   #  #  # #  # ###  ##   ###
 #   # #####   #        #  ## #      #   # # # # #  # #  # # #     #
 ####  #   #   #        #   # ####   #    #   #   ##  #  # #  # ###

         ###  #  # ###  ## #  #      ### #####   #   ###  #####
        #   # #  #  #  #   # #      #      #    # #  #  #   #
        #   # #  #  #  #   ##   ### ###    #   #   # ###    #
        # # # #  #  #  #   # #         #   #   ##### #  #   #
         ###  #### ###  ## #  #     ###    #   #   # #  #   #
            #
```

```
                           Version 1.166
                        Copyright 1993-1996
-----------------------------------------------------------------

                           Introduction
                           ------------

This part of the Quick-Start procedure configures the initial IP
network interface on the router.  You perform this procedure so
that
the router can communicate with the network management station

Each step of this procedure is further described in the Quick-
Start Guide.
As you perform the procedure, refer to the Quick-Start Guide for
additional helpful information and examples.

When you are finished with this procedure, the router will be able
to
communicate with the network management station over the IP
network.  You
are then ready to install the network management software, as
described in the Quick-Start Guide.

Each procedure step requires you to do one of the following things:
      1. Enter a number that corresponds to a selection.
      2. Enter ‘y’ for Yes;  ‘n’ for No; ‘q’ for Quit.
      3. Enter a word or phrase referred to as a “text string”
      4. Enter <Return> to accept default displayed in [].

You must press the <Return> key after entering one of the above
responses.

Press <Return> to Continue, q<Return> to Quit: <Press enter or return key>
-----------------------------------------------------------------

             Preliminary Information You Need to Know
             ----------------------------------------

Before you begin this procedure, you should gather the network
information listed below:

You Need to Know This Information:               For Example:
----------------------------------               ------------
Type of Link Module connecting the router’s      DSDE
IP network interface to the Site Manager.

Slot number where the Link Module resides.       2

Communication type and connector number          Ethernet XCVR1

IP address of initial IP network interface       192.32.10.189
```

```
Subnet mask of initial IP network interface      255.255.255.0

IP address of Site Manager workstation           192.32.10.100

Do you wish to continue? (y/n)[y]: y <Continue with configuration>
-----------------------------------------------------------------

Step 1.  Specify the slot number where the Link Module resides.

                     Slot Menu for Link Module
                     -------------------------

Slot    Link Module        Processor Module
----    -----------        ----------------
1       ANSEDSG            Access Node

Note: AN system, default is slot 1.
Slot 1 selected. <The ANH provides this information - no user input required>

-----------------------------------------------------------------
Step 2. Specify the Link Module and network interface information
for
      the initial IP connection to the Site Manager.

Link Module: ANSEDSG

Driver Type Menu
----------------
1. Ethernet
2. Synchronous

Enter driver type number [1]: 1 <Configure the ethernet port>

Recommended Circuit Name: E11

Enter circuit name [E11]: E11 <Select default value>
-----------------------------------------------------------------

Step 3. Specify the IP configuration information for the network
      interface.

                       IP Configuration Menu
                       ---------------------

IP address format: ###.###.###.###

IP subnetwork mask format: ###.###.###.###
      Example:           255.255.255.0
```

<Enter the appropriate IP address for the specific router. This address will be different for each router in the Remote Access network. IP addresses should be consistent with the recommendations in this document.>

```
Enter IP address in dotted decimal notation: 192.168.1.2
```

<If the recommendations of this document are followed, the following netmask should be used for all router ethernet ports in the Remote Access network.>

```
Enter IP subnetwork mask in dotted decimal notation:
255.255.255.192

You have entered a zero subnet.
Would you like to enable zero subnetting? (y/n)[n]: y <Enable zero
subnetting>
```

<The Remote Access network does not employ a Site Manager workstation.>

```
Is the router connected to the same local area network as
the Site Manager workstation? (y/n)[n]: n

Since the router is not on the same network as the Site
Manager workstation an IP Routing Protocol must be
configured in order to manage the box remotely

              IP Routing Protocol Configuration Menu
              --------------------------------------
      1. RIP
      2. OSPF
      3. Static Route to Site Manager.

Enter Routing Protocol Number [1]: 2 <Select OSPF as the IP routing protocol>

-----------------------------------------------------------------

                     OSPF Configuration Menu
                     ------------------------


OSPF Router ID
--------------

The Router ID uniquely identifies this router in the OSPF
domain. By convention, and to ensure uniqueness, one of the
router’s IP interface addresses should be used as the Router
ID.

The Router ID will determine the Designated Router on a
broadcast link if the priority values of the routers being
considered are equal. The higher the Router ID, the greater
priority.
```

<Accept the default parameter provided by the ANH. It must be the same as the IP address entered for the ethernet port in Step 3 above.>

```
Enter OSPF router ID in dotted decimal notation [192.168.1.2]:
192.168.1.2
```

```
OSPF Area ID
------------

Next configure the OSPF Area ID. Remember that it must match the
Area ID of this router’s neighbor.

Note: The backbone area ID is always 0.0.0.0.
```

<Accept the default parameter provided by the ANH. Do not change this value.>

```
Enter the OSPF area ID in dotted decimal notation [0.0.0.0]:
0.0.0.0


OSPF Authentication Type
------------------------

Enable or disable password authentication for the area.  If
you select Simple Password (enabling password authentication),
only those routers sharing the correct password will be able to
communicate with each other. If you accept the default None,
password authentication is disabled for this area.
```

<The Remote Access network does not use password authentication.>

```
Enable Simple Password authentication? (y/n)[n]: n

Default Route For Unknown Subnets
---------------------------------

The default route will not apply for subnets unless
default route for unknown subnets is enabled.
```

<Enable default routes for unknown subnets.>

```
Follow default paths for unknown subnets? (y/n)[n]: y


OSPF MTU Configuration
----------------------

Select the MTU size for OSPF packets sent out this interface

      1. Default
      2. Ethernet-size (Bay Networks 5-series compatible)
      3. User Defined MTU

Enter the OSPF MTU size selection [1]: 2 <Select ethernet size MTU>


OSPF Interface Type Menu
------------------------
```

```
Select the interface’s type (the type of network to
which it is attached). Set this parameter to Broadcast if
this network is a broadcast LAN, such as Ethernet. Set it
to NBMA for an X.25 or similar type of interface. Set it
to point-to-point for a synchronous, point-to-point
interface.  OR, set it to point-to-multipoint for a star
Frame Relay topology

      1. Broadcast
      2. NBMA
      3. Point to Point
      4. Point to MultiPoint (Proprietary)
      5. Point to MultiPoint (Per OSPF Standard)

Enter OSPF interface type selection [1]: 1 <The ethernet port is a broadcast
network>

OSPF Hello Interval
-------------------

The Hello Interval Indicates the number of seconds between the
Hello Packets that the router sends on the interface.  Set this
to the value that the other router(s) on the network are using.

Enter decimal value in seconds for Hello Interval [10]: 10 <Accept
default>

OSPF Router Dead Interval
-------------------------

The Dead Interval Indicates the number of seconds that a
router’s Hello packets have not been seen before it’s neighbors
declare the router down. Set this to the value that the other
router(s) on the network are using.

Enter decimal value in seconds for Router Dead Interval [40]: 40
<Accept default>

OSPF Router Priority
--------------------

Select the priority of this interface. The Router Priority value
is
used in multi-access networks (Broadcast, NBMA, or Point-to-
multipoint),
for the election of the designated router. If this parameter is set
to 0, this router is not eligible to become the designated router
on
this particular network.

In the case of equal Router Priority values, the router ID will
determine which router will become designated router. However, if
there already is a designated router on the network when you boot
up, it will remain the designated router no matter what your
```

```
priority or router ID.

Enter decimal value for Router Priority [1]: 1 <Accept default>

-----------------------------------------------------------------

                    OSPF Configuration Summary
                   ---------------------------

1. Router ID                              192.168.1.2
2. Area ID                                0.0.0.0
3. Stub Area                              No
4. Authentication                         No
5. Configured Password                    ""
6. OSPF MTU Size                          Ethernet-size
7. Interface Type                         Broadcast
8. Hello Interval                         10
9. Router Dead Interval                   40
10. Router Priority                       1
11. Poll Interval                         ------
12. Configured Neighbors                  ------

Are the values specified correct? (y/n)[y]: y <Accept if information is
correct>
OSPF Configuration Complete

-----------------------------------------------------------------

                  SNMP Community Management Menu
                  ------------------------------

Setting up SNMP community management is optional.

It allows you to limit control of this router to a single
Site Manager workstation at a given IP address.  The default
is to allow any Site Manager from any workstation to manage
and to configure the router.

Note: You can later configure this using Site Manager.

      <SNMP is not currently used in the Remote Access network.>

Do you wish to set SNMP community management? (y/n)[n]: n
-----------------------------------------------------------------
Step 4. Select TFTP default volume.

                     TFTP Default Volume Menu
                     ------------------------

NVFS File System:

VOL    STATE        TOTAL SIZE   FREE SPACE      CONTIG   FREE
SPACE
-----------------------------------------------------------------
```

```
 1:    FORMATTED    4194304      1782843          1782843

Enter volume number [1]: 1 <Accept default value>


TFTP default volume is 1:

-----------------------------------------------------------------
Step 5. Select FTP default volume.

                             FTP Menu
                             --------

Do you want to enable FTP? (y/n)[n]: y <Enable FTP>

NVFS File System:

VOL    STATE        TOTAL SIZE   FREE SPACE       CONTIG   FREE
SPACE
-----------------------------------------------------------------
 1:    FORMATTED    4194304      1782843          1782843

Enter volume number [1]: 1 <Accept default>


FTP default volume is 1:

-----------------------------------------------------------------
Step 6. Enable TELNET

            Enable the Technician Interface via TELNET
            ------------------------------------------

Do you want to enable TI TELNET? (y/n)[n]: y <Enable telnet to Technician
Interface>

TI TELNET enabled.

-----------------------------------------------------------------
                      Configuration Summary
                      ---------------------

Link Module:                  ANSEDSG
Connector:                    1
Slot:                         1
Circuit Name:                 E11
IP address:                   192.168.1.2
IP subnetwork mask:           255.255.255.192
Routing Protocol:             OSPF
TFTP Default Volume:          1:
FTP Default Volume:           1:
TI TELNET:                    Yes

Press [RETURN] to continue: <Press enter or return>
```

```
-----------------------------------------------------------------
Step 7. Specify a name for the configuration file.

                  Save configuration to a file.
                  -----------------------------

The Quick-Start configuration of the router is now complete and
active.
```

<Save the configuration to a temporary file.>

```
Do you wish to save this configuration to a file? (y/n)[y]: y

Default file name is startup.cfg on the current volume.

NOTE: Do *NOT* name this file ‘config’. Later, you may wish to
rename
      this file ‘config’ after you perform a named boot and verify
its
      operation.
```

<Enter a name for the temporary configuration file. It will be renamed later.>

```
Enter file name [startup.cfg]: temp.cfg
-----------------------------------------------------------------
Step 8. Test this initial IP interface configuration.

                        TEST IP Interface
                        -----------------

IP Interface 192.168.1.2 is up.
```

<The ANH will automatically attempt to test the interface just configured.>

```
Testing local IP interface.

ping -IP 192.168.1.2 -r5
IP ping: 192.168.1.2 is alive (size = 16 bytes)
IP ping: 192.168.1.2 is alive (size = 16 bytes)
IP ping: 192.168.1.2 is alive (size = 16 bytes)
IP ping: 192.168.1.2 is alive (size = 16 bytes)
IP ping: 192.168.1.2 is alive (size = 16 bytes)
```

<After the interface is successfully tested, the ANH will attempt to ping the Site Manager workstation. The Remote Access network does not employ a Site Manager workstation, so do not perform the following test.>

```
This test attempts to ping the Site Manager workstation.
```

```
NOTE: If routing has not yet converged, an attempt
      to ping the Site Manager workstation may fail.  If
      this happens, you may either enter a new IP address or
      quit and wait a short period of time and try again from
      the TI command line.

Type q<return> to cancel this test.

Enter IP address of Site Manager workstation: q <Do not perform this test>

Exiting...
      More Mode: ON
      Lines per screen: 24
```

<This completes the configuration of the ethernet hub port. Both WAN ports and the ethernet hub port should now be properly configured.>

<The correct configuration of the router can now be confirmed by querying some of the router configuration parameters>

<Query the the base IP configuration by entering the following command.>

```
[1:1]$ show ip base
Protocol:                      IP
State:                         Up
Forwarding Mode:               Enabled
Zero/All Ones Subnetting:      Enabled
Default TTL:                   30

RIP Diameter:                  15
Route Cache Size:              60
MIB Tables Maintained:         Route
Classless:                     Enabled
Route Filters:                 Enabled

Route pools contain 6 [est. 0] networks/subnets and 1 [est. 0]
hosts.
Maximum policy rules per type per protocol: 32
```

<Query correct configuration of the interfaces by entering the following command.>

```
[1:1]$ show ip circuits
Circuit Circuit #     State IP Address Mask
-------- ----------- -------- --------------- ----------------
S11          1          Up    192.168.0.5      255.255.255.252
S12          2          Up    192.168.0.18     255.255.255.252
E11          3          Up    192.168.1.2      255.255.255.192
3 circuit(s) found
```

<Query the routing table by entering the following command.>

```
[1:1]$ show ip routes
```

```
Destination Mask  Proto Age Cost NextHop Addr/AS
--------------- -------------- ----- -------- --------
192.168.0.4     255.255.255.252 LOCAL   75        0
192.168.0.5
192.168.0.16    255.255.255.252 LOCAL   75        0
192.168.0.18
192.168.1.0     255.255.255.192 LOCAL   75        0
192.168.1.2
192.168.2.0     255.255.255.192 OSPF    40        2
192.168.0.6
192.168.4.0     255.255.255.192 OSPF    40        2
192.168.0.17
5 route(s) found
```

<Confirm proper operation of the WAN ports.>

```
[1:1]$ ping 192.168.0.5
IP ping: 192.168.0.5 is alive (size = 16 bytes)

[1:1]$ ping 192.168.0.18
IP ping: 192.168.0.18 is alive (size = 16 bytes)
```

<Confirm proper operation of the ethernet hub port.>

```
[1:1]$ ping 192.168.1.2
IP ping: 192.168.1.2 is alive (size = 16 bytes)
```

<Confirm correct local routing by pinging the local workstation.>

```
[1:1]$ ping 192.168.1.1
IP ping: 192.168.1.1 is alive (size = 16 bytes)
```

<Confirm correct remote routing by pinging a remote workstation on each WAN port.>

```
[1:1]$ ping 192.168.2.1
IP ping: 192.168.2.1 is alive (size = 16 bytes)

[1:1]$ ping 192.168.4.1
IP ping: 192.168.4.1 is alive (size = 16 bytes)
```

<After confirmation of the correct configuration and operation of the router, rename the temporary configuration file so it will be executed whenever the ANH is reset or booted.>

```
[1:1]$ copy 1:temp.cfg 1:config <Copy the temporary file to the default configuration file>
[1:1]$ delete 1:temp.cfg <Delete the temporary configuration file>
```

<Quit this session and exit the ANH.>

```
[1:1]$ logout


Bay Networks, Inc. and its Licensors.
Copyright 1992, 1993, 1994, 1995, 1996. All rights reserved.
```

5.0 Diagnostics for Remote Access

5.1 Workstation

The ADAS/Remote Access workstation supports numerous commands and utilities to assist in isolating and debugging problems with the remote access network. Two classes of debugging facilities exist on the workstation--those provided by native unix commands, and extended diagnostics provided by special hpux operating system utilities.

5.1.1 Base Unix Commands

The unix operating system has inherent networking support that is very flexible and robust. Unix provides a number a native commands specifically oriented to the configuration and verification of the networking components. The commands of most importance and usefulness to the Remote Access feature are listed below:

1. ***arp;*** The ***arp*** command displays and permits modification of the internet-to-ethernet address translation tables. This command is useful for determining which nodes the host has been communicating with, and for extracting the station address of a node for use with the ***linkloop*** diagnostic.
2. ***ifconfig;*** The ***ifconfig*** command is used to display and configure the ethernet LAN interface(s) in an ADAS workstation. In addition to configuring the LAN interfaces, the ***ifconfig*** command is also useful for displaying the current hardware/software status and ethernet configuration details of a LAN interface.
3. ***ioscan;*** The ***ioscan*** command displays the software drivers configured in the hpux kernel (the operating system) currently running on the workstation. This command is useful for confirming that the necessary software drivers required by the hardware peripherals are present in the current operating system.
4. ***lanconfig;*** The ***lanconfig*** command is used to configure and display the network interface protocol (IEEE 802.3 and/or ethernet) used by a LAN interface.
5. ***lanscan;*** The ***lanscan*** command displays the configuration details of the LAN interface(s) in an ADAS workstation. The ***lanscan*** command differs from the ***ifconfig*** command in that ***lanscan*** returns software related configuration details about a LAN interface and ***ifconfig*** returns ethernet related details about a LAN interface. The ***ifconfig*** command can also be used to modify the configuration of a LAN interface, whereas the ***lanscan*** command can only display information about a LAN interface.
6. ***netstat;*** The ***netstat*** command provides statistics about various network related data structures associated with the LAN interface(s). Such information includes data packets received and transmitted by a LAN interface and the contents of the routing tables in the workstation.

7. ***ping;*** The ***ping*** command is a fundamental, low-level command very useful for testing the network connectivity between two nodes. ***Ping*** sends *echo_request* packets to the specified host and records the reception of the *echo_response* packets from the same host. The ***ping*** command is one of the first tools used to debug network connectivity problems in the ADAS Remote Access network.
8. ***route;*** The ***route*** command is used to add and delete routes from the routing tables in the ADAS workstation. The route entries tell the workstation where to send data packets for any/all nodes connected to the workstation via the LAN interface(s).

If a problem is encountered with the installation and/or configuration of the Remote Access components or network, the commands listed above should be the first tools employed in an attempt to identify and resolve the problems.

5.1.1.1arp

The ***arp*** (Address Resolution Protocol) command provides two important pieces of information (IP and MAC address details) about the network that can be very useful for resolving problems with the Remote Access network.

The man page entry for the ***arp*** command is listed below. Following this listing is an example of how the ***arp*** command can be used to help resolve network problems.
arp(1M) arp(1M)

NAME
arp - address resolution display and control

SYNOPSIS
arp hostname
arp -a [system] [core]
arp -d hostname
arp -s hostname address [temp] [pub] [trail] [rif rifAddress]
arp -f filename

DESCRIPTION
arp displays and modifies the Internet-to-Ethernet address translation tables used by the Address Resolution Protocol.

OPTIONS

none — If no options are specified (first form above), ***arp*** displays the current ARP entry for hostname. The hostname must either appear in the hostname database (see hosts(4)), or be a DARPA Internet address expressed in Internet standard “dot notation”.

-a — Display all current ARP entries by reading the table from file core (default /dev/kmem) based on the kernel file system (default /hp-ux).

-d — If an ARP entry exists for the host called hostname, delete it. This requires super-user privileges.

-s — Create an ARP entry for the host called hostname with the hardware station address. The hardware station address is given as six hexadecimal bytes separated by colons.

If an ARP entry already exists for hostname, the existing entry is updated with the new information. The entry is permanent unless the word temp is given in the command. If the word pub is specified, the entry is published, which means that this system will act as an ARP server responding to requests for hostname even though the host address is not its own. The word ***trail*** indicates that trailer encapsulations can be sent to this host. The word ***rif*** specifies source routing information used for token ring networks. This information allows a user to specify the particular bridge route which the token ring packet should be delivered. ***rifAddress*** is given as an even number of hexadecimal bytes separated by colons, up to a maximum of 16 bytes. This requires super-user privileges.

-f Read file filename and set multiple entries in the ARP tables. Entries in the file should be of the form:

hostname address [temp] [pub] [trail] [rif rifAddress]

Argument meanings are the same as for the -s option.

AUTHOR

arp was developed by the University of California, Berkeley.

WARNINGS

HP 9000 systems can receive trailer packets but do not send them. Setting the trailers flag has no effect.

SEE ALSO

ifconfig(1M), inet(3N), lanconfig(1M), arp(7P).

From the point of view of the Remote Access feature, the most useful form of the ***arp*** command will be the ***-a*** option. Issuing this command as follows:

/etc/arp -a

may provide a response similar to the following:

```
? (192.1.1.65) at 0:0:75:f0:73:16 ether
? (192.168.1.2) at 0:0:a2:ce:57:7 ether
```

The above response indicates the workstation has recently been communicating with two nodes via ethernet protocol encapsulation. The leading question marks are placeholders for the hostnames associated with the IP addresses. These two entries have no associated hostnames. The two important pieces of information provided by each entry are the IP address (enclosed in parentheses) and the MAC address (the number separated by colons).

The first node has an IP address of 192.1.1.65 and a MAC address of 0:0:75:f0:73:16. In this instance, this address would be that the EIU on the LAN0 interface. This is the LAN interface to the DMS components. In a properly installed ADAS system, there should only be one ***arp*** entry (the EIU) associated with the LAN0 interface.

The second node has an IP address of 192.168.1.2 and a MAC address of 0:0:a2:ce:57:7. In this instance, this address would be that of the external router/

hub providing access to the ADAS Remote Access network. Depending on the office configuration for Remote Access, there could be multiple ***arp*** entries for the LAN1 interface for Remote Access.

A null response to the ***arp*** command indicates no entries in the address translation tables and would indicate an inactive or unconnected LAN interface.

Other than providing an indication of which (if any) nodes the workstation has been communicating with and the associated IP addresses, the MAC addresses provided in the response to the ***arp*** command can be used by the ***linkloop*** command to assist in diagnosing LAN problems.

5.1.1.2ifconfig

The ***ifconfig*** command can be used as both in a passive mode to query the configuration of a LAN interface, or in an active mode to modify the configuration of a network interface. ***ifconfig*** is the primary command used to configure the LAN interface cards in the ADAS and Remote Workstations.

The man page entry for the ***ifconfig*** command is listed below. Following this listing is an example of how the ***ifconfig*** command can be used to help resolve network problems.
ifconfig(1M)ifconfig(1M)

NAME
ifconfig - configure network interface parameters

SYNOPSIS
ifconfig interface address_family [address [dest_address] [parameters]
ifconfig interface [address_family]

DESCRIPTION
ifconfig is used to assign an address to a network interface and/or configure network interface parameters. ***ifconfig*** must be used at boot time to define the network address of each interface present on a machine. It can also be used at other times to redefine an interface’s address or other operating parameters.

Command-Line Arguments

interface A string of the form name unit, such as ***lan0***. (See DEPENDENCIES.)

address_family Name of protocol on which naming scheme is based. An interface can receive transmissions in differing protocols, each of which may require separate naming schemes. Therefore, it is necessary to specify the ***address_family***, which may affect interpretation of the remaining parameters on the command line. The only ***address_family*** currently supported is ***inet*** (DARPA-Internet family).

address Either a host name present in the host name database (see hosts(4)), or a DARPA Internet address expressed in Internet standard “dot notation”. The host number can be omitted on 10-Mbit/second Ethernet interfaces (which use the hardware physical address), and on interfaces other than the first.

dest_address Address of destination system. Consists of either a host name present in the host name database (see hosts(4)), or a DARPA Internet address expressed in Internet standard “dot notation”.

parameters The following operating parameters can be specified:

up Mark an interface “up”. Enables interface after an “ifconfig down”. Occurs automatically when setting the address on an interface. Setting this flag has no effect if the hardware is “down”.

down Mark an interface “down”. When an interface is marked “down”, the system will not attempt to transmit messages through that interface. If possible, the interface will be reset to disable reception as well. This action does not automatically disable routes using the interface.

trailers Request the use of a “trailer” link-level encapsulation when sending. If a network interface supports trailers, the system will, when possible, encapsulate outgoing messages in a manner that minimizes the number of memory-to-memory copy operations performed by the receiver. On networks that support Address Resolution Protocol, this flag indicates that the system should request that other systems use trailers when sending to this host. Similarly, trailer encapsulations will be sent to other hosts that have made such requests. Currently used by Internet protocols only (see NOTES).

-trailers Disable the use of a “trailer” link-level encapsulation (default).

arp Enable the use of Address Resolution Protocol in mapping between network level addresses and link-level addresses (default). This is currently implemented for mapping between DARPA Internet addresses and 10-Mbit/second Ethernet addresses.

-arp Disable the use of Address Resolution Protocol.

metric n Set the routing metric of the interface to n, default 0. The routing metric is used by the routing protocol (see gated(1m)). Higher metrics have the effect of making a route less favorable; metrics are counted as additional hops to the destination network or host.

debug Enable driver-dependent debugging code. This usually turns on extra console error logging.

-debug Disable driver-dependent debugging code.

netmask mask (Inet only) Specify how much of the address to reserve for subdividing networks into sub-networks. ***mask*** includes the network part of the local address, and the subnet part which is taken from the host field of the address. ***mask*** can be specified as a single hexadecimal number with a leading 0x, with a dot-notation Internet address, or with a pseudo-network name listed in the network table (see networks(4)). ***mask*** contains 1’s for each bit position in the 32-bit address that are to be used for the network and subnet parts, and 0’s for the host part. ***mask*** should contain at least the standard network portion, and the subnet field should be contiguous with the network portion.

broadcast (Inet only) Specify the address that represents broadcasts to the network. The default broadcast address is the address with a host part of all 1’s.

ipdst (NS only) This is used to specify an Internet host that is willing to receive IP packets encapsulating NS packets bound for a remote network. In this case, an apparent point-to-point link is constructed, and the address specified is taken as the NS address and network of the destination.

The command:

ifconfig interface

with no optional command arguments supplied displays the current configuration for interface. If ***address_family*** is specified, ***ifconfig*** reports only the details specific to that address family. Only a user who has appropriate privileges can modify the configuration of a network interface.

DEPENDENCIES

The name of an interface associated with a LAN card is ***lan***, and its unit is determined as follows:

The LAN card in the lowest hardware module in the backplane is given interface unit number 0; the LAN card in the next higher hardware module is given interface unit number 1; and so on. When there are two or more LAN cards in a module (e.g. CIO), interface unit numbers are assigned to LAN cards in slot order before being assigned to cards in the next higher module. For example, consider a system with two LAN cards in CIO module 4 (slot 3 and slot 7) and one LAN card in CIO module 8 (slot 5). The three cards are assigned interface unit numbers 0, 1, and 2, respectively.

The ***lanscan*** command can be used to display the name and unit number of each interface that is associated with a LAN card (see lanscan(1M)).

NOTES

Currently, all HP 9000 systems can receive trailer packets but do not send them. Setting the trailers flag has no effect.

DIAGNOSTICS

Messages indicating that the specified interface does not exist, the requested address is unknown, or the user is not privileged and tried to alter an interface’s configuration.

SEE ALSO

netstat(1), lanconfig(1m), lanscan(1m) hosts(4), routing(7).

From the point of view of the Remote Access feature, the ***ifconfig*** command can be used in a couple of fashions. The passive form of the ***ifconfig*** command, entered without any additional parameters as follows:

/etc/ifconfig lan0

simply returns the configuration of the lan0 interface as follows:

```
lan0: flags=63<UP,BROADCAST,NOTRAILERS,RUNNING>
        inet 192.1.1.66 netmask ffffffc0 broadcast 192.1.1.127
```

Similarly, the configuration for the lan1 interface could be queried as follows:

/etc/ifconfig lan1

```
lan1: flags=63<UP,BROADCAST,NOTRAILERS,RUNNING>
        inet 192.168.1.1 netmask ffffffc0 broadcast 192.168.1.63
```

Looking at the first response, the following can be observed regarding ***lan0***:

- the interface is up (enabled)
- the hardware is running
- the protocol family is internet
- the IP address is 192.1.1.66
- the subnet mask is 0xffffffc0 or 255.255.255.192
- the broadcast address is 192.1.1.127

Similar observations can be made for the ***lan1*** interface.

Using the appropriate options, the ***ifconfig*** command can be used to manually configure or modify the network interfaces in a workstation. This form of the command appears as follows:

```
/etc/ifconfig lan1 inet 192.168.2.1 netmask 255.255.255.192 up
```

which configures the ***lan1*** interface as follows:

- selects the internet protocol family
- sets the IP address to 192.168.2.1
- sets the subnet mask to 255.255.255.192 or 0xffffffc0
- enables the interface

A response to the ***ifconfig*** command similar to the following:

```
ifconfig: no such interface
```

or

```
lan1: flags=62<BROADCAST,NOTRAILERS,RUNNING>
```

or any other response that is not similar to the responses shown previously, indicate a problem with the network interface. In these instances, connectivity to the ethernet network will not be possible.

5.1.1.3ioscan

The ***ioscan*** command provides both the software and hardware status of the hardware components in the workstation. In addition to hardware and software status information, ***ioscan*** also indicates the software drivers present in the current kernel operating system.

The man page entry for the ***ioscan*** command is listed below. Following this listing is an example of how the ***ioscan*** command can be used to help resolve network problems.

ioscan(1M) ioscan(1M)

NAME

ioscan - scan I/O system

SYNOPSIS

/etc/ioscan [-d driver|-C class] [-H hw_path] [-f] [-h]

DESCRIPTION

Depending on system hardware architecture, ***ioscan*** scans system hardware, usable I/O system devices, or kernel I/O system data structures as appropriate, and lists the results.

By default, ***ioscan*** scans the system, and lists all reportable hardware found. The types of hardware reported vary according to what series computer is in the system (see DEPENDENCIES) and can include processors, memory, interface cards and I/O devices. Entities that cannot be scanned are not listed.

Options

ioscan recognizes the following options:

-C class — Restrict the output listing to those devices belonging to the specified class. Cannot be used with -d.

-H hw_path — Restrict the scan and output listing to those devices connected at the specified hardware path.

-d driver — Restrict the output listing to those devices controlled by the specified driver. Cannot be used with -C.

-h — Return hardware hversion identifier information.

The -d and -C options can be used to obtain listings of subsets of the I/O system, but the entire system is still scanned. Specifying -H causes ***ioscan*** to restrict both the scan and the listing to the hardware subset indicated.

Output Format

The default output format for ***ioscan*** lists the class of each hardware module, the hardware path to the module, and the hardware status. If the -f option is used, ***ioscan*** produces a 'full' listing, giving the module's class, hardware path, module path, any logical unit, and hardware and software status values. Output fields are as follows:

class — Indicates the device class, such as disk, printer, graphics, lan, and tape_drive.

hw_path — A hardware path specifies the address of the hardware components leading to a device. It consists of a string of numbers separated by periods (.). Hardware components suffixed by (.) indicate the addresses of the hardware components on the path to the device.

module_path Lists the driver controlling a hardware component. The string ? indicates there is no driver available in the system to control that hardware component.

hardware_status

Entity identifier for the hardware component. It is one of the following strings:

ok(0xidy_value)
The hexadecimal value of the entity identifier.

Unrecognized_HW
There is hardware present but no entity identifier is available.

Driver_Won't_Probe
No identifier can be obtained because the parent driver does not support probing.

Probe_Failed
An attempt to probe the hardware failed for some reason.

hversion Hardware hversion identifier.

software_status
The state of the software driver controlling this hardware component. Consists of one of the following strings:

ok
The driver is bound.

Unbound
The driver is unbound because another driver has been bound in its place.

Too_Many_Devices
The driver is not bound because the maximum number of bound instances has been reached.

Out_of_Memory
The driver is not bound because the required system resources (such as dynamically allocated memory) could not be obtained.

HW/Driver_Mismatch
The driver is not bound because some other driver should be unbound and this one bound in its place, but the other driver cannot unbind.

No_Driver
There is no driver available in the system that matches this hardware component.

Driver_Failure
The attempt to bind the driver failed for some reason.

Not_Configured
The driver does not support dynamic configuration.

RETURN VALUE

ioscan returns 0 upon normal completion and 1 if an error occurred.

DIAGNOSTICS

Most of the diagnostic messages from ***ioscan*** are self-explanatory. Listed below are some messages deserving further clarification. Errors cause ***ioscan*** to halt immediately.

Errors

Device driver name is not in the kernel
Device class name is not in the kernel

The indicated device driver or device class is not present in the kernel. Add the appropriate device driver and/or device class to the uxgen(1M) input file and generate a new kernel.

Invalid module path - name1 cannot connect to name2

The indicated drivers cannot be adjacent in a module path. Driver connectivity is determined by the /etc/master file and the drivers “included’ in the uxgen input file (see uxgen(1M)

No such device in the system

No device in the system matched the options specified. Use a less specific set of options to list the devices in the system.

EXAMPLES

Scan the system hardware and list all the devices belonging to the disk device class.

ioscan -C disk

DEPENDENCIES

The only types of hardware reported are interface cards, disk devices and tape drives. ***ioscan*** scans the kernel data structures for interface cards, then probes for any disk or tape devices attached.

AUTHOR

ioscan was developed by HP.

FILES

/dev/config
/dev/*

SEE ALSO

lsdev(1M), ioconfig(4).

The ***ioscan*** command is useful to determine the hardware components installed in the workstation, and the software drivers bound into the kernel. With regards to Remote Access, the value of the ***ioscan*** command is to confirm that two lan interface cards are installed in the workstation, and both have the necessary software driver installed in the kernel. Obviously, the LAN interface hardware has to be installed to get access to the ADAS and Remote Access networks, but the appropriate software driver also has to be installed in the operating system (the kernel) in order for the applications to make use of the hardware. The ***ioscan*** command readily displays the status of the necessary hardware and software drivers.

The format of the ***ioscan*** command to determine the system configuration is as follows:

/etc/ioscan -fh

which provides a response similar to the following:

```
Class        H/W Path     Driver     H/W Status        hversion   S/W
Status
==========================================================
=====
graphics     1.0.0        graph3     ok(0x785)         0x0        ok
scsi         2.0.1        c700       ok(0x7082)        0x160      ok
tape_drive   2.0.1.3.0    scsitape   ok(0x1800202)     0x0        ok
disk         2.0.1.6.0    scsi       ok(0x202)         0x0        ok
lan          2.0.2        lan01      ok(0x708a)        0x160      ok
serial       2.0.4        asio0      ok(0x708c)        0x160      ok
parallel     2.0.6        parallel   ok(0x7074)        0x160      ok
audio        2.0.8        audio      ok(0x707b)        0x0        ok
ps2          2.0.11       ps2        ok(0x7084)        0x160      ok
ps2          2.0.12       ps2        ok(0x7084)        0x160      ok
lan          6.0.2        lan01      ok(0x708a)        0x180      ok
serial       6.0.4        asio0      ok(0x708c)        0x180      ok
```

From the above response to the ***ioscan*** command, it is evident that two lan interface cards are installed in the system (one at hardware location 2.0.2 and the other at hardware location 6.0.2) and that both have the necessary software driver (lan01) bound into the operating system kernel.

5.1.1.4lanscan

The ***lanscan*** command provides detailed information about the LAN network interface cards installed in the workstation. Additionally, ***lanscan*** can be used to quickly determine the state (up/down) of any LAN cards in the system to establish a go/no-go status for further diagnostic actions.

The man page entry for the ***lanscan*** command is listed below. Following this listing is an example of how the ***lanscan*** command can be used to help resolve network problems.
lanscan(1M) lanscan(1M)

NAME
lanscan - display LAN device configuration and status

SYNOPSIS
lanscan [system [core]]

DESCRIPTION
lanscan displays the following information about each LAN device that has software support on the system:

+ Series 700/800 Hardware Path or Series 300/400 Select Code.

+ Active Station Address (also known as Physical Address).

+ Device lu (logical unit).

+ Hardware State.

+ Network Interface “Name Unit” and State.

+ Network Management ID.

+ Encapsulation Methods configured for the Network Interface.

+ Major Number of the lan device file.
A -- implies that a major number does not apply to this LAN device.

The arguments system and core allow substitution for the default values /hp-ux and /dev/kmem.

WARNINGS

lanscan does not display information about LAN devices that do not have software support such as LAN interface cards that fail to bind properly at boot-up time.

AUTHOR

lanscan was developed by HP.

SEE ALSO

ifconfig(1M), lanconfig(1M).

From the point of view of the Remote Access feature, the ***lanscan*** command provides a concise indication of the status and configuration of all the network interface cards in a system. As illustrated below, the ***lanscan*** command

/etc/lanscan

provides the following information about the network interface cards in the workstation:

```
Hardware   Station          Dev  Hardware   Net-Interface      NM  Encapsulation
Mjr
Path       Address          lu   State      NameUnit State     ID  MethodsNum
2.0.2      0x080009835949   0    UP         lan0    UP         4   ETHER52
6.0.2      0x08000983594A   1    DOWN       lan1    DOWN       5   ETHER52
```

The above response provides the following information regarding the ***lan0*** network interface card:

- the hardware path is 2.0.2
- the MAC address is 0x080009835949
- the hardware state is UP
- the interface state is UP
- the protocol encapsulation is ethernet

and the following information regarding the ***lan1*** network interface card:

- the hardware path is 6.0.2

- the MAC address is 0x08000983594A
- the hardware state is DOWN
- the interface state is DOWN
- the protocol encapsulation is ethernet

From the above, it can be concluded that the ***lan0*** interface is up and running, but the ***lan1*** interface is not. Additional actions must be pursued to determine if the problem lies in the external LAN network or the network interface card itself.

5.1.1.5netstat

The ***netstat*** command is useful for interrogating the workstation's routing tables and LAN card usage. The ***netstat*** command is most useful for confirming the correct static routes required for DMS switch access and Remote Access network connectivity.

The man page entry for the ***netstat*** command is listed below. Following this listing is an example of how the ***netstat*** command can be used to help resolve network problems.

netstat(1) netstat(1)

NAME

netstat - show network status

SYNOPSIS

netstat [-Aan] [-f address_family] [system] [core]
netstat [-R] [system] [core]
netstat [-himnrs] [-f address_family] [system] [core]
netstat [-n] [-I interface] interval [system] [core]

DESCRIPTION

netstat symbolically displays the contents of various network-related data structures. Output format varies according to options selected. The ***netstat*** command takes one of the three forms shown above:

+ The first form of the command displays a list of active sockets for each protocol.

+ The second form presents the contents of one of the other network data structures according to the option selected.

+ The third form causes netstat to display updated packet traffic data on configured network interfaces. The display is updated at each interval.

Options are interpreted as follows:

-A Use the default display to show the address of any protocol control blocks associated with sockets. This option is used for debugging.

-R Lists all socket names in the socket registry for NetIPC applications. netstat -R returns only NetIPC information, not BSD IPC ("Berkeley Sockets") information.

-a Use the default display to show the state of all sockets. Normally sockets used by server processes are not shown.

-h Show the state of the IMP host table.

-i Show the state of auto-configured interfaces (interfaces statically configured into a system, but not located at boot time are not shown).

-I interface Show information about this interface only. This option is used with an interval as described below.

-m Show statistics recorded by memory management routines (the network manages a private pool of memory buffers).

-n Show network addresses as numbers (normally netstat interprets addresses and attempts to display them symbolically). This option can be used with any available display format.

-s Show per-protocol statistics.

-r Show the routing tables. If -s is also present, show routing statistics instead.

-f address_family Limit statistics or address control block reports to those of the specified address_family. The following address families are recognized: inet for AF_INET, and unix for AF_UNIX.

The arguments, system and core allow substitutes for the defaults /hp-ux and /dev/kmem.

The default display, for active sockets, shows the local and remote addresses, send and receive queue sizes (in bytes), protocol, and the internal state of the protocol. Address formats are of the form host.port or network.port if a socket's address specifies a network but no specific host address. When known, the host and network addresses are displayed symbolically using gethostbyname() and getnetbyname(), respectively (see gethostbyname(3N) and getnetbyname(3N)). If a symbolic name for an address is unknown, or if the -n option is specified, the address is displayed numerically according to the address family. For more information regarding the Internet "dot format", refer to inet(3N). Unspecified or "wildcard" addresses and ports appear as *.

The interface display provides a table of cumulative statistics regarding packets transferred, errors, and collisions. The network addresses of the interface and the maximum transmission unit (mtu) are also displayed.

The routing table display indicates the available routes and their status. Each route consists of a destination host or network and a gateway to use in forwarding packets. The flags field shows the state of the route (U if up), whether the route is to a gateway (G), and whether the route was created dynamically by a redirect (D). Direct routes are created for each interface attached to the local host. The gateway field for such entries shows the address of the outgoing interface. The refcnt field gives the current number of active uses of the route. Connection-oriented protocols normally hold on to a single route for the duration of a connection while connectionless protocols obtain a route while sending to the same destination. The use field provides a count of the number of packets sent using that route. The interface entry identifies which network interface was used for the route.

When ***netstat*** is invoked with an ***interval*** argument, it displays a running count of statistics related to network interfaces. This display consists of a column for the primary interface (the first interface found during autoconfiguration) and a column summarizing information for all interfaces. To replace the primary interface with another interface, use the -I option. The first line of each screen of information contains a summary since the system was last rebooted. Subsequent lines of output show values accumulated over the preceding interval.

AUTHOR

netstat was developed by the University of California, Berkeley.

SEE ALSO

hosts(4), networks(4), gethostbyname(3N), getnetbyname(3N), protocols(4), services(4).

The ***netstat*** command has two options that make it useful for diagnosing network problems with the Remote Access product. The first option provides information regarding the network interface cards in the workstation and has the following format:

/usr/bin/netstat -ni

and provides a response similar to the following:

```
Name    Mtu     Network     Address       Ipkts    Ierrs   Opkts
Oerrs   Coll
lo0     1500    127         127.0.0.1     58816    0       58816    00
lan0    1500    192.1.1     192.1.1.66    4621     0       3538     00
lan1    1500    192.168.1   192.168.1.1   578      0       345      00
```

which indicates two active LAN cards, plus the application loopback address. Activity on the LAN interface cards is indicated by non-zero fields in the ***Ipkts*** (input packets received) and ***Opkts*** (output packets transmitted) fields. This information implies both LAN interface cards are functioning properly.

Using the ***netstat*** command with the ***-r*** option provides routing table information that can be used to confirm that the route entries are correct. This form of the ***netstat*** command appears as follows:

/usr/bin/netstat -r

and provides a response similar to the following:

```
Routing tables
Destination     Gateway        Flags    Refs   Use      Interface
localhost       localhost      UH       0      25983    lo0
default         192.168.1.2    UG       4      3473     lan1
192.1.1.128     192.1.1.65     UG       2      5714     lan0
192.168.1       192.168.1.1    U        7      3        lan1
192.1.1.64      oam_ws         U        10     372      lan0
```

Presuming the ***lan0*** interface is connected to the ADAS LAN and the ***lan1*** interface is connected to the Remote Access LAN, the above response provides the following information:

- the IP address of the CM network is 192.168.1.128

- all data sent to the DMS switch (APUs and CM) are sent to the EIU at IP address 192.1.1.65 (the ***‘G’*** in the Flags field indicates this address is a ***gateway***)
- access to the ethernet LAN (network IP address of 192.1.1.64) between the workstation and the EIU is provided by the ***lan0*** interface of the workstation
- the ADAS workstation’s hostname is ***oam_ws***, and this entry is in the ***/etc/hosts*** table
- any data that is sent to an address that ***is not*** on the 192.1.1.128 network, will be sent out the ***lan1*** interface (this is indicated by the ***default*** keyword in the Destination field, which implies that unless a specific route exists in the routing tables for a given destination, the data will be sent to this address)
- all data sent out the ***lan1*** interface (IP address of 192.168.1.1) goes to the ***gateway*** at IP address 192.168.1.2, which is the Remote Access router/hub

Any time it is determined that a network interface card is up and running, but a specific node cannot be accessed, the ***netstat -r*** command should be issued to insure a route has been properly defined between the two nodes.

5.1.1.6ping

The ***ping*** command is perhaps the most useful tool for debugging LAN network connectivity issues. If the user knows the nodes in a system, ***ping*** can be used to confirm connectivity to each node in the network.

The man page entry for the ***ping*** command is listed below. Following this listing is an example of how the ***ping*** command can be used to help resolve network problems.
ping(1M) ping(1M)

NAME

ping - send ICMP ECHO_REQUEST packets to network hosts

SYNOPSIS

ping [-r] [-v] [-o] host [packetsize] [count]

DESCRIPTION

ping sends an ICMP echo (ECHO_REQUEST) packet to host once per second. Each packet that is echoed back (via an ECHO_RESPONSE packet) is reported on the screen, including round-trip time.

ECHO_REQUEST datagrams (“pings”) have an IP and ICMP header, followed by a struct timeval, and an arbitrary number of “pad” bytes used to fill out the packet. Default datagram length is 64 bytes, but this can be changed by using the command-line option.

Other options and parameters are:

-r Bypass the normal routing tables and send directly to a host on an attached network. If the host is not on a directly-attached network, an error is returned. This option can be used to ***ping*** a local host through an interface that has no route through it (such as after the interface was dropped by gated (see gated(1M)).

-v

Verbose output. ICMP packets other than ECHO_RESPONSE that are received are listed.

-o

Insert “record route” IP option in outgoing packets, summarizing routes taken when the program exits. It may not be possible to get the round-trip path if all hosts on the route taken do not implement the “record route” IP option. A maximum of nine Internet addresses can be displayed due to the maximum length of the IP option area.

host

host can be a hostname or an Internet address. All symbolic names specified for a host are looked up using gethostbyname() (see gethostbyname(3N)). If ***host*** is an Internet address, it must be in “dot” notation (see inet_addr(3N)).

packetsize

By default (when ***packetsize*** is not specified), the size of transmitted packets is 64 bytes. The minimum value allowed for ***packetsize*** is eight bytes, and the maximum is 4096 bytes. Also, if ***packetsize*** is smaller than 16 bytes, there is not enough room for timing information. In this case the round-trip times are not displayed.

count

The number of packets ping will transmit before terminating. Range: 1 to (2**31 -1), decimal. Default: ping sends packets until interrupted.

When using ***ping*** for fault isolation, it should first be run on the local host to verify that the local network interface is working correctly, then hosts and gateways further and further away should be pinged. ***ping*** sends one datagram per second, and prints one line of output for every ECHO_RESPONSE returned. No output is produced if there is no response. If an optional count is given, only the specified number of requests is sent. Round-trip times and packet loss statistics are computed. When all responses have been received or the program times out (with a count specified), or if the program is terminated with a SIGINT, a brief summary is displayed.

This program is intended for use in testing, managing, and measuring network performance. It should be used primarily for manual fault isolation. Because of the load it could impose on the network, it is considered discourteous to use ***ping*** unnecessarily during normal operations, or from automated scripts.

AUTHOR

ping was developed in the Public Domain.

FILES

/etc/hosts

SEE ALSO

gethostbyname(3N), rlb(1M), inet(3N).

With respect to Remote Access, the ***ping*** command is perhaps the most powerful command to diagnose network problems. The ***ping*** command provides a means to perform an end-to-end connectivity test between two nodes in a network, and provides positive confirmation of connectivity (or lack thereof). The ***ping*** command has numerous options--the most useful for Remote Access are illustrated below.

```
/etc/ping 192.168.1.2
```

will return the following:

```
PING 192.168.1.2: 64 byte packets
64 bytes from 192.168.1.2: icmp_seq=0. time=2. ms
```

```
64 bytes from 192.168.1.2: icmp_seq=1. time=1. ms
64 bytes from 192.168.1.2: icmp_seq=2. time=1. ms
64 bytes from 192.168.1.2: icmp_seq=2. time=0. ms

----192.168.1.2 PING Statistics----

4 packets transmitted, 4 packets received, 0% packet loss

round-trip (ms)  min/avg/max = 0/1/2
```

As issued above, the ping command will run forever until terminated by entering ***cntrl-c*** or by killing the process. In the case above, the ***ping*** command was terminated after four loops. Once ***ping*** terminates, the statistics are displayed four the number of packets sent, received, and lost.

The above command was obviously successful, indicating the network connectivity between the two nodes is okay. If no connectivity exists between the two nodes, the response will be similar to the following:

```
/etc/ping 192.168.1.2

PING 192.168.1.2: 64 byte packets

----192.168.1.2 PING Statistics----

4 packets transmitted, 0 packets received, 100% packet loss
```

This response obviously indicates the transmitted data packets were not received from the remote destination for some reason. Note, the command as issued had to be terminated via ***cntrl-c*** or by killing the process.

The ***ping*** command is very flexible, providing numerous options. A more advanced version of ***ping*** may appear as follows:

```
/etc/ping 192.168.1.2 512 3
```

which sends three packets of 512 bytes to 192.168.1.2 and stops. The response to the above command is as follows:

```
PING 192.168.1.2: 512 byte packets

512 bytes from 192.168.1.2: icmp_seq=0. time=3. ms

512 bytes from 192.168.1.2: icmp_seq=1. time=2. ms

512 bytes from 192.168.1.2: icmp_seq=2. time=2. ms

----192.168.1.2 PING Statistics----

3 packets transmitted, 3 packets received, 0% packet loss

round-trip (ms)  min/avg/max = 2/2/3
```

The general form of the ***ping*** command is as follows:

/etc/ping host <packet_size> <count>

where <***packet_size***> can be from 8 to 4096 bytes (but should be limited to 1024 bytes due to other network restrictions) and <***count***> can be from 1 to 2,147,483,647.

When troubleshooting an intermittent network problem, it is sometimes helpful to try the ***ping*** command with different packet sizes, as some noisy networks may successfully pass small packets (64 bytes) but not larger packets (>256 bytes).

5.1.1.7route

The ***route*** command is an active command only. It is used to manually add or delete static routes from the workstation's routing tables. These routing tables tell the workstation how to connect to any other node in the network.

The man page entry for the ***route*** command is listed below. Following this listing is an example of how the ***route*** command can be used to help resolve network problems.

route(1M) route(1M)

NAME

route - manually manipulate the routing tables

SYNOPSIS

```
/etc/route [-f] [-n] add [net | host] destination gateway [count]
/etc/route [-f] [-n] delete [net | host] destination gateway [count]
/etc/route -f [-n]
```

route is used to manipulate the network routing tables manually, and accessible only by users who have appropriate privileges. ***route*** supports two commands:

add	Add a route.
delete	delete a route.

When adding a route, if the route already exists, a message is printed and nothing changes.

Other command line arguments are:

net or host — specifies the type of destination address. If not specified, routes to a particular host are distinguished from those to a network by interpreting the Internet address associated with destination. If the destination has a "local address part" of INADDR_ANY(0), the route is assumed to be to a network; otherwise, it is treated as a route to a host.

destination — destination host system where the packets will be routed. ***destination*** can be either a host name (the official name or an alias, see gethostbyname(3N)), a network name (the official name or an alias, see getnetbyname(3N)), an Internet address in "dot" notation (see inet(3N)), or the keyword default, which signifies the wildcard gateway route (see routing(7)).

gateway The gateway through which the destination is reached. ***gateway*** can be either a host name (the official name or an alias, see gethostbyname(3N)), or an Internet address in “dot” notation.

count An integer that indicates whether the gateway is a remote host or the local host. If the route leads to a destination via a remote gateway, ***count*** should be a number greater than 0. If the route leads to destination and the gateway is the local host, ***count*** should be 0. The default for ***count*** is zero. The result is not defined if ***count*** is negative.

All symbolic names specified for a ***destination*** or ***gateway*** are looked up first as a hostname using gethostbyname(); if the hostname is not found, the destination is searched as a network name using getnetbyname(). ***destination*** and ***gateway*** can be in dot notation (see inet(3N)). If the -n option is not specified, any host and network addresses are displayed symbolically according to the name returned by gethostbyaddr() and getnetbyaddr(), respectively, except for the default network address (printed as default) and addresses that have unknown names. Addresses with unknown names are printed in Internet dot notation (see inet(3N) for more information regarding this format). If the -n option is specified, any host and network addresses are printed in Internet dot notation except for the default network address which is printed as default.

If the -f option is specified, ***route*** deletes all route table entries that specify a remote host for a ***gateway***. If this is used with one of the commands described above, the entries are deleted before the command’s application.

Output

add ***destination***: gateway ***gateway*** flags ***flags***

The specified route is being added to the tables.

delete ***destination***: gateway ***gateway*** flags ***flags***

The specified route is being deleted from the tables.

Flags

The following truth table can be used to help understand the relationship between count, destination type, flags, and route type.

Count	Destination Type	Flags	Route Type
=0	network	1=U	route to a network via a gateway which is the local host itself
>0	network	3=UG	route to a network via a gateway which is a remote host
=0	host	5=UH	route to a host via a gateway which is the local host itself
>0	host	7=UGH	route to a host via a gateway which is a remote host
=0	“default”	1=U	wildcard route via the local host
>0	“default”	3=UG	wildcard route via a remote gateway

DIAGNOSTICS

delete a route that does not exist

The specified route was not in the route table.

add a route that already exists

The specified entry is already in the route table.

add too many routes
The routing table is full.

WARNINGS

Reciprocal route commands must be executed on the local host, the destination host, and all intermediate hosts if routing is to succeed in the cases of virtual circuit connections or bidirectional datagram transfers.

DEPENDENCIES

The HP-UX implementation of ***route*** does not presently support a change command argument.

AUTHOR

route was developed by the University of California, Berkeley.

FILES

/etc/networks
/etc/hosts

SEE ALSO

netstat(1), ifconfig(1M), inet(3N), gethostbyname(3N), gethostbyaddr(3N), getnetbyname(3N), getnetbyaddr(3N), routing(7).

The ***route*** command is used to manually add or remove routes from the workstations routing tables. This may be required for debugging purposes, or because a route was not properly added by the automated install scripts. Note, however, changes made by the ***route*** command are not permanent. When the workstation is rebooted, changes made by the ***route*** command will be lost. Permanent route entries should be added by making the appropriate modifications in the ***/etc/netlinkrc*** file. Routes in the workstation’s routing tables inform the workstation how to send data to other nodes in the network. If the workstation does not know how to reach a node, it cannot communicate with that node. Examples of the ***route*** command more clearly illustrate what ***route*** does.

The route command to add a routing entry appears as follows:

/etc/route add default 192.168.1.2 1

The above command adds an external ***default gateway*** at IP address 192.168.1.2 to the workstation’s routing tables. The digit ***1*** at the end of the command adds the default route as a gateway instead of simply another route. The ***default*** entry in the workstation’s routing table has a special meaning. The ***default*** route is used for any/all data that is transmitted by the workstation to a node that does not have an explicit route defined in the routing tables. In other words, when transmitting a message, the workstation first looks for a explicit route to the remote node. If a route has been defined, the workstation will use it. If no route exists to the specified node, the workstation will use the route defined as ***default***, if it exists. The default route is therefore the fall-through, or catch-all.

The following command

/etc/route add net 192.1.1.128 192.1.1.65 1

adds an external ***gateway*** at IP address 192.1.1.65 for the network with an IP address of 192.1.1.128.

The command

/etc/route add 192.1.1.135 192.1.1.65

adds an entry to the routing tables that says, “to get to the ***host*** at IP address 192.1.1.135, use the ***host*** at IP address 192.1.1.65”.

5.1.2 Extended Unix Diagnostic Tools

The commands listed in the previous section represent some of the more useful, base unix commands supporting configuring and diagnosing problems with the LAN interface cards in the ADAS or Remote Workstations. These are basic operating system commands supported, in one form or another, by most all versions of unix.

For the ADAS and Remote Workstations, there are some extended LAN diagnostic utilities developed by Hewlett-Packard specifically for the HP-UX operating system. Targeted specifically for Hewlett-Packard products, these extended diagnostics provide additional resources for troubleshooting and isolating problems with the workstation LAN interface cards and Remote Access network.

Two extended diagnostic utilities useful for troubleshooting problems with the ADAS Remote Access network are

1. ***landiag;*** The ***landiag*** command permits the user to query, reset, and test the LAN interface cards in the ADAS or Remote Workstations. ***landiag*** is actually a utility, which when invoked, provides a menu driven interface permitting the user the ability to identify problems with the LAN interface cards. ***landiag*** is the only method for a user to test the LAN interface hardware itself.
2. ***linkloop;*** The ***linkloop*** command is similar to the ***ping*** command, in that linkloop also tests for connectivity between two nodes in the network. However, whereas ***ping*** uses ***ICMP echo-request packet***s to test continuity between two nodes, ***linkloop*** use ***IEEE 802.3 link-level test frames*** to test connectivity. Since ***linkloop*** is a vendor-specific diagnostic implementation, it will only be 100% successful when both nodes being tested are Hewlett-Packard products. However, even with this limitation, since ***linkloop*** is based on standard IP packet structures that all standards-based nodes support, it can still provide valuable information about the connectivity between two nodes.

5.1.2.1landiag

The ***landiag*** utility is the only method for the user to test the actual LAN interface hardware itself. This utility is vendor-specific to Hewlett-Packard hardware and operating systems only. ***landiag*** provides a method of determining a go/no-go status of the LAN interface cards in the ADAS and Remote Access Workstations.

The man page entry for the ***landiag*** command is listed below. Following this listing is an example of how the ***landiag*** command can be used to help resolve network problems.
landiag(1M) landiag(1M)

NAME
landiag - local area network diagnostic

SYNOPSIS
landiag [-e] [-t]

DESCRIPTION
landiag is a diagnostic program for testing the Local Area Network (LAN). It allows the user to examine and reset the status of the LAN interface card. LAN interface status includes the LAN interface card self-test completion code, the local network address, and various LAN interface card statistics. users with appropriate privileges can reset the local LAN interface card, causing it to execute its self-test.

landiag reads commands from the standard input, writes prompts and error messages to standard error, and writes status to the standard output. The Break key generates an interrupt of a currently executing command if initiated interactively. If commands are read from a file, Ctrl-| returns control to the shell.

landiag accepts either complete command words or unique abbreviations, and no distinction is made between uppercase and lowercase letters in commands. Multiple commands can be entered on one line if they are separated by spaces, tabs, or commas.

Options
landiag recognizes the following command-line options:

-e Echo the input commands on the output device.

-t Suppress the display of the command menu before each command prompt. This option functions identically to the Test Selection Mode terse command. The default for ***landiag*** is verbose.

Test Selection Mode
The available Test Selection Mode commands are:

lan Puts landiag in LAN Interface Test Mode.

menu Displays the Test Selection Mode command menu.

quit Terminates the ***landiag*** program.

terse Suppresses display of command menus.

verbose Restores default display of command menus.

LAN Interface Test Mode
The following commands are available:

clear Clears the LAN interface card network statistics registers to zero. Requires superuser privileges to execute.

display Displays the local LAN interface card status and statistics registers.

end Returns landiag to Test Selection Mode.

menu Displays the LAN Interface Test Mode command menu.

name Prompts for a device file name for the LAN interface card. Default is /dev/lan0.

quit Terminates the ***landiag*** program.

reset Resets the local LAN interface card which causes it to execute its self-test. Local access to the network is interrupted during execution of reset. Requires superuser privileges to execute.

AUTHOR

landiag was developed by HP.

FILES

/usr/bin/landiag

SEE ALSO

linkloop(1M), lanscan(1M) ping(1M), lan(7).

When attempting to isolate a fault with the Remote Access network, ***landiag*** can be used to confirm the correct operation of the network interface hardware in the workstation. This utility permits the user to query, reset, and test the LAN cards in the workstation, and either eliminate or identify the hardware as the point of failure.

The ***landiag*** utility can be invoked at any time, however, there are instances in which problems with the network external to the workstation can cause the ***landiag*** utility to function erroneously and incorrectly indicate a problem with the hardware. Therefore, unless the user is absolutely certain the local network is functional, ***landiag*** should only be executed with the workstation disconnected from the network and a LAN loopback connector installed in the MAU, and the ***Loopback Test*** switch on the MAU enabled. Please refer to ***Section 6.3.5 on page 195*** for complete details on performing loopback tests employing the ***landiag*** command and an external loopback connector.

The ***landiag*** utility is invoked in the following manner (***landiag*** must be executed by a user with ***root*** privileges):

/usr/bin landiag

which will provide the follow menu:

```
          LOCAL AREA NETWORK ONLINE DIAGNOSTIC, Version 1.0
                       Wed, Aug 27,1997 10:53:59

                 Copyright 1985 Hewlett Packard Company.
                         All rights are reserved.

Test Selection mode.

    lan          = LAN Interface Diagnostic
    menu         = Display this menu
    quit         = Terminate the Diagnostic
    terse        = Do not display command menu
    verbose      = Display command menu

Enter command:
```

Entering the ***lan*** selection will provide the following sub-menu:

```
Enter command: lan

LAN Interface test mode. LAN Interface device file = /dev/lan0

    clear        = Clear statistics registers
    display      = Display LAN Interface status and statistics registers
    end          = End LAN Interface Diagnostic, return to Test Selection
    menu         = Display this menu
    name         = Name of the LAN Interface device file
    quit         = Terminate the Diagnostic, return to shell
    reset        = Reset LAN Interface to execute its selftest

Enter command:
```

Selecting the ***clear*** command will clear the statistics registers and provide the following response:

```
Enter command: clear

Clearing LAN Interface statistics registers.

LAN Interface test mode. LAN Interface device file = /dev/lan0

    clear       = Clear statistics registers
    display     = Display LAN Interface status and statistics registers
    end         = End LAN Interface Diagnostic, return to Test Selection
    menu        = Display this menu
    name        = Name of the LAN Interface device file
    quit        = Terminate the Diagnostic, return to shell
    reset       = Reset LAN Interface to execute its selftest

Enter command:
```

To reset the LAN interface card and perform a self test, enter the ***reset*** selection. This selection results in the following response.

```
Enter command: reset

Resetting LAN Interface to run selftest.

LAN Interface test mode. LAN Interface device file = /dev/lan0

    clear       = Clear statistics registers
    display     = Display LAN Interface status and statistics registers
    end         = End LAN Interface Diagnostic, return to Test Selection
    menu        = Display this menu
    name        = Name of the LAN Interface device file
    quit        = Terminate the Diagnostic, return to shell
    reset       = Reset LAN Interface to execute its selftest

Enter command:
```

Note: The above response indicates a successful selftest, i.e., no hardware faults found. Any other response indicates a selftest failure. A failed selftest would result in a response similar to the following:

```
Enter command: reset

Unable to reset LAN Interface.
errno = 6

LAN Interface test mode. LAN Interface device file = /dev/lan0

    clear       = Clear statistics registers
    display     = Display LAN Interface status and statistics registers
    end         = End LAN Interface Diagnostic, return to Test Selection
    menu        = Display this menu
    name        = Name of the LAN Interface device file
    quit        = Terminate the Diagnostic, return to shell
    reset       = Reset LAN Interface to execute its selftest

Enter command:
```

To display the LAN interface status and statistics registers, select the ***display*** command, as follows:

```
Enter command: display


                    LAN INTERFACE STATUS DISPLAY
                     Wed, Aug 27,1997 10:53:59

    Device file                          = /dev/lan0
    Lu number                            = 0
    Current state                        = active
    LAN station address, hex             = 0x0800093525FD
    Number of multicast addresses        = 2
    Frames received                      = 31
    Frames transmitted                   = 15
    Undelivered received frames          = 0
    Untransmitted frames                 = 0
    CRC errors received                  = 0
    Transmit collisions                  = 0
    One transmit collision               = 0
    More transmit collisions             = 0
    Excess retries                       = 0
    Deferred transmissions               = 1
    Carrier lost when transmitting       = 0
    No heartbeat after transmission      = 0
    Frame alignment errors               = 0
    Late transmit collisions          = 0
    Frames lost                       = 0
    Unknown protocol                  = 0
    Bad control field                 = 0
    IEEE 802.3 XID packets            = 0
    IEEE 802.3 TEST packets           = 0
    Unable to respond TEST/XID pkts   = 0

LAN Interface test mode. LAN Interface device file = /dev/lan0

    clear      = Clear statistics registers
    display    = Display LAN Interface status and statistics registers
    end        = End LAN Interface Diagnostic, return to Test Selection
    menu       = Display this menu
    name       = Name of the LAN Interface device file
    quit       = Terminate the Diagnostic, return to shell
    reset      = Reset LAN Interface to execute its selftest

Enter command:
```

To test the second LAN interface, it must first be selected from the menu. This is accomplished via the ***name*** selection, as indicated below:

```
Enter command: name

Enter LAN Interface device file name. Currently /dev/lan0: /dev/lan1

LAN Interface test mode. LAN Interface device file = /dev/lan1

    clear      = Clear statistics registers
    display    = Display LAN Interface status and statistics registers
    end        = End LAN Interface Diagnostic, return to Test Selection
    menu       = Display this menu
    name       = Name of the LAN Interface device file
    quit       = Terminate the Diagnostic, return to shell
    reset      = Reset LAN Interface to execute its selftest

Enter command:
```

To terminate and exit the ***landiag*** utility, enter the ***quit*** command.

```
LAN Interface test mode. LAN Interface device file = /dev/lan1

    clear          = Clear statistics registers
    display        = Display LAN Interface status and statistics registers
    end            = End LAN Interface Diagnostic, return to Test Selection
    menu           = Display this menu
    name           = Name of the LAN Interface device file
    quit           = Terminate the Diagnostic, return to shell
    reset          = Reset LAN Interface to execute its selftest

Enter command: quit

Diagnostic terminated by operator.
```

5.1.2.2linkloop

The ***linkloop*** command is similar in affect as the ***ping*** command, in that ***linkloop*** also attempts to confirm connectivity between two nodes in the network. Whereas ***ping*** uses **echo-request** packets to confirm connectivity between two nodes anywhere in the network, ***linkloop*** uses **test frames** to confirm connectivity between any two local nodes on the same network segment. Therefore, the ***linkloop*** command is only useful for confirming connectivity between the ADAS or Remote Workstation and the local router/hub LAN equipment. Furthermore, since ***linkloop*** is a vendor-dependent implementation, it will only work fully when both nodes are Hewlett-Packard products. Even with these limitations, ***linkloop*** can still be valuable for isolating LAN problems with the Remote Access Network.

The man page entry for the ***linkloop*** command is listed below. Following this listing is an example of how the ***linkloop*** command can be used to help resolve network problems.

linkloop(1M) linkloop(1M)

NAME

linkloop - verify LAN connectivity with link-level loopback

SYNOPSIS

linkloop [-n count] [-f devfile] [-t timeout] [-s size] [-r rif] [-v] linkaddr1 [linkaddr2 ...]

DESCRIPTION

linkloop uses IEEE 802.2 link-level test frames to check connectivity within a local area network (LAN). This program differs from the remote loopback capability of ***rlb*** (see rlb(1M)) in that it tests only link-level connectivity; not transport-level connectivity.

The required parameter is the hardware station address (linkaddr1, linkaddr2, ...) of a remote node. ***linkloop*** tests the connectivity of the local node and the remote node specified by the hardware station address. The hardware station address of a remote node can be found by executing ***lanscan*** on the remote node. This hardware station address is usually represented as a hexadecimal string prefixed with 0x, but can also be represented as a octal string prefixed with 0 or as a decimal string. The hardware station address must not be a multicast or broadcast address.

Options
linkloop recognizes the following options:

-n count — Sets the number of frames to transmit. If count is 0, ***linkloop*** transfers frames indefinitely until an interrupt signal (defined by the user shell) is received. The default value for count is 1.

-t timeout — Sets the amount of time (in seconds) to wait for a reply from the remote node before aborting. If timeout is 0, linkloop waits indefinitely for a reply. The default value for timeout is 2 seconds.

-s size — Sets the size of the data message to send. The maximum data size is dependent on the type of LAN link being used. The default value is the maximum data byte count that can be used for the particular link.

-v — Sets the verbose option. In addition to the regular summary of test results, this option displays more extensive error information. If there are header or length errors, appropriate messages are displayed. All verbose output is preceded by the number of replies accepted before an error occurred.

-f devfile — Specifies which device file to use. The device file must be an LAN device file. If no device file is entered, ***linkloop*** uses /dev/lan0 as the default. If /dev/lan0 is not present or is the wrong type of device file, ***linkloop*** uses /dev/ieee0 as the default.

-r rif — The rif is the routing information field used for token-ring networks. This information allows a user to specify the particular bridge route over which the token ring packet should be delivered. The rif value is given as an even number of hexadecimal bytes separated by colons, up to a maximum of 16 bytes.

Connectivity Test Results
linkloop aborts upon receipt of an interrupt signal. If aborted, the current results are printed.

linkloop prints the result of the link-level connectivity test. If the test fails, it prints a summary of the test and indicates the type of error. The possible messages are:

address has bad format
Incorrect hardware station address was entered on the command line.

address is not individual
Station address entered on the command line is either a multicast or broadcast address.

frames sent
Total number of frames sent.

frames received correctly
Total number of frames received without errors.

frames with length error
Received frame length does not match transmitted frame length. If the verbose option is set, the length received is printed.

frames with data error

Received frame does not match transmitted frame.

frames with header error

Number of frames received containing unexpected frame header information. Either the source address does not match the remote address, the destination address does not match the local address, or the control field is not the TEST frame control field. These frames are ignored. ***linkloop*** continues to try to receive the reply frame until the read operation times out.

reads that timed out

Count of how many read operations timed out before the reply was received.

DIAGNOSTICS

illegal count parameter

count is not a non-negative integer, or the number specified is too large for the local computer.

illegal timeout parameter

timeout is not a non-negative integer, or the number specified multiplied by 1000 is too large for the local computer.

illegal size parameter

Size specified is not in the range from 0 to the maximum link data size. Remember that the maximum link data size may vary in value between different LAN connection types.

unable to use device file

Specified device file does not exist or is of the wrong type, or ***linkloop*** is already being executed by another user.

unable to use default device file

Default device file does not exist or is of the wrong type, or ***linkloop*** is already being executed by another user.

invalid rif parameter

Values in the rif parameter were invalid.

rif parameter too long

The number of bytes in the rif parameter exceeded the maximum allowed (16).

rif parameter length must be even

The number of bytes in the rif parameter was odd. The number of bytes must be even.

AUTHOR

linkloop was developed by HP.

FILES

/dev/lan0	first default device file for Series 700
/dev/ieee0	second default device file for Series 700

SEE ALSO

lanscan(1M), lan(7).

The linkloop command will only provide meaningful results when used with two nodes on the same LAN segment. Attempting to run ***linkloop*** over the WAN link to a remote node will not be successful (in this case, the ***ping*** command should be used instead). The linkloop command is invoked as follows:

```
/etc/linkloop -v 0x0800096265ef
```

where ***0x0800096265ef*** is the station address of the remote node (the more common form of the above station address, as produced by the ***arp*** command, is 8:0:9:62:65:ef). The ***-v*** option signifies ***verbose*** mode. If successful, ***linkloop*** will respond as follows:

```
Link connectivity to LAN station: 0x0800096265ef -
- OK
```

An unsuccessful execution of ***linkloop*** will result in a response similar to the following:

```
/etc/linkloop -v 0x0800096265ef
```

```
Link connectivity to LAN station: 0x0800096265ef -
- FAILED
     frames sent :                   1
     frames received correctly :     0
     reads that timed out :          1
```

When using ***linkloop*** to test connectivity with the Remote Access router/hub, the following form of the command should be used:

```
/etc/linkloop -f /dev/lan1 -s3 -t10 -v 0x0000a2ce5707
```

which tells ***linkloop*** to use the second lan interface (***-f /dev/lan1***), send one (***default***) three-byte packet (***-s3***), wait ten seconds (***-t10***) for a response, and use verbose mode (***-v***). Since the router hub is not a Hewlett-Packard product, this command will not be fully successful. However, the response returned will indicate the data was received by the remote node and transmitted back, as shown below:

```
Link connectivity to LAN station: 0x0000a2ce5707 -
- FAILED
     frames sent :                   1
     frames received correctly :     0
     frames with data error :        1
```

The response above indicates the remote node responded with erred data. This indicates that connectivity does in fact exist between the two nodes, as compared to the previous example where the failure was the result of ***"reads that timed out"*** (indicating no response from the remote node).

The ***linkloop*** command has a number of limitations for use in the ADAS Remote Access Network as compared to the ***ping*** command. And in fact, ***ping*** will be the more usable command in most cases. However, whereas ***ping*** provides simple go/no-go type results of an attempt to establish network connectivity, ***linkloop*** can provide additional details about a failed attempt to establish connectivity.

5.2 Networking Equipment: Router/Hub

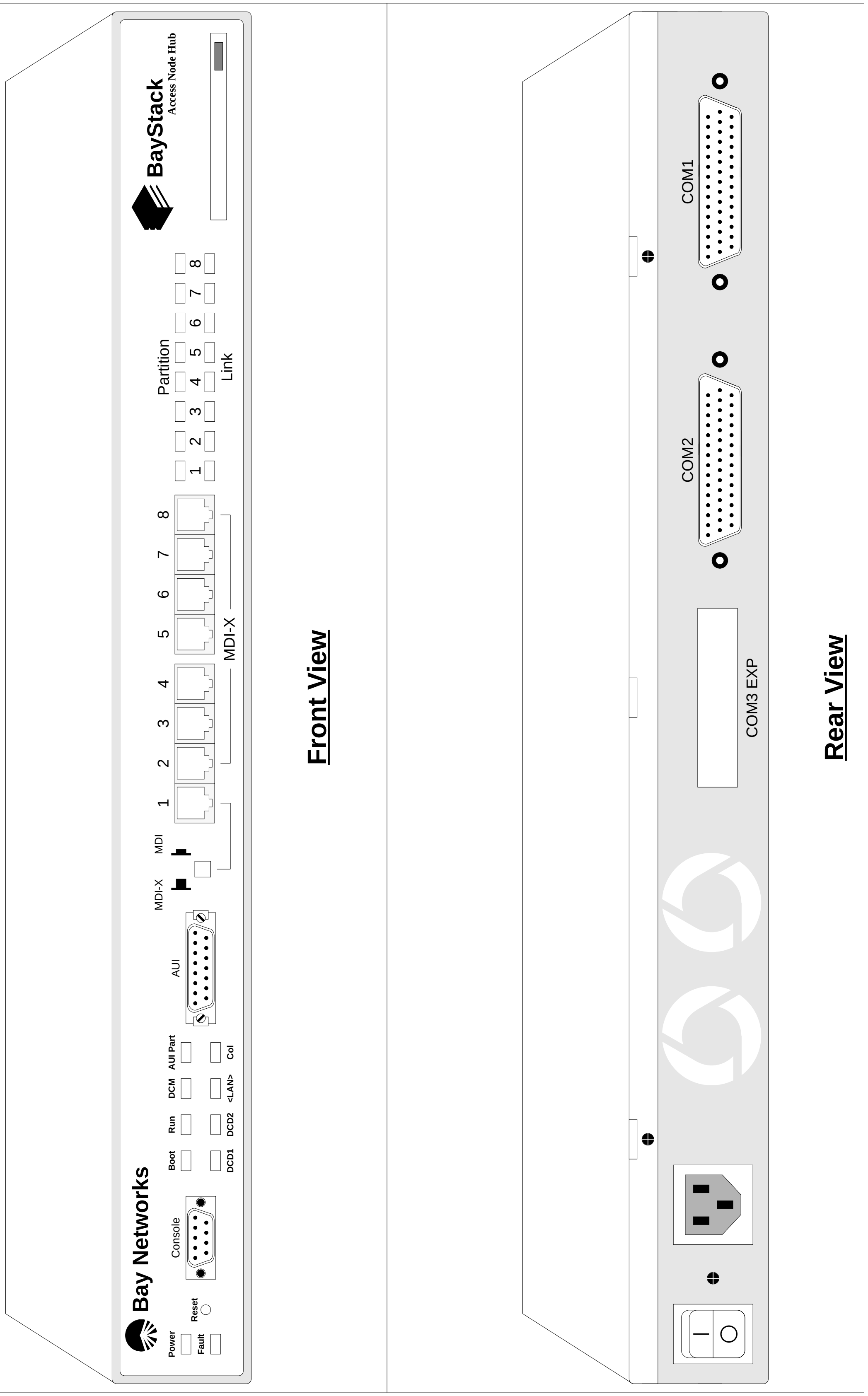
In addition to the network diagnostic capabilities available on the workstation, the routing equipment also provides diagnostic and fault isolation capabilities. These include rudimentary front panel visual indicators, as well as on-line software commands that can be issued from the terminal interface or via a login session from the ADAS or Remote Access Workstation.

Note: The following details are specific to the equipment validated by Northern Telecom. Equipment provided by alternate vendors may have different, but possibly equivalent, commands and/or capabilities.

5.2.1 Front Panel Indicators

Figure 26 on page 158 illustrates the LAN/WAN routing equipment validated and recommended by Northern Telecom for use in the ADAS Remote Access Network. As can be seen in this figure, the routing equipment contains a number of front panel visual indicators that provide rudimentary status and fault information. A brief description of the front panel indicators is provided in the following sections.

Figure 26 Remote Access Network Equipment: Router/Hub (ANH)



ANH Routing Equipment Front Panel Indicators

LED Label	Color	Meaning
Power	Green	Lights after DC power is delivered to internal circuitry and remains on while the ANH is powered on.
Fault	Amber	Lights briefly when the ANH is powered on. Remains on if a diagnostic failure occurs. If the Run LED is also on, indicates that the ANH is running its self-test.
Boot	Green	Lights for 1 to 3 minutes while the ANH is booting to indicate that diagnostic tests were successful after power up.
Run	Green	Flashes for 1 to 3 minutes while diagnostic tests are running; this indicates that the ANH has not yet started to execute the runtime image software code. The light remains on to indicate that the ANH has begun to execute the software image. If the Fault LED is also on, this indicates that the ANH is running its self-test.
DCM	Green	Lights to indicate an RMON data collection module (DCM) is installed.
AUI Part (Partition)	Amber	Lights to indicate that the AUI port has been partitioned from the repeater due to a disruption in transmission. Autopartitioning occurs after an excessive number of consecutive collisions or an excessively long single-collision signal.
DCD1 (Data Carrier Detect 1)	Green	Lights to indicate that the first synchronous port (COM1) is active.
DCD2 (Data Carrier Detect 2)	Green	Lights to indicate that the second synchronous port (COM2) is active.
<LAN>	Green	Lights after each data transmission to indicate data present in the repeater. The signal is longer than the duration of data, allowing the eye to perceive the occurrence of very short transmissions.
Col (Collision)	Amber	Lights to indicate a collision in the AUI repeater.
Partition (8 LEDs)	Amber	Lights when a specific repeater port (1-8) has been partitioned due to a disruption in transmission. Autopartitioning occurs after an excessive number of consecutive collisions or an excessively long single-collision signal. All 8 LEDs flash when the ANH is reset.
Link (8 LEDs)	Green	Lights to indicate that a repeater port registers a connection; that is, the port is currently connected to another powered 10Base-t port. All 8 LEDs flash when the ANH is reset.

5.2.2 LED Power Up Sequence

When the ANH is powered up, the following LED sequence should occur:

1. The ***POWER LED*** lights and remains on.
2. ***RUN***, ***BOOT***, and ***FAULT LEDs*** light for approximately 1 second, then turn off, indicating that the ANH is functioning.
3. The ***RUN LED*** blinks for 1 to 3 minutes, indicating that the diagnostic tests are running. (The ***RUN LED*** blinks quickly during the initial 17-second memory test, then slows down during the interface tests.)
4. The ***BOOT LED*** lights for 1 to 3 minutes, indicating that the diagnostic tests were successful and the ANH is booting.
5. The ***RUN LED*** lights and the ***BOOT LED*** turns off, indicating that the ANH is operational.

5.2.3 ANH Controls

The ANH has minimal controls, all of which are listed below:

1. The ***Power Switch***, which is located in the right rear of the ANH (when viewed from the front). To power on the ANH, press the power switch to the ‘ON’ (1) position. To power off the ANH, press the power switch to the ‘OFF’ (0) position.
2. The ***Reset Switch***, which is used to reboot the ANH without cycling the power, is recessed into the left front of the unit. To ***warm-boot*** the ANH, i.e., no diagnostic tests run, press the reset button for ***less*** than 3 seconds. To ***cold-boot*** the ANH, i.e., diagnostic tests run, press the reset button for ***more*** than 3 seconds.

   To activate the reset switch, use a small pointed object, such as a paper clip, to press in the button.

   Note: The ***Link*** and ***Partition LEDs*** for Repeater Ports 1-8 flash when the ANH is reset, whether or not a port has attached cables.
3. The ***MDI-X/MDI Switch***, located near the front center of the ANH, permits the 10Base-T Repeater Port 1 to be changed from a ***MDI-X*** (media-dependent interface with crossover) configuration to a ***MDI*** (media-dependent interface) configuration. With this switch pressed ***IN***, Repeater Port 1 is configured as a ***MDI*** connector. In this position, two ANH routers could be daisy-chained together using a straight-through ethernet cable instead of a crossover cable. With switch in the ***OUT*** position, the transmit and receive signal connections are reversed and Repeater Port 1 becomes a ***MDI-X*** connector.

Even though the Remote Access configuration does not use Repeater Port 1, the ***MDI-X/MDI Switch*** should be left in the ***OUT*** position (***MDI-X***) to prevent confusion and maintain a consistent configuration for all 8 repeater ports.

4. The ***Flash Card Eject Button***, located adjacent to the flash card receptacle on the right front of the ANH, is used to eject the flash memory card from the unit. Depressing this button unlocks and unseats the flash card from its connector so the flash card can be removed from the ANH.

   Caution! Do not remove the flash memory card while the ANH is operating, as this will interfere with the current networking operations.

5.2.4 Terminal Diagnostics

The terminal diagnostics are available via a telnet session to the router/hub, or by a direct connection to the Technician Interface with an ascii terminal. Numerous commands are available via a terminal session to the router/hub, most of which pertain the to configuration of the hardware, but many of which provide status information about the various protocols and interfaces. The two primary commands useful for diagnostic purposes are displayed below.

1. ***ping;*** The ***ping*** command functions in a similar fashion as ***ping*** on the workstation. ***Ping*** is an active command, in that the execution of this command initiates activity on the Remote Access Network.
2. ***show;*** The ***show*** command is used to display the status of various registers and configuration options in the router hub. ***Show*** is a passive command, in that it only provides status information and does not initiate network activity.

To initiate any of the terminal commands, the user must first log into the router/hub. If the workstation is connected to the hub and the network is active, the user can simply initiate a ***telnet*** session to the router and log in from the workstation.

In many cases, however, the network will not be active (hence the need to perform some fault isolation activities), so access to the terminal commands will not be available from the workstation. In these cases, the user must connect an ascii terminal to the Technician Interface port (located on the front of the router/hub) to log into the router to initiate the diagnostic functions.

Details regarding the functionality of the above two commands are provided in the following sections.

5.2.4.1Ping

The ***ping*** command on the router/hub is very similar to the ***ping*** command on the workstation. And like the workstation version, ***ping*** will be the most important and useful diagnostic tool for diagnosing problems with the Remote Access Network.

The help entry for the ***ping*** command is listed below.

ping

NAME

ping - send ICMP ECHO_REQUEST packets to network hosts

SYNOPSIS

ping <address> [-t<timeout>] [-r<repeat count>] [-s<size>] [-p] [-a<address>] [-v]

DESCRIPTION

ping allows the execution of an ICMP Echo Request/Reply handshake with the specified IP Address <address>.

The options and parameters to ***ping*** are:

-t<timeout> Specifies the amount of time to wait before deciding success/failure of the ***ping*** command.

-r<repeat count> Specifies the number of times to issue the ***ping*** message to <address>. The default <repeat count> is 1.

-s<size> Specifies the size of the ***ping*** data in bytes (maximum size is 1024 bytes).

-p Enables the ‘path report’ option, which displays the intervening hops used in reaching the destination.

-a<address> Specifies an optional source address from which to issue the ***ping*** command.

-v Displays statistical information about the ***ping***, including the success rate and the round-trip times of the ICMP Echo Request/Reply.

With the exception of some minor syntax differences, the ***ping*** command on the router/hub functions essentially the same as the ***ping*** command on the workstation, and would be used in a similar fashion to isolate network problems as shown ***Section 5.1.1.6 on page 140***.

5.2.4.2Show

The ***show*** command is used to view the MIB records associated with a particular entry. ***show*** is very useful for querying the configuration of the router/hub and for displaying various statistical information about the network activity. The options and parameters for ***show*** are too numerous to easily detail, so instead, the following examples are provided to illustrate how ***show*** can be used to illicit pertinent information about the Remote Access Network.

***Note**: Comments are enclosed in brackets and italicized like <this>.*

```
[1:1]$ show <show with no parameters simply provides the available options>

    show <entity>
    Where <entity> is one of the following:

      csmacd fr                    ftp ip snmp
      sync  tcp                    telnet tftp

[1:1]$ show ip <display all the options for the show ip command>

    Usage: show ip <option>
    where <option> is one of the following:
      ?
      adjacent hosts
      alerts
      arp
      base
      circuits            [<circuit_name>]
      disabled
      enabled
      rfilters            [<export | import>] [<PROTO>]
      rip
      rip alerts
      rip disabled
      rip enabled
      rip timers
      rip auth
      routes              [type [local|bgp|rip|egp|ospf|<etc.>] |
                          <ip address> | find <search pattern>]
      static
      stats
      stats cache         [<circuit_name>]
      stats fragments     [<circuit_name>]
      stats icmp client   [<circuit_name>]
      stats icmp in       [<circuit_name>]
      stats icmp misc     [<circuit_name>]
      stats icmp out      [<circuit_name>]
      stats icmp server   [<circuit_name>]
      stats security in   [<circuit_name>]
      stats security out  [<circuit_name>]
      traffic filters

[1:1]$ show ip alerts <display alert conditions relating to the IP protocol>

    No Circuits found

[1:1]$ show ip base <display the base IP routing configuration>

      Protocol:                     IP
      State:                        Up
      Forwarding Mode:              Enabled
      Zero/All Ones Subnetting:     Enabled
      Default TTL:                  30

      RIP Diameter:                 15
      Route Cache Size:             60
```

```
       MIB Tables Maintained:        Route
       Classless:                    Enabled
       Route Filters:                Enabled

    Route pools contain 6 [est. 0] networks/subnets and 1 [est. 0] hosts.
    Maximum policy rules per type per protocol: 32

[1:1]$ show ip circuits <display the configuration of the IP interfaces>

     Circuit   Circuit #  State     IP Address          Mask
    ---------  ---------- -----  ---------------  ---------------
       S11          1       Up     192.168.0.5    255.255.255.252
       S12          2       Up    192.168.0.18    255.255.255.252
       E11          3       Up     192.168.1.2    255.255.255.192
    3 circuit(s) found

[1:1]$ show ip routes <display the routing table entries>

     Destination        Mask        Proto    Age     Cost   NextHop Addr/AS
     ------------ ---------------  ------- ------- -------  ---------------
     192.168.0.4  255.255.255.252   LOCAL     75      0       192.168.0.5
    192.168.0.16  255.255.255.252   LOCAL     75      0      192.168.0.18
     192.168.1.0  255.255.255.192   LOCAL     75      0       192.168.1.2
     192.168.2.0  255.255.255.192    OSPF     40      2       192.168.0.6
     192.168.4.0  255.255.255.192    OSPF     40      2      192.168.0.17
    5 route(s) found

[1:1]$ show ip stats <display IP packet statistics>

                               In        Out                In       Out
    Circuit   IP Address    Receives  Requests  Forwards Discards  Discards
    -------  -------------  --------- --------- -------- --------  --------
      S12     192.168.0.5     51094     50596     537        0        0
      E11     192.168.1.2      923      51218     498        0        0
    2 entries found

[1:1]$ show ip stats datagrams <display IP datagram statistics>

                             Header    Address  Unknown     In       Out       No
    Circuit   IP Address     Errors    Errors   Protocol Discards  Discards  Routes
    -------  -------------  --------- --------- -------- --------  -------- --------
      S12     192.168.0.5      0         0         0        0        0         0
      E11     192.168.1.2      0         0         0        0        0         0
    2 entries found

[1:1]$ show ip stats cache <display IP routing cache statistics>

                                  Cache     Cache     Cache
     Circuit     IP Address     Networks   Misses    Removes
    ---------  ---------------  --------- --------- ---------
       S12       192.168.0.5        5         4         0
       E11       192.168.1.2        4         2         0
    2 entries found
```

```
[1:1]$ show ip stats fragments <display IP fragment statistics>

                          Fragment  Sucssful   Failed  Fragmnt  Fragmnt    Total
    Circuit   IP Address  Received   Reassem  Reassem    Sent    Failed  Fragmnts
    ------- ------------- --------- --------- -------- -------- -------- --------
      S12    192.168.0.5      0         0         0        0        0        0
      E11    192.168.1.2      0         0         0        0        0        0
    2 entries found
```

```
[1:1]$ show ip stats icmp client <display IP ICMP client statistics>

                            Echo      Echo    Timestmp Timestmp Addrmask Addrmask
    Circuit   IP Address  Requests   Replies  Requests  Replies Requests  Replies
    ------- ------------- --------- --------- -------- -------- -------- --------
      S12    192.168.0.5      0         0         0        0        0        0
      E11    192.168.1.2     11         0         0        0        0        0
    2 entries found
```

```
[1:1]$ show ip stats icmp in <display IP ICMP receive packet statistics>

                            ICMP     ICMP In  Destint   Rcv.Tim Rcv.Parm
    Circuit   IP Address  Received   Errors   Unreach  Exceeded  Problem
    ------- ------------- --------- --------- -------- -------- --------
      S12    192.168.0.5      0         0         0        0        0
      E11    192.168.1.2     11         0         0        0        0
    2 entries found
```

```
[1:1]$ show ip stats icmp out <display IP ICMP transmit packet statistics>

                            ICMP    ICMP Out  Destint   Snd.Tim Snd.Parm
    Circuit   IP Address    Sent     Errors   Unreach  Exceeded  Problem
    ------- ------------- --------- --------- -------- -------- --------
      S12    192.168.0.5      0         0         0        0        0
      E11    192.168.1.2     11         0         0        0        0
    2 entries found
```

```
[1:1]$ show ip stats icmp misc <display miscellaneous IP ICMP statistics>

                          SrcQunch   Messages   Redirect   Messages
    Circuit   IP Address     In         In         In        Out
    ------- ------------- ---------  ---------  ---------  ---------
      S12    192.168.0.5      0          0          0          0
      E11    192.168.1.2      0          0          0          0
    2 entries found
```

```
[1:1]$ show ip stats icmp server <display IP ICMP server statistics>

                            Echo      Echo    Timestmp Timestmp Addrmask Addrmask
    Circuit   IP Address  Requests   Replies  Requests  Replies Requests  Replies
    ------- ------------- --------- --------- -------- -------- -------- --------
      S12    192.168.0.5      0         0         0        0        0        0
      E11    192.168.1.2     11         0         0        0        0        0
    2 entries found
```

```
[1:1]$ show csmacd <display all the options for the show csmacd command>
```

```
    Usage: show csmacd  <option>
    Where <option> is one following:
      alerts
      autoneg
      base
      base circuit        <circuit name>
      collisions
      collisions circuit  <circuit name>
      disabled
      enabled
      hwfilters
      rx_errors
      rx_errors circuit   <circuit name>
      tx_errors
      tx_errors circuit   <circuit name>
      sys_errors
      sys_errors circuit  <circuit name>
      sample
      sample              [period <seconds> | circuit <circuit name>]
      stats
      stats circuit       <circuit number>

      [1:1]$ show csmacd alerts <display ethernet hub alerts>

    No CSMACD Modules Found.

      [1:1]$ show csmacd collisions <display ethernet hub collisions>

    CSMACD Module Collision Information:

                         Single    Multiple                Late      Late
                       Collision  Collision  Excessive  Collision  Collision
     Slot Conn Circuit   Frames     Frames  Collisions   Transmit   Receive
     ---- ---- ------- ---------  --------- ----------  ---------  ---------
      1     1    E11        0          0          0          0         0
     1 entry(s) found
```

```
[1:1]$ show csmacd rx_errors <display ethernet receive errors>

    CSMACD Module Receive Errors:

                                                                    Internal    Late
                       Checksum  Aligment Overflow  Frames  Symbol    MAC    Collision
     Slot Conn Circuit  Errors    Errors   Errors  Too Long Errors   Errors   Received
     ---- ---- ------- --------  -------- --------  ------- ------- -------- ---------
      1     1    E11       0         0        0        0       0        0         0
     1 entry(s) found

[1:1]$ show csmacd tx_errors <display ethernet hub transmit errors>
```

```
    CSMACD Module Transmit Errors:

                                                       Internal
                          Frames   Underflow     MAC      Deadlock  Excessive     Late
     Slot  Conn  Circuit Too Long   Errors     Errors     Errors   Collisions Collisions
     ----  ----  ------- -------- --------- --------- --------- ---------- ----------
       1     1     E11       0         0         0         0          0          0
     1 entry(s) found
```

```
[1:1]$ show csmacd sys_errors <display ethernet hub system errors>

    CSMACD Module System Errors:

                          Memory   Collision  Internal   Loss of
     Slot  Conn  Circuit  Errors    Errors     Buffer    Carrier
     ----  ----  ------- -------- --------- --------- ---------
       1     1     E11       0         0         0         0
     1 entry(s) found
```

```
[1:1]$ show csmacd stats <display ethernet hub statistics>

    CSMACD Module I/O Statistics:

                         Received  Received  Transmitted Transmitted  Transmit    Total
     Slot  Conn  Circuit   Bytes    Frames      Bytes       Frames    Deferred   Errors
     ----  ----  ------- -------- --------- ----------- ----------- --------- ---------
       1     1     E11    516776     978     10790232     147143        0          0
     1 entry(s) found
```

```
[1:1]$ show csmacd sample <sample ethernet hub traffic and display results>

    Taking the First Sample.
    Waiting 10 Seconds ...
    Taking the Second  Sample.
    CSMACD Sampled Data Over 10 Seconds.

                            Rx        Tx      Rx Lack of  Tx Lack of
     Slot  Conn  Circuit  Frames    Frames    Resources   Resources
     ----  ----  ------- -------- --------- ----------- -----------
       1     1     E11       0         3          0           0
     1 entry(s) found
```

```
[1:1]$ show sync <display all the options for the show sync command>

       ?
       alerts
       base                 [circuit <circuit name>]
       disabled
       enabled
       rx_errors            [circuit <circuit name>]
       sample               [period <seconds> | circuit <circuit name>]
       stats                [circuit <circuit name>]
       sys_errors           [circuit <circuit name>]
       tx_errors            [circuit <circuit name>]
```

```
[1:1]$ show sync alerts <display WAN port alerts>
```

```
    No entries found

[1:1]$ show sync rx_errors <display WAN port receive errors>

                         Bad       Runt      Frame     Frames   Overflow
    Slot Conn Circuit   Frames    Frames    Rejects   Too Long   Frames
    ---- ---- ------- --------- ---------  ---------- --------- ---------
     1    2     S12       0         0          0         0          0
    1 entry(s) found

[1:1]$ show sync tx_errors <display WAN port transmit errors>

                      Underflow
    Slot Conn Circuit   Frames
    ---- ---- ------- ---------
     1    2     S12       0
    1 entry(s) found

[1:1]$ show sync sys_errors <display WAN port system errors>

                       Receive   Transmit     T1       Memory
    Slot Conn Circuit  Rejects   Rejects   Timeouts    Errors
    ---- ---- ------- --------- ---------  ---------- ---------
     1    2     S12       0         0          0         0
    1 entry(s) found

[1:1]$ show sync stats <display WAN port statistics>

                       Receive   Receive   Transmit   Transmit   Total
    Slot Conn Circuit   Bytes     Frames     Bytes     Frames    Errors
    ---- ---- ------- --------- ---------  ---------- --------- ---------
     1    2     S12    14669831   377420   14679603    377462       0
    1 entry(s) found
```

```
[1:1]$ show sync sample <sample WAN port traffic and display results>

    Taking the First Sample.
    Waiting  10 Second(s) ...
    Taking the Second Sample.

                           Rx        Tx      Rx Lack of  Tx Lack of
     Slot  Conn Circuit  Frames    Frames    Resources   Resources
     ---- ---- ------- -------- --------- ----------- -----------
       1     2     S12       7         7          0           0
     1 entry(s) found

[1:1]$ show tcp <display the tcp configuration>

    Tcp protocol is enabled.
       The Time Out Minimum: 250 milliseconds
       The Time Out Maximum: 240000 milliseconds
       The Maximum Window Size: 4096 in octets
       The Time Out Algorithm: Van_Jacobson
       The Number of Segments Sent: 1170
       The Number of Segments Received: 382
       The Number of Segments Retransmitted: 4
       The Number of Bad Segments Received: 0
       The Number of Segments Sent Containing the Reset Flag: 0
       The Number of Established Connections: 0

    The current TCP connections :

                        Local                   Remote
         Local IP        Port     Remote IP      Port    State
     --------------- ------- --------------- ------- ---------
         0.0.0.0         21       0.0.0.0         0     Listen
         0.0.0.0         23       0.0.0.0         0     Listen

[1:1]$ show ftp <display the ftp configuration>

    Ftp enabled.
       Default Volume:     1
       Idle Timeout:       900 minutes
       Max. Sessions:      3
       Max. Login Retries: 3
       Transfer Type:      Binary
       Control Type:       Low Delay
       Data Type:          Hi Thru Put
       TCP Window Size:    60000
       Logins:             0
       Logins Failed:      0
       Files Received:     0
       Avg In Rate Kb/s:   0
       In Errors:          0
       Files Sent:         0
       Avg Out Rate Kb/s:  0
       Out Errors:         0
```

```
[1:1]$ show telnet <display the telnet configuration>

    Telnet Server enabled.
       TI/Telnet Prompt: “[%slot%:TN]$ “
       Screen Size: 24
       Max. Login Retries: 3
       Login Time Out: 1 minutes
       Password Time Out: 1 minutes
       Command Time Out:  15 minutes

    Telnet Client enabled.
       Telnet Command Prompt: ““
       Remote Telnet/Tcp Port: 23

[1:1]$ show tftp <display the tftp configuration>

    TFTP protocol is enabled.
       The Default Volume: 1
       Retransmit Timeout Value: 5 Seconds
       Max Number of Retransmits: 5
       Number of Writes Received: 0
       Number of Reads Received: 0
       Number of Retransmits: 0

[1:1]$ show snmp <display the snmp configuration>

    Snmp protocol is enabled.
       Authentication Type: Trivial
       Received PDUs: 0
       Transmitted PDUs: 0
       MIB Objects Retrieved: 0
       MIB Objects Set: 0
       Get Request PDUs Accepted & Processed: 0
       Get Next Request PDUs Accepted & Processed: 0
       Get Response PDUs Generated: 0
       Set Request PDUs Accepted & Processed: 0
       Trap PDUs Generated: 0
       Decoding ANS.1 Parsing Errors: 0
       Recieved Bad Community Name PDUs: 0
       Recieved Unsupported Operation PDUs: 0
       Generated PDUs with “tobig” Error: 0
       Generated PDUs with “noSuchName” Error: 0
       Generated PDUs with “badValue” Error: 0
       Generated PDUs with “readOnly” Error: 0
       Generated PDUs with “genErr” Error: 0
```

5.3 Ethernet Twisted-Pair Media Attachment Unit

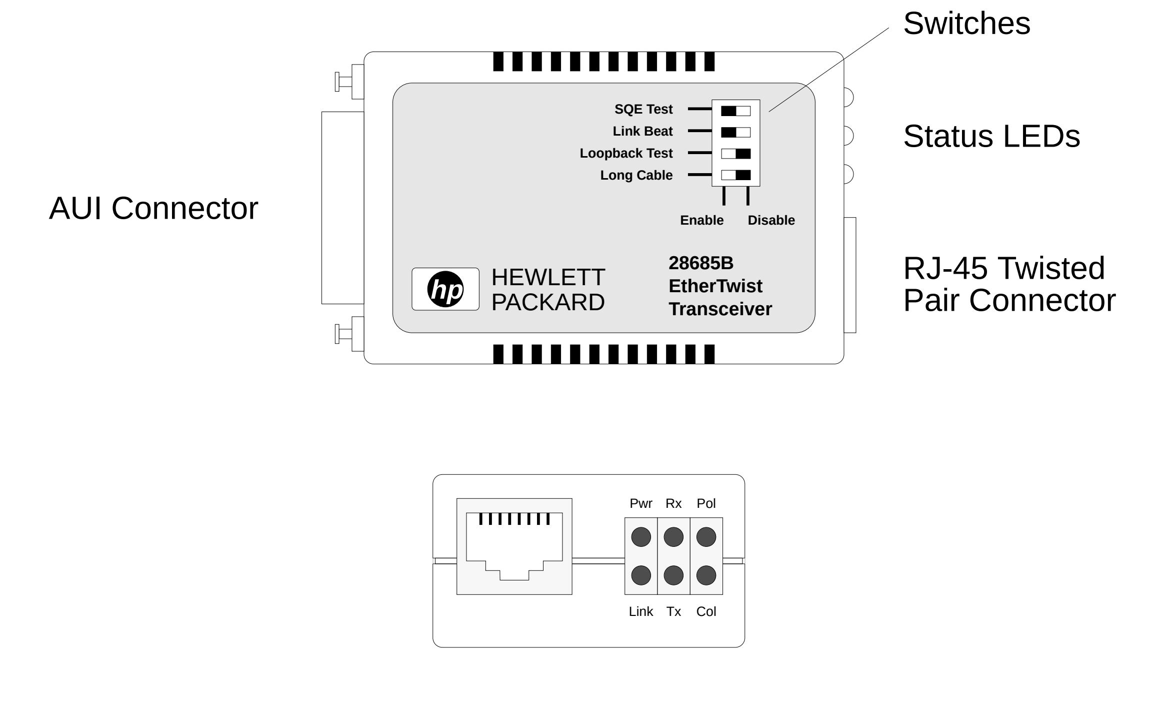
The physical and electrical connectivity to the ADAS Remote Access Network by a workstation is provided by a Media Attachment Unit (MAU). The twisted pair ethernet cable plugs directly into one end of the MAU via a RJ-45 connector, and the MAU plugs into the workstation via a 15-pin delta-subminiature (D-Sub) connector.

On the ADAS Workstation, the Remote Access Network connects to the second ethernet port (LAN1), which is located on an add-in card. The use of a MAU is mandatory on this port, as no other access point is provided for the LAN.

Since only one ethernet LAN is necessary for the Remote Access Workstation, the Remote Access Network connects to the first ethernet interface (LAN0). In addition to the standard AUI (attachment unit interface) port, the first ethernet interface has a built-in MAU, which permits the ethernet cable to be plugged directly into the workstation without the use of an external MAU. However, it is highly recommended that an external MAU be used anyway, as the external MAU provides additional diagnostic capabilities that assist in the isolation and identification of problems with the Remote Access Network.

The MAU used in the ADAS products is available from Hewlett-Packard, as part number 28685B, EtherTwist Transceiver. ***Figure 27 on page 172*** provides an illustration of the HP 28685B MAU.

Figure 27 HP28685B Ethernet MAU



5.3.1 MAU Switch Settings

The operation of the MAU is controlled by four switches located on the top of the unit as illustrated in ***Figure 27 on page 172***. A description of the switches is provided in the table below.

Switch Label	Meaning
SQE Test	Enables or disables the signal quality error (SQE) test signal. The factory setting is **ENABLE**. Keep at **ENABLE** if attaching the transceiver to a LAN card in a computer, or to another device that expects the SQE test signals such as a bridge or router. Also **ENABLE** if running a loopback test. Set to **DISABLE** if attaching the transceiver to a hub’s or repeater’s AUI port.
Link Beat	Enables or disables the link beat signal. The factory setting is **ENABLE**. Keep at **ENABLE** if the transceiver is used in a 10Base-T network. Set to **DISABLE** if the transceiver is used in a network that is not compatible with the 10Base-T standard (for example, HP StarLAN 10).
Loopback Test	Enables or disables the loopback test mode for troubleshooting the transceiver. The factory setting is **DISABLE**. Set to **ENABLE *only*** when troubleshooting with a loopback test.
Long Cable	Allows the transceiver to be used with twisted-pair cable lengths of over 100 meters (when used with high-grade, low-crosstalk cable). The factory setting is **DISABLE**. Keep at **DISABLE** if high-grade twisted-pair cable is ***not*** being used to connect the transceiver to another device. Set to **ENABLE *only*** if high-grade twisted-pair cable is used.

5.3.2 MAU LED Status Indicators

The MAU provides six LED status indicators, which are located on the end of the MAU adjacent to the RJ-45 ethernet connector. These LEDs provide information about both the MAU and the link activity. The MAU LED indicators are described below.

LED	State	Meaning
Pwr (Power)	ON	Power is being received from the network device.
Color: green	OFF	Power is not being received.

LED	State	Meaning
Link (Link Beat) Color: green	ON	If in a 10Base-T network (the Link Beat switch set to **ENABLE**), a good link has been established with a functioning 10Base-T device over the twisted-pair cable. If in a non-10Base-T network (the Link Beat switch is set to **DISABLE**), this LED is always **ON**.
	OFF	The Link Beat switch is set to **ENABLE** and for some reason the link beat signal is not being received.
Rx (Receive) Color: green	FLASH	Data is being received on the twisted-pair port. Under many normal traffic LAN traffic loads, this LED may appear **ON** continuously.
	OFF	No data is being received from the twisted-pair port.
Tx (Transmit) Color: green	FLASH	Data is being transmitted on the twisted-pair port. In heavy traffic, this LED may appear **ON** continuously.
	OFF	No data is being transmitted on the twisted-pair port.
Pol (Polarity) Color: green	ON	The UTP cable is wired correctly with respect to the polarity of the “receive” pair of wires.
	OFF	The polarity of the “receive” pair of wires is reversed from normal. The LED indicates this wiring error, but the transceiver automatically corrects for the error.
Col (Collision) Color: green	FLASH	A collision has been detected on the twisted-pair port. (A collision occurs when two or more devices try to transmit on the twisted-pair network at the same time.) If the collisions are infrequent (which is normal), this LED will blink very faintly with each collision. If it appears **ON** continuously, there may be a problem with the twisted pair cable segment attached to the transceiver.
	OFF	No collisions are being detected.

5.3.3 MAU Troubleshooting

The following table provides a quick troubleshooting reference for resolving network problems using the MAU LED indicators.

LED	State	Troubleshooting Tips
Pwr	OFF	Confirm that the device supplying power to the transceiver is powered on. Confirm that the AUI connector is secure.

LED	State	Troubleshooting Tips
Link	OFF	Confirm that the device at the other end of the twisted-pair cable is powered on, and that it is configured to send the link beat signal. Confirm that all the connections in the twisted-pair cable are secure and the cable is not broken.
Col	ON (continuous)	Either the device attached to the AUI port or the device at the other end of the twisted-pair cable is causing or propagating continuous collisions. Confirm correct operation of these devices. During a loopback test, collisions appear if you do not enable the Loopback Test switch.

6.0 Troubleshooting: How to Solve Problems

The customer is responsible for the implementation, maintenance, and ownership of the Remote Access Network. Therefore, the customer should be knowledgeable about identifying, isolating, and resolving problems with this network. The Northern Telecom Customer Support group will provide assistance to the customer in resolving problems with Remote Access to the point of confirming that the ADAS or Remote Workstation is functioning correctly or not. However, once it is confirmed that the workstation is properly configured and functional, it will be the customer's responsibility to conduct additional corrective actions to isolate, identify, and resolve problems with the network.

As a final note, it cannot be stressed enough that before the Remote Access Network is even installed, the customer should set aside some time to properly plan, engineer, and configure the implementation of the network. Equally important, the customer should fully document the network, including such things as hostnames and IP addresses, network addresses, subnet masks, link speeds, physical location of nodes,... No tools or diagnostics will be able to correct problems with a poorly or improperly implemented network, nor can any logical attempt be made to isolate a problem without accurate documentation of the network.

6.1 Tools for Fault Isolation and Identification

There are numerous tools available to the customer for identifying and correcting problems with the Remote Access Network. All of the diagnostics detailed in ***Section 5.0 on page 125*** are available to the customer for problem resolution. These many tools include the base unix networking commands and the extended diagnostic utilities on the workstation, the diagnostics on the network equipment, and even the status and fault indicators on the ethernet hubs and MAUs. Special purpose LAN diagnostic equipment may also be used to isolate and identify problems with the Remote Access Network, but this equipment should rarely be required if the tools documented herein are properly applied.

As documented in the previous section, the various tools available for resolving problems with the Remote Access network are listed below:

- Workstation based tools (base unix commands)
    1. arp
    2. ifconfig
    3. ioscan
    4. lanconfig
    5. lanscan
    6. netstat

7. ping
8. route

- Workstation based utilities (extended diagnostic functions)

  1. landiag
  2. linkloop

- Networking equipment terminal diagnostics

  1. ping
  2. show

- Networking equipment LED status indicators

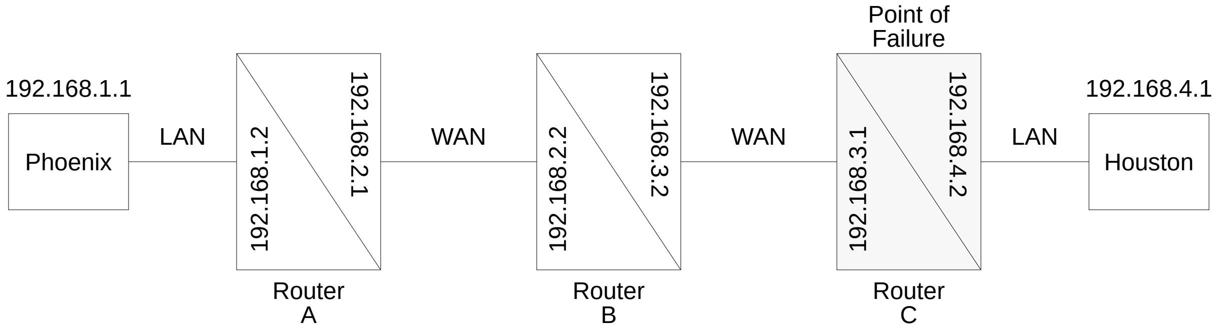
- Ethernet MAU LED status indicators

With the intelligent use of the above tools, there are very few network problems that the customer should not be able to isolate and resolve.

Also, for additional information and troubleshooting tips, the user is encouraged to reference the Hewlett-Packard document *"Installing and Administering LAN/9000 Software"*, HP Customer Order Number 98194-60530. This document is included in the documentation set provided with the ADAS/Remote Workstation and contains all the details and guidelines for installing, maintaining, and troubleshooting network interface cards in the Hewlett-Packard Series 700 workstations.

6.2 Problem Resolution Strategy

The general philosophy that is normally followed when attempting to resolve a network problem, is to start by confirming one known properly configured and functional host, and then expanding out the fault isolation process from that node in ever widening circles. For example, suppose we are trying to isolate a network problem that is preventing node Phoenix from communicated with node Houston, as illustrated in the figure below.



First, it must be established that node Phoenix is properly configured and functional, and that the local LAN connectivity is correct. Once it is confirmed that node Phoenix is correctly configured, node connectivity is tested in ever an ever widening diameter, starting with node Phoenix itself. The ***ping*** command is the best tool for doing this.

First, node Phoenix's own ethernet address is pinged (***ping*** 192.168.1.1). If this is successful, Router A's LAN port would be pinged (***ping*** 192.168.1.2), then Router A's WAN port (***ping*** 192.168.2.1). With each successful ping, a node (or router port) farther away from node Phoenix would be attempted, i.e., ***ping*** 192.168.2.2, ***ping*** 192.168.3.2, ***ping*** 192.168.3.1... In the hypothetical example above, the ***ping*** commands were successful up to (and including) Router C's WAN port. However, ***ping*** commands from node Phoenix to Router C's LAN port were not successful. This does not necessarily mean that Router C is the problem, only that the problem lies somewhere between Router C and node Houston. Additional diagnostics could be invoked by logging in to Router C, or from node Houston to fully isolate and identify the failure.

Although the example above is rather trivial, it illustrates a logical procedure to isolate and identify a network failure. This is by no means the only correct methodology for resolving network issues, but it provides a very straightforward, procedural method for troubleshooting problems with the Remote Access Network.

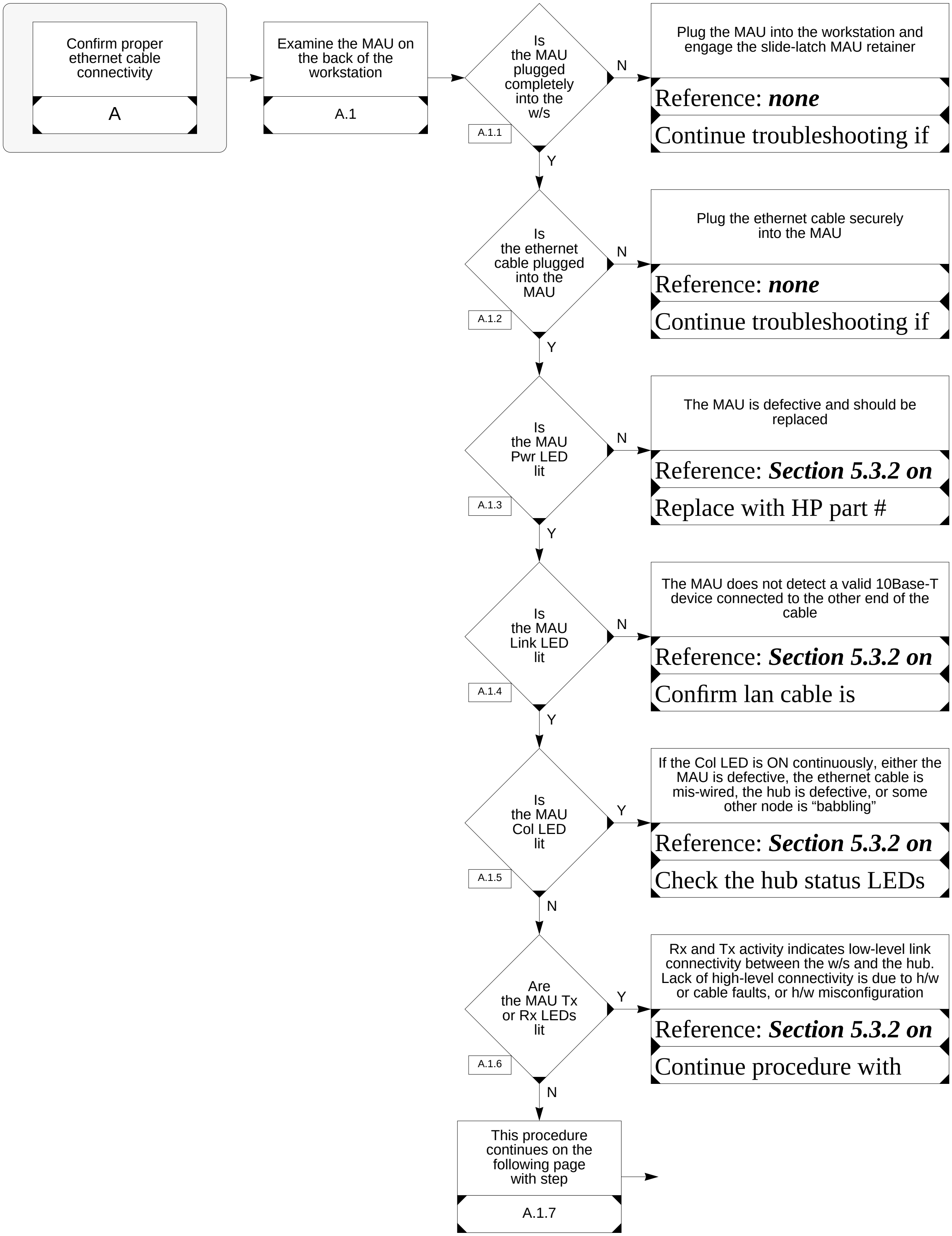
6.3 Fault Isolation Procedure

The following pages provide troubleshooting guides, including fault isolation flowcharts, that can be used to assist the user in resolving common problems that may be encountered with the Remote Access Network.

6.3.1 Remote Access Workstation Cannot Connect to Any Other Node

Issue		Troubleshooting Tip
	Remote Access Workstation Cannot Connect to Any Other Node in the Network ***Assumptions:*** None--the remote nodes to which connectivity is being attempted may or may not be properly configured, and may or may not be accessible by other nodes in the network.	***Observations:*** The problem could be configuration issues with the workstation ethernet interface or hub/router, hardware failures, or cabling faults. • Before proceeding with any in-depth troubleshooting activities, confirm that the IP addresses of the remote nodes are correct and have been entered correctly in the Remote Access site configuration panel. Also confirm the remote node names are correct and have been entered correctly in the Remote Access site configuration panel. ***Typographical errors should always be suspected as the first source of problems!*** • Confirm proper ethernet cable connectivity (Reference ***Flow Chart A on page 181***) - check workstation MAU status indicators - check workstation MAU switch settings - check ethernet hub/router status indicators - check ethernet cabling between workstation MAU and ethernet hub/router • Confirm proper configuration of the ethernet interface card in the workstation (Reference ***Flow Chart B on page 183***) - execute ***lanscan*** command - execute ***ifconfig*** command - execute ***netstat*** command - execute ***ioscan*** command • Perform workstation-based ethernet interface diagnostics (Reference ***Flow Chart C on page 184***) - execute ***ping*** connectivity tests - execute ***landiag*** utility - perform ***loopback*** test • Confirm proper configuration of the ethernet hub/router (Reference ***Flow Chart D on page 186***) - check configuration details via ***show*** command - execute ***ping*** connectivity tests

Flow Chart A Confirm Proper Ethernet Cable Connectivity



Confirm proper ethernet cable connectivity

Examine the MAU on the back of the workstation

Is the MAU plugged completely into the w/s

Plug the MAU into the workstation and engage the slide-latch MAU retainer

Reference: ***none***

Continue troubleshooting if

Is the ethernet cable plugged into the MAU

Plug the ethernet cable securely into the MAU

Reference: ***none***

Continue troubleshooting if

Is the MAU Pwr LED lit

The MAU is defective and should be replaced

Reference: ***Section 5.3.2 on***

Replace with HP part #

Is the MAU Link LED lit

The MAU does not detect a valid 10Base-T device connected to the other end of the cable

Reference: ***Section 5.3.2 on***

Confirm lan cable is

Is the MAU Col LED lit

If the Col LED is ON continuously, either the MAU is defective, the ethernet cable is mis-wired, the hub is defective, or some other node is “babbling”

Reference: ***Section 5.3.2 on***

Check the hub status LEDs

Are the MAU Tx or Rx LEDs lit

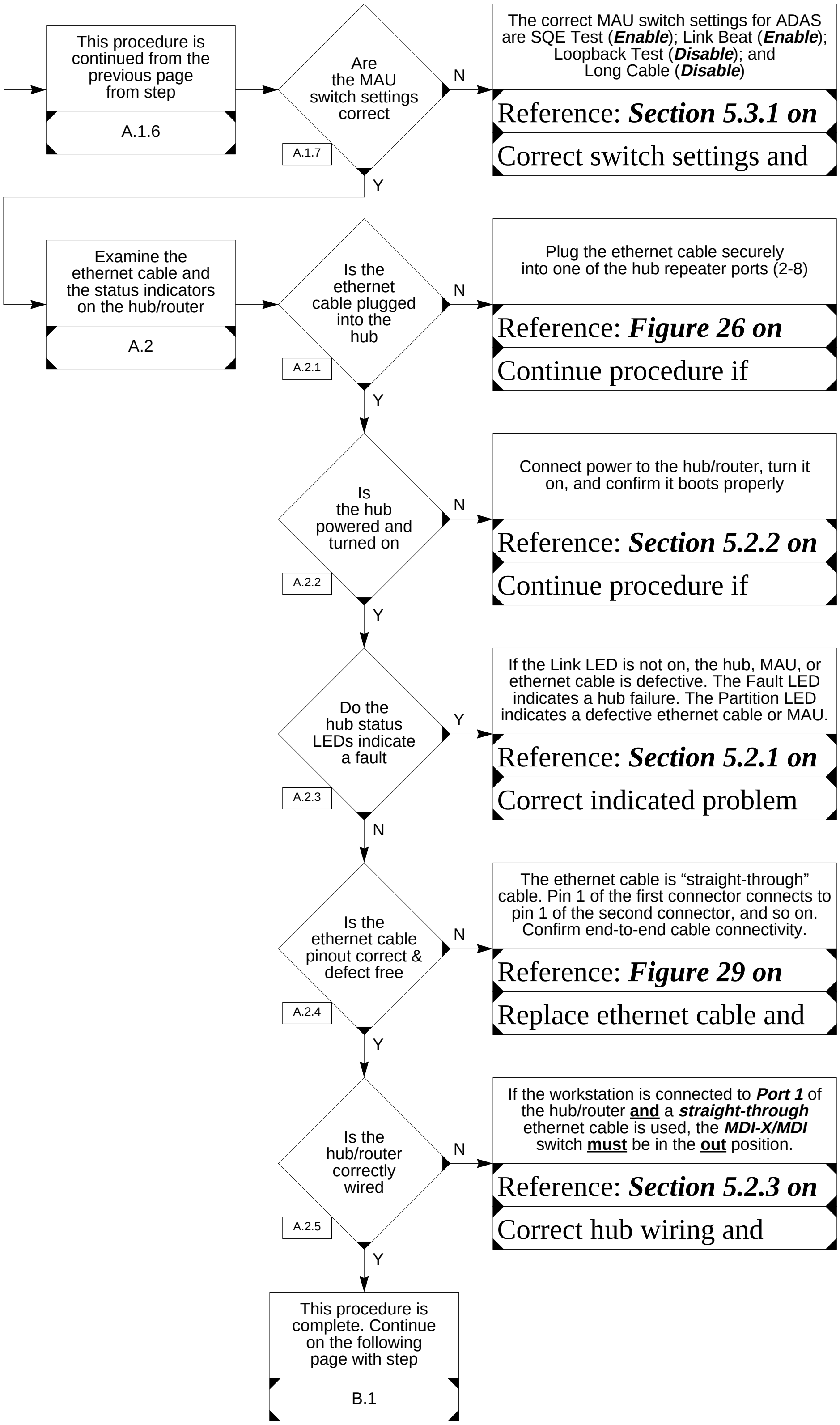
Rx and Tx activity indicates low-level link connectivity between the w/s and the hub. Lack of high-level connectivity is due to h/w or cable faults, or h/w misconfiguration

Reference: ***Section 5.3.2 on***

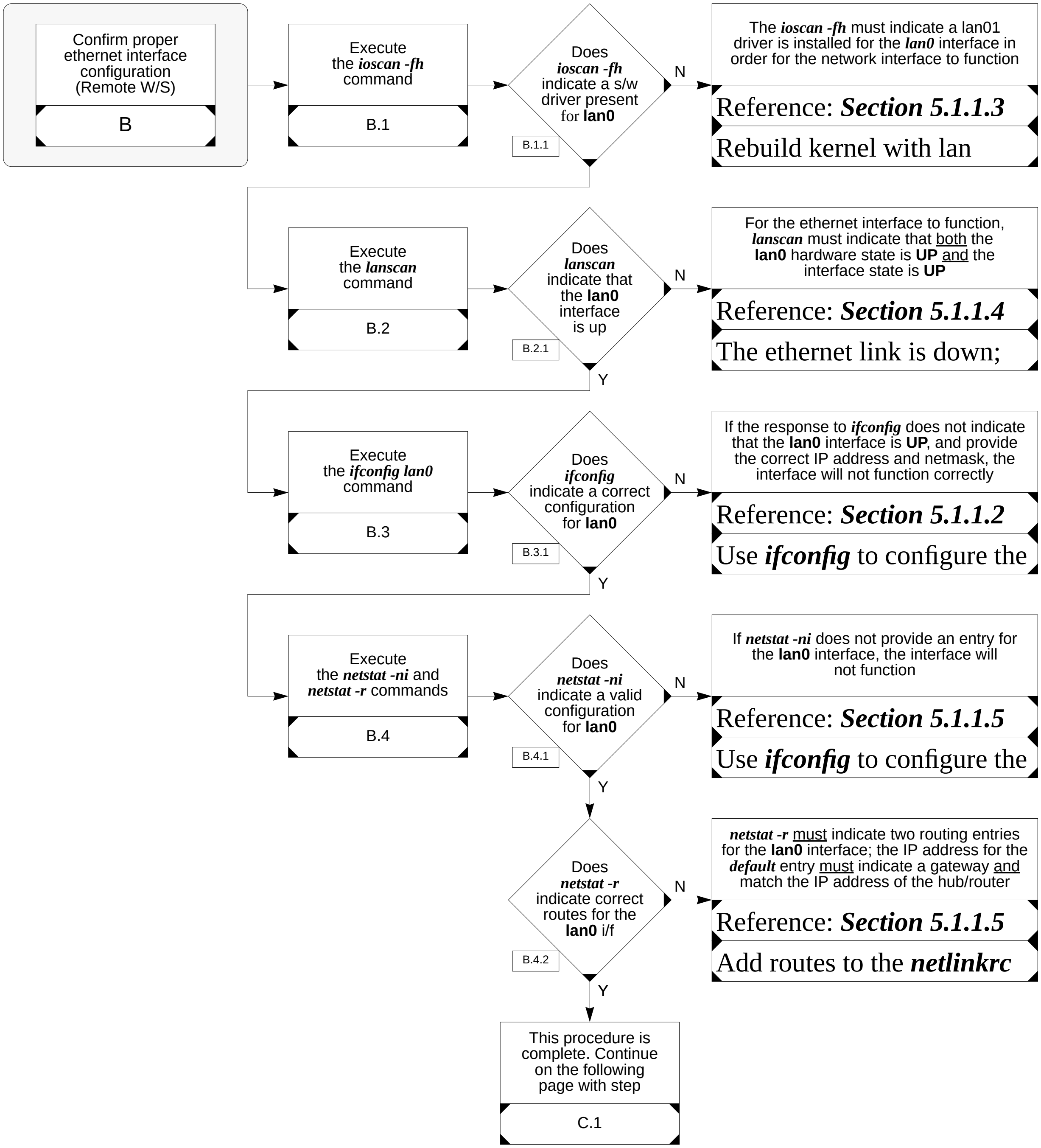
Continue procedure with

This procedure continues on the following page with step

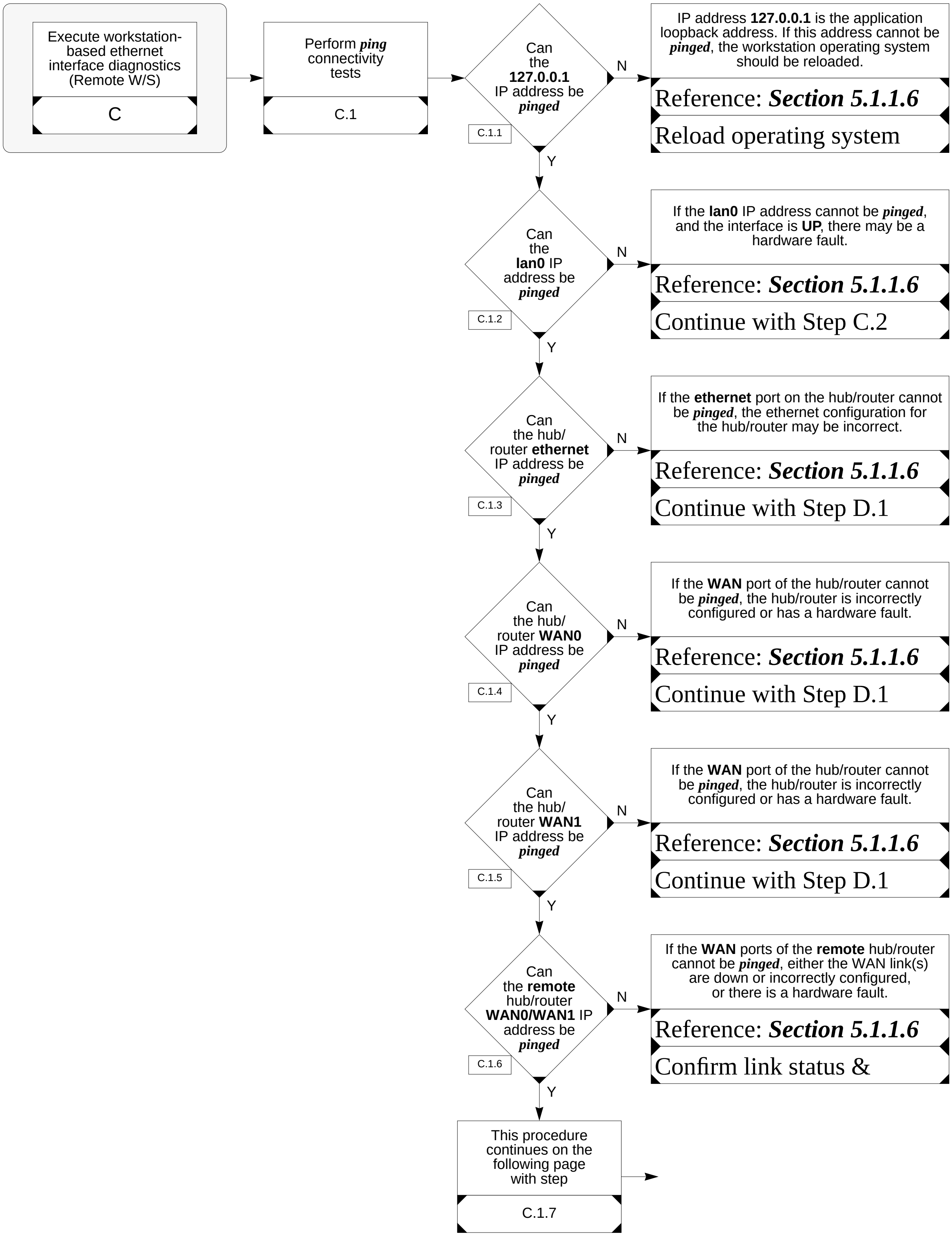
Flow Chart A (cont) Confirm Proper Ethernet Cable Connectivity


This procedure is continued from the previous page from step
A.1.6
Are the MAU switch settings correct
A.1.7
N
The correct MAU switch settings for ADAS are SQE Test (Enable); Link Beat (Enable); Loopback Test (Disable); and Long Cable (Disable)
Reference: Section 5.3.1 on
Correct switch settings and
Y
Examine the ethernet cable and the status indicators on the hub/router
A.2
Is the ethernet cable plugged into the hub
A.2.1
N
Plug the ethernet cable securely into one of the hub repeater ports (2-8)
Reference: Figure 26 on
Continue procedure if
Y
Is the hub powered and turned on
A.2.2
N
Connect power to the hub/router, turn it on, and confirm it boots properly
Reference: Section 5.2.2 on
Continue procedure if
Y
Do the hub status LEDs indicate a fault
A.2.3
Y
If the Link LED is not on, the hub, MAU, or ethernet cable is defective. The Fault LED indicates a hub failure. The Partition LED indicates a defective ethernet cable or MAU.
Reference: Section 5.2.1 on
Correct indicated problem
N
Is the ethernet cable pinout correct & defect free
A.2.4
N
The ethernet cable is "straight-through" cable. Pin 1 of the first connector connects to pin 1 of the second connector, and so on. Confirm end-to-end cable connectivity.
Reference: Figure 29 on
Replace ethernet cable and
Y
Is the hub/router correctly wired
A.2.5
N
If the workstation is connected to Port 1 of the hub/router and a straight-through ethernet cable is used, the MDI-X/MDI switch must be in the out position.
Reference: Section 5.2.3 on
Correct hub wiring and
Y
This procedure is complete. Continue on the following page with step
B.1

Flow Chart B   Confirm Proper Ethernet Interface Configuration (Remote W/S)


Confirm proper ethernet interface configuration (Remote W/S)
B
Execute the *ioscan -fh* command
B.1
Does *ioscan -fh* indicate a s/w driver present for **lan0**
B.1.1
N
The *ioscan -fh* must indicate a lan01 driver is installed for the *lan0* interface in order for the network interface to function
Reference: *Section 5.1.1.3*
Rebuild kernel with lan
Execute the *lanscan* command
B.2
Does *lanscan* indicate that the **lan0** interface is up
B.2.1
N
Y
For the ethernet interface to function, *lanscan* must indicate that both the **lan0** hardware state is **UP** and the interface state is **UP**
Reference: *Section 5.1.1.4*
The ethernet link is down;
Execute the *ifconfig lan0* command
B.3
Does *ifconfig* indicate a correct configuration for **lan0**
B.3.1
N
Y
If the response to *ifconfig* does not indicate that the **lan0** interface is **UP**, and provide the correct IP address and netmask, the interface will not function correctly
Reference: *Section 5.1.1.2*
Use *ifconfig* to configure the
Execute the *netstat -ni* and *netstat -r* commands
B.4
Does *netstat -ni* indicate a valid configuration for **lan0**
B.4.1
N
Y
If *netstat -ni* does not provide an entry for the **lan0** interface, the interface will not function
Reference: *Section 5.1.1.5*
Use *ifconfig* to configure the
Does *netstat -r* indicate correct routes for the **lan0** i/f
B.4.2
N
Y
netstat -r must indicate two routing entries for the **lan0** interface; the IP address for the *default* entry must indicate a gateway and match the IP address of the hub/router
Reference: *Section 5.1.1.5*
Add routes to the *netlinkrc*
This procedure is complete. Continue on the following page with step
C.1

Flow Chart C Execute Workstation-Based Ethernet Interface Diagnostics (Remote W/S)



Execute workstation-based ethernet interface diagnostics (Remote W/S)

C

Perform ***ping*** connectivity tests

C.1

Can the **127.0.0.1** IP address be ***pinged***

C.1.1

IP address **127.0.0.1** is the application loopback address. If this address cannot be ***pinged***, the workstation operating system should be reloaded.

Reference: ***Section 5.1.1.6***

Reload operating system

Can the **lan0** IP address be ***pinged***

C.1.2

If the **lan0** IP address cannot be ***pinged***, and the interface is **UP**, there may be a hardware fault.

Reference: ***Section 5.1.1.6***

Continue with Step C.2

Can the hub/router **ethernet** IP address be ***pinged***

C.1.3

If the **ethernet** port on the hub/router cannot be ***pinged***, the ethernet configuration for the hub/router may be incorrect.

Reference: ***Section 5.1.1.6***

Continue with Step D.1

Can the hub/router **WAN0** IP address be ***pinged***

C.1.4

If the **WAN** port of the hub/router cannot be ***pinged***, the hub/router is incorrectly configured or has a hardware fault.

Reference: ***Section 5.1.1.6***

Continue with Step D.1

Can the hub/router **WAN1** IP address be ***pinged***

C.1.5

If the **WAN** port of the hub/router cannot be ***pinged***, the hub/router is incorrectly configured or has a hardware fault.

Reference: ***Section 5.1.1.6***

Continue with Step D.1

Can the **remote** hub/router **WAN0/WAN1** IP address be ***pinged***

C.1.6

If the **WAN** ports of the **remote** hub/router cannot be ***pinged***, either the WAN link(s) are down or incorrectly configured, or there is a hardware fault.

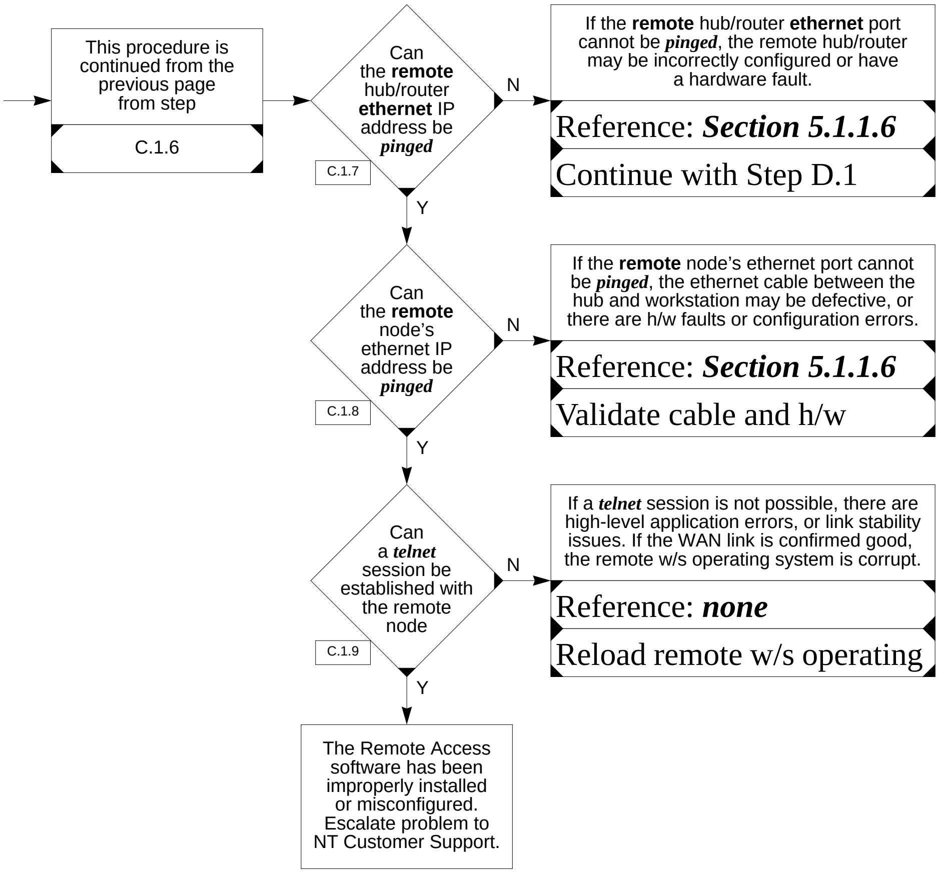
Reference: ***Section 5.1.1.6***

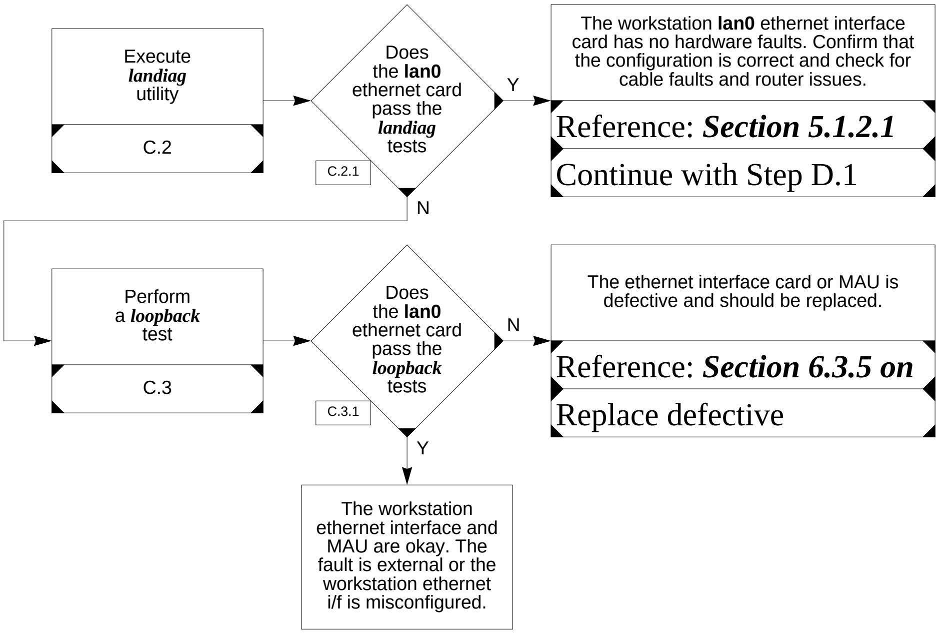
Confirm link status &

This procedure continues on the following page with step

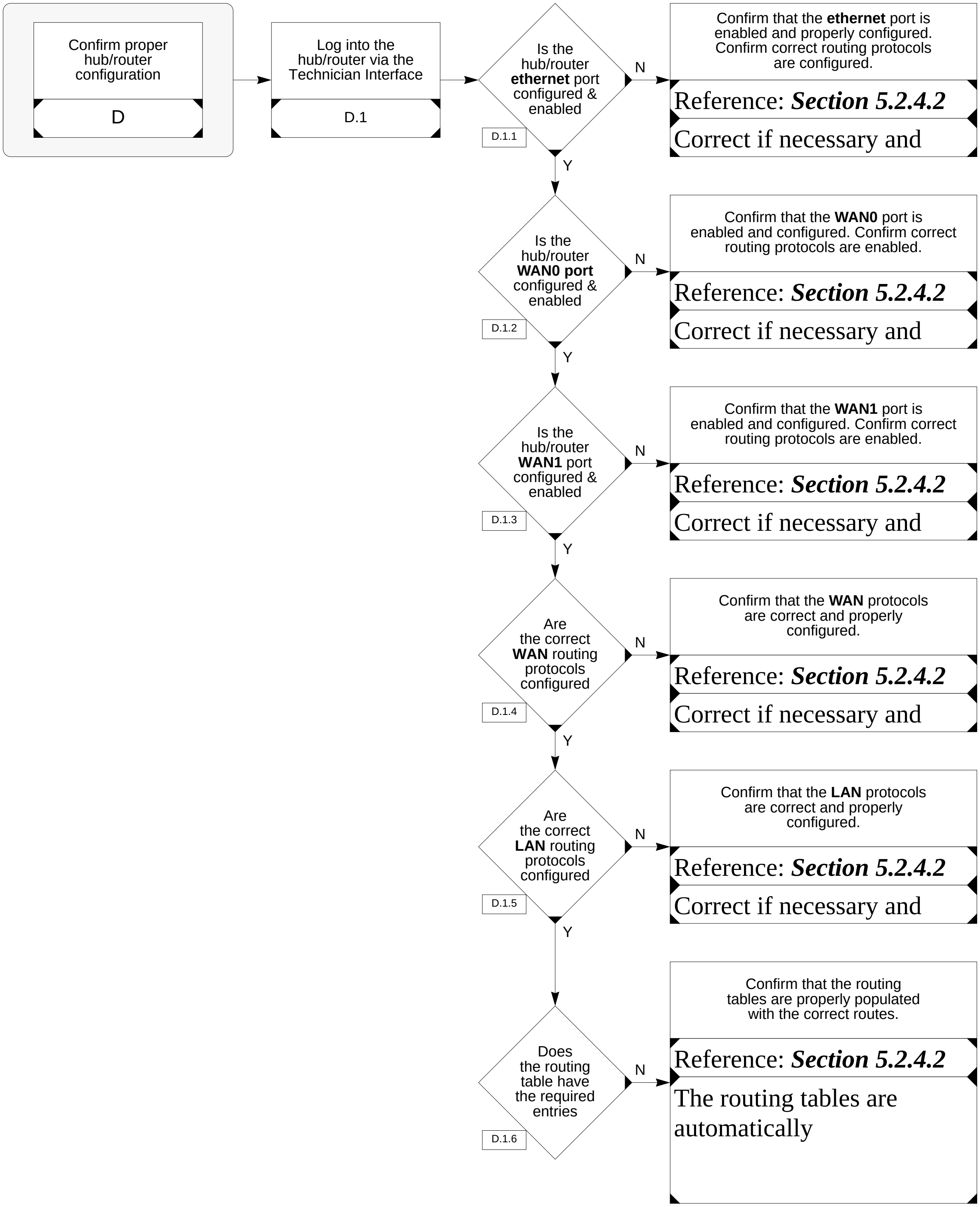
C.1.7

Flow Chart C (cont) Execute Workstation-Based Ethernet Interface Diagnostics (Remote W/S)


This procedure is continued from the previous page from step
C.1.6
Can the **remote** hub/router **ethernet** IP address be ***pinged***
C.1.7
N
Y
If the **remote** hub/router **ethernet** port cannot be ***pinged***, the remote hub/router may be incorrectly configured or have a hardware fault.
Reference: ***Section 5.1.1.6***
Continue with Step D.1
Can the **remote** node's ethernet IP address be ***pinged***
C.1.8
N
Y
If the **remote** node's ethernet port cannot be ***pinged***, the ethernet cable between the hub and workstation may be defective, or there are h/w faults or configuration errors.
Reference: ***Section 5.1.1.6***
Validate cable and h/w
Can a ***telnet*** session be established with the remote node
C.1.9
N
Y
If a ***telnet*** session is not possible, there are high-level application errors, or link stability issues. If the WAN link is confirmed good, the remote w/s operating system is corrupt.
Reference: ***none***
Reload remote w/s operating
The Remote Access software has been improperly installed or misconfigured. Escalate problem to NT Customer Support.


Execute ***landiag*** utility
C.2
Does the **lan0** ethernet card pass the ***landiag*** tests
C.2.1
Y
N
The workstation **lan0** ethernet interface card has no hardware faults. Confirm that the configuration is correct and check for cable faults and router issues.
Reference: ***Section 5.1.2.1***
Continue with Step D.1
Perform a ***loopback*** test
C.3
Does the **lan0** ethernet card pass the ***loopback*** tests
C.3.1
N
Y
The ethernet interface card or MAU is defective and should be replaced.
Reference: ***Section 6.3.5 on***
Replace defective
The workstation ethernet interface and MAU are okay. The fault is external or the workstation ethernet i/f is misconfigured.

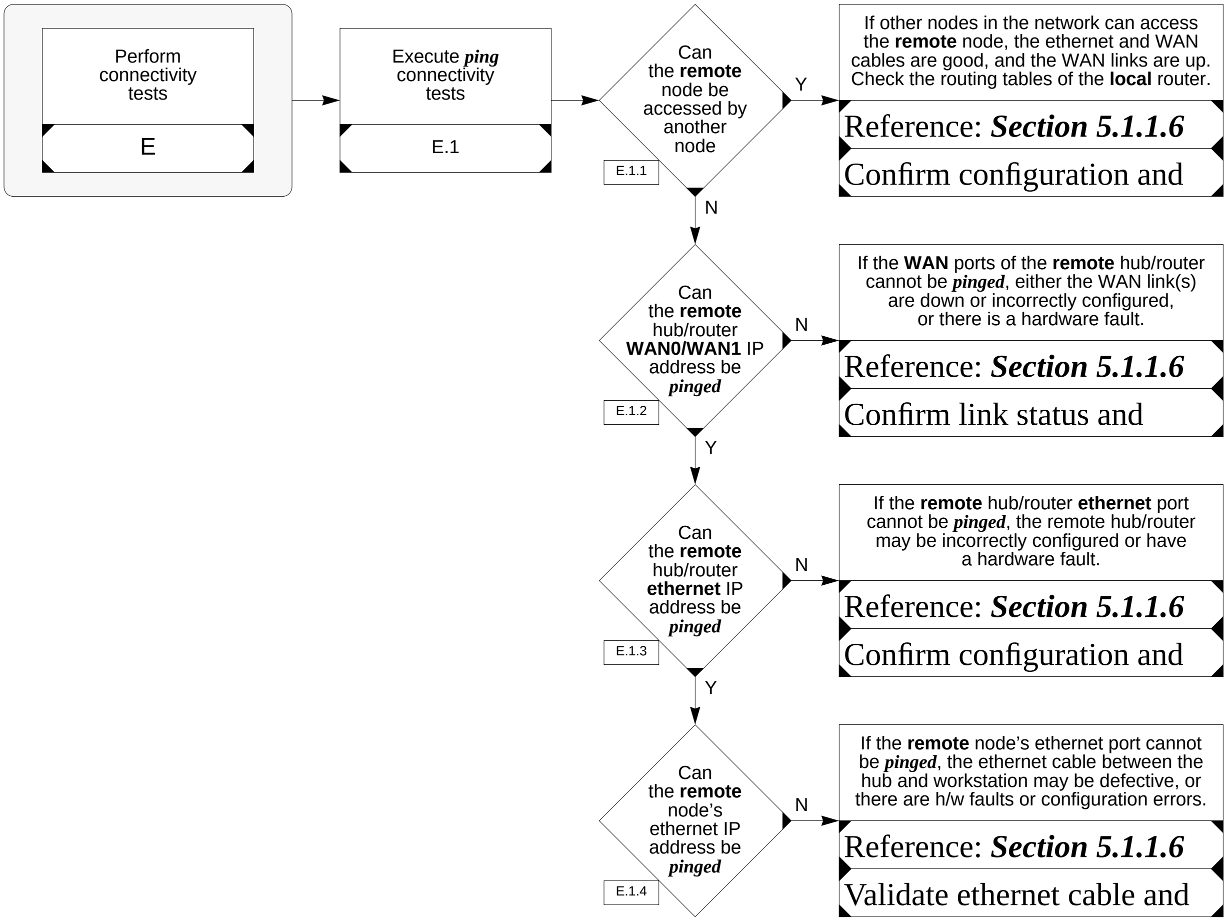
Flow Chart D Confirm Proper Hub/Router Configuration


Confirm proper hub/router configuration
D
Log into the hub/router via the Technician Interface
D.1
Is the hub/router ethernet port configured & enabled
D.1.1
N
Y
Confirm that the ethernet port is enabled and properly configured. Confirm correct routing protocols are configured.
Reference: Section 5.2.4.2
Correct if necessary and
Is the hub/router WAN0 port configured & enabled
D.1.2
N
Y
Confirm that the WAN0 port is enabled and configured. Confirm correct routing protocols are enabled.
Reference: Section 5.2.4.2
Correct if necessary and
Is the hub/router WAN1 port configured & enabled
D.1.3
N
Y
Confirm that the WAN1 port is enabled and configured. Confirm correct routing protocols are enabled.
Reference: Section 5.2.4.2
Correct if necessary and
Are the correct WAN routing protocols configured
D.1.4
N
Y
Confirm that the WAN protocols are correct and properly configured.
Reference: Section 5.2.4.2
Correct if necessary and
Are the correct LAN routing protocols configured
D.1.5
N
Y
Confirm that the LAN protocols are correct and properly configured.
Reference: Section 5.2.4.2
Correct if necessary and
Does the routing table have the required entries
D.1.6
N
Confirm that the routing tables are properly populated with the correct routes.
Reference: Section 5.2.4.2
The routing tables are automatically

6.3.2 Remote Access or ADAS Workstation Cannot Connect to All Nodes

Issue		Troubleshooting Tip
	Remote Access/ADAS Workstation Can Connect to Some Nodes in the Network, but Not to All Nodes ***Assumptions:*** The remote nodes to which connectivity is being attempted are properly configured and accessible by other nodes in the network.	***Observations:*** Since some nodes are accessible to the Remote Access Workstation, the local ethernet and WAN cabling is correct, the workstation ethernet interface is properly configured, the hub/router ethernet interface is properly configured, and the WAN port on the local hub/router is properly configured. • Confirm that the IP addresses of the remote nodes are correct and have been entered correctly. Also confirm the remote node names are correct and have been entered correctly. ***Before any in-depth troubleshooting, check for typographical errors!*** • Perform connectivity tests (Reference ***Flow Chart E on page 188***) - execute ***ping*** connectivity tests • Confirm proper configuration of the ethernet hub/router (Reference ***Flow Chart D on page 186***) - check configuration details via ***show*** command - execute ***ping*** connectivity tests

Flow Chart E Perform Connectivity Tests


Perform connectivity tests
E
Execute *ping* connectivity tests
E.1
Can the **remote** node be accessed by another node
E.1.1
Y
N
Can the **remote** hub/router **WAN0/WAN1** IP address be *pinged*
E.1.2
N
Y
Can the **remote** hub/router **ethernet** IP address be *pinged*
E.1.3
N
Y
Can the **remote** node's ethernet IP address be *pinged*
E.1.4
N

If other nodes in the network can access the **remote** node, the ethernet and WAN cables are good, and the WAN links are up. Check the routing tables of the **local** router.

Reference: ***Section 5.1.1.6***

Confirm configuration and

If the **WAN** ports of the **remote** hub/router cannot be *pinged*, either the WAN link(s) are down or incorrectly configured, or there is a hardware fault.

Reference: ***Section 5.1.1.6***

Confirm link status and

If the **remote** hub/router **ethernet** port cannot be *pinged*, the remote hub/router may be incorrectly configured or have a hardware fault.

Reference: ***Section 5.1.1.6***

Confirm configuration and

If the **remote** node's ethernet port cannot be *pinged*, the ethernet cable between the hub and workstation may be defective, or there are h/w faults or configuration errors.

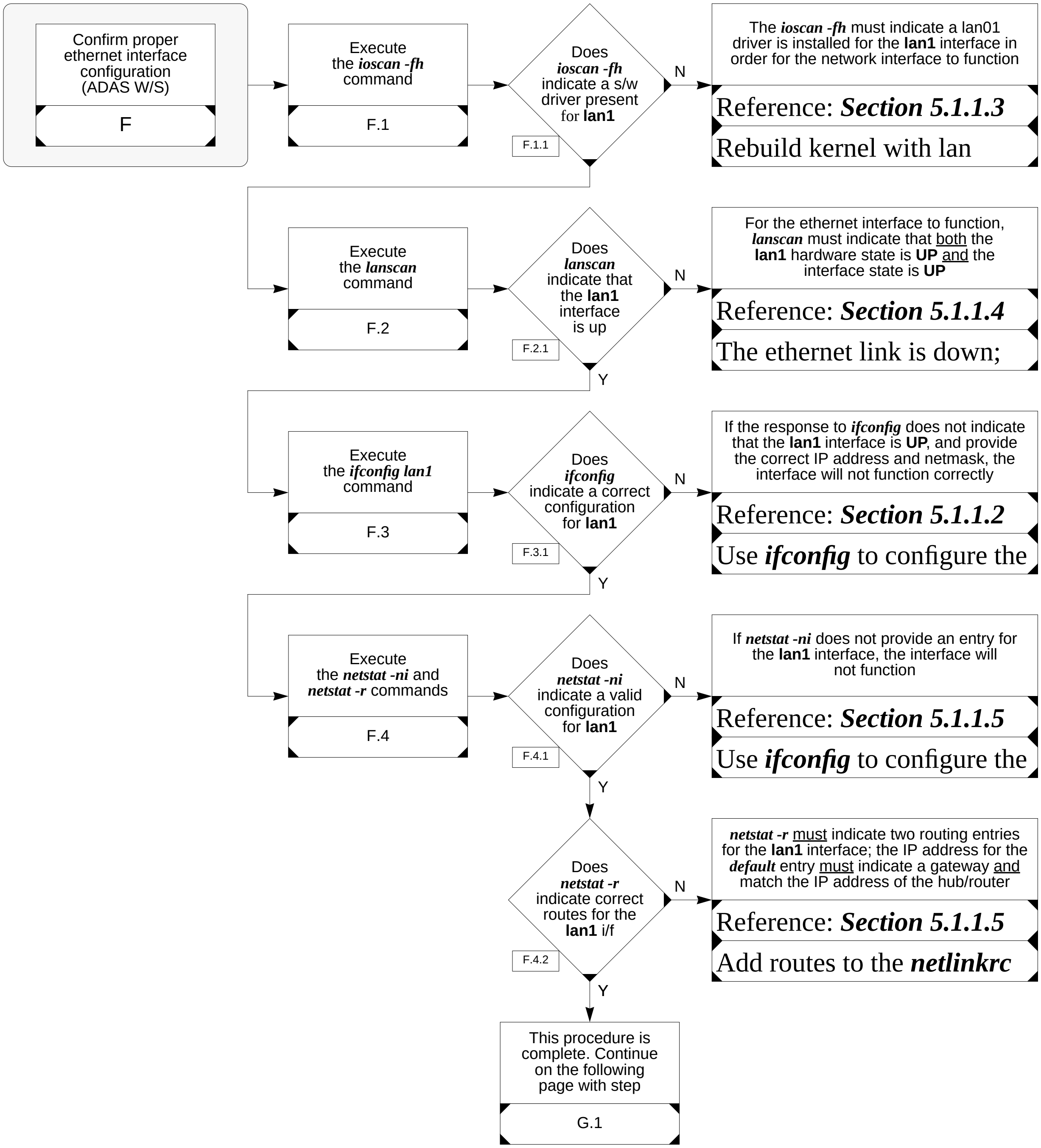
Reference: ***Section 5.1.1.6***

Validate ethernet cable and

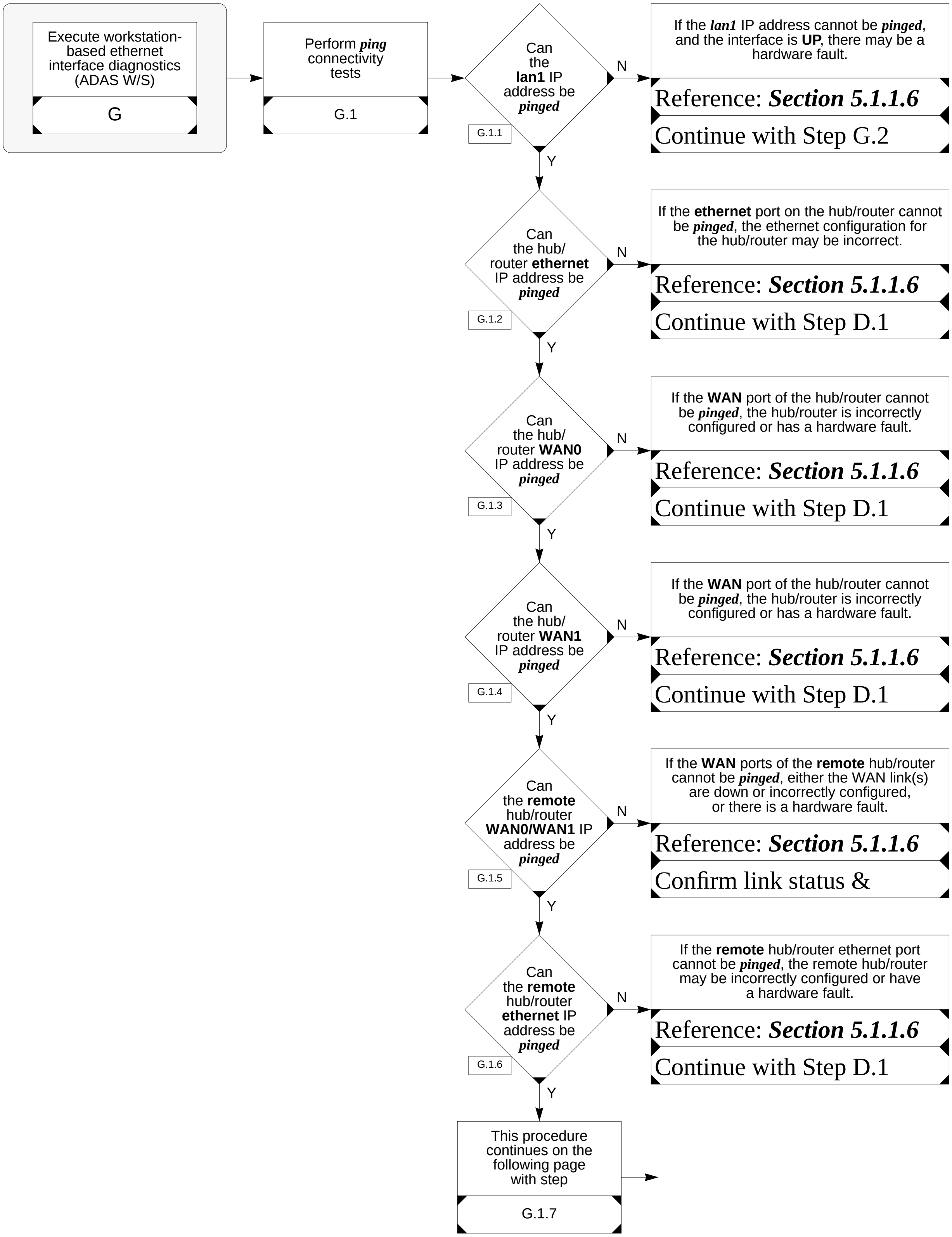
6.3.3 ADAS Workstation Cannot Connect to Any Other Node

Issue		Troubleshooting Tip
	ADAS Workstation Cannot Connect to Any Other Node in the Network ***Assumptions:*** The ADAS Workstation can access nodes (APUs, EIU, & CM) on the DMS Switch.	***Observations:*** The workstation LAN0 ethernet interface is properly configured and functional. However, there could still be workstation configuration issues with the LAN1 interface, hub/router configuration issues, hardware failures, or cabling faults. • Confirm that the IP addresses of the remote nodes are correct and have been entered correctly. Also confirm the remote node names are correct and have been entered correctly. ***Before any in-depth troubleshooting, check for typographical errors!*** • Confirm proper ethernet cable connectivity (Reference ***Flow Chart A on page 181***) - check workstation MAU status indicators - check workstation MAU switch settings - check ethernet hub/router status indicators - check ethernet cabling between workstation MAU and ethernet hub/router • Confirm proper configuration of the ethernet interface card in the workstation (Reference ***Flow Chart F on page 190***) - execute ***ioscan*** command - execute ***lanscan*** command - execute ***ifconfig*** command - execute ***netstat*** command • Perform workstation-based ethernet interface diagnostics (Reference ***Flow Chart G on page 191***) - execute ***ping*** connectivity tests - execute ***landiag*** utility - perform ***loopback*** test • Confirm proper configuration of the ethernet hub/router (Reference ***Flow Chart D on page 186***) - check configuration details via ***show*** command - execute ***ping*** connectivity tests

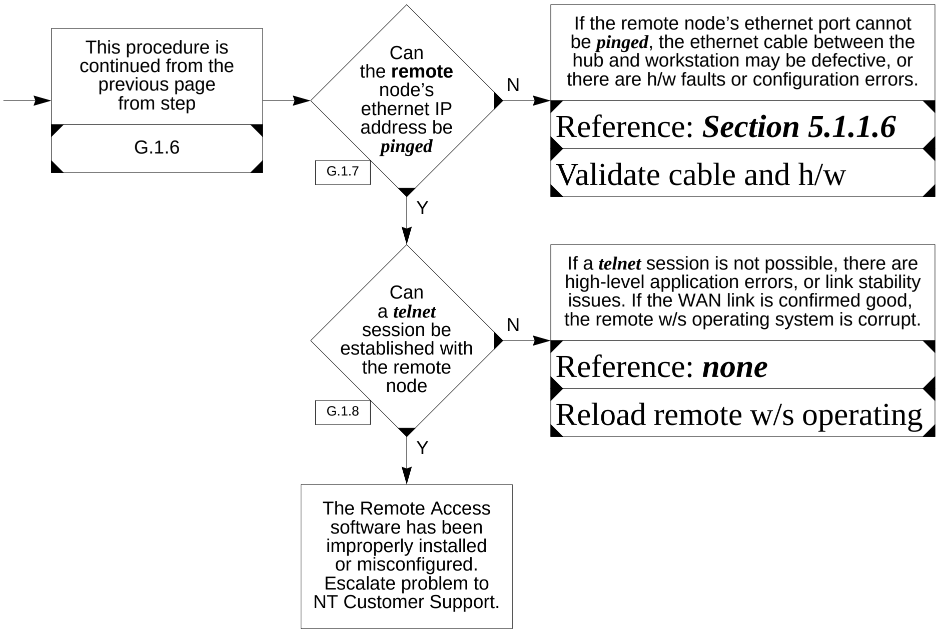
Flow Chart F Confirm Proper Ethernet Interface Configuration (ADAS W/S)

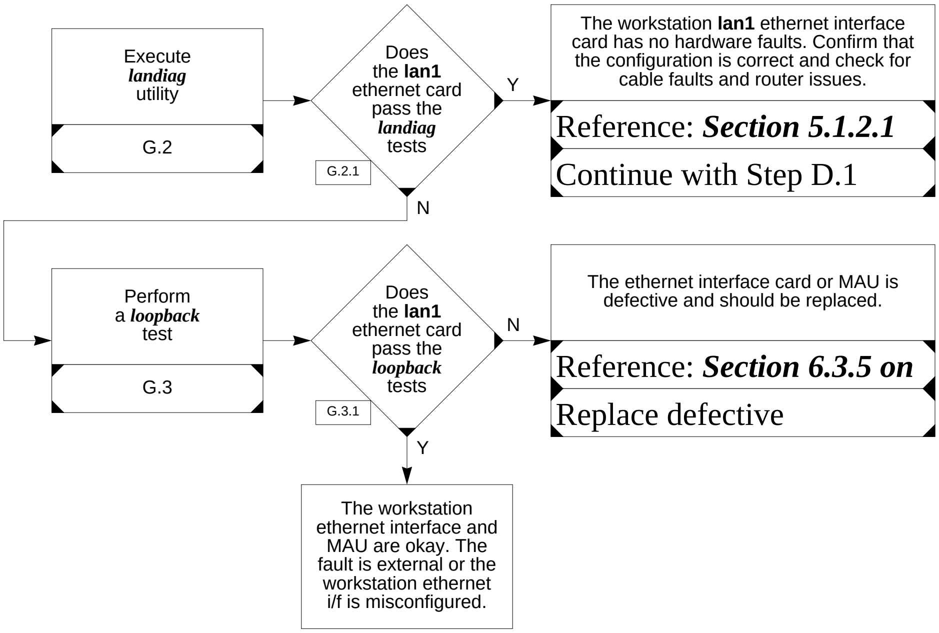

Confirm proper ethernet interface configuration (ADAS W/S)
F
Execute the *ioscan -fh* command
F.1
Does *ioscan -fh* indicate a s/w driver present for **lan1**
F.1.1
N
The *ioscan -fh* must indicate a lan01 driver is installed for the **lan1** interface in order for the network interface to function
Reference: *Section 5.1.1.3*
Rebuild kernel with lan
Execute the *lanscan* command
F.2
Does *lanscan* indicate that the **lan1** interface is up
F.2.1
N
Y
For the ethernet interface to function, *lanscan* must indicate that both the **lan1** hardware state is **UP** and the interface state is **UP**
Reference: *Section 5.1.1.4*
The ethernet link is down;
Execute the *ifconfig lan1* command
F.3
Does *ifconfig* indicate a correct configuration for **lan1**
F.3.1
N
Y
If the response to *ifconfig* does not indicate that the **lan1** interface is **UP**, and provide the correct IP address and netmask, the interface will not function correctly
Reference: *Section 5.1.1.2*
Use *ifconfig* to configure the
Execute the *netstat -ni* and *netstat -r* commands
F.4
Does *netstat -ni* indicate a valid configuration for **lan1**
F.4.1
N
Y
If *netstat -ni* does not provide an entry for the **lan1** interface, the interface will not function
Reference: *Section 5.1.1.5*
Use *ifconfig* to configure the
Does *netstat -r* indicate correct routes for the **lan1** i/f
F.4.2
N
Y
netstat -r must indicate two routing entries for the **lan1** interface; the IP address for the *default* entry must indicate a gateway and match the IP address of the hub/router
Reference: *Section 5.1.1.5*
Add routes to the *netlinkrc*
This procedure is complete. Continue on the following page with step
G.1

Flow Chart G Execute Workstation-Based Ethernet Interface Diagnostics (ADAS W/S)


Execute workstation-based ethernet interface diagnostics (ADAS W/S)
G
Perform *ping* connectivity tests
G.1
Can the **lan1** IP address be *pinged*
G.1.1
N
Y
If the *lan1* IP address cannot be *pinged*, and the interface is **UP**, there may be a hardware fault.
Reference: *Section 5.1.1.6*
Continue with Step G.2
Can the hub/router **ethernet** IP address be *pinged*
G.1.2
N
Y
If the **ethernet** port on the hub/router cannot be *pinged*, the ethernet configuration for the hub/router may be incorrect.
Reference: *Section 5.1.1.6*
Continue with Step D.1
Can the hub/router **WAN0** IP address be *pinged*
G.1.3
N
Y
If the **WAN** port of the hub/router cannot be *pinged*, the hub/router is incorrectly configured or has a hardware fault.
Reference: *Section 5.1.1.6*
Continue with Step D.1
Can the hub/router **WAN1** IP address be *pinged*
G.1.4
N
Y
If the **WAN** port of the hub/router cannot be *pinged*, the hub/router is incorrectly configured or has a hardware fault.
Reference: *Section 5.1.1.6*
Continue with Step D.1
Can the **remote** hub/router **WAN0/WAN1** IP address be *pinged*
G.1.5
N
Y
If the **WAN** ports of the **remote** hub/router cannot be *pinged*, either the WAN link(s) are down or incorrectly configured, or there is a hardware fault.
Reference: *Section 5.1.1.6*
Confirm link status &
Can the **remote** hub/router **ethernet** IP address be *pinged*
G.1.6
N
Y
If the **remote** hub/router ethernet port cannot be *pinged*, the remote hub/router may be incorrectly configured or have a hardware fault.
Reference: *Section 5.1.1.6*
Continue with Step D.1
This procedure continues on the following page with step
G.1.7

Flow Chart G (cont) Execute Workstation-Based Ethernet Interface Diagnostics (ADAS W/S)

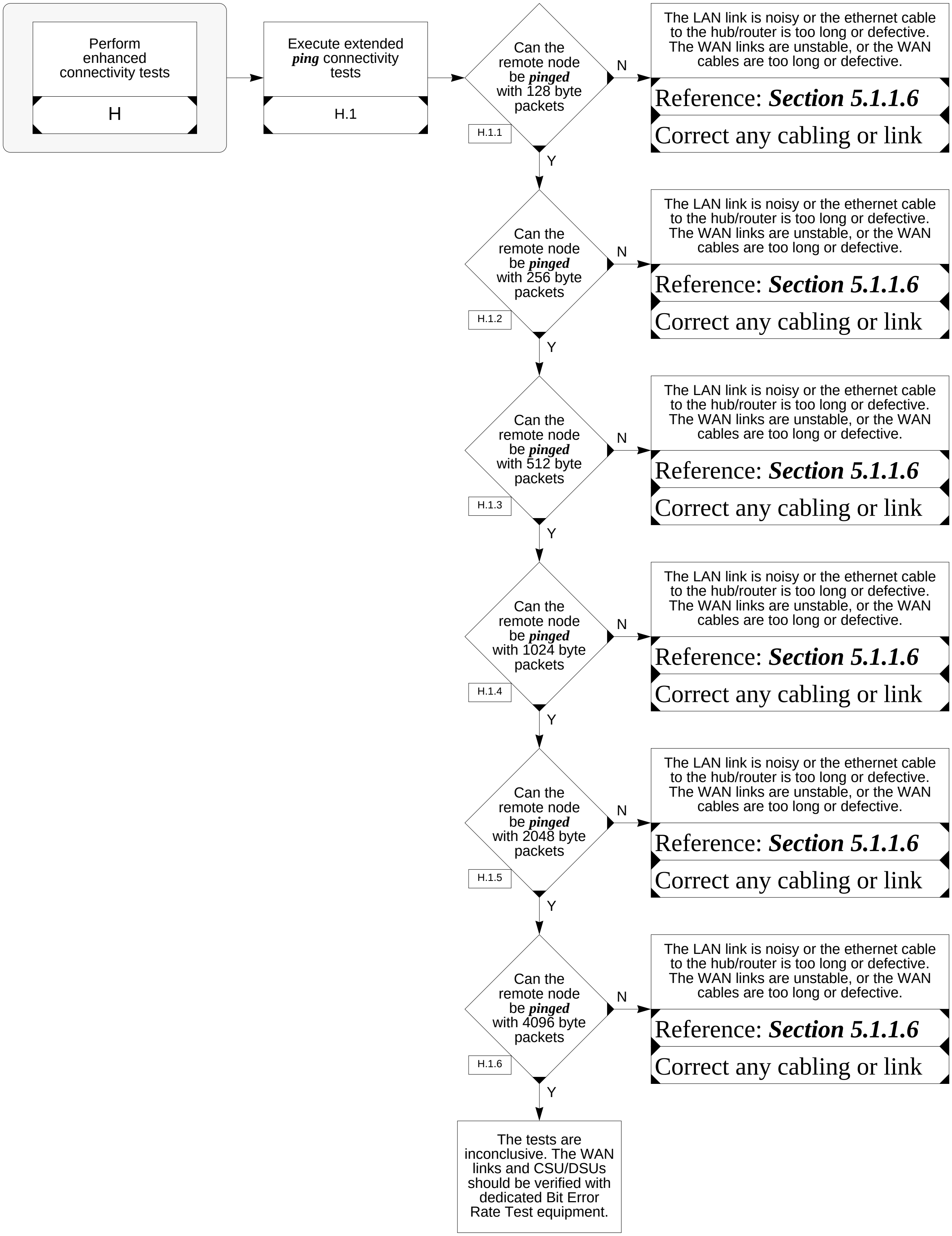

This procedure is continued from the previous page from step
G.1.6
Can the **remote** node's ethernet IP address be ***pinged***
G.1.7
N
If the remote node's ethernet port cannot be ***pinged***, the ethernet cable between the hub and workstation may be defective, or there are h/w faults or configuration errors.
Reference: ***Section 5.1.1.6***
Validate cable and h/w
Y
Can a ***telnet*** session be established with the remote node
G.1.8
N
If a ***telnet*** session is not possible, there are high-level application errors, or link stability issues. If the WAN link is confirmed good, the remote w/s operating system is corrupt.
Reference: ***none***
Reload remote w/s operating
Y
The Remote Access software has been improperly installed or misconfigured. Escalate problem to NT Customer Support.


Execute ***landiag*** utility
G.2
Does the **lan1** ethernet card pass the ***landiag*** tests
G.2.1
Y
The workstation **lan1** ethernet interface card has no hardware faults. Confirm that the configuration is correct and check for cable faults and router issues.
Reference: ***Section 5.1.2.1***
Continue with Step D.1
N
Perform a ***loopback*** test
G.3
Does the **lan1** ethernet card pass the ***loopback*** tests
G.3.1
N
The ethernet interface card or MAU is defective and should be replaced.
Reference: ***Section 6.3.5 on***
Replace defective
Y
The workstation ethernet interface and MAU are okay. The fault is external or the workstation ethernet i/f is misconfigured.

6.3.4 Remote Access Connectivity Appears Intermittent

Issue		Troubleshooting Tip
	Remote Access Connectivity Appears to be Intermittent or Connectivity is Dropped on Some Applications ***Assumptions:*** Connections to remote nodes can readily be established, but are dropped when certain applications are executed.	***Observations:*** Since only certain applications experience problems, the configuration of all the workstations and routing equipment is probably correct. • Perform enhanced connectivity tests (Reference ***Flow Chart H on page 194***) - execute extended ***ping*** connectivity tests

Flow Chart H   Perform Enhanced Connectivity Tests


Perform enhanced connectivity tests
H
Execute extended *ping* connectivity tests
H.1
Can the remote node be ***pinged*** with 128 byte packets
H.1.1
N
Y
The LAN link is noisy or the ethernet cable to the hub/router is too long or defective. The WAN links are unstable, or the WAN cables are too long or defective.
Reference: ***Section 5.1.1.6***
Correct any cabling or link
Can the remote node be ***pinged*** with 256 byte packets
H.1.2
N
Y
The LAN link is noisy or the ethernet cable to the hub/router is too long or defective. The WAN links are unstable, or the WAN cables are too long or defective.
Reference: ***Section 5.1.1.6***
Correct any cabling or link
Can the remote node be ***pinged*** with 512 byte packets
H.1.3
N
Y
The LAN link is noisy or the ethernet cable to the hub/router is too long or defective. The WAN links are unstable, or the WAN cables are too long or defective.
Reference: ***Section 5.1.1.6***
Correct any cabling or link
Can the remote node be ***pinged*** with 1024 byte packets
H.1.4
N
Y
The LAN link is noisy or the ethernet cable to the hub/router is too long or defective. The WAN links are unstable, or the WAN cables are too long or defective.
Reference: ***Section 5.1.1.6***
Correct any cabling or link
Can the remote node be ***pinged*** with 2048 byte packets
H.1.5
N
Y
The LAN link is noisy or the ethernet cable to the hub/router is too long or defective. The WAN links are unstable, or the WAN cables are too long or defective.
Reference: ***Section 5.1.1.6***
Correct any cabling or link
Can the remote node be ***pinged*** with 4096 byte packets
H.1.6
N
Y
The LAN link is noisy or the ethernet cable to the hub/router is too long or defective. The WAN links are unstable, or the WAN cables are too long or defective.
Reference: ***Section 5.1.1.6***
Correct any cabling or link
The tests are inconclusive. The WAN links and CSU/DSUs should be verified with dedicated Bit Error Rate Test equipment.

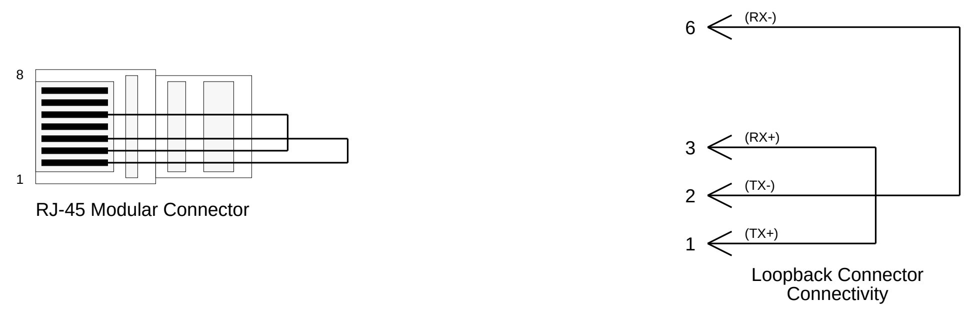
6.3.5 Workstation Loopback Testing

Troubleshooting problems in a networked system is often very tedious, as it can be very difficult to isolate the source of the failure. Is the failure in the interface hardware or in the link interconnecting the two nodes? Is is the interface configuration, or the router configuration?

Recognizing the difficulties in isolating faults in a networked system, the workstations employed in the ADAS system provide a convenient means of providing positive go/no-go confirmation of the LAN interface hardware. By following a simple diagnostic test procedure, the user can positively eliminate, or identify, the LAN interface card as the source of the problem in a failed system.

The diagnostic routine employed for identifying hardware faults with the LAN interface card in the workstation uses the ***landiag*** utility and an external loopback connector. The loopback connector is available from Hewlett-Packard as HP part number 5061-4977. Alternatively, the customer may fabricate their own loopback connector using two short pieces of wire and a single RJ-45 modular connector, as illustrated in the figure below.

Figure 28 Ethernet Loopback Connector



The requirements for performing the loopback test are listed below:

- Remote Access or ADAS Workstation
- HP EtherTwist MAU (HP part number 28685B)
- Loopback connector (HP part number 5061-4977, or customer provided)

The following steps detail how to perform a loopback test:

1. Disconnect the lan cable from the MAU
2. Enable both the MAU Loopback Test switch and the SQE Test switch (Reference ***Section 5.3.1 on page 173***)--note that the MAU ***Link*** LED will turn **ON**

3. Attach the loopback connector to the MAU
4. Open an ***xterm*** window on the workstation
5. Perform a loopback test by entering the ***landiag*** command in the ***xterm*** window as shown below (Reference ***Section 5.1.2.1 on page 147***):

   ```
   /usr/bin/landiag
   ```

   When the *"Test Selection mode."* menu is displayed, enter

   ```
   lan
   ```

   The default lan interface is ***/dev/lan0***. If the loopback test is being performed on the second lan interface, enter

   name

   and then enter

   ```
   /dev/lan1
   ```

   when prompted to enter the LAN Interface device file. Once the desired lan interface has been selected, execute the lan interface selftest by entering

   reset

   The workstation should provide the following response to the ***reset*** command:

   Resetting LAN Interface to run selftest.

   and then display the previous menu.

   Any response other than the above, such as

   ```
   Unable to reset LAN Interface.
   errno = 6
   ```

   indicates a **failure** of the loopback test. In this case, either the MAU is defective, or the ethernet interface card has a hardware fault.[1]

   To terminate the ***landiag*** utility, enter

   quit

   which will exit the diagnostic utility to the shell (***xterm*** window).

[1] It is not be possible to execute the ***reset*** command on a lan interface card that is not in the **UP** state (Reference ***Section 5.1.1.4 on page 135***). Attempting to execute the ***reset*** command on a lan card that is in the **DOWN** state will always result in the response *"Unable to reset LAN Interface. errno = 6"*. If the workstation was powered up without a valid lan connection to the ethernet interface card, the interface will remain in the **DOWN** state until the workstation is rebooted with a valid ethernet link. To circumvent this problem and permit testing of the ethernet interface card using the ***landiag*** utility, the workstation can be rebooted with the loopback connector installed and the MAU switches set as detailed above. If both the MAU and ethernet card are good, the lan interface will then come in-service in the **UP** state and the loopback tests can proceed. (At this point, however, these tests will not be necessary, as the lan interface selftest is performed when the workstation reboots and the fact that the interface card comes up in the **UP** state indicates that the hardware is okay.)

6. In the case of a failure of the loopback test, the MAU should first be replaced with a known good unit and the above test re-executed. If the loopback test now passes, the old MAU is defective and should be disposed. If the loopback test still fails, the ethernet interface card should be replaced.
7. Once loopback testing is complete, return the MAU switches to the normal positions, i.e., SQE Test (Enable), Link Beat (Enable), Loopback Test (Disable), and Long Cable (Disable).

6.4 Know Problems and Workarounds

6.4.1 Incorrect Link Status Indication from the Remote Access Dialog Box

There is a known problem with the Remote Access feature relating to the status of a remote access connection as displayed by the Remote Access dialog box. In some instances, when the status of the connection is queried by selecting ***"Status"*** under the **Connection** Menu, Remote Access will respond with a dialog box stating "You are currently connected to the ***xxx*** site." when in fact, the connection has not been made. This is an inter-process communications issue between the various routines of the Remote Access feature.

If an attempt to connect to a remote site does not appear to be successful, and the user wants to ascertain the status of the connection, the following steps should be taken to confirm the state of the connection attempt.

- First, are the ADAS icons displayed across the top of the screen? A successful remote access connection will present the same ADAS screen to the user as does the ADAS Workstation. The five ADAS icons should appear across the top of the screen, the current call scenario should be displayed, and an xterm window should open. If these actions do not occur, then the remote access attempt was not successful, regardless of what is displayed in the link status dialog box.
- If the user invoked Remote Access from an xterm window by typing ***Remote Access***, then the user should monitor the xterm window from which the Remote Access was launched for information regarding the status of the remote connection. A few examples will illustrate the response a user should expect when invoking Remote Access.

  Remote Access is invoked from an xterm window in the following manner:

  > **Remote Access** <*return*> (This is the command invoking Remote Access. It is entered from an xterm window.)

When the above command is executed, a GUI-based interface is opened on the screen to allow a user to connect to a remote site. When the user selects a remote site from the site list and selects the ‘*Connect*’ action, the following text is displayed in the same xterm window in which the above **Remote Access** command was issued.

```
Password (192.168.2.1:admin):
WARNING: You are Admin!
:Can’t assign requested address
/iws/usm/usmstup - Error, unknown HP-UX version
(A.09.03)
```

The above response represents a **successful** connection attempt. The last two messages are interprocess status messages and can be ignored.

Unsuccessful attempts to connect to a remote site will result in error messages similar to the following when the *‘Connect’* action is selected from the Remote Access dialog box. These messages will appear in the same xterm window in which the **Remote Access** command was issued. Also note, due to network latencies and time-out values, it may take a couple of minutes for some of the following status messages to be displayed in the xterm window.

```
rexec: connect: 191.173.1.2: Network is
unreachable
```

The above response indicates that the Remote Access Workstation does not know how to reach the node at the specified IP address. This is probably not a network problem, but rather an issue with the Remote Access Workstation configuration. The problem here is that the Remote Access Workstation does not have a ***route*** entry to detail how to access this node. The user should confirm that the IP address has been entered correctly for the remote site to which access is being attempted.

```
rexec: connect: 192.168.2.3: No route to host
```

The above response indicates that the IP address provided is a valid address for a node to be connected to the Remote Access Network, but there are no routing table entries in any of the network equipment routing tables or host routing tables. The user should confirm that the IP address is correct and that the remote node using this IP address is actually connected to the Remote Access Network.

```
rexec: connect: 192.168.2.1: Connection timed out
```

The above response indicates that the remote connection attempt timed out at the Remote Access Workstation due to no response from the remote site. This would occur due to any of the following reasons:

1. the LAN connection to the Remote Access Workstation is down

2. the LAN is not connected to the Remote Access Workstation
3. the LAN connection to the remote site is down
4. the ADAS Workstation at the remote site is down
5. or any other network problem that would prevent the two nodes from establishing a two-way connection

For any of the above cases, the troubleshooting guide in the previous section should be used to identify and isolate the problem.

Note: The above messages will only be displayed if the Remote Access program was invoked from an xterm window by typing ***Remote Access***. However, the Remote Access Workstation is configured to bring up the Remote Access dialog box upon login by the user. In this case, the above error messages will not be displayed since Remote Access is not executed from an xterm. If the user desires to see the above ***rexec*** status messages, Remote Access can be exited, and then re-launched from an xterm window, which will then display the status messages in the xterm window.

If none of the above responses are provided in the xterm window in which the ***Remote Access*** command was issued (or if Remote Access was not invoked from an xterm) and the link status still incorrectly indicates a connection is up, the following **unix** command can be issued to query if the appropriate processes are running.

- From a different xterm window from which the ***Remote Access*** command was issued, enter the following **unix** command:

  ps -eaf | grep rexec

  The above command will result in a response similar to the following for a **successful** remote access connection.

  ```
  root 703 699 0 08:13:23 ttyp7 0:00 rexec
  192.168.2.1 -l admin /iws/asm/startenet.text
  192.168.1

  root 704 703 0 08:13:34 ttyp7 0:00 rexec
  192.168.2.1 -l admin /iws/asm/startenet.text
  192.168.1

  root 723 695 0 08:15:49 ttyp7 0:00 grep rexec
  ```

  Most of the fields in the above response will be different when executed by a user. However, what the user should look for in the above response is the **two** lines containing the ***rexec*** process. These **two** lines will have the following form:

```
<immaterial fields> rexec <IP address> -l admin /
iws/asm/startenet.text <network address>
```

The third line in the above response is immaterial and can be ignored.

In the case of an **unsuccessful** remote connection attempt, issuing the

ps -eaf | grep rexec

command will result in a response similar to the following:

```
root 723 695 0 08:15:49 ttyp7 0:00 grep rexec
```

This response indicates that the two required ***rexec*** processes are not running and the connection attempt to the remote site was **not successful**.

7.0 Glossary

7.1

ADAS	*Automated Directory Assistance Service*
APU	*Application Processor Unit*
ARP	*Address Resolution Protocol*
AUI	*Attachment Unit Interface*
CM	*Compute Module*
CSU	*Channel Service Unit*
DDS	*Digital Data Service*
DSU	*Data Service Unit*
DS0	*64 kb/s (56 kb/s) Timeslot (channel) of a T1 Span*
EIU	*Ethernet Interface Unit*
FDDI	*Fiber Distributed Data Interface*
FTP	*File Transfer Protocol*
FT1	*Fractional T1*
GUI	*Graphical User Interface*
HDLC	*High-level Data Link Control*
ICMP	*Internet Control Message Protocol*
IEEE	*Institute of Electrical and Electronic Engineers*
IP	*Internet Protocol*
ISO	*International Standards Organization*
LAN	*Local Area Network*
MAC	*Media Access Control*
MAP	*Maintenance Administration Position*
MAU	*Media Attachment Unit*
MDF	*Main Distribution Panel*
MIB	*Management Information Base*
OAM	*Operations, Administration, & Maintenance*
OSI	*Open System Interconnection*
PPP	*Point-to-Point Protocol*
RARP	*Reverse Address Resolution Protocol*
RARPD	*Reverse Address Resolution Protocol Daemon*
SAM	*System Administration Manager*
SMTP	*Simple Mail Transfer Protocol*

SNMP	*Simple Network Management Protocol*
TCP	*Transmission Control Protocol*
TFTP	*Trivial File Transfer Protocol*
TOPS	*Traffic Operator Position System*
T1	*1.544 Mb/s (1.536 Mb/s) Framed (24 channels) WAN Link*
UDP	*User Datagram Protocol*
WAN	*Wide Area Network*

8.0 Appendix A: Cable Diagrams

The following figures provide the generic design details for the cables required by the LAN/WAN network equipment that has been validated by Northern Telecom for the Remote Access product.

Note: Networking equipment other than that specified in this document will require cables different than those detailed herein. ***It is the customer's responsibility to provide all cables required for the Remote Access product, regardless of whether or not the networking equipment actually used is that validated by Northern Telecom.***

The following cable diagrams are provided for informational purposes only. Customers wishing to fabricate these cables should consult the formal cable design/drawing documents available from Northern Telecom.

Specifications are provided for the following cables:

1. ***NT9X8511*** *Ethernet Cable*
2. ***NTGB0182*** *T1 Cable*
3. ***NTNX36SE*** TIA-530 High Speed WAN Cable
4. ***NTNX36SG*** TIA-530 Low Speed WAN Cable
5. ***NTNX36SK*** RS-422 Low Speed WAN Cable

Note: The following cable drawings are provided as a courtesy only. This information is intended to help the customer understand the connectivity between the routing equipment used in the Remote Access product, and to assist the customer with the fabrication/procurement of these cables. ***Northern Telecom does not supply any of the above cables, nor any other cables that may be required for the Remote Access product.***

Figure 29 NT9X8511 Ethernet Cable

NOTES

1. ALL DIMENSIONS ARE IN INCHES,UNLESS OTHERWISE SPECIFIED.
2. TO CONVERT FEET TO MILLIMETERS, USE 1 INCH = 25.40 MILLIMETERS.
3. () INDICATES DESIGNATIONS TO BE STAMPED IN ACCORDANCE WITH JOB INFORMATION.
4. [] INDICATES DESIGNATIONS SHOWN FOR INFORMATION ONLY AND ARE NOT TO BE STAMPED.
5. <> INDICATES DESIGNATIONS WHICH ARE PROVIDED IN ACCORDANCE WITH OTHER INFORMATION.
6. PRODUCT INFORMATION CODE IDENTIFIER AND RELEASE NUMBER SHALL BE MARKED ON LABEL WITH .10 INCH HIGH CHARACTERS. REFER TO HB0X0102, FIGURE 28B, FOR CABLE IDENTIFICATION MARKINGS.
7. CABLE ASSEMBLY SHALL BE TESTED FOR SHORTS AND CONTINUITY.
8. THIS CABLE ASSEMBLY HAS LENGTH RESTRICTION OF 1 METER MINIMUM AND 100 METERS MAXIMUM.
9. INFORMATION IN PARENTHESES ON SHEET 2 SHALL BE MARKED PER HANDBOOK, HB0X0102, FIGURE 28B AND LOCATE AS SHOWN.
10. TO INSURE PROPER STRAIN RELEIF, A MINIMUM OF .30 INCHES OF THE CABLE JACKETING SHALL BE INSIDE OF THE RJ-45 CONNECTOR HOUSING.

WIRING TABLE

FROM MAU						TO HUB						WIRE/CABLE DATA		
TERM. DESIG.	SHELF POS.	CP/ CONN	PIN	COLOR	TERMINAL CPC	TERM. DESIG.	SHELF POS.	CP/ CONN	PIN	COLOR	TERMINAL CPC	CABLE CPC	NOTE	LENGTH
TX+		RJ-45	1	WIO		RX+		RJ-45	1	WIO			8	JOB ENG
TX-			2	OIW		RX-			2	OIW				
RX+			3	WIG		TX+			3	WIG				
RX-			6	GIW		TX-			6	GIW				
SPARE			4	BLIW		SPARE			4	BLIW			SPARE	
			5	WIBL					5	WIBL				
			7	WIBR					7	WIBR				
			8	BRIW					8	BRIW				

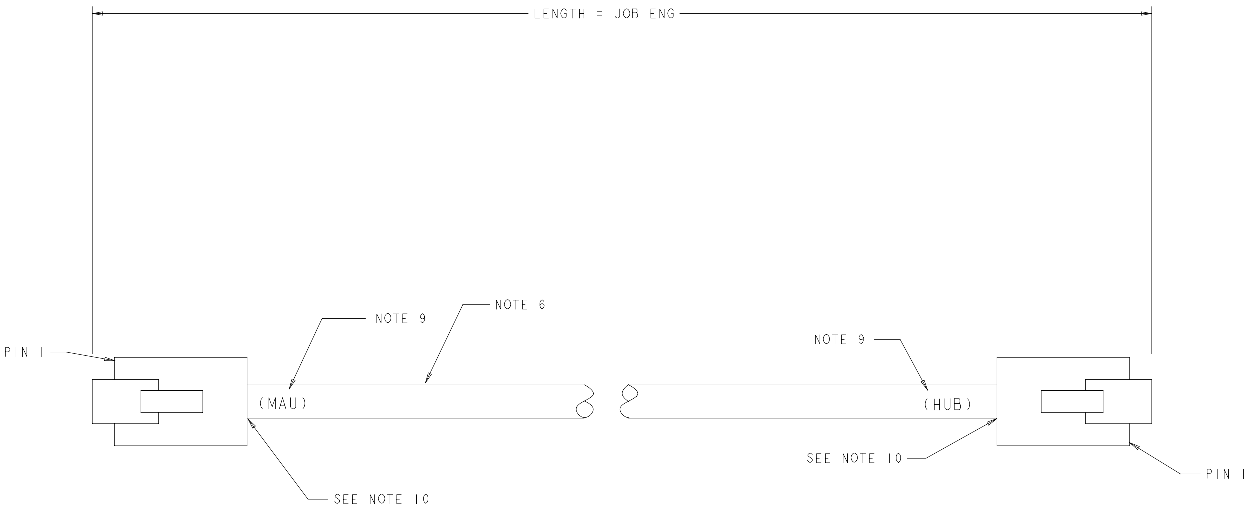
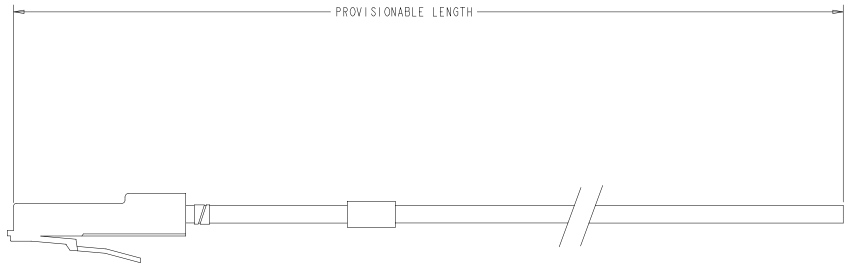


Figure 30 NTGB0182 T1 Cable

EC # 006-28495
96-09-30

JAT JAT 01 01

NOTES:
1. UNLESS OTHERWISE SPECIFIED, ALL DIMENSIONS ARE IN INCHES.
2. TO CONVERT INCHES TO MILLIMETRES USE 1 INCH = 25.40 MILLIMETRES.
3. UNASSIGNED
4. STAMP PRODUCT ENGINEERING CODE AND RELEASE NUMBER ON THE LABEL USING BLACK CHARACTERS, 0.10 HIGH, PER SPEC 91256, METHOD 3.
5. THE CONNECTOR AND STRAIN RELIEF SHOULD BE ASSEMBLED ACCORDING TO CONNECTOR MANUFACTURER'S INSTRUCTION.
6. THE LEFT END OF THE CABLE IS PREPARED AS SHOWN IN FIG A.
7. ALL CABLES SHOULD BE 100% TESTED FOR SHORTS AND CONTINUITY PER TEST SPEC XSIX2345.



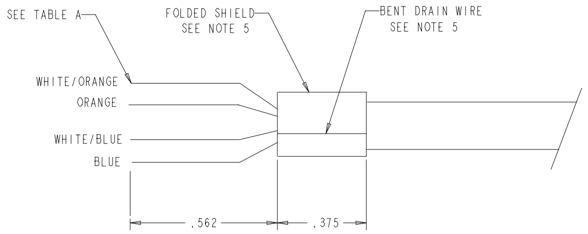


FIG A

METHOD OF PREPARING LEFT
END OF THE CABLE
SCALE NONE

TABLE A	
COLOR	PIN (CONNECTOR)
BLUE	1
WHITE/BLUE	2
NOT CONNECTED	3
ORANGE	4
WHITE/ORANGE	5
NOT CONNECTED	6
NOT CONNECTED	7
NOT CONNECTED	8

DESIGN AUTHORITY
NORTHERN TELECOM

DSI CABLE ASSY
RJ-48C CONN

NTGB0182 NTS

1

Figure 31 NTNX36SE TIA-530 High Speed WAN Cable

MANUFACTURING NOTES:

1. UNLESS OTHERWISE SPECIFIED ALL DIMENSIONS ARE IN INCHES.
2. TO CONVERT INCHES TO MILLIMETERS USE 1 INCH = 25.40 MILLIMETERS.
3. ( ) INDICATES DESIGNATIONS TO BE STAMPED IN ACCORDANCE WITH OTHER INFORMATION.
4. [ ] INDICATES DESIGNATIONS SHOWN FOR INFORMATION ONLY AND ARE NOT TO BE STAMPED.
5. <> INDICATES DESIGNATIONS WHICH ARE PROVIDED IN ACCORDANCE WITH OTHER INFORMATION.
6. REFER TO HANDBOOK HB0X0102; FIGURE 28B FOR CABLE IDENTIFICATION MARKINGS.

WIRING TABLE A

PIN	COLOR	SIGNAL	PIN	COLOR	SIGNAL	PIN	COLOR	SIGNAL
1	BK(BL)	SHIELD	16	BL(W)	SEND DATA -	31		
2	W(BL)	SEND DATA +	17	O(W)	RECEIVE DATA -	32		
3	W(O)	RECEIVE DATA +	18	G(W)	REQUEST TO SEND -	33		
4	W(G)	REQUEST TO SEND +	19	BR(W)	CLEAR TO SEND -	34		
5	W(BR)	CLEAR TO SEND +	20	SL(W)	DATA SET READY -	35		
6	W(SL)	DATA SET READY +	21			36		
7			22	BL(R)	DATA TERMINAL READY -	37		
8	R(BL)	DATA TERMINAL READY +	23	O(R)	DATA CARRIER DETECT -	38		
9	R(O)	DATA CARRIER DETECT +	24	G(R)	SEND TIMING -	39		
10	R(G)	SEND TIMING +	25	BR(R)	RECEIVE TIMING +	40		
11	R(BR)	RECEIVE TIMING +	26	SL(R)	TRANSMITTER TIMING -	41	BL(BK)	SIGNAL GROUND
12	R(SL)	TRANSMITTER TIMING +	27			42		
13			28			43		
14			29			44		
15			30					

JUMPER 3 PLACES

WIRING TABLE B

PIN	COLOR	SIGNAL	PIN	COLOR	SIGNAL	PIN	COLOR	SIGNAL	PIN	COLOR	SIGNAL
B	BL(BK)	SIGNAL GROUND				A	BK(BL)	SHIELD			
F	R(O)	DATA CARRIER DETECT +	D	W(BR)	CLEAR TO SEND +	E	W(SL)	DATA SET READY +	C	W(G)	REQUEST TO SEND +
L			J	SL(W)	DATA SET READY -	K			H	R(BL)	DATA TERMINAL READY +
R	W(O)	RECEIVE DATA +	N			P	W(BL)	SEND DATA +	M		
V	BR(W)	CLEAR TO SEND -	T	O(W)	RECEIVE DATA -	U	G(R)	SEND TIMING -	S	BL(W)	SEND DATA -
Z	BL(R)	DATA TERMINAL READY -	X	R(BR)	RECEIVE TIMING +	Y	G(W)	REQUEST TO SEND -	W	R(SL)	TRANSMITTER TIMING +
DD			BB			CC	SL(R)	TRANSMITTER TIMING -	AA	R(G)	SEND TIMING +
JJ			FF			HH	O(R)	DATA CARRIER DETECT -	EE		
NN			LL			MM	BR(R)	RECEIVE TIMING -	KK		

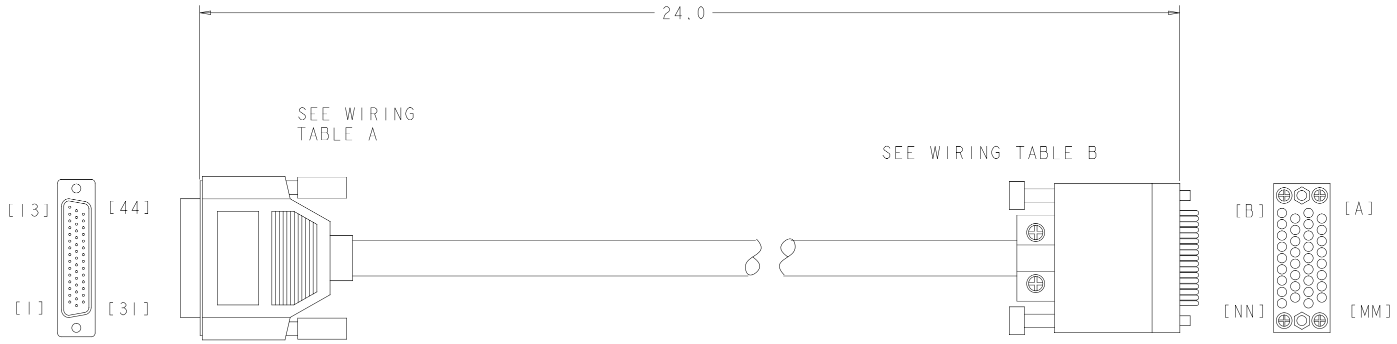


Figure 32 NTNX36SG TIA-530 Low Speed WAN Cable

MANUFACTURING NOTES:

1. UNLESS OTHERWISE SPECIFIED ALL DIMENSIONS ARE IN INCHES.
2. TO CONVERT INCHES TO MILLIMETERS USE 1 INCH = 25.40 MILLIMETERS.
3. ( ) INDICATES DESIGNATIONS TO BE STAMPED IN ACCORDANCE WITH OTHER INFORMATION.
4. [ ] INDICATES DESIGNATIONS SHOWN FOR INFORMATION ONLY AND ARE NOT TO BE STAMPED.
5. <> INDICATES DESIGNATIONS WHICH ARE PROVIDED IN ACCORDANCE WITH OTHER INFORMATION.
6. REFER TO HANDBOOK HB0X0102; FIGURE 28B FOR CABLE IDENTIFICATION MARKINGS.

WIRING TABLE A

PIN	COLOR	SIGNAL	PIN	COLOR	SIGNAL	PIN	COLOR	SIGNAL
1			16	BL(W)	SEND DATA -	31		
2	W(BL)	SEND DATA +	17	O(W)	RECEIVE DATA -	32		
3	W(O)	RECEIVE DATA +	18	G(W)	REQUEST TO SEND -	33		
4	W(G)	REQUEST TO SEND +	19	BR(W)	CLEAR TO SEND -	34		
5	W(BR)	CLEAR TO SEND +	20	S(W)	DATA SET READY -	35		
6	W(S)	DATA SET READY +	21			36		
7			22	BL(R)	TERMINAL READY -	37		
8	R(BL)	TERMINAL READY +	23	O(R)	DATA CARRIER DETECT -	38		
9	R(O)	DATA CARRIER DETECT +	24	G(R)	SEND TIMING -	39		
10	R(G)	SEND TIMING +	25	BR(R)	RECEIVE TIMING -	40		
11	R(BR)	RECEIVE TIMING +	26	S(R)	TERMINAL TIMING -	41	BK(BL)	SIGNAL GROUND
12	R(S)	TERMINAL TIMING +	27			42		
13			28			43		
14			29			44	BL(BK)	SEND COMMON
15			30					

JUMPER
3 PLACES

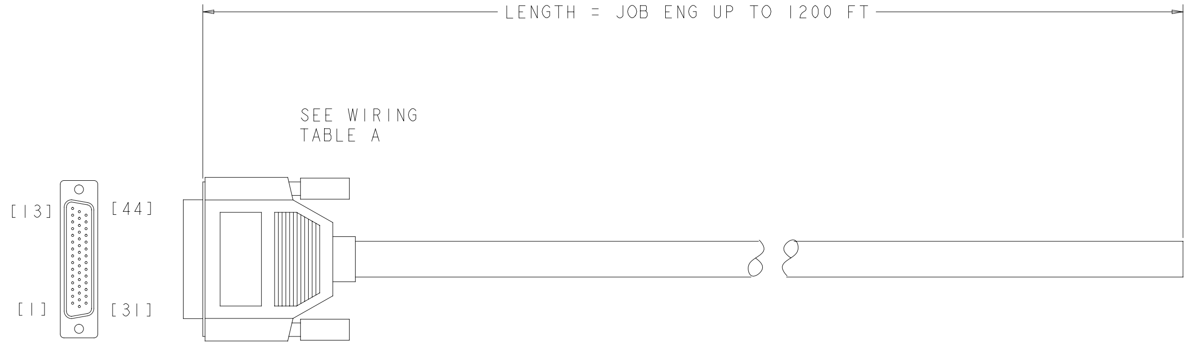


Figure 33 NTNX36SK RS-422 Low Speed WAN Cable

WIRING TABLE A

PIN	COLOR	SIGNAL	PIN	COLOR	SIGNAL
1			14	BL(W)	TRANSMIT DATA B
2	W(BL)	TRANSMIT DATA A	15	R(BR)	TRANSMIT CLOCK A
3	W(O)	RECEIVE DATA A	16	O(W)	RECEIVE DATA B
4	W(G)	RTS A	17	R(O)	RECEIVE CLOCK A
5	W(BR)	CTS A	18		
6	W(SL)	DATA SET READY A	19	G(W)	RTS B
7	R(SL)	SIGNAL GROUND	20		
8	R(BL)	CARRIER DETECT A	21		
9	O(R)	RECEIVE CLOCK B	22	SL(W)	DATA SET READY B
10	BL(R)	CARRIER DETECT B	23		
11	G(R)	TERMINAL TIMING B	24	R(G)	TERMINAL TIMING A
12	BR(R)	TRANSMIT CLOCK B	25		
13	BR(W)	CTS B			

MANUFACTURING NOTES:

1. UNLESS OTHERWISE SPECIFIED ALL DIMENSIONS ARE IN INCHES.

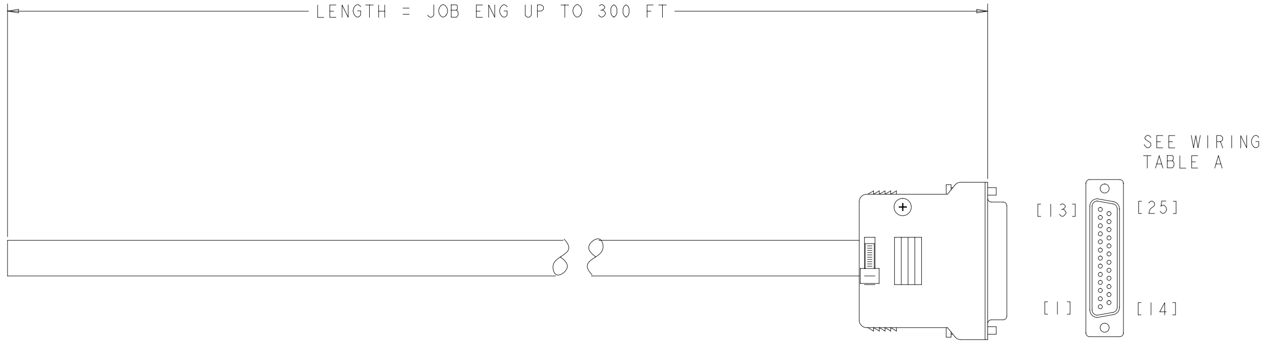
2. TO CONVERT INCHES TO MILLIMETERS USE 1 INCH = 25.40 MILLIMETERS.

3. () INDICATES DESIGNATIONS TO BE STAMPED IN ACCORDANCE WITH OTHER INFORMATION.

4. [] INDICATES DESIGNATIONS SHOWN FOR INFORMATION ONLY AND ARE NOT TO BE STAMPED.

5. <> INDICATES DESIGNATIONS WHICH ARE PROVIDED IN ACCORDANCE WITH OTHER INFORMATION.

6. REFER TO HANDBOOK HB0X0102; FIGURE 28B FOR CABLE IDENTIFICATION MARKINGS.



DMS-100 Family

TOPS ADAS Network Configuration

Reference Guide

NORTHERN TELECOM CONFIDENTIAL: The information contained in this document is the property of Northern Telecom. Except as specifically authorized in writing by Northern Telecom, the holder of this document shall keep the information contained herein confidential and shall protect same in whole or in part from disclosure and dissemination to third parties and use same for evaluation, operation, and maintenance purposes only.

Information subject to change without notice. Northern Telecom reserves the right to make changes in design or components as progress in engineering and manufacturing may warrant.

Publication number: 297-2211-800
Product release: ADAS07 and up
Loadname: ADAS07 and up
Document release: Standard 01.04
Date: October 1997
Printed in the United States of America

NORTEL
NORTHERN TELECOM