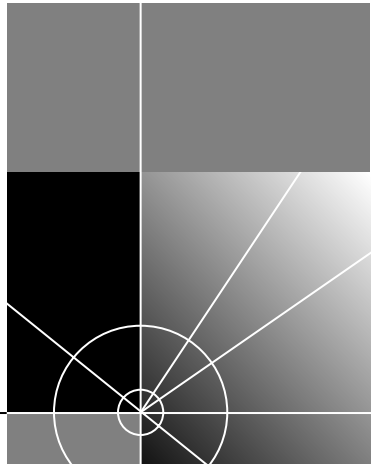
HiPer Network Management Card

Network Application Card Product Reference

Part No. 1.024.1660-00
Version Numbers
6.0, 6.1, 6.2

HiPer Network Management Card

Network Application Card Product Reference Version 6.0, 6.1, 6.2



http://www.3com.com/

Part No. 1.024.1660-00

3Com Corporation
5400 Bayfront Plaza
Santa Clara, California
95052-8145

YEAR 2000 INFORMATION:

For information on Year 2000 compliance and 3Com products, visit the 3Com Year 2000 web page:

http://www.3Com.com/products/yr2000.html

CONTENTS

ABOUT THIS REFERENCE

PART I PRODUCT OVERVIEW

NEW THIS RELEASE

CONFIGURING THE NMC SLIP PORT FOR REMOTE CHASSIS MANAGEMENT

ENABLING FEATURES AND ACTIVATING CHANGES

PLANNING FOR ADDITIONAL NMC CONFIGURATION

PART III TROUBLE CLEARING

USING THE FRONT PANEL INDICATORS TO DIAGNOSE PROBLEMS AND TROUBLE CLEAR

PART IV APPENDICES

HIPER NMC TECHNICAL SPECIFICATIONS

HiPer NMC System Block Diagram

10/100 Ethernet Aux I/O Network Interface Card Inputs and Outputs

Upgrading Software using Software Download-2

SINGLE GENERIC CONFIGURATION FILE

IP ADDRESSING

INDEX

About This Reference

This section provides an overview of this reference, describes reference conventions, tells you where to look for specific information, and lists other publications that may be useful.

This reference is intended for network administrators with some training or experience working in a data center using Total Control equipment. Prior experience with SNMP is recommended. This reference is most useful if you are already familiar with using an SNMP browser.

This reference mentions several different network devices and software applications. However, it does not provide an extensive discussion of each piece of software and each device mentioned. Please refer to the documentation provided for a particular device or piece of software for a complete description.

3Com® ships release notes with some products. If the information in the release notes differs from the information in this reference, follow the instructions in the release notes.

This document was written with the assumption that the user has some knowledge of data processing, telecommunications, and networking.

About configuring the Network Management Card

You must use the Network Management Card (NMC) Network Interface Card (NIC) RS–232 port and the User Interface (UI) to complete initial NMC configuration. This guide includes instructions for completing that configuration.

Use Total Control Manager or another SNMP manager to complete additional configuration.

Conventions

These tables list conventions that are used throughout this guide.

Notice Icons

Icon	Notice Type	Description
	Information note	Information that contains important features or instructions.
	Caution	Information to alert you to potential damage to a program, system, or device.
	Warning	Information to alert you to potential personal injury or fatality. May also alert you to potential electrical hazard.
	ESD	Information to alert you to take proper grounding precautions before handling a product.

Text Conventions

Convention	Description
Text represented as a `screen display`	This typeface represents displays that appear on your terminal screen, for example: `Netlogin:` This typeface also represents objects written in a MIB text (.txt) file.
Text represented as **`commands`**	**`This typeface`** represents commands that you enter, for example: **`setenv TCMHOME directory`**
Text represented as **menu** or **sub-menu names**.	**This typeface** represents all menu and sub-menu names within procedures, for example: On the **File** menu, click **New**.

Compatibility

You must use the 10/100 Ethernet Aux I/O Network Interface Card (NIC) with the HiPer NMC Network Access Card (NAC).

The NMC software is compatible with this NMC hardware:

NMC software version	NMC hardware
6.0	4MB 486-based NMC
6.1	16MB 486-based NMC
6.2	HiPer NMC

The NMC version 6.x software is compatible with these Total Control products:

Product	Version
HiPer DSP	2.0 (North America) 2.0 (International) 2.1
Quad modem	6.0 (Double Sided) 6.1 (Single Sided)
Total Control Manager	Windows 6.0 UNIX 6.0
Security and Accounting	Windows 6.0 UNIX 6.0
T1–386	4.3
T1 / PRI	3.2
T1–186	3.5
E1 / PRI	3.1
E1–R2	1.3
NETServer	3.8 (Ethernet)
NETServer Manager	3.4
HiPer ARC	4.1 (Ethernet)
HiPer ARC Manager (HARM)	Windows 1.1 UNIX 1.1 HP 1.1
EdgeServer	1.6
EdgeServerPRO	1.6

Related Documentation

Complete HiPer NMC documentation is available on the Total Control Documentation Library CD-ROM. The HiPer NMC documentation set includes these documents:

- **HiPer NMC NAC Getting Started Guide —** this document contains installation and trouble clearing information for the HiPer NMC NAC
- **NMC Parameter Reference —** this document contains a complete tabular listing of NMC MIBs and their related data, plus a cross reference to Total Control Manager commands
- **NMC SNMP and MIB Reference —** this document provides additional information about SNMP, NMC MIBs, and their application to the Total Control chassis
- **Software Download-2 (SDL-2) Instructions —** this document provides instructions for downloading new software to the HiPer NMC NAC
- **10/100 Ethernet Aux I/O NIC Getting Started Guide —** this document contains installation and troubleshooting information for the 10/100 Ethernet Aux I/O NIC that is used with the HiPer NMC NAC

Additional documentation for 486-based NMC releases is available on the Total Control Documentation Library CD-ROM. Documentation for other previous NMC releases is available at ***http://totalservice.3Com.com***.

The Total Control Documentation Library CD-ROM

The Total Control Documentation Library CD–ROM contains documentation for:

- Chassis and Fan Tray
- Network Management Card (NMC)
- Quad Modem Card
- NETServer
- Security and Accounting
- HiPer DSP Card
- HiPer ARC
- E1 Card
- T1 Card
- EdgeServer

- X.25 Card

Contacting 3Com

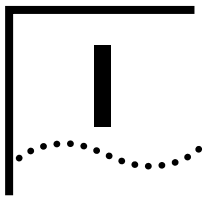
Call the appropriate toll free number listed below for technical support.

For European countries that do not have a toll free number listed, call +31 30 602 9900.

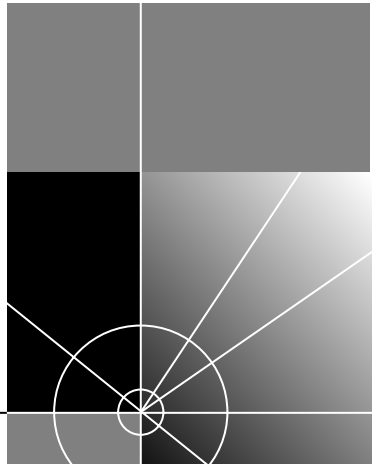
Country	Toll Free Number	Country	Toll Free Number
Austria	06 607468	**Netherlands**	0800 0227788
Belgium	0800 71429	**Norway**	800 11376
Canada	1800 2318770	**Poland**	00800 3111206
Denmark	800 17309	**Portugal**	0800 831416
Finland	0800 113153	**South Africa**	0800 995014
France	0800 917959	**Spain**	900 983125
Germany	0800 1821502	**Sweden**	020 795482
Hungary	00800 12813	**Switzerland**	0800 553072
Ireland	1800 553117	**UK**	0800 966197
Israel	0800 9453794	**United States**	1800 2318770
Italy	1678 79489	***All Other Locations*** ***(Outside Europe)***	1847 7976600

Refer to the Total Control Hub Documentation CD-ROM for more information regarding product warranty.

For information about Customer Service, including support, training, contracts, and documentation, visit our website at ***http://totalservice.3com.com***



PRODUCT OVERVIEW



1 NEW THIS RELEASE

This chapter describes the new features in this release of the NMC. The beginning of the chapter includes a features availability matrix for the software builds.

Overview of NMC release version 6.0

NMC version 6.x.x software is part of the Total Control System (TCS) 3.5 system release. The major additions to this NMC release are:

- Single configuration file (CFM) upload and download
- HISTORY MIB
- FILE MIB
- Additional SNMP management support for
  - Quad modems
  - Trunk applications
  - HiPer ARC
  - HiPer DSP
  - HiPer TRAX

NMC version 6.x.x supports the 386-based NMC, the 4MB and 16MB 486-based NMC, and the HiPer NMC. The release version numbers are:

Product	Software version
HiPer NMC	6.2.13
16MB 486-based NMC	6.1.13
4MB 486-based NMC	6.0.6

NMC Software Build Compatibility Matrix

This table defines the features that are included in the three new releases of NMC software:

Feature	4MB 486-based NMC version 6.0.6	16MB 486-based NMC version 6.1.13 HiPer NMC version 6.2.13
NMC Features		
Single Generic Config file	NO	YES
Unique Call Reference Number	NO	YES
Dynamic Slot Assignment	NO	YES
History MIB Support	NO	YES
File MIB Support	NO	YES
AUX-IO Support	NO	HiPer NMC only
QUAD 6.0 Modem Support		
All features as mentioned in this document	YES	YES
Trunk applications		
All features as mentioned in this document	YES	YES
HiPer ARC 4.1		
All features as mentioned in this document	NO	YES
Hiper DSP Modem		
All features as mentioned in this document	NO	YES
TRAX Support		
All features as mentioned in this document	NO	YES

New NMC Features

New NMC features include:

- A single generic configuration file (CFM)
- Unique generic call reference number for RADIUS
- Dynamic slot assignments for HiPer ARC load balancing
- New HISTORY MIB for supporting chassis-wide history information logging and reporting
- Auxiliary I/O support
- Synchronized NMC and HiPer DSP default values
- Removal of support for the PB MIB

Details of these features are in this section.

Single generic configuration file

You can now use the NMC to create and store a single file in the 3Com-proprietary CFM format that contains configuration information for all Network Application Cards (NACs) in the chassis (including HiPer ARC). The format of this file is located in the new FILE.MIB.

This file allows you to use SNMP to:

- Configure an entire chassis with a single file (by downloading the file from a MS to the NMC
- Compare different chassis configurations
- View the entire chassis configuration (by uploading the file from the NMC to a MS)

This feature is triggered through an SNMP browser and requires a TFTP script for file transfer. This feature is not available through Total Control Manager.

Refer to the appendix in this document titled *Single Generic Configuration File* for additional information, theory of operation, and sample scripts.

Unique call reference number for RADIUS

A new call reference number format allows the NMC and HiPer ARC cards to generate a unique call reference number (session ID) in RADIUS logging messages for chassis-wide call events/traps.

Previously, the NMC, NETServer, and HiPer ARC provided different call reference number formats to their own accounting servers. With the new format, both accounting servers will provide the reference numbers in the same format.

Backward compatibility

To allow backward compatibility, enable/disable this feature with the new NMC MIB configurable object *nmcCfgSessionIDNewFmt* . The default is disabled.

For all call events where session ID is applicable, the NMC checks this object to determine data format.

- If *nmcCfgSessionIDNewFmt* is disabled (default), the session ID will be generated to the pre-NMC version 6.0 format.
- If *nmcCfgSessionIDNewFmt* is enabled, the new format will be generated.

Setting the Unique Call Reference number through Total Control Manager

1. Select the NMC card.
2. From the **Configure** menu, select **Programmed Settings**.
3. From the **Parameter Group**, select **Configuration Group**.
4. From **Unique Call Reference Number**, select **enable** or **disable**.
5. Click **Set**.

Dynamic Slot Assignments for HiPer ARC Load Balancing

The dynamic slot assignment (DSA) feature allows the NMC to manage automatic load balancing when operating two or more HiPer ARC cards in the same chassis. DSA is supported by HiPer DSP modem cards only.

DSA is an automatic feature, and does not have a user setting.

This is how DSA works:

1. The NMC periodically polls all chassis HiPer ARC NACs to determine which cards support DSA. The NMC then compiles a list of all relevant NACs.
2. Next, the NMC periodically polls each DSA-supporting HiPer ARC to determine what modems are assigned to each HiPer ARC, how many

modems the NAC can support, and whether it can accept any more modem assignments.

3 As a result of these polls, a new statically load balanced slot assignment schedule is automatically computed for each HiPer ARC. A new schedule is also computed every time a modem or HiPer ARC is removed or inserted.

- If a HiPer DSP modem card reboots and it is not statically assigned to a particular HiPer ARC card, the modem slot is assigned to the HiPer ARC with the least load.
- If a HiPer ARC reboots, a period of time is allowed for the reboot, and then a new load schedule is automatically determined for the chassis.
- If an additional HiPer ARC is inserted into the chassis, and all modems are currently assigned to existing HiPer ARCs, no modem will be assigned to the new HiPer ARC until a modem reboots.

You may still statically assign modem slots to a particular HiPer ARC.

HISTORY MIB

The new HISTORY MIB supports chassis-wide information logging and reporting for Quad and HiPer DSP modems. This MIB is only supported in the NMC 16M 486/HiPer NMC builds.

The HISTORY MIB is not supported through Total Control Manager.

You may use the HISTORY MIB to see card-level and modem-level chassis trends, including modem use, call statistics, etc. History data for all modems is stored in 104 timeslot intervals, plus an additional interval for the current timeslot. Each interval contains information about all channels on each modem. Each channel holds data for all objects defined in the HISTORY MIB. The interval time span is configurable, with 15 minutes as the recommended default value (equalling 26 hours of chassis history).

History data is maintained in the NMC's DRAM memory. All history will be reset if the NMC or the chassis reboots. Use the **bulkFileUpload** NMC command to obtain the .hst file, which contains both the card-level and modem-level interval statistics.

At this release, you may only query the HISTORY MIB using a standard SNMP browser; Total Control Manager does not support this feature. The NMC also allows you to save the history data to a single configuration file format.

Refer to the *NMC SNMP and MIB Reference* for a complete listing of all HISTORY MIB objects.

FILE MIB

The new FILE MIB was designed to be used as a template for the Single Configuration File Format (CFM). This MIB is comprised of objects which represent the stored CFM data format which has been created by NMC. Contents of this MIB serve as template for parsing a file already created.

The FILE MIB is not supported through Total Control Manager.

Refer to the *NMC SNMP and MIB Reference* for a complete listing of all FILE MIB objects.

Auxiliary I/O Support

The Auxiliary (Aux) I/Os are the two input and two output ports located on the back of the 10/100 Ethernet AUX I/O NIC. Status and command functions are associated with these ports. This feature is only supported in the HiPer NMC.

The auxiliary I/O feature is not supported in the current release of Total Control Manager.

New Aux I/O Status Objects

Four read-only objects are added to the NMC MIB to report current status of each port:

- *nmcAuxIn1Sts*
- *nmcAuxIn2Sts*
- *nmcAuxOut1Sts*
- *nmcAuxOut2Sts*

For non-HiPer NMC cards, the status query will always return as "portNotApplicable".

```
nmcAuxIn1Sts OBJECT-TYPE
SYNTAX INTEGER{
  portOpen(1),
  portClosed(2),
  portNotApplicable(3)
}
ACCESS read-only
STATUS mandatory
```

```
DESCRIPTION
"This Object shows status of Aux Input port1."
::= { nmcStat 13 }

nmcAuxIn2Sts OBJECT-TYPE
SYNTAX INTEGER{
  portOpen(1),
  portClosed(2),
  portNotApplicable(3)
}
ACCESS read-only
STATUS mandatory
DESCRIPTION
"This Object shows status of Aux Input port2."
::= { nmcStat 14 }

nmcAuxOut1Sts OBJECT-TYPE
SYNTAX INTEGER{
  portOpen(1),
  portClosed(2),
  portNotApplicable(3)
}
ACCESS read-only
STATUS mandatory
DESCRIPTION
"This Object shows status of Aux Output port1."
::= { nmcStat 15 }

nmcAuxOut2Sts OBJECT-TYPE
SYNTAX INTEGER{
  portOpen(1),
  portClosed(2),
  portNotApplicable(3)
}
ACCESS read-only
STATUS mandatory
DESCRIPTION
"This Object shows status of Aux Output port2."
::= { nmcStat 16 }
```

Aux I/O Commands

Four response handlers are added to the AutoResponse mechanism to drive the Aux I/O commands. These commands allow you to open or close the ports.

These handlers are the new commands in *nmcCmdFunction* (shown in bold):

```
nmcCmdFunction OBJECT-TYPE
SYNTAX INTEGER{
  noCommand(1),
  saveToNvram(2),
  restoreFromDefaults(3),
  restoreFromNvram(4),
  nonDisruptSelfTest(5),
  softwareReset(6),
  saveUiParmsToEEPROM(7),
  restoreNmcFromDefaults(8),
  restoreNmcFromNvram(9),
  bulkFileUpload(10),
  bulkFileDownload(11),
  openAuxOutputPort1(12),
  openAuxOutputPort2(13),
  closeAuxOutputPort1(14),
  closeAuxOutputPort2(15)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"This object contains the value which decsribe the command
which is being invoked."
::= { nmcCmd 3 }
```

Change port status by issuing the appropriate Aux I/O command. The values of these commands reflect the actual operation values needed to be sent via an SNMP message. When the port status is successfully changed, a trap is sent from the NMC to the MS. Refer to "Aux I/O Traps" for a list of the traps sent.

Aux I/O Traps

Traps are generated automatically after you issue a command to open or close an Aux I/O port.

These Aux I/O traps are added to the CHS_TRAP MIB:

- *AuxIN1Open*
- *AuxIN2Open*
- *AuxIN1Closed*

- *AuxIN2Closed*
- *AuxOut1Open*
- *AuxOut1Closed*
- *AuxOut2Open*
- *AuxOut2Closed*

AuxIN1Open Trap

```
AuxIN1Open TRAP-TYPE
ENTERPRISE usr
DESCRIPTION
"Aux In1 Port Open."
--#TYPE "Aux In port 1 open"
--#SUMMARY "%d; Aux In Port 1 Open"
--#ARGUMENTS { 1 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 178
```

AuxIN2Open Trap

```
AuxIN2Open TRAP-TYPE
ENTERPRISE usr
DESCRIPTION
"Aux In2 Port Open."
--#TYPE "Aux In port 2 open"
--#SUMMARY "%d; Aux In Port 2 Open"
--#ARGUMENTS { 1 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 179
```

AuxIN1Closed Trap

```
AuxIN1Closed TRAP-TYPE
ENTERPRISE usr
DESCRIPTION
"Aux In1 Port Closed."
```

```
--#TYPE "Aux In port 1 closed"
--#SUMMARY "%d; Aux In Port 1 Closed"
--#ARGUMENTS { 1 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 180
```

AuxIN2Closed Trap

```
AuxIN2Closed TRAP-TYPE
ENTERPRISE usr
DESCRIPTION
"Aux In2 Port Closed."
--#TYPE "Aux In port 2 closed"
--#SUMMARY "%d; Aux In Port 2 Closed"
--#ARGUMENTS { 1 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 181
```

AuxOut1Open Trap

```
AuxOut1Open TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcAuxOut1Sts
}
DESCRIPTION
"Aux Output Port1 Open."
--#TYPE "Aux Out port 1 open"
--#SUMMARY "%d; Aux Out Port 1 Open"
--#ARGUMENTS { 1 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 182
```

AuxOut1Closed Trap

```
AuxOut1Closed TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcAuxOut1Sts
}
DESCRIPTION
"Aux Output Port1 Closed."
--#TYPE "Aux Out port 1 closed"
--#SUMMARY "%d; Aux Out Port 1 Closed"
--#ARGUMENTS { 1 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 183
```

AuxOut2Open Trap

```
AuxOut2Open TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcAuxOut2Sts
}
DESCRIPTION
"Aux Output Port2 Open."
--#TYPE "Aux Out port 2 open"
--#SUMMARY "%d; Aux Out Port 2 Open"
--#ARGUMENTS { 1 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 184
```

AuxOut2Closed Trap

```
AuxOut2Closed TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcAuxOut2Sts
}
DESCRIPTION
"Aux Output Port2 Closed."
--#TYPE "Aux Out port 2 closed"
--#SUMMARY "%d; Aux Out Port 2 Closed"
--#ARGUMENTS { 1 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 185
```

Synchronized NMC and HiPer DSP Default Values

NMC and HiPer DSP default values are synchronized at both the channel- and template-level to ensure efficient product configuration. This feature does not have any user configuration.

NMC Support for Quad Modem

New NMC support for the Quad modems includes:

- 105 responder test enhancements
- Bidirectional trap enables for connection attempt fails
- LLC IE message for ISDN HDLC protocols
- Data over voice bearer service (DOVBS)
- Support for min/max speed per session
- DTMF detection
- Online configuration

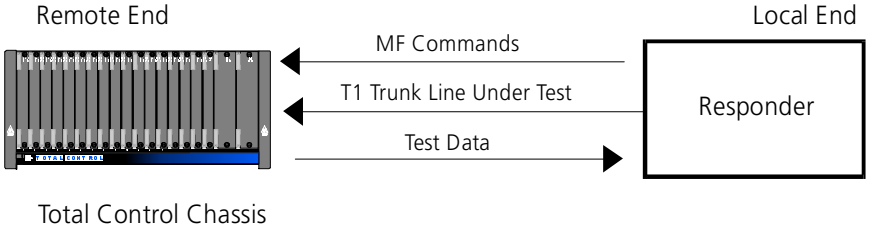
Details of these features are in this section.

105 Responder Test Enhancements

Responder tests are used to evaluate the performance of point-to-point circuits and test the quality of a T1 DS0 channel using the T1 trunks. It is a bi-directional test performed in these directions: near-to-far and far-to-near.

The near-end is connected to the responder unit box on the trunk to be tested (also known as the director). It establishes the connection over the trunk to the DWAN chassis, acting as a true responder via the 105 type test line.

This diagram shows the mode in which the test operates from the responder unit to the chassis.



105 Responder Test objects

New tables and objects are added to the MDM MIB to support the 105 responder test:

- *mdmTfDialInToneTest*
- *mdmTfToneTestCallRef*
- *mdmTfToneTable*
  - *mdmTfToneEntry*
  - *mdmTfToneIndex*
  - *mdmTfTxFreq*
  - *mdmTfTxAmpl*
  - *mdmTfRxFreq*
  - *mdmTf*RxAmpl
- *mdmTfRspndrTable*
  - *mdmTfRspndrEntry*
  - *mdmTfRspndrIndex*
  - *mdmTf404FarNearLvl*
  - *mdmTf404NearFarLvl*
  - *mdmTf1004FarNearLvl*
  - *mdmTf1004NearFarLvl*
  - *mdmTf2804FarNearLvl*
  - *mdmTf2804NearFarLvl*
  - *mdmTfCmsgFarNearLvl*
  - *mdmTfCmsgNearFarLvl*
  - *mdmTfCnotchFarNearLvl*
  - *mdmTfCnotchNearFarLvl*
  - *mdmtTfSigNoiseFarNearLvl*
  - *mdmtTfSigNoiseNearFarLvl*
  - *mdmTf404FarNearSts*
  - *mdmTf404NearFarSts*
  - *mdmTf1004FarNearSts*

- *mdmTf1004NearFarSts*
- *mdmTf2804FarNearSts*
- *mdmTf2804NearFarSts*
- *mdmTfCmsgFarNearSts*
- *mdmTfCmsgNearFarSts*
- *mdmTfCnotchFarNearSts*
- *mdmTfCnotchNearFarSts*
- *mdmTfSigNoiseFarNearSts*
- *mdmTfSigNoiseNearFarSts*
- *mdmTf0dB1004FarNearLvl*
- *mdmTf0dB1004NearFarLvl*
- *mdmTf0dB1004FarNearSts*
- *mdmTf0dB1004NearFarSts*

```
mdmTfDialInToneTest OBJECT-TYPE
SYNTAX INTEGER{
  disable(1),
  enable(2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"This object is used to enable Dial in Tone Tests."
::= { mdmTfEntry 7 }

mdmTfToneTestCallRef OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(0..53))
ACCESS read-only
STATUS mandatory
DESCRIPTION
"This object is used to send the call reference info with
the tone test."

::= { mdmTfEntry 8 }

mdmTfToneTable OBJECT-TYPE
SYNTAX SEQUENCE OF MdmTfToneEntry
ACCESS not-accessible
```

```
STATUS optional
DESCRIPTION
"Table for modem tone tests"
::= { mdmTf 2 }

mdmTfToneEntry OBJECT-TYPE
SYNTAX MdmTfToneEntry
ACCESS not-accessible
STATUS optional
DESCRIPTION
"Objects for modem tone tests"
INDEX { mdmTfToneIndex }
::= { mdmTfToneTable 1 }

MdmTfToneEntry ::=
SEQUENCE {
mdmTfToneIndex
    INTEGER,
mdmTfTxFreq
    INTEGER,
mdmTfTxAmpl
    INTEGER,
mdmTfRxFreq
    INTEGER,
mdmTfRxAmpl
    INTEGER
}

mdmTfToneIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"A value used to ID this modem"
::= { mdmTfToneEntry 1 }

mdmTfTxFreq OBJECT-TYPE
SYNTAX INTEGER (300..4000)
ACCESS read-write
STATUS optional
DESCRIPTION
"Modem tone test TX frequency"
::= { mdmTfToneEntry 2 }

mdmTfTxAmpl OBJECT-TYPE
SYNTAX INTEGER (-40..0)
```

```
ACCESS read-write
STATUS optional
DESCRIPTION
"Modem tone test TX amplitude"
::= { mdmTfToneEntry 3 }

mdmTfRxFreq OBJECT-TYPE
SYNTAX INTEGER (0..5000)
ACCESS read-only
STATUS optional
DESCRIPTION
"Modem tone test RX frequency"
::= { mdmTfToneEntry 4 }

mdmTfRxAmpl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"Modem tone test RX amplitude"
::= { mdmTfToneEntry 5 }

mdmTfRspndrTable OBJECT-TYPE
SYNTAX SEQUENCE OF MdmTfRspndrEntry
ACCESS not-accessible
STATUS optional
DESCRIPTION
"Table for 105/102 responder test results."
::= { mdmTf 3 }

mdmTfRspndrEntry OBJECT-TYPE
SYNTAX MdmTfRspndrEntry
ACCESS not-accessible
STATUS optional
DESCRIPTION
"Objects containing the 105/102 responder test result dB
levels and test status values."
INDEX { mdmTfRspndrIndex }
::= { mdmTfRspndrTable 1 }

MdmTfRspndrEntry ::=
SEQUENCE {
mdmTfRspndrIndex
    INTEGER,
mdmTf404FarNearLvl
    INTEGER,
```

```
mdmTf404NearFarLvl
    INTEGER,
mdmTf1004FarNearLvl
    INTEGER,
mdmTf1004NearFarLvl
    INTEGER,
mdmTf2804FarNearLvl
    INTEGER,
mdmTf2804NearFarLvl
    INTEGER,
mdmTfCmsgFarNearLvl
    INTEGER,
mdmTfCmsgNearFarLvl
    INTEGER,
mdmTfCnotchFarNearLvl
    INTEGER,
mdmTfCnotchNearFarLvl
    INTEGER,
mdmtTfSigNoiseFarNearLvl
    INTEGER,
mdmtTfSigNoiseNearFarLvl
    INTEGER,
mdmTf404FarNearSts
    INTEGER,
mdmTf404NearFarSts
    INTEGER,
mdmTf1004FarNearSts
    INTEGER,
mdmTf1004NearFarSts
    INTEGER,
mdmTf2804FarNearSts
    INTEGER,
mdmTf2804NearFarSts
    INTEGER,
mdmTfCmsgFarNearSts
    INTEGER,
mdmTfCmsgNearFarSts
    INTEGER,
mdmTfCnotchFarNearSts
    INTEGER,
mdmTfCnotchNearFarSts
    INTEGER,
mdmTfSigNoiseFarNearSts
    INTEGER,
mdmTfSigNoiseNearFarSts
    INTEGER,
```

```
mdmTf0dB1004FarNearLvl
    INTEGER,
mdmTf0dB1004NearFarLvl
    INTEGER,
mdmTf0dB1004FarNearSts
    INTEGER,
mdmTf0dB1004NearFarSts
    INTEGER
}

mdmTfRspndrIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"A value used to ID this modem."
::= { mdmTfRspndrEntry 1 }

mdmTf404FarNearLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"404 Hz tone at -16 dB loss test level. Far to near. Level
in 0.1 dB resolution."
::= { mdmTfRspndrEntry 2 }

mdmTf404NearFarLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"404 Hz tone at -16 dB loss test level. Near to far. Level
in 0.1 dB resolution."
::= { mdmTfRspndrEntry 3 }

mdmTf1004FarNearLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"1004 Hz tone at -16 dB loss test level. Far to near. Level
in 0.1 dB resolution."
::= { mdmTfRspndrEntry 4 }

mdmTf1004NearFarLvl OBJECT-TYPE
```

```
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"1004 Hz tone at -16 dB loss test level. Near to far. Level
in 0.1 dB resolution."
::= { mdmTfRspndrEntry 5 }

mdmTf2804FarNearLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"2804 Hz tone at -16 dB loss test level. Far to near. Level
in 0.1 dB resolution."
::= { mdmTfRspndrEntry 6 }

mdmTf2804NearFarLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"2804 Hz tone at -16 dB loss test level. Near to far. Level
in 0.1 dB resolution."
::= { mdmTfRspndrEntry 7 }

mdmTfCmsgFarNearLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"C-message test level. Far to near. Level in 0.1 dB
resolution."
::= { mdmTfRspndrEntry 8 }

mdmTfCmsgNearFarLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"C-message test level. Near to far. Level in 0.1 dB
resolution."
::= { mdmTfRspndrEntry 9 }

mdmTfCnotchFarNearLvl OBJECT-TYPE
SYNTAX INTEGER
```

```
ACCESS read-only
STATUS optional
DESCRIPTION
"C-notch test level. Far to near. Level in 0.1 dB
resolution."
::= { mdmTfRspndrEntry 10 }

mdmTfCnotchNearFarLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"C-notch test level. Near to far. Level in 0.1 dB
resolution."
::= { mdmTfRspndrEntry 11 }

mdmtTfSigNoiseFarNearLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"Signal to noise ratio level. Far to near. Level in 0.1 dB
resolution."
::= { mdmTfRspndrEntry 12 }

mdmtTfSigNoiseNearFarLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"Signal to noise ratio level. Near to far. Level in 0.1 dB
resolution."
::= { mdmTfRspndrEntry 13 }

mdmTf404FarNearSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
```

```
"404 Hz tone at -16 dB loss test status. Far to near.

Default=noTest(1)."
::= { mdmTfRspndrEntry 14 }

mdmTf404NearFarSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"404 Hz tone at -16 db loss test status. Near to far.

Default=noTest(1)."
::= { mdmTfRspndrEntry 15 }

mdmTf1004FarNearSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"1004 Hz tone at -16 dB loss test status. Far to near.

Default=noTest(1)."
::= { mdmTfRspndrEntry 16 }

mdmTf1004NearFarSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
```

```
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"1004 Hz tone at -16 dB loss test status. Near to far.

Default=noTest(1)."
::= { mdmTfRspndrEntry 17 }

mdmTf2804FarNearSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"2804 Hz tone at -16 dB loss test status. Far to near.

Default=noTest(1)."
::= { mdmTfRspndrEntry 18 }

mdmTf2804NearFarSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"2804 Hz tone at -16 dB loss test status. Near to far.

Default=noTest(1)."
::= { mdmTfRspndrEntry 19 }

mdmTfCmsgFarNearSts OBJECT-TYPE
SYNTAX INTEGER{
```

```
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"C-message test status. Far to near.
Default=noTest(1)."
::= { mdmTfRspndrEntry 20 }

mdmTfCmsgNearFarSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"C-message test status. Near to far.
Default=noTest(1)."
::= { mdmTfRspndrEntry 21 }

mdmTfCnotchFarNearSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"C-notch test status. Far to near.
Default=noTest(1)."
::= { mdmTfRspndrEntry 22 }
```

```
mdmTfCnotchNearFarSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"C-notch test status. Near to far.
Default=noTest(1)."
::= { mdmTfRspndrEntry 23 }

mdmTfSigNoiseFarNearSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"Signal to noise ratio status. Far to near.

Default=noTest(1)."
::= { mdmTfRspndrEntry 24 }

mdmTfSigNoiseNearFarSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"Signal to noise ratio status. Near to far.
```

```
Default=noTest(1)."
::= { mdmTfRspndrEntry 25 }

mdmTf0dB1004FarNearLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"1004 Hz tone at 0 db loss test level. Far to near. Level in
0.1 dB resolution."
::= { mdmTfRspndrEntry 26 }

mdmTf0dB1004NearFarLvl OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS optional
DESCRIPTION
"1004 Hz tone at 0 db loss test level. Near to far. Level in
0.1 dB resolution."
::= { mdmTfRspndrEntry 27 }

mdmTf0dB1004FarNearSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"1004 Hz tone at 0 db loss test status. Far to
near.
Default=noTest(1)."
::= { mdmTfRspndrEntry 28 }

mdmTf0dB1004NearFarSts OBJECT-TYPE
SYNTAX INTEGER{
  noTest(1),
  success(2),
  noResponder(3),
  unsupported(4),
  timeOut(5),
```

```
  noToneDetected(6)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"1004 Hz tone at 0 db loss test status. Near to
Far.
Default=noTest(1)."
::= { mdmTfRspndrEntry 29 }
```

Enabling the Responder Test through Total Control Manager

Follow these steps to enable the 105 Responder test through Total Control Manager:

1. On the Quad modem, select the channel.
2. From the **Fault** menu, select **Remote Testing**, **Responder Test(s)**, **102/105 Dial-Out** or **105 Dial-In**.
3. In the **Dial string** box, enter a dial string.
4. From **Select Test**, select **105 Responder**.
5. Click **Start**.

105 Responder Test trap

A new trap enable, *mdmTe105ResponderTest,* is added to the MDM MIB to support the 105 responder test:

This trap is not supported by Total Control Manager.

```
mdmTe105ResponderTest OBJECT-TYPE
SYNTAX INTEGER{
  enableTrap(1),
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Enables generation of a SNMP trap upon detection of a 105
responder test on the specified modem."
::= { mdmTeEntry 25 }
```

A new trap, *mdm105responderTest*, is added to the CHS_TRAP MIB to support the 105 responder test:

This event is not reported in Total Control Manager.

mdm105responderTest Trap

```
mdm105responderTest TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
uchasEntityIndex,
uchasEntityObjectID,
mdmTfToneTestCallRef,
mdmTf404NearFarLvl,
mdmTf1004NearFarLvl,
mdmTf0dB1004NearFarLvl,
mdmTf2804NearFarLvl,
mdmTfCmsgNearFarLvl,
mdmTfCnotchNearFarSts,
mdmTfSigNoiseNearFarSts
}
DESCRIPTION
"This event is sent to indicate the 105 Responder test is
active."

--#TYPE "Modem 105 responder test"
--#SUMMARY "%d; A Modem 105 Responder test occured in Slot
%d, Channel %d"
--#ARGUMENTS { 1 3 4 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 125
```

Bidirectional traps for failed connection attempts

New trap enables

Two new trap enables, *mdmTeInConnAttemptFail* and *mdmTeOutConnAttemptFail*, are added in the MDM MIB for inbound and outbound connection attempt fail events:

```
mdmTeInConnAttemptFail OBJECT-TYPE
SYNTAX INTEGER{
  enableTrap(1),
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Enables generation of a SNMP trap upon detection of an
inbound connection attempt failure on the specified modem."
::= { mdmTeEntry 23 }

mdmTeOutConnAttemptFail OBJECT-TYPE
SYNTAX INTEGER{
  enableTrap(1),
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Enables generation of a SNMP trap upon detection of an
outbound connection attempt failure on the specified modem."
::= { mdmTeEntry 24 }
```

New traps

Two new traps, *Incoming Connections Failed* and *Outgoing Connections Failed,* are added to the CHS_TRAP MIB for inbound and outbound connection attempt fail events:

These events are reported in Total Control Manager through ***Performance****,* ***Modem Events****.*

Incoming Connections Failed Trap

```
inconnectAttemptFailure TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
```

```
uchasEntityIndex,
uchasEntityObjectID,
mdmCsConnectFailReason,
mdmCsCallRefNum
}
DESCRIPTION
"In bound call attempt failed to connect with the modem"
--#TYPE "In Connection Attempt Failure"
--#SUMMARY "%d; In bound Connection Attempt Failure on Slot
%d, Chan %d, Fail %d, Ref %d"
--#ARGUMENTS { 1 3 4 6 7}
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 86
```

Outgoing Connections Failed Trap

```
outconnectAttemptFailure TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
uchasEntityIndex,
uchasEntityObjectID,
mdmCsConnectFailReason,
mdmCsCallRefNum
}
DESCRIPTION
"Out bound call attempt failed to connect with the modem"
--#TYPE "Out Connection Attempt Failure"
--#SUMMARY "%d; Out bound Connection Attempt Failure on Slot
%d, Chan %d, Fail %d, Ref %d"
--#ARGUMENTS { 1 3 4 6 7}
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 87
```

Enabling bidirectional traps through Total Control Manager

1. On the Quad modem, select the channel.
2. From the **Fault** menu, select **Trap Settings**.
3. From the **Parameter Group**, select **Packet Bus Traps**.
4. For **Incoming Connections Failad** and **Outgoing Connections Failed**, enable the appropriate settings.
5. Click **Set**.

LLC IE message for ISDN HDLC protocols

A new object, *mdmCcHdlcLicIe*, is added to the MDM MIB to support the LLC IE (lower layer compatibility information element) message for ISDN HDLC protocols:

This object is not supported in Total Control Manager.

```
mdmCcHdlcLicIe OBJECT-TYPE
SYNTAX INTEGER{
  disable(1),
  enable(2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"This object is to Enable\Disable LIC IE message for HDLC
protocols. S Register S68.3"
::= { mdmCcEntry 60 }
```

Enabling this feature allows the modem to use the Telco's LLC IE message to determine its training sequence. Disabling this feature causes the modem to autodetect the incoming data transfer protocol. The default is disabled. This object supports S68.3 register.

Data Over Voice Bearer Service

New objects, *mdmCcDataOverVoice* and *mdmCc2100AnswerTone,* are added to the MDM MIB to support data over voice bearer service (DOVBS) for T1 applications. These objects are needed for S68.1 and S68.2 registers to enable and disable support for DOBVS for outgoing T1 calls:

mdmCcDataOverVoice

```
mdmCcDataOverVoice OBJECT-TYPE
SYNTAX INTEGER{
```

```
  disable(1),
  enable(2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Data Over Voice support originate S68.1."
::= { mdmCcEntry 57 }
```

Use *mdmCcDataOverVoice* to enable or disable DOVBS. If DOVBS is enabled on the originating side, the modem is instructed to originate speech bearer capability call types.

mdmCc2100AnswerTone

```
mdmCc2100AnswerTone OBJECT-TYPE
SYNTAX INTEGER{
  disable(1),
  enable(2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Data Over Voice support answer S68.2."
::= { mdmCcEntry 58 }
```

Use *mdmCc2100AnswerTone* to enable or disable DOVBS. If DOVBS is enabled on the answering side, the modem is instructed to originate speech bearer capability call types.

Setting DOVBS through Total Control Manager

1. On the Quad modem, select the channel.
2. From the **Configure** menu, select **Programmed Settings**.
3. From the **Parameter Group**, select **Data Over Voice Bearer Service (DOVBS)**, then make the appropriate settings.
4. Click **Set**.

Support for min/max speed per session

New query inbound/outbound objects for min/max speeds per session are added to the MDM MIB:

- *mdmCsRxMinSpeed*
- *mdmCsRxMaxSpeed*
- *mdmCsTxMinSpeed*
- *mdmCsTxMaxSpeed*

```
mdmCsRxMinSpeed OBJECT-TYPE
SYNTAX INTEGER{
  bps110(1),
  bps300(2),
  bps600(3),
  bps1200(4),
  bps2400(5),
  bps4800(6),
  bps7200(7),
  bps9600(8),
  bps12K(9),
  bps14K(10),
  bps16K(11),
  bps19K(12),
  bps38K(13),
  bps75(14),
  bps450(15),
  unknown(16),
  bps57K(17),
  bps21K(18),
  bps24K(19),
  bps26K(20),
  bps28K(21),
  bps31K(23),
  bps33K(24),
  bps25333(25),
  bps26666(26),
  bps28000(27),
  bps29333(28),
  bps30666(29),
  bps32000(30),
  bps33333(31),
  bps34666(32),
  bps36000(33),
  bps37333(34),
  bps38666(35),
```

```
  bps40000(36),
  bps41333(37),
  bps42666(38),
  bps44000(39),
  bps45333(40),
  bps46666(41),
  bps48000(42),
  bps49333(43),
  bps50666(44),
  bps52000(45),
  bps53333(46),
  bps54666(47),
  bps56000(48),
  bps57333(49),
  bps58666(50),
  bps60000(51),
  bps61333(52),
  bps62666(53),
  bps64000(54)
}
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Current/Last Session MIN Rx Speed."
::= { mdmCsEntry 89 }

mdmCsRxMaxSpeed OBJECT-TYPE
SYNTAX INTEGER{
  bps110(1),
  bps300(2),
  bps600(3),
  bps1200(4),
  bps2400(5),
  bps4800(6),
  bps7200(7),
  bps9600(8),
  bps12K(9),
  bps14K(10),
  bps16K(11),
  bps19K(12),
  bps38K(13),
  bps75(14),
  bps450(15),
  unknown(16),
  bps57K(17),
  bps21K(18),
```

```
  bps24K(19),
  bps26K(20),
  bps28K(21),
  bps31K(23),
  bps33K(24),
  bps25333(25),
  bps26666(26),
  bps28000(27),
  bps29333(28),
  bps30666(29),
  bps32000(30),
  bps33333(31),
  bps34666(32),
  bps36000(33),
  bps37333(34),
  bps38666(35),
  bps40000(36),
  bps41333(37),
  bps42666(38),
  bps44000(39),
  bps45333(40),
  bps46666(41),
  bps48000(42),
  bps49333(43),
  bps50666(44),
  bps52000(45),
  bps53333(46),
  bps54666(47),
  bps56000(48),
  bps57333(49),
  bps58666(50),
  bps60000(51),
  bps61333(52),
  bps62666(53),
  bps64000(54)
}
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Current/Last Session MAX Rx Speed."
::= { mdmCsEntry 90 }

mdmCsTxMinSpeed OBJECT-TYPE
SYNTAX INTEGER{
  bps110(1),
  bps300(2),
```

```
bps600(3),
bps1200(4),
bps2400(5),
bps4800(6),
bps7200(7),
bps9600(8),
bps12K(9),
bps14K(10),
bps16K(11),
bps19K(12),
bps38K(13),
bps75(14),
bps450(15),
unknown(16),
bps57K(17),
bps21K(18),
bps24K(19),
bps26K(20),
bps28K(21),
bps31K(23),
bps33K(24),
bps25333(25),
bps26666(26),
bps28000(27),
bps29333(28),
bps30666(29),
bps32000(30),
bps33333(31),
bps34666(32),
bps36000(33),
bps37333(34),
bps38666(35),
bps40000(36),
bps41333(37),
bps42666(38),
bps44000(39),
bps45333(40),
bps46666(41),
bps48000(42),
bps49333(43),
bps50666(44),
bps52000(45),
bps53333(46),
bps54666(47),
bps56000(48),
bps57333(49),
```

```
  bps58666(50),
  bps60000(51),
  bps61333(52),
  bps62666(53),
  bps64000(54)
}
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Current/Last Session MIN Tx Speed."
::= { mdmCsEntry 91 }

mdmCsTxMaxSpeed OBJECT-TYPE
SYNTAX INTEGER{
  bps110(1),
  bps300(2),
  bps600(3),
  bps1200(4),
  bps2400(5),
  bps4800(6),
  bps7200(7),
  bps9600(8),
  bps12K(9),
  bps14K(10),
  bps16K(11),
  bps19K(12),
  bps38K(13),
  bps75(14),
  bps450(15),
  unknown(16),
  bps57K(17),
  bps21K(18),
  bps24K(19),
  bps26K(20),
  bps28K(21),
  bps31K(23),
  bps33K(24),
  bps25333(25),
  bps26666(26),
  bps28000(27),
  bps29333(28),
  bps30666(29),
  bps32000(30),
  bps33333(31),
  bps34666(32),
  bps36000(33),
```

```
    bps37333(34),
    bps38666(35),
    bps40000(36),
    bps41333(37),
    bps42666(38),
    bps44000(39),
    bps45333(40),
    bps46666(41),
    bps48000(42),
    bps49333(43),
    bps50666(44),
    bps52000(45),
    bps53333(46),
    bps54666(47),
    bps56000(48),
    bps57333(49),
    bps58666(50),
    bps60000(51),
    bps61333(52),
    bps62666(53),
    bps64000(54)
  }
  ACCESS read-only
  STATUS mandatory
  DESCRIPTION
  "Current/Last Session MAX Tx Speed."
  ::= { mdmCsEntry 92 }
```

Monitoring minimum and maximum RX speed through Total Control Manager

1. On the Quad modem, select the channel.
2. From the **Performance** menu, select **Session Monitor**.
3. From the **Functional Group**, select **Call Statistics**.
4. Select **Rx Minimum Speed** and/or **Rx Maximum Speed**, then click **add**.
5. Click **OK**.

DTMF detection

New objects, *mdmCsCollectedDtmfDigits* and *mdmCcDtmfTerminationTone***,** are added to the MDM MIB to support DTMF (Dual Tone Multi-Frequency) detection:

mdmCsCollectedDtmfDigits

```
mdmCsCollectedDTMFDigits OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(0..64))
ACCESS read-only
STATUS mandatory
DESCRIPTION
"This object is an ASCII array of the DTMF digits collected
from the client."
::= { mdmCsEntry 93 }
```

The Quad modem supports the ability to detect, collect, and route the DTMF tones sent from a client device to a software application that is either included in the chassis (such as the EdgeServer), or through RS–232 or packet bus to an application sitting outside of the chassis.

Use *mdmCsCollectedDtmfDigits* to query a modem to gather all collected DTMF tones in ASCII data format.

Monitoring collected DTMF digits through Total Control Manager

1. On the Quad modem, select the channel.
2. From the **Performance** menu, select **Session Monitor**.
3. From the **Functional Group**, select **Call Statistics**.
4. Select **Collected DTMF digits**, then click **add**.
5. Click **OK**.

mdmCcDtmfTerminationTone

```
mdmCcDtmfTerminationTone OBJECT-TYPE
SYNTAX INTEGER (0..16)
ACCESS read-write
STATUS mandatory
DESCRIPTION
"This object is to set the DTMF Termination Tone."
::= { mdmCcEntry 61 }
```

Use *mdmCcDtmfTerminationTone* as the equivalent of **AT%Gn** , where n = 0–16. The digits are translated as:

- 0–9; DTMF tones "0" to "9"
- 10–15; DTMF tones "A", "B", "C", "D", "E", and "F"
- 16; Intertone timeout (2 seconds)

Activate this feature by sending **ATH1** when receiving a call, followed by **AT%T**. Three beeps can be sent to alert the caller to send the DTMF digits by setting the S-register 72.2 = 1. Use **AT%G** to indicate the DTMF termination character.

Online Configuration

The Quad modem now accepts configuration objects (parameters) from the NMC while it is online. All changes are applied immediately, and queries will return the current configuration.

Most configuration changes will not affect the current call because the settings are only used while the modem is training or in an offline state. A few configuration changes may affect the current call, so be aware of this if completing online configuration. All configuration changes will be in effect for the following calls.

NMC Support for Trunk Applications

New NMC support for the trunk applications includes:

- Enhanced status for call event traps
- Increased number of modem pools
- Australian switch type configuration
- Configurable "wink to seizure" delay
- Support for BRI over R2
- R2-Venezuela configuration

Details of these features are in this section.

Enhanced Status for Call Event Traps

The new directional switch (variable binding) *ids0StatCallDir* is added to these CHS_TRAP MIB traps to indicate the direction of the call:

- callArriveEvent
- callConnectEvent
- callTermNormalEvent
- callTermFailedEvent

Increased number of modem pools

Eight modem pools are added to the existing pools allowed in the idt1PlTable. This change increases the number of available modem pools to 12.

IAustralian Switch Type Configuration

A new enumeration value is added to *uds1CfgPriSwitchType* to support the Australian switch type (new value is in bold):

```
uds1CfgPriSwitchType OBJECT-TYPE
SYNTAX INTEGER{
  priSw4ESS(1),
  priSw5ESS(2),
  priSwDMS100(3),
  priSwICTR4(4),
  priSwVn4(5),
  priSwNI2(6),
  priSwINS1500(7),
  priSwDASS2(8),
  priSwTSO14(9)
}
ACCESS read-write
STATUS mandatory
```

```
DESCRIPTION
"This sets the primary switch type for the T1-PRI ISDN NAC.
The setting takes effect at NAC boot time.
Default =
priSw5ESS(2)."
::= { uds1ConfigEntry 13 }
```

Setting the switch type configuration through Total Control Manager

1. On the PRI card, select the span.
2. From the **Configure** menu, select **Programmed Settings**.
3. From the **Parameter Group**, select **PRI Trunk Settings**.
4. From **Active Primary Switch Type**, select the appropriate switch type. For the Australian switch, select **priSwTSO14**.
5. Click **Set**.

Configurable "Wink to Seizure" Delay

A new configurable object, *uds1CfgSeizureWinkDly*, is added to the UDS1 MIB to support SNMP configuration of the delay from Telco seizure to T1 wink signaling:

```
uds1CfgSeizureWinkDly OBJECT-TYPE
SYNTAX INTEGER (70..3000)
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Allows an adjustable delay on sending out address
information from the T1 NAC to the TELCO."
::= { uds1ConfigEntry 30 }
```

This new object supplements the previous console port configurable value. The default is 200 ms.

Setting the wink to seizure delay through Total Control Manager

1. On the T1 card, select the span.
2. From the **Configure** menu, select **Programmed Settings**.
3. From the **Parameter Group**, select **DS1 Trunk Settings**.
4. From **Seizure to Wink Delay**, set the appropriate delay.
5. Click **OK**.

Support for BRI over R2

A new object, *idt1CfgNoBptyMethod*, is added to the IDT1 MIB to support BRI over R2:

```
idt1CfgNoBptyMethod OBJECT-TYPE
SYNTAX INTEGER{
  analog(1),
  digital(2),
  unAllocatedNum(3)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"This object sets the method for handling an incoming call
when the B-party (Called Number) is not found within the
<inPhoneNum> table."
::= { idt1CfgEntry 10 }
```

This object is not supported in Total Control Manager.

R2-Venezuela Configuration

A new enumeration value is added to *idt1CfgProjectSelectionR2* in the IDT1 MIB to support Venezuela (new value is in bold):

```
idt1CfgProjectSelectionR2 OBJECT-TYPE
SYNTAX INTEGER{
  q421(1),
  r2(2),
  r2Korea(3),
  p7(4),
  r2Malaysia(5),
  r2Brazil(6),
  r2Mexico(7),
  r2China(8),
  r2LME(9),
  r2Venezuela(10),
  notApplicable(255)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"This object lets user select E1 CAS project options.
Default = mfcBkR2(3)."
::= { idt1CfgEntry 5 }
```

Setting project selection through Total Control Manager

1. On the E1/R2 card, select the span.
2. From the **Configure** menu, select **Programmed Settings**.
3. From the **Parameter Group**, select **E1 R2 Configuration Group**.
4. From **Project Selection**, select the appropriate setting.
5. Click **Set**.

NMC support for HiPer ARC version 4.1

New NMC support for HiPer ARC version 4.1 includes a single configuration file. An entry is also added to the Chassis MIB (CHS.MIB) to support the latest HiPer ARC software.

Details of these features are in this section.

Single Configuration File

You may create and store a single file in the 3Com-proprietary CFM format that contains configuration information for the HiPer ARC. This file allows you to use SNMP to:

- Configure one or more HiPer ARC NACs by downloading the file from a MS to the NMC, and then to the HiPer ARC
- Compare different HiPer ARC configurations
- View the HiPer ARC configuration (by uploading the file from the NMC to a MS)

This feature is triggered through an SNMP browser and requires a TFTP script for file transfer. After the CFM file is transported to the NMC, the NMC will append additional data to the file as opaque data.

The HiPer ARC configuration file is present in the single generic file as opaque data (refer to the FILE MIB in the *NMC SNMP and MIB Reference*).

Refer to *Single generic configuration file* in this chapter for additional information.

New Chassis MIB Entry

This entry was added to the known entities in the Chassis MIB:

```
uchasVPNGwyEntity OBJECT IDENTIFIER::={uchasKnownEntities 60}
```

There are no MIBs for the HiPer ARC software, and the NMC only identifies this software type to be reported to Total Control Manager.

NMC Support for HiPer DSP

New NMC support for the HiPer DSP modems includes:

- New hardware and software ID number
- Support for new HiPer DSP protocols
- Support for NFAS
- Support for E1-R2 CAS cards
- Synchronized NMC and HiPer DSP default values
- Adjustable transmit level option
- Support for V.90 power level setting configuration

Details of these features are in this section.

New Hardware and Software ID Numbers

New hardware and software ID numbers are added. These numbers are only visible to the chassis. The new hardware ID is 51 decimal. The new software ID is 46 decimal.

Support for New HiPer DSP Protocols

A new object, hdmCcEnableV120v42Bis, is added to support V.42bis over V.120 and the LLC IE message for ISDN HDLC protocols.

V.42bis over V.120

```
hdmCcEnableV120v42Bis OBJECT-TYPE
SYNTAX INTEGER{
  disable(1),
  enable(2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"V.42bis Compression over V.120."
::= { hdmCcEntry 43 }
```

Setting V.42bis over V.120 through Total Control Manager

1. On the HiPer DSP card, select the template.
2. From the **Configure** menu, select **Programmed Settings**.
3. From the **Parameter Group**, select **Call Control Options**.
4. Select **V.42bis Compression over V.120**, then select **enable**.
5. Click **Set**.

LLC IE message for ISDN HDLC protocols

The HiPer DSP supports the LLC IE message for ISDN HDLC protocols.

Support for NFAS

NFAS (Non-Facility Associated Signaling) is a mechanism that allows multiple PRI span lines to negotiate their call set-up via a single D channel. You can also use NFAS to add a back-up D channel. NFAS is required for Switched-1536 data service. It also provides cost savings over standard ISDN by minimizing the number of D channels that need to be routed while keeping the out-of-band signaling advantage offered by ISDN.

New objects

These objects are added to the RDS1 MIB to support NFAS:

- *usrds1CfgNFASInterfaceId*
- *usrds1CfgSigGroupNumber*
- *usrds1CfgNFASSpanType*
- *usrds1CfgSigGroupType*
- *usrds1StatNFASSpanState*

```
usrds1CfgNFASInterfaceId OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-write
STATUS optional
DESCRIPTION
"This object indicates external ( assigned by Telco )
interface id. It is one of the entries in the NFAS group
table. "
::= { usrds1ConfigEntry 28 }

usrds1CfgSigGroupNumber OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-write
STATUS optional
DESCRIPTION
"Indicates NFAS ( or SS7 ) logical group number to be
monitored ( configured).  This variable serves as an index
for entering NFAS group table. "
::= { usrds1ConfigEntry 29 }

usrds1CfgNFASSpanType OBJECT-TYPE
SYNTAX INTEGER{
```

```
  fas(1),
  dChannelPrimary(2),
  dChannelBackUp(3),
  dChannelNone(4)
}
ACCESS read-write
STATUS optional
DESCRIPTION
"Span type with regard to the D-Channel: Primary, Back-up,
None or FAS (the latter in case if NFAS is not configured).
It is one of the entries in the NFAS group table. "
::= { usrds1ConfigEntry 30 }

usrds1CfgSigGroupType OBJECT-TYPE
SYNTAX INTEGER{
  fas(1),
  nfas(2),
  ss7(3)
}
ACCESS read-write
STATUS optional
DESCRIPTION
"This variable indicates one of the three possible
application types: FAS, NFAS, or FAS. FAS stands for
facility associated signaling. "
::= { usrds1ConfigEntry 31 }

usrds1StatNFASSpanState OBJECT-TYPE
SYNTAX INTEGER{
  none(1),
  is(2),
  stby(3),
  mb(4),
  moos(5),
  wait(6),
  oos(7)
}
ACCESS read-only
STATUS optional
DESCRIPTION
"This object displays the span's current state with regard
to D-channel. "
::= { usrds1StatEntry 27 }
```

Setting NFAS configuration through Total Control Manager

1. On the HiPer DSP card, select the span line settings.
2. From the **Configure** menu, select **Programmed Settings**.
3. From the **Parameter Group**, select **NFAS Settings**.
4. Enable the appropriate options.
5. Click **Set**.

New trap enables

These trap enables are added to the RDS1 MIB to support NFAS:

- *usrds1EventNfasDchSwStart*
- *usrds1EventNfasDchSwEnd*
- *usrds1EventNfasDchSwfail*

```
usrds1EventNfasDchSwStart OBJECT-TYPE
SYNTAX INTEGER{
  enableTrap(1),
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS optional
DESCRIPTION
"Enable the generation of an SNMP trap upon detection of a
begining of D-channel switch-over process. "
::= { usrds1EventCfgEntry 23 }

usrds1EventNfasDchSwEnd OBJECT-TYPE
SYNTAX INTEGER{
  enableTrap(1),
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS optional
DESCRIPTION
"Enable the generation of an SNMP trap upon detection of the
ending of D-channel switch-over process. "
::= { usrds1EventCfgEntry 24 }
```

```
usrds1EventNfasDchSwfail OBJECT-TYPE
SYNTAX INTEGER{
  enableTrap(1),
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS optional
DESCRIPTION
"Enable the generation of an SNMP trap upon detection of a
failure of the D-channel switch-over process. "
::= { usrds1EventCfgEntry 25 }
```

Setting NFAS trap enables through Total Control Manager

1. On the HiPer DSP card, select the span line settings.
2. From the **Fault** menu, select **Trap Settings**.
3. From the **Parameter Group**, select **NFAS**.
4. Enable the appropriate traps and/or logs.
5. Click **OK**.

New traps

These traps are added to the CHS_TRAP MIB to support NFAS:

- *rds1EvDchSwitchOverStart*
- *rds1EvDchSwitchOverEnd*
- *rds1EvDchSwitchOverFailure*

rds1EvDchSwitchOverStart Trap

```
rds1EvDchSwitchOverStart TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
uchasEntityIndex,
uchasEntityObjectID
}
```

```
DESCRIPTION
"This event is sent to indicate the begining of D-channel
switch-over process."

--#TYPE "NFAS D-channel switch-over start"
--#SUMMARY "%d; A D-Channel switch-over process started in
Slot %d, Channel %d"
--#ARGUMENTS { 1 3 4 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 122
```

rds1EvDchSwitchOverEnd Trap

```
rds1EvDchSwitchOverEnd TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
uchasEntityIndex,
uchasEntityObjectID
}
DESCRIPTION
"This event is sent to indicate the End of D-channel
switch-over process."

--#TYPE "NFAS D-channel switch-over end"
--#SUMMARY "%d; A D-Channel switch-over process ended in Slot
%d, Channel %d"
--#ARGUMENTS { 1 3 4 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 123
```

rds1EvDchSwitchOverFailure Trap

```
rds1EvDchSwitchOverFailure TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
```

```
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
uchasEntityIndex,
uchasEntityObjectID
}
DESCRIPTION
"This event is sent to indicate the Failure of D-channel
switch-over process."

--#TYPE "NFAS D-channel switch-over failure"
--#SUMMARY "%d; A D-Channel switch-over process failed in
Slot %d, Channel %d"
--#ARGUMENTS { 1 3 4 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE DEGRADED
 ::= 124
```

Support for E1-R2 CAS Cards

The HDR2 MIB is added to support E1-R2 CAS (Channel Associated Signaling) cards. The HiPer DSP employs R2 line signaling to achieve basic call set-up and teardown. Line signaling performs no transmission of numbers or other call details.

HiPer DSP employs Multi-Frequency Compelled (MFC) register signalaing to transmit the called numbers, calling numbers, and other information between exchanges using in-band multi-frequency signaling.

Refer to the *NMC SNMP and MIB Reference* for more information and the *NMC Parameter Reference* for a complete listing of all HDR2 MIB objects.

New Trap Enables

These trap enables in the HDR2 MIB support E1-R2 CAS cards:

- *hdr2TeMultiFrame*
- *hdr2TeMultiFrameClr*
- *hdr2TeRemMultiFrame*
- *hdr2TeRemMultiFrameClr*

These traps are enabled automatically in Total Control Manager. this may not be true!!!

```
hdr2TeTable OBJECT-TYPE
SYNTAX SEQUENCE OF Hdr2TeEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION
"A table containing the Trap Enable configurations for the
Hiper E1/R2 Module."
::= { hdr2Te 1 }

hdr2TeEntry OBJECT-TYPE
SYNTAX Hdr2TeEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION
"Trap Enable Configuration parameters for the E1/R2 module.
"
INDEX { hdr2TeIndex }
::= { hdr2TeTable 1 }

Hdr2TeEntry ::=
SEQUENCE {
hdr2TeIndex
    INTEGER,
hdr2TeMultiFrame
    INTEGER,
hdr2TeMultiFrameClr
    INTEGER,
hdr2TeRemMultiFrame
    INTEGER,
hdr2TeRemMultiFrameClr
    INTEGER
}

hdr2TeIndex OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
"The index which uniquely identifies the Hiper E1/R2 CARD to
which this entry is applicable. "
::= { hdr2TeEntry 1 }

hdr2TeMultiFrame OBJECT-TYPE
```

```
SYNTAX INTEGER{
  enableTrap(1),
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Enable the generation of an SNMP trap upon detection of a
Multi-Frame Alignment condition on the specified DS1. "
::= { hdr2TeEntry 2 }

hdr2TeMultiFrameClr OBJECT-TYPE
SYNTAX INTEGER{
  enableTrap(1),
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Enable the generation of an SNMP trap upon detection of
clearing a Multi-Frame Alignment condition on the specified
DS1. "
::= { hdr2TeEntry 3 }

hdr2TeRemMultiFrame OBJECT-TYPE
SYNTAX INTEGER{
  enableTrap(1),
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Enable the generation of an SNMP trap upon detection of a
Remote Multi-Frame Alignment condition on the specified
DS1. "
::= { hdr2TeEntry 4 }

hdr2TeRemMultiFrameClr OBJECT-TYPE
SYNTAX INTEGER{
  enableTrap(1),
```

```
  disableAll(2),
  enableLog(3),
  enableAll(4)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Enable the generation of an SNMP trap upon detection of
clearing a Remote Multi-Frame Alignment condition on the
specified DS1. "
::= { hdr2TeEntry 5 }
```

New traps

These traps are added to the CHS_TRAP MIB to support R2:

- *uds1MultiFrame*
- *uds1RemoteMultiFrame*
- *uds1MultiFrameClr*
- *uds1RemoteMultiFrameClr*

uds1MultiFrame Trap

```
uds1MultiFrame TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
uchasEntityIndex,
uchasEntityObjectID
}
DESCRIPTION
"A Multi Frame Misalignment has occured on the specified
DS1."
--#TYPE "Multi Frame Misalignment"
--#SUMMARY "%d; Multi Frame Misalignment occurred on Slot %d,
DS1 Channel %d"
--#ARGUMENTS { 1 3 4 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE DEGRADED
```

```
 ::= 104
```

uds1RemoteMultiFrame Trap

```
uds1RemoteMultiFrame TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
uchasEntityIndex,
uchasEntityObjectID
}
DESCRIPTION
"The remote terminal has detected a Multi Frame Misalignment
occurence on the specified DS1."
--#TYPE "Remote Multi Frame Misalignment"
--#SUMMARY "%d; Remote Multi Frame Misalignment occurred on
Slot %d, DS1 Channel %d"
--#ARGUMENTS { 1 3 4 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE DEGRADED
 ::= 105
```

uds1MultiFrameClr Trap

```
uds1MultiFrameClr TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
uchasEntityIndex,
uchasEntityObjectID
}
DESCRIPTION
"A Multi Frame Misalignment condition has cleared on the
specified DS1."
--#TYPE "Multi Frame Misalignment Cleared"
--#SUMMARY "%d; Multi Frame Misalignment cleared on Slot %d,
DS1 Channel %d"
--#ARGUMENTS { 1 3 4 }
--#SEVERITY INFORMATIONAL
```

```
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 106
```

uds1RemoteMultiFrameClr Trap

```
uds1RemoteMultiFrameClr TRAP-TYPE
ENTERPRISE usr
VARIABLES{
nmcTrapSequenceNumber,
nmcStatEventId,
nmcGmtime,
uchasSlotIndex,
uchasEntityIndex,
uchasEntityObjectID
}
DESCRIPTION
"The remote terminal had detected a clearing of the Multi
Frame Misalignment condition on the specified DS1."
--#TYPE "Remote Multi Frame Misalignment Cleared"
--#SUMMARY "%d; Multi Frame Misalignment cleared on Slot %d,
DS1 Channel %d"
--#ARGUMENTS { 1 3 4 }
--#SEVERITY INFORMATIONAL
--#TIMEINDEX 2
--#HELP "nmm.hlp"
--#HELPTAG 9999
--#STATE OPERATIONAL
 ::= 107
```

Synchronized NMC and HiPer DSP Default Values

NMC and HiPer DSP default values are synchronized at both the channel- and template-level to ensure efficient product configuration.

Adjustable Transmit Level Option

The HiPer DSP supports S.39, transmit level adjust. The object *hdmLiTransmitLevel* is added to the hdmLiTable to support this option at the HiPer DSP template-level:

```
hdmLiTransmitLevel OBJECT-TYPE
SYNTAX INTEGER (0..20)
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Specifies the signal level of the modem transmitter in
```

```
negative db."
::= { hdmLiEntry 9 }
```

Setting the transmit level through Total Control Manager

1. On the HiPer DSP card, select the template settings.
2. From the **Configure** menu, select **Programmed Settings**.
3. From the **Parameter Group**, select **Line Interface Options**.
4. Select **Transmit Level (-db)**, then make the appropriate setting.
5. Click **Set**.

Support for V.90 Power Level Setting Configuration

Two objects, *hdmScTxPwrLvl* and *hdmScTxPwrLvlApplied,* are added to the hdmScTable to allow configuration of V.90 power level settings:

```
hdmScTxPwrLvl OBJECT-TYPE
SYNTAX INTEGER (0..31)
ACCESS read-write
STATUS mandatory
DESCRIPTION
"This object configures the power level. "
::= { hdmScEntry 62 }

hdmScTxPwrLvlApplied OBJECT-TYPE
SYNTAX INTEGER{
  inputToFarEndOfCodec(1),
  outputOfTheServerModem(2)
}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"This object configures where the power level is applied."
::= { hdmScEntry 63 }
```

NMC Support for HiPer TRAX

The TRAX MIB is added to support the HiPer TRAX application on HiPer ARC. Refer to the *NMC SNMP and MIB Reference* for additional information.

A new hardware and software ID is also added to support the HiPer TRAX application. The hardware ID is 53 decimal. The software ID is 49 decimal. These ID numbers are only visible to the chassis.

2 NETWORK MANAGEMENT CARD OVERVIEW

This chapter provides an overview of the HiPer Network Management Card (NMC).

Product Description

The NMC Network Application Card (NAC) manages all of the devices installed in the Total Control chassis. The NMC operates under the direction of management software running on a PC/workstation (referred to as the Management Station, or MS).

Two protocols are used to implement management functions: one between the NMC and the MS (called Simple Network Management Protocol), and a second between the NMC and the managed devices (called Management Bus Protocol).

Simple Network Management Protocol

The NMC communicates with the MS by way of Simple Network Management Protocol (SNMP) version 1. THE NMC acts as a proxy agent for other NACs in the chassis that are not running an SNMP agent directly.

The NMC receives requests from the MS. These requests are articulated by Management Information Bases (MIBs) defined for each device in the chassis. In this capacity, the NMC acts as a proxy agent for the other NACs in the chassis. The NMC then carries out the requests and obtains results using the proprietary 3Com Management Bus Protocol (MBP), and uses SNMP to return the results to the MS.

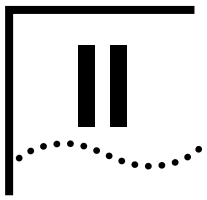
Event Management

The NMC also can perform event management. Standard SNMP traps can be enabled to send a trap message or event notification to one or more Management Stations. The MS uses these traps for logging and alarms.

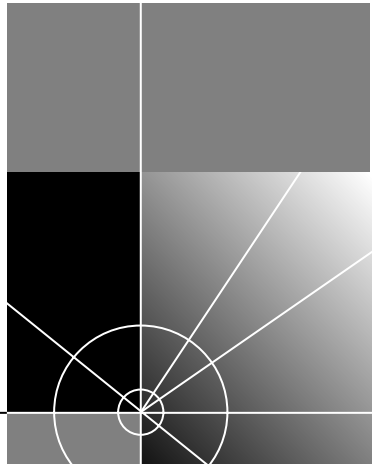
Management Bus Protocol

The NMC uses the 3Com proprietary MBP to communicate to the installed chassis devices. The NMC provides this functionality within the chassis via MBP:

- NAC configuration management
- Automatic NAC configuration upon installation
- NAC configuration queries
- NAC software upgrades
- Security and accounting management (RADIUS)
- Performance management
- Fault management



CUSTOMIZING NMC CONFIGURATIONS WITH THE USER INTERFACE



3 SETTING NMC SECURITY

NMC security settings include SNMP read and write strings, user interface (UI) password protection, the IP address authorized access list, and the RADIUS secret key.

This chapter provides instructions for using the NMC's console port to set basic card security. Typically, once you make these settings, you do not need to change them for daily operation.

Refer to the *HiPer NMC Getting Started Guide* for instructions about how to connect to the NMC's console port and establish a terminal session.

Setting the SNMP Read and Write Strings

Configure the community strings to allow or restrict NMC access through both the UI and Total Control Manager.

"Public" allows read-only access. "Private" allows read and write access.

1. From the **Configuration** menu, type **5**, then press **ENTER** to access the **Local SNMP Community Strings** menu.
2. Type **1**, then press **ENTER** to access the **SNMP Read-Only (Public) Community String** menu.
3. Accept the default community string ("public") by pressing **ESC**.
   or
   Type the New Public Community String (up to 32 characters), then press **ENTER**.
4. Type **2**, then press **ENTER** to access the **SNMP Read-Write (Private) Community String** menu.
5. Accept the default community string ("private") by pressing **ESC**.
   or
   Type the New Private Community String (up to 32 characters), then press **ENTER**.

6 From the **Configuration** menu, type **9** to access the **Save Configuration to Non-Volatile Memory** menu, then press **ENTER** to save the configuration to NVRAM.

Enabling UI Password Protection

You may enable a password to restrict access to the NMC UI and the configuration menus. You must set the community strings before you can enable UI password protection. Refer to "*Setting the SNMP read and write strings*".

When the password feature is enabled, two levels of security exist: read-only access (SNMP read string) and read and write access (SNMP write string).

DIP switch 6 must be ON to enable a password. Refer to the HiPer NMC Getting Started Guide.

1 From the **Configuration** menu, type **14**, then press **ENTER** to access the **Password Screen Enable/Disable** menu.

2 Accept the current status by pressing **ESC**.
or
Type **1** to enable or **2** to disable the password on power-up, then press **ESC** twice.

Re-initializing the Authorized Access List

The authorized access list is a restricted list of those IP addresses that are allowed to access the NMC. IP addresses that are not on this list are not allowed to access the NMC. Maintain this list through Total Control Manager or another SNMP manager.

The only way to delete IP addresses or reset this list is to re-initialize (delete) it. You must use the NMC console port to re-initialize the list; you cannot re-initialize the list from Total Control Manger or an SNMP browser. Once you re-initialize the list, use Total Control Manager or an SNMP browser to enter authorized IP addresses.

Until you set a new authorized access list, any IP address will have full access rights to the NMC. This will make your network less secure.

CAUTION: *This procedure will clear the authorized access list. All IP addresses will have full access to the NMC until you set a new authorized access list.*

1. From the **Configuration** menu, type **8**, then press **ENTER** to access the **Reinitialize Authorized Access List** menu.
2. Press **ESC** to exit without changing the authorized access list.
   or
   Press **ENTER** to reinitialize the authorized access list, then press **ESC**.
3. From the **Configuration** menu, type **9** to access the **Save Configuration to Non-Volatile Memory** menu, then press **ENTER** to save the configuration to NVRAM.

Enabling the RADIUS Secret Key

If you are using the additional-cost RADIUS security feature, you may choose to enable a security secret key to encrypt data sent between the server and client. The RADIUS secret key is a series of up to 64 keystrokes known only to the NMC and RADIUS security server. The secret key is never displayed, so you must remember what it is, note it, and keep the information in a secure place.

Refer to the RADIUS documentation for additional information about the secret key.

1. From the **Configuration** menu, type **7**, then press **ENTER** to access the **RADIUS Secret Key** menu.
2. Type the New Secret Key, then press **ESC**.
3. From the **Configuration** menu, type **9** to access the **Save Configuration to Non-Volatile Memory** menu, then press **ENTER** to save the configuration to NVRAM.

4 Configuring the NMC SLIP Port for Remote Chassis Management

This chapter provides instructions for configuring the NMC's Serial Line Internet Protocol (SLIP) port to allow remote chassis management through Total Control Manager or an SNMP browser. The SLIP port uses the Wide Area Network (WAN) protocol.

Typically, once you make these settings, you do not need to change them for daily operation.

Refer to the *HiPer NMC Getting Started Guide* for instructions about how to connect to the NMC's console port and establish a terminal session.

Guidelines for Preparing to Use the SLIP Port

This section includes guidelines for preparing to use the NMC SLIP port for remote chassis management. Actual set-up may vary based on your installation and facilities.

Assign the SLIP Port's IP Address and Subnet Mask

From the NMC's console port (UI), assign the IP address and the subnet mask that you will use to access the NMC from a remote location.

1. From the **Configuration** menu, type **2**, then press **ENTER** to access the **Local WAN IP Address** menu.

The SLIP port uses WAN protocol. The WAN IP address and subnet mask represent the IP address and subnet mask of the NMC SLIP port.

2. Type **1**, then press **ENTER** to access the **WAN IP Address** menu.
3. Accept the Current WAN IP Address by pressing **ESC**.
   or
   Type the New WAN IP Address, then press **ENTER**.
4. Type **2**, then press **ENTER** to access the **WAN IP Subnet Mask** menu.
5. Accept the Current WAN IP Subnet Mask by pressing **ESC**.
   or
   Type the New WAN IP Subnet Mask, then press **ENTER**.

Enable LAN/WAN Routing

Set routing to "Enable" to allow remote access to the entire LAN through the NMC SLIP port. Set routing to "Disable" to disable LAN routing and only allow remote access to the NMC for chassis management.

1 From the **Configuration** menu, type **10**, then press **ENTER** to access the **Enable/Disable Routing between LAN&WAN** menu.

2 Accept the current status by pressing **ESC**.
or
Type **1** to enable or **2** to disable routing between the LAN and WAN, then press **ESC** twice.

Set the Inactivity Timer

If there is no activity on a SLIP port for the specified time duration, the port drops the connection. This applies to both dialed-out and dialed-in calls. If the console remains inactive for the duration, the display reverts back to the password request screen.

If Total Control Manager or another management device connected to the physical port is polling at rates lower than the specified inactivity time, the connection will not be dropped.

The timer default is three minutes. The maximum value you may set is 1440 minutes. To disable the local inactivity timer feature, enter a value of zero.

1 From the **Configuration** menu, type **13**, then press **ENTER** to access the **Local INACTIVITY TIME** menu.

2 Accept the current UI/SLIP port inactivity timer setting by pressing **ESC**.
or
Type the new timer value, then press **ENTER**.

Save the Configuration to NVRAM

From the **Configuration** menu, type **9** to access the **Save Configuration to Non-Volatile Memory** menu, then press **ENTER** to save the configuration to NVRAM.

Prepare the NMC

Complete these additional steps to prepare the NMC (refer to the *NMC Getting Started Guide* for illustrations, DIP switch locations, and cabling information):

- Set NMC DIP switch 6 to OFF to allow the console port to be used for SLIP operation. At this point, you will not be able to use the console port as a user interface port.

- Set NMC DIP switches 3 and 4 to the appropriate baud rate for the SLIP port
- Connect the NMC SLIP port to an external modem
- Connect the external modem to an analog POTS line

Prepare the Remote Management Station

Complete these items to prepare the remote management station:

- Connect the management station to an external modem
- Connect the external modem to an analog POTS line
- Set up a new dial-up connection to the NMC
  - Set the server type port option on the management station to a SLIP connection
  - In the management station's TCP/IP settings, set the IP address of the management station to one number higher than that of the NMC's SLIP port
  - Set the baud rate to match the NMC's SLIP port baud rate
  - Establish a dial-up connection with the NMC
- Open Total Control Manager or an SNMP browser

Configuring the second SLIP port

If you are using an Ethernet Network Interface Card (NIC) with the NMC, you may configure the CH1 port to serve as a second SLIP port. This feature allows you to manage the chassis from two remote locations.

Configure CH1 as the Second SLIP Port

DIP switch 6 must be set to ON to enable CH1 to serve as a second SLIP port. Refer to the NMC Getting Started Guide.

1. From the **Configuration** menu, type **11**, then press **ENTER** to access the **UI/SLIP Port Selection** menu.
2. Accept the current status by pressing **ESC**.
   or
   Type **1** to set as UI or **2** to set as SLIP, then press **ENTER**.

Set the IP Address and Subnet Mask

When CH1 is configured to be a SLIP port, the NMC NAC recognizes it as WAN2. For this port to function correctly, you must configure the WAN2 IP address and subnet mask. This is the address and subnet you will use to access the NMC remotely.

The SLIP port uses WAN protocol. The WAN IP address and subnet mask represent the IP address and subnet mask of the NMC SLIP port.

1. From the **Configuration** menu, type **12**, then press **ENTER** to access the **Local WAN2 IP Address** menu.
2. Type **1**, then press **ENTER** to access the **WAN2 IP Address** menu.
3. Accept the Current WAN2 IP Address by pressing **ESC**.
   or
   Type the New WAN2 IP Address, then press **ENTER**.
4. Type **2**, then press **ENTER** to access the **WAN2 IP Subnet Mask** menu.
5. Accept the Current WAN2 IP Subnet Mask by pressing **ESC**.
   or
   Type the New WAN2 IP Subnet Mask, then press **ENTER**.
6. From the **Configuration** menu, type **9** to access the **Save Configuration to Non-Volatile Memory** menu, then press **ENTER** to save the configuration to NVRAM.

5 Enabling Features and Activating Changes

You may use the console port or Total Control Manager to enable additional-cost features.

This chapter provides instructions for using the NMC's console port to enable additional-cost features. Typically, once you make these settings, you do not need to change them for daily operation.

Refer to the *HiPer NMC Getting Started Guide* for instructions about how to connect to the NMC's console port and establish a terminal session.

About Additional-cost Features

Several NMC features are available at additional cost from 3Com.These include (but are not limited to) hub security (RADIUS), cellular support, I-modem, Quad Modem Analog Fax for CDMA Circuit, and PIAFs for Quad Modem.

If you ordered any of these features, the NMC should ship from the factory with the features enabled. Use the **Feature Enable** menu to view purchased and enabled features. You may also add new added-cost features through this menu. Some features also require you to set an additional enable command through Total Control Manager.

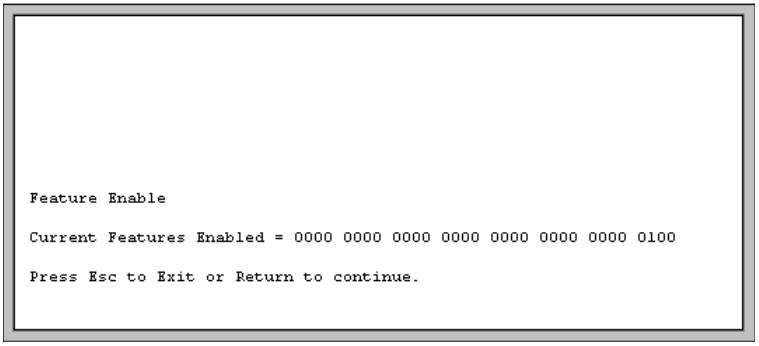
*You may also view enabled features through the **Added Cost** parameter group in Total Control Manager.*

Enabling Additional-cost Features in NMC

If you need to enable additional-cost features, contact your sales representative to obtain the appropriate feature enable string. You will need to provide the serial number of the NMC on which you are enabling the feature.

The feature enable string is also called an enable key.

1 From the **Main** menu, type **3**, then press **ENTER** to access the **Feature Enable** menu.



2 View the currently-enabled features, then press **ESC** to exit without change.
or
Type the New Feature Enable String, then press **ENTER**.

3 From the **Main** menu, type **1**, then press **ENTER** to access the **Configuration** menu.

4 From the **Configuration** menu, type **9** to access the Save Configuration to Non-Volatile Memory menu, then press **Enter** to save the configuration to NVRAM.

5 Reset/reboot the NMC to activate the configuration changes. Refer to *"Activating configuration changes"*.

Enabling Hub Security (RADIUS)

If you enable hub security, you must set the RADIUS secret key. Refer to "*Enabling the RADIUS Secret Key* in Chapter 3.

Activating Configuration Changes

Use the **Command Menu** to reset (reboot) the NMC.

1. From the **Main Menu**, type **2**, then press **ENTER** to access the **Command** menu.
2. Press **ESC** to exit with out change.
   or
   Type **1**, then press **ENTER** to access the **Reset** menu.
3. Press **ESC** to exit without resetting.
   or
   Type **Y** to reset the NMC.

6 PLANNING FOR ADDITIONAL NMC CONFIGURATION

Use Total Control Manager or an SNMP browser to complete additional NMC configuration. Refer to the *NMC Product Reference*, *NMC SNMP and MIB Reference*, and the Total Control documentation set for additional information.

When to use these planning tables

Use these planning tables before configuring additional NMC features. They are intended as a job aid to help you to gather all of the information you will need for configuration. Write the settings you will need in these tables. It is not mandatory that you complete these tables.

These are generic features for the NMC. Not all features included may apply to your installation.

NMC additional-cost features

Feature	Installed?
Cellular support	
Hub security (RADIUS)	
I-modem	
PIAFS for Quad Modem	
other	

AutoResponse Timer parameters

Feature	Default
Enable auto response timers	disabled

Use this table to record the appropriate times you plan to set for AutoResponse timers.

Timers	Start Date	Stop Date	Start Time	Stop Time	Timer Interval
Timer					
Timer					
Timer					
Timer					

Security Server parameters

Use these tables to record the values you plan to set.

Parameter	Value	Default
IP address		N/A
UDP port number		1645
Retries allowed client to reach server (1-3)		1

Hub security prompts

Prompt	Value
User name	
User password	
Dialback name	
Dialback password	
Dialback phone number	
Dialback pending	
Modem select	
Restricted phone number	

Hub security messages

Feature	Message
"Invalid login attempt"	
"Invalid modem" selected	
"Selected modem unavailable"	
"Security connection unsuccessful"	
"New password"	
"Password expired"	

Hub security responses

Parameter	Value	Default
Timeout (time users have to respond to prompt)		30 sec
Attempt limit (allowed user tries per prompt) (1–16)		1
Echo enable (send user responses back to user screen)		disabled

Other hub security parameters

Parameter	Value	Default
Dialback delay (wait between dialback attempts)		30 sec
Dialback attempt limit (1–100)		1
Modem login attempt limit (1–16)		3

Logging Group parameters

Use these tables to record the values you plan to set.

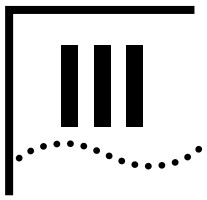
Parameter	Value	Default
Primary RADIUS accounting server IP address		N/A
Secondary RADIUS accounting server IP address		N/A
Event logging server (primary | secondary | none)		none
UDP port number of accounting server		1646
Logging client TX retry		3
Log group selection (none | 2 | 3 | 4 | 23 | 24 | 34 | all)		none (log group #1 is always sent, but you can send others)

Configuration Group parameters

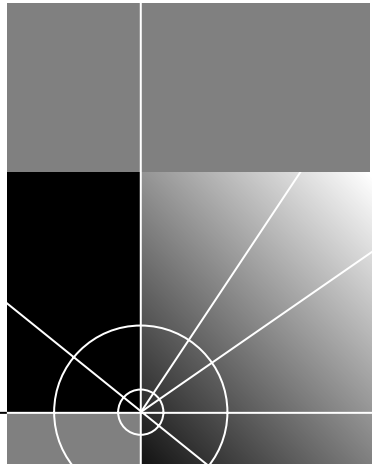
Parameter	Value	Default
System time / date / zone / GMT		N/A
Daylight savings time		enabled
Auto configuration card from NVRAM on power-up (recommended to be set to **disabled**)		disabled
Chassis name (string)		N/A
NMC LED display (4–character string)		3COM

Dial-out parameters

Parameter	Value	Default
AT init string		AT&FE0Q0&H &R2&B1V1
WAN connect number		null
WAN dialout attempt limit (0 = try until successful)		null
Pause between attempts to transmit packet		20
WAN port retry suspension interval		300 (sec)
Connection failures limit before suspension interval start		3



TROUBLE CLEARING



7 Using the Front Panel Indicators to Diagnose Problems and Trouble Clear

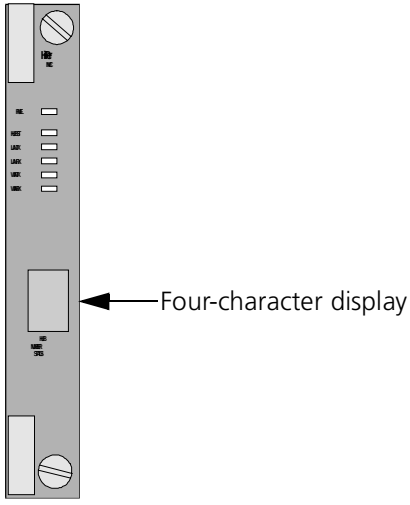
This chapter contains information about using the NMC front panel indicators to diagnose and clear operational problems.

HiPer NMC Front Panel Indicators

The HiPer NMC front panel contains two types of indicators that are useful for monitoring and troubleshooting: the chassis: status light emitting diodes (LEDs) and a four-character display.

The status LEDs are red, green, and orange, and are used to indicate status within a chassis environment. The four-character display is used to identify the hub name or number in a multi-chassis environment. You can also use the display for four-character messages.

Alphanumeric Four-character Display



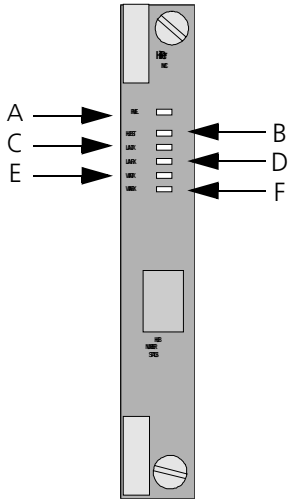
Four-character Display During NMC Power-up Sequence

This table shows the status of the four-character display during the NMC power-up sequence.

Event	Display
Power on	Blank
Hardware reset	Blank
Software download (SDL)	Blank
Internal diagnostics	TEST
Initializing LAN interface	WAIT
Normal operation	User-defined display

Diagnosing Trouble with the HiPer NMC LEDs

Use the HiPer NMC LEDs to diagnose power-up, boot, and connectivity errors. This table lists the NMC NAC LEDs and their possible conditions.



LED	Description	Status	Meaning
A	Run/Fail (RN/FL)	Solid green	Normal/diagnostics mode/boot-up self-test
		Solid red	Critical failure
		Flashing red/green	Non-critical failure on initial power-up
		Flashing green	Testing or software download (required or in process)/also during boot-up sequence
B	Hub Status	Solid green	Chassis normal/diagnostics mode
		Solid red	Chassis critical failure, including: ■ Fan failure ■ Temperature alert ■ Power supply failure ■ Management operational failure of chassis NACs (other than the NMC) ■ NMC operational status (try to perform a non-destructive self test through TCM or an SNMP browser)
		Flashing red	Management bus failure with card in chassis

LED	Description	Status	Meaning (continued)
C	LAN TX	Solid green	NMC transmitting data on LAN port
		OFF	No data being transmitted on LAN port
D	LAN RX	Solid green	NMC receiving data on LAN port
		OFF	No data being received on LAN port
E	WAN TX	Solid green	NMC transmitting data on WAN port
		OFF	No data being transmitted on WAN port
F	WAN RX	Solid green	NMC receiving data on WAN port
		OFF	No data being received on WAN port

LED States During NMC Power-up

This table details the status LEDs during the NMC power-up sequence. Identifying the correct portion of the start-up sequence can help when trouble clearing start-up problems and when talking with 3Com technical support.

Event	RN/FL	Hub Status	LAN TX	LAN RX	WAN TX	WAN RX
Power on	Solid red	Solid red	off	off	off	off
Hardware reset	Solid red	Solid red	off	off	off	off
Software download (SDL)	Flashing green	off	off	off	off	off
Diagnostics	Solid green	off	off	off	off	off
Critical failure	Solid red	N/A	N/A	N/A	N/A	N/A
Non-critical failure	Flashing red/green	N/A	N/A	N/A	N/A	N/A
Chassis critical failure	N/A	Solid red	N/A	N/A	N/A	N/A
Management bus failure	N/A	Flashing red	N/A	N/A	N/A	N/A
Normal operation	Solid green	Solid green	N/A	N/A	N/A	N/A
LAN transmitting	N/A	N/A	Solid green	N/A	N/A	N/A
LAN receiving	N/A	N/A	N/A	Solid green	N/A	N/A
WAN transmitting	N/A	N/A	N/A	N/A	Solid green	N/A
WAN receiving	N/A	N/A	N/A	N/A	N/A	Solid green

Diagnosing Power-up Problems with the RN/FL LEDs

This table provides information on diagnosing power-up problems with the HiPer NMC RN/FL LEDs.

LED Status	Cause	Trouble Clearing Action
RN/FL LED is solid green	The condition is normal	No action required.
RN/FL LED is solid red	There is a critical failure	Re-install the NMC according to the instructions in this manual and refer to the "[Critical Failure Debug Procedure" on page 6 of this chapter.
RN/FL LED is flashing red and green	There is no NIC installed behind the HiPer NMC NAC	Install the NIC. Refer to the appropriate *Getting Started Guide*. *Note: If the NIC is installed after the NMC, reboot the NMC by removing and reseating the NMC card.*
RN/FL LED is not lit	There is no power to the NAC	■ Make sure the NMC is installed properly. ■ Make sure the chassis is powered on. ■ Make sure power supply status LED is green.

Diagnosing with the Hub Status LED

The red hub status LED can indicate a status failure for many areas of the chassis, including:

- Fan failure
- Temperature alert
- Power supply failure
- Management operational failure of chassis NACs (other than the NMC)
- NMC operational status

Checking NMC operational status

To further diagnose NMC operational status, perform a non-disruptive self-test using TCM or an SNMP browser. Non-disruptive means that you can perform this test while the NMC is online.

If this test fails, contact 3Com Technical Support.

Performing the non-disruptive self-test from TCM

1. Select the NMC card.
2. From the **Configure** menu, select **Actions/Commands**.
3. From the **Command to Execute:** menu, select **Software**, then select **Non-Disruptive Self-Test**.
4. Select the **Polling Interval**.
5. Click **Execute**.
6. Refer to the Total Control Manager documentation set for additional information.

Performing the non-disruptive self-test from an SNMP browser

Two objects are included in the NMC MIB for diagnosing the red hub status LED: use *nmcCmdFunction* and choose **nonDisruptSelfTest(5)** to perform the non-disruptive self-test; use *nmcStatRedLed* to gain additional information about the hub status LED.

Refer to the *NMC SNMP and MIB Reference* for additional information.

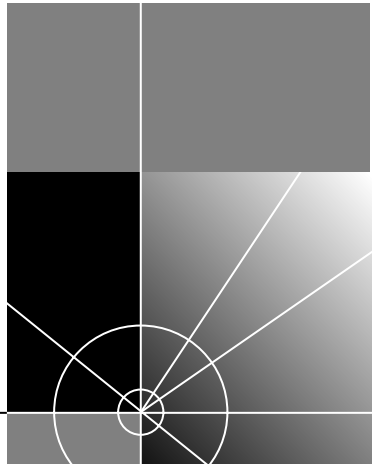
[Critical Failure Debug Procedure

Follow this procedure if you suspect a critical failure at start-up. A critical failure is indicated by a solid red RN/FL LED, and may be caused by a bus communication failure.

1. Pull the HiPer NMC NAC forward to unplug it from the midplane.
2. Pull the 10/100 Ethernet Aux I/O NIC backward to unplug it from the midplane.
3. Reseat the NIC.
4. Reseat the NAC.
5. Check to see if the RN/FL LED becomes green. If reseating the NIC and NAC does not solve the problem, call 3Com Technical Support.

IV

APPENDICES



A

HiPer NMC Technical Specifications

This appendix includes the HiPer NMC technical specifications.

Certifications

Complies with:

- FCC Part 15, Class A
- EN 55022, Class A
- EN 50082

Physical Specifications

Length:	12.95 in. (32.89 cm)
Width:	0.79 in. (2.0 cm)
Height:	6.9 in. (17.53 cm)

Environmental Specifications

Shipping and storage specifications

Temperature:	-13° to 167°F (-25° to 75° C)
Humidity:	0 to 100%, Non-condensing

Operating specifications

Temperature:	32° to 104°F (0° to 40° C)
Humidity:	0 to 95%, Non-condensing

Additional Specifications

Processor Pentium processor (P5) at 133 MHz

Operational Memory

Data Retention Method	Clock and CMOS retained by 3V microlithium cell
Dynamic Random Access Memory (DRAM)	16 MB
Flash Memory	8 MB

Measurement Accuracy

	Range	Tolerance
+5.2VDC	4.5 to 5.5VDC	± 45mv
-5VDC	-1.0 to -6.0VDC	± 45mv
+12.2VDC	6.0 to 13.4VDC	± 45mv
-12.2VDC	-6.0 to -13.4VDC	± 45mv
Temperature	+5° to +300°F (-15° to +148.5°C)	2.5° C

Midplane Connector 180-position, 4-row, high-density connector

NAC Management Bus 256 KHz (data clock), synchronous

PCI Bus 33 MHz

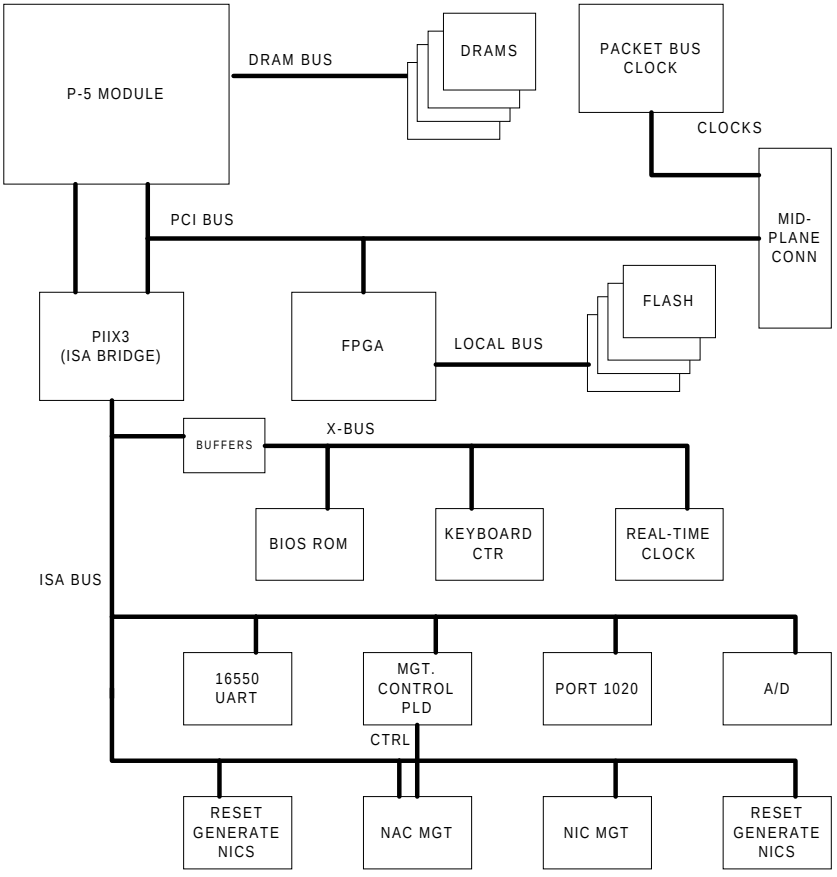
Power Requirements

Voltage	Current Draw (Typical Maximum*)
+5VDC	2.3 A
-5VDC	0.010 A
+12VDC	0.021 A
-12VDC	0.010 A

** "Typical Maximum" refers to the maximum current draw under most typical configurations.*

B

HiPer NMC System Block Diagram



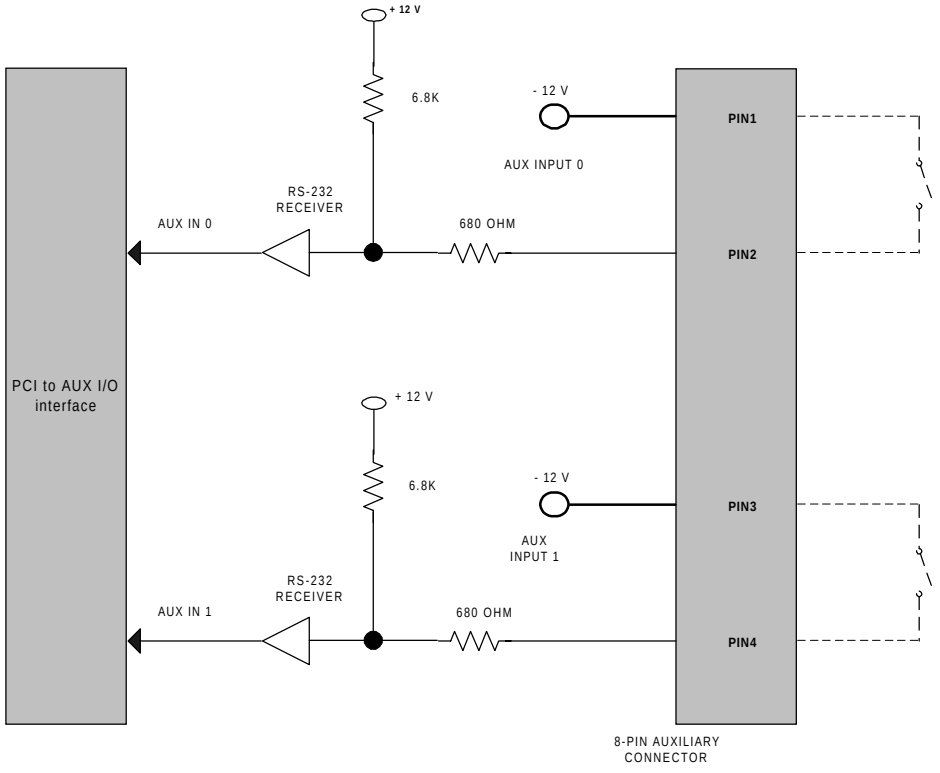
C

10/100 Ethernet Aux I/O Network Interface Card Inputs and Outputs

This appendix contains illustrations of the auxiliary inputs and outputs for the 10/100 Ethernet Aux I/O NIC.

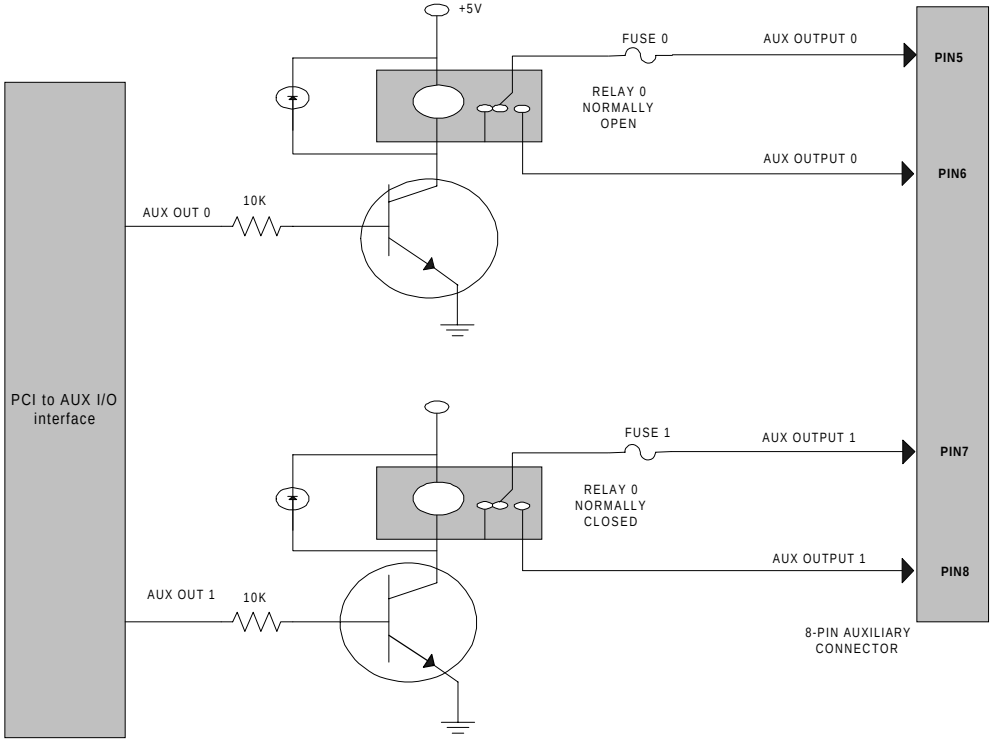
Auxiliary Inputs

This diagram shows the input ports. Use this information when specifying an auxilliary input connector.



Auxiliary Outputs

This diagram shows the output ports. Use this information when specifying an auxilliary output connector.



D UPGRADING SOFTWARE USING SOFTWARE DOWNLOAD-2

This appendix contains the instructions for using Software Download-2 (SDL-2) to upgrade HiPer NMC software.

Overview

SDL-2 is a utility that allows software download to a HiPer Network Application Card (NAC) to a Total Control chassis locally through the console port or remotely across a LAN or WAN.

Only use SDL-2 when downloading to HiPer NACs. All other software downloads use SDL-1 (PCSDL).

For more information on Total Control Manager, refer to the *Total Control Manager/SNMP Software Guide*.

The Download Process

This section describes what occurs during the download process. You may use one of these methods to complete an SDL-2:

SDL-2 Method	Use
Z-Modem download	Console port downlod using Z-Modem protocol directly from the host PC to the NAC
SNMP or Total Control Manager	Download across a LAN or WAN to the NMC using TFTP protocol through a MIB browser or Total Control Manager. The file then transfers automatically to the NAC using SNMP .

Typical Z-Modem Download

The typical Z-Modem download is a Z-Modem file transfer from the host PC to the target NAC. Use a terminal emulator package that supports Z-Modem Upon the completion of the Z-Modem transfer, a result message will be displayed regarding the success or failure of the file transfer. Successful transfer does not necessarily mean that the entire process has completed successfully, since the device may be done with

the file transfer, but it still could encounter an error programming NVRAM.

After the NAC is finished processing the download file, it will wait about five seconds, and an ASCII result string is sent to the serial port. This string displays on the emulator screen.

If the SDL-2 agent aborts the download operation, the final ASCII result message displays after the five-second delay.

Typical SNMP/Total Control Manager TFTP Download

You may also use an SNMP browser or Total Control Manager to download to a chassis NAC. In this case, the download occurs through the NMC, which then distributes the code to the appropriate NAC. When you select multiple devices, they must be of the same type (all HiPer DSP NACs, for example).

When completing this type of SDL-2, select the slots to which you want to download. If you are using a MIB browser, you issue a **set** command in the slot command table for each device selected.

If no TFTP occurs within the timeout, the SDL-2 trigger command aborts. As the TFTP progresses, the NMC distributes the download file contents to each of the selected devices. The final result is indicated in the SNMP command table's command result and command code objects. If you are using Total Control Manager, a "Success" message indictes the download is complete.

Downloading HiPer NAC Software using Z-Modem on the Console Port

You may use the Z-Modem transfer to download software through the console port.

Use this Software Download-2 (SDL-2) procedure if you are installing code to the NAC through the console port. SDL-2 is a utility that allows you to download software to a NAC via a serial link, both locally and remotely.

What You Will Need

To send the new code to the NAC from the console port, you need a standard terminal program that can send files using the Z-Modem protocol.

Checking the Software Version

From the console interface, type **ATI7** at a modem prompt to display product configuration.

To determine the version of software, view the date and revision of the Boot Block, Board Manager, ACP, and DSP.

System Requirements

You need a computer with a serial port capable of link rates up to 115200 bps, and a null modem cable with RJ45 and RS–232 on each end.

Preparing to Download Software

Set your terminal program to 115200 bps, 8 bits per character, no parity, one stop bit, and hardware flow control on.

*HiPer DSP link port speeds are fixed and cannot be changed. Although either port can be used, 3Com recommends using the **Auxiliary port**, which has a standard 115200 bps interface, when downloading software.*

Downloading the Software

Complete these steps to download software:

If power to the NAC is removed during any of these software download steps, you must restart the procedure.

1. Run a terminal program that supports the Z-Modem transfer protocol (such as HyperTerminal).
2. Manually reboot the NAC by pulling the card forward and re-inserting it.
3. When the "Enter Download Trigger" prompt appears, type **AT{Z}** in all capital letters. Using your terminal program, select and send the boot file.

*Use all capital letters when typing **AT{Z}.** Begin sending the files before the hardware reboot cycle begins. The hardware reboot cycle begins about 10 seconds after you manually reboot the card.*

The download is complete when you see this message:

```
"0;Download successful: File=file.dmf"
Programming flash
"!!-----> SDL2 for the PPC403 <-------!!"
"__ Enter Download Trigger __"
```

Do not input the trigger when prompted the second time unless you want to re-download the code.

The software download is now complete.

Downloading HiPer NAC Software through an SNMP MIB Browser

You can use an SNMP MIB browser to transfer software across a LAN or WAN to a chassis NAC.

System Requirements

You will need a MIB browser and a TFTP server. Both must be able to communicate with the NMC NIC over an Ethernet or SLIP connection.

Before You Start

1. Make a backup copy of the software disk (if you received a disk).
2. Set up the MIB browser. Select the uchasCmdTable (1.3.6.1.4.1.429.1.1.7.1) table within the uchasCmd group of the CHS MIB. This MIB is in the private.enterprises.usr.nas.MIB.tree.
3. Set up the TFTP server. Set the server to binary transfer mode with a connection to the IP address of the NMC.
4. Verify that the NMC NIC is configured for remote access. Refer to the appropriate NIC getting started guide.

Temporarily Copy the Software onto the Management Station

If you are downloading the software from the 3Com website, you need to temporarily copy the files to the management station. If you are using Total Control Manager on the management station, install to C:\USRSUITE\SDL2. Otherwise, install the software to an appropriate temporary directory.

Start the Download

Complete these steps:

1. Determine the slot number of the HiPer card to which you will download.
2. Connect to the chassis.
3. Using the MIB browser, select the *uchasCmdFunction* object.
4. Choose the index for the slot number of the card to which you will download.
5. Issue a **SET** request command to set the object to enumeration 6, which corresponds to softwareDownload2. For example, if the card to which you are downloading is in slot number 3, SET *uchasCmdFunction.3* to "6".

6 Once the **SET** is issued, and the response is received, the *uchasCmdResult* object will indicate whether the download can proceed.

7 Issue a **GET** or **GET-NEXT** request to the *uchasCmdResult* object that contains the index number used in step 3. For example, if the card to which you are downloading is in slot number 3, the **GET** or **GET-NEXT** request will be completed on *uchasCmdResult*.*3*.

8 If the result of the GET or GET-NEXT request is "3", which corresponds to the InProcess enumeration, then continue this procedure. Otherwise, check the status of the card to which you are downloading and begin again at step 3.

9 Start the TFTP session.

Checking status of the download

Determine status of the download by checking *uchasCmdResult* as explained in step 4.

The possible *uchasCmdResult* values are:

Enumeration	Download status
none (1)	No result
success (2)	Successful download
inProgress (3)	Download in progress
notSupported (4)	Not supported
unAbleToRun (5)	Unable to complete the download - card may be in a bad state or not accepting the download
aborted (6)	Download file was aborted
failed (7)	Download failed

Downloading HiPer NAC Software through Total Control Manager

You can use Total Control Manager to transfer software across a LAN or WAN to a chassis NAC.

Connect to the NMC

Connect the management station to the NMC NIC through a LAN, WAN, or SLIP connection. Refer to the *NMC Getting Started Guide*. You may also need to configure the NMC SLIP port. Refer to the *NMC Getting Started Guide* or the *Total Control Manager Getting Started Guide*.

Temporarily Copy the Software onto the Management Station

If you are downloading the software from the 3Com website, you need to temporarily copy the files to the management station. If you are using Total Control Manager on the management station, install to C:\USRSUITE\SDL2. Otherwise, install the software to an appropriate temporary directory.

Checking the Software Version

Follow these steps to check the currently-installed version of software:

1. Open Total Control Manager and select the appropriate chassis.
2. From the **Configure** menu, select **Programmed Settings**, then select **Inventory...**.
3. Check the software version of the appropriate NAC after the inventory is complete.

Start the Software Download

Complete these steps:

Before starting a Total Control Manager SDL-2 , unload any terminate-and-stay-resident (TSR) programs, including Windows. TSRs will greatly slow SDL-2 .

1. Launch the Total Control Manager application.
2. Establish a connection with the chassis.
3. From the device display on the Total Control Manager/SNMP console, select the card(s) to which you want to perform the download.

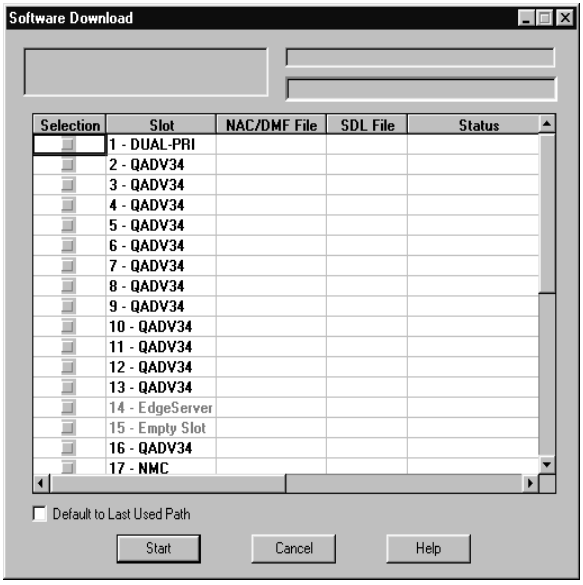
Once you enter the Software Download window, you may select or deselect cards by their card number.

4. On the **Configure** menu, click **Software Download**. Or, click the **Software Download** icon on the toolbar. Refer to *About the Software Download window* for more information.
5. Click **Start** to begin the download.
6. During the download, progress messages are displayed in the upper-left corner of the Software Download window. A "Success" message indicates the download is complete.

*If you selected more than one card to which to download, and you want to cancel the download before it completes, click **STOP**. The in-progress SDL-2 continues, but the remaining downloads are canceled. Corresponding checkmarks are removed.*

About the Software Download window

This is the software download window that you see when completing SDL-2 from Total Control Manager.



Function	Column name
Allows you to select multiple cards on which to perform downloads. A checkmark appears for every selected card. The device display also shows a checkmark for each selected card.	Selection
Lists the chassis slot number and card name.	Slot
Lists the most current .NAC or .DMF file in the C:\USRSUITE\SDL2 directory. If the software version is more recent than the NAC itself, the entry appears in red.	NAC/DMF File
Lists the most current .SDL file in the C:\USRSUITE\SDL-2 directory. If the software version is more recent than the NAC itself, the entry appears in red.	SDL File
Lists the status of each software download on a card-by-card basis. Messages include “In Progress” and “Complete”.	Status

If you have loaded or moved the .NAC, .DMF, or .SDL files to a directory other than C:\USRSUITE\SDL , position the cursor in the appropriate column in the same row as the desired card. Then, double-click the left mouse button. The ***Open*** *window appears. Select the appropriate file.*

Trouble clearing the Software Download

What Happens During a Download?

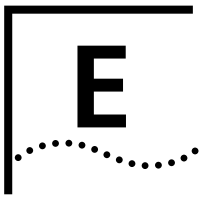
When the SDL-2 program begins, the management station sends an AT command sequence to the appropriate NAC to enable SDL-2 mode. Control then transfers to the software loader. While in SDL-2 mode, the NAC's RN/FL LED flashes green. Once the NAC enters SDL-2 mode, no other code applications can run and the NAC is entirely devoted to performing the SDL-2 .

Failure may occur if the NAC is powered up without a NIC installed behind it. Failure is indicated by the NAC RN/FL LED flashing green and red.

The SDL-2 program verifies the initialization and operation software, then begins the download. As the program executes, status messages are displayed.

"Device Not Responding" Error Message

If you receive a "Device Not Responding" error message while trying to download, too much other management traffic may be routing to the chassis. Reduce other operations (including multiple management station access) and try the download again.



SINGLE GENERIC CONFIGURATION FILE

Overview

You can use the NMC to create and store a single file in the 3Com-proprietary CFM format that contains configuration information for all NACs in the chassis (including HiPer ARC). This file allows you to use SNMP to:

- Configure an entire chassis with a single file (by downloading the file from a MS to the NMC)
- Compare different chassis configurations
- View the entire chassis configuration (by uploading the file from the NMC to a MS)

Use an SNMP browser to create the file. You will need a TFTP script for file transfer (sample scripts are provided in this appendix). You can also parse or edit this file using custom-developed tools conforming to the CFM format, and then download the file to the NMC. The NMC will then populate the configurations to all the NACs on the chassis.

Theory of operation

This is how the single configuration file works:

1. Each NAC stores its SNMP object IDs and configurations in the 3Com-proprietary CFM format. The NAC uses the new File MIB to define the format. (Refer to the "File MIB" chapter in the *NMC SNMP and MIB Reference* for additional information about this MIB).
2. When the NMC requests configuration information, the NAC sends a CFM file to the NMC.
3. The NMC builds a single configuration file containing the data configuration information from its NVRAM, plus the configuration information from each NAC.

Uploading and downloading the configuration file

The NMC uses the existing **bulkFileUpload(10)** and **bulkFileDownload(11)** commands for transferring single configuration files. These commands are enhanced to accept a parameter that contains the file name extension (.xxx) and slot number(s) of each NAC. The NMC also now can convert its NVRAM data into the CFM file format (and vice versa).

Uploading the configuration file from the NMC

This is what happens when uploading the configuration file from the NMC to a MS:

1. The user issues the **Save Chassis to NVRAM** command before getting the configuration file from the NMC.
2. The user issues a **bulkFileUpload(10)** command.
3. After the NMC receives the **get** command, it converts its NVRAM files to CFM format.
4. The NMC gets the configuration data files from the respective NACs across the management bus.
5. The NMC compiles these configuration files and a slot header for the data into CFM format.
6. The NMC adds a file header.
7. The NMC transfers the single configuration file to the MS.

Downloading the configuration file to the NMC

This is what happens when downloading the configuration file from a MS to the NMC:

1. The user issues the **bulkFileDownload(11)** command.
2. The NMC receives the single configuration file.
3. The NMC parses the file.
4. The NMC simulates the SNMP **set** commands for all objects in the NMC portion of the file.
5. The NMC then detaches the NAC data files and sends them to the respective NACs over the management bus.

Sample upload shell script

This is a typical shell script that you can modify and use to transfer the CFM configuration file from the NMC to the management station:

```
#!/sbin/sh
##############################################################
#  Dependency: #
##############################################################

# First arg is filename (?)
# Second arg is the IP address of the NMC card

if [ $# = 2 ]
then

/opt/OV/bin/snmpset -v1 -cprivate $2
.1.3.6.1.4.1.429.1.2.4.3.0 integer32 10
.1.3.6.1.4.1.429.1.2.4.5.0 octetstring $1



while [ TRUE ]
do

   sleep 10
   nmcCmdCode=`/opt/OV/bin/snmpget   -v1  -cprivate $2
.1.3.6.1.4.1.429.1.2.4.7 | awk '{printf  $4}'`
   nmcCmdResult=`/opt/OV/bin/snmpget -v1  -cprivate $2
.1.3.6.1.4.1.429.1.2.4.6 | awk '{printf  $4}'`
   echo "nmcCmdCode =" $nmcCmdCode
   echo "nmcCmdResult =" $nmcCmdResult

   if [ $nmcCmdResult = "inProgress" ]
   then

      case $nmcCmdCode in
      pendingFileTransfer)
            echo connect $2 > _TMP_TFTP_FILE
            echo verbose >> _TMP_TFTP_FILE
            echo trace  >> _TMP_TFTP_FILE
            echo timeout 300 >> _TMP_TFTP_FILE
            echo bin  >> _TMP_TFTP_FILE
            echo get $1 >> _TMP_TFTP_FILE
            tftp < _TMP_TFTP_FILE
            exit 0
      ;;
      fileTransferTimedOut)
```

```
            echo Try increasing the timeout
            exit 0
      ;;
      noError)
      ;;
      abort)
            echo aborting....
            exit 0
      ;;
      *)
            echo unsupported status \"$nmcCmdCode\"
            exit -1
      ;;
      esac

   else

      echo "terminated with nmcCmdResult =" $nmcCmdResult
      exit -1

   fi
done

else
   echo usage: "upload filename ip-address"
fi
```

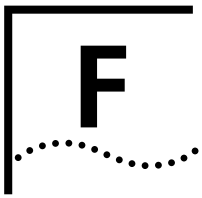
Sample download shell script

This is a typical shell script that you can modify and use to transfer the CFM configuration file from the management station to the NMC, and then to the chassis NACs.

```
#!/sbin/sh
################################################################
#  Dependency: #
################################################################

# First arg is filename
# Second arg is IP address of the NMC

if [ $# = 2 ]
then

/opt/OV/bin/snmpset -v1 -cprivate $2
.1.3.6.1.4.1.429.1.2.4.3.0 integer32 11
.1.3.6.1.4.1.429.1.2.4.5.0 octetstring $1
```

```
while [ TRUE ]
do

   sleep 10
   nmcCmdCode=`/opt/OV/bin/snmpget   -v1  -cprivate $2
.1.3.6.1.4.1.429.1.2.4.7 | awk '{printf  $4}'`
   nmcCmdResult=`/opt/OV/bin/snmpget -v1  -cprivate $2
.1.3.6.1.4.1.429.1.2.4.6 | awk '{printf  $4}'`
   echo "nmcCmdCode =" $nmcCmdCode
   echo "nmcCmdResult =" $nmcCmdResult

   if [ $nmcCmdResult = "inProgress" ]
   then

      case $nmcCmdCode in
      pendingFileTransfer)
            echo connect $2 > _TMP_TFTP_FILE
            echo verbose >> _TMP_TFTP_FILE
            echo trace  >> _TMP_TFTP_FILE
            echo timeout 300 >> _TMP_TFTP_FILE
            echo bin  >> _TMP_TFTP_FILE
            echo put $1 >> _TMP_TFTP_FILE
            tftp < _TMP_TFTP_FILE
            exit 0
      ;;
      fileTransferTimedOut)
            echo Try increasing the timeout
            exit 0
      ;;
      noError)
      ;;
      abort)
            echo aborting....
            exit 0
      ;;
      *)
            echo unsupported status \"$nmcCmdCode\"
            exit -1
      ;;
      esac

   else

      echo "terminated with nmcCmdResult =" $nmcCmdResult
```

```
      exit -1

   fi
done

else
   echo usage: "upload filename ip-address"
fi
```



IP ADDRESSING

Setting IP addresses to route data to and from the Total Control chassis is one of the first things a Network Manager should consider before cabling the chassis to a network. This appendix contains overview information about IP addressing.

Overview

Default IP addresses are associated with the NMC Network Interface Card (NIC) ports. These default addresses are not appropriate for connection to the Internet. It is recommended that default addresses be changed before establishing the network connection for the chassis. Use either the NMC User Interface software or the Total Control Manager/SNMP software.

You should take into account the size of the network, the number of physical networks, expected growth, and maintenance.

Internet Naming

TCP/IP had its origins in the U.S. Department of Defense. Naming provides a human-readable, user-friendly scheme by which to refer to hosts and routers. Names administered by the InterNIC Registration Services are organized in a tree structure, under a number of major nodes, called *domains*. The major naming domains include the following:

- COM for commercial organizations
- EDU for education institutions
- GOV for government bodies
- MIL for military organizations
- NET for systems performing network services
- ORG for non-profit organizations
- Country-specific domains

Once a domain has been assigned and registered to an organization, that organization is responsible for maintaining all subunits within the domain.

IP Address Structure

Each host on a TCP/IP internet is assigned a unique 32-bit address. This address is expressed as four decimal integers separated by decimal points, such as 192.77.203.193. Each integer gives the value of one octet (eight-digit binary number) of the IP address. This number should be thought of as its binary equivalent, with leading 0s inserted to form four 8-bit bytes. The address 192.77.203.193 would therefore be expressed as:

11000000 01001101 11001011 11000001

IP addressing includes a coded reference to the network to which a host attaches, as well as a reference to a unique host on that network. An IP address specifies an attachment to a network rather than an individual machine.

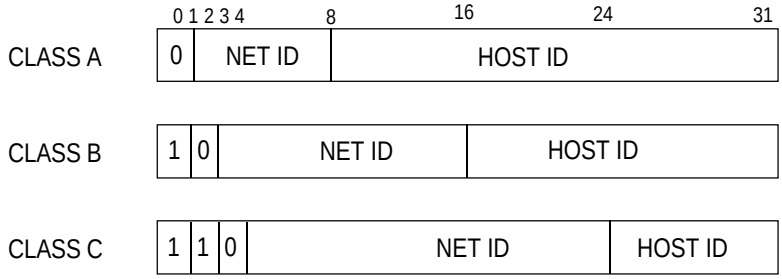
Each address consists of a Net ID and a Host ID. A Net ID may be assigned to a network by the InterNIC Registration Services, and depends on the network size. It is the responsibility of a Network Manager to assign the Host IDs with a view to how the devices must communicate with each other. The proportion of bits devoted to the Net ID and Host ID depends on the address class.

IP Address Classes

The three primary classes of IP address are: Class A, Class B, and Class C. The class concept distinguishes among networks of varying sizes. This table shows how these classes are divided:

Class	Network Size (Number of Hosts)	Net ID Size	Host ID Size
A	Over 2^{16} (Over 65,536)	7 bits	24 bits
B	Between 2^{8} and 2^{16} (256 to 65,536)	14 bits	16 bits
C	Under 2^{8} (Under 256)	21 bits	8 bits

This is a graphic representation of this concept. The class of an IP address can be determined from the first two bits.



Subnet Addressing

To use subnet addressing, a host computer must be able to identify which bits of the 32-bit internet address correspond to the physical network and which correspond to host identifiers. This information is contained in a 32-bit quantity called the subnet mask. Subnet addressing allows a Network Manager to divide the available Host IDs along logical lines that represent the physical connections.

The subnet mask makes it possible for a Network Manager to partition the Host ID of the IP address into a Physical Network portion and a Host portion. Bits in the subnet mask are set to 1 if the network is to treat the corresponding bit in the IP address as part of the network address, and 0 if it is to treat the bit as part of the host identifier. The IP address and the subnet mask are computed together with a Boolean *and* operation to determine the subnet on which a given IP address resides.

This example shows both the decimal integer IP address and subnet mask, and their binary equivalents.

	Decimal Integer	**Binary Equivalent**
IP Address	192.77.203.65	11000000 01001101 11001011 01000001
Subnet Mask	255.255.255.192	11111111 11111111 11111111 11000000

The first three octets of the subnet mask are set to 1; this is the Net ID portion of the IP address. The fourth octet of the subnet mask has 1's in the first two positions; this indicates that the first two digits will be used to indicate physical networks, allowing four subnets: 00, 01, 10 and 11. The remaining six digits in the last octet can be used for the host identifier.

Subnet Packet Forwarding

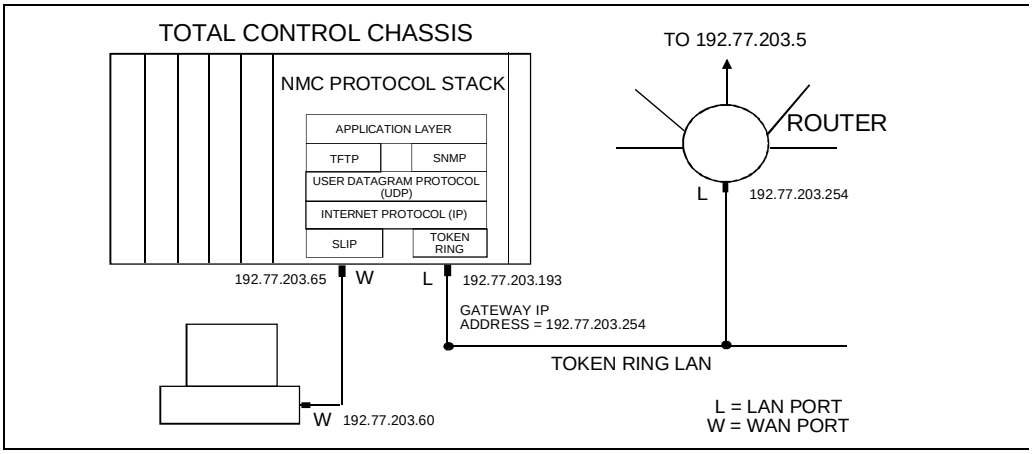
The NMC can forward data from one subnet to another. If the NMC has different subnetworks attached to its LAN and WAN ports, then any packets arriving at one port — but targeted at the network on the other port — are forwarded to the other port. This process is analogous to bridging, although it is based on IP addresses, which bridges do not typically examine.

Gateway Addressing

The gateway addressing concept provides a catch-all address for packets that the NMC does not know how to handle. It establishes a default path for data to follow when the destination IP address of a packet is not on the same subnet as either the LAN or WAN port IP address. This allows packets to be forwarded to a bridge or router (or even another NMC).

The NMC should be on an isolated LAN segment that does not receive too much traffic; this will prevent the NMC from expending too much time on nonproductive traffic and concentrate its resources on its other real-time management functions.

This figure illustrates the gateway concept. The subnet mask is 255.255.255.192, and the gateway address for this NMC is 192.77.203.254, which places the gateway on the same physical network (or subnet) as the LAN port IP address.



This example follows this path:

1. A data packet is sent from a PC through the WAN connection, with a destination address of 192.77.203.5.
2. It arrives at the Total Control chassis, which has both LAN and WAN connections. The routing is handled at the IP layer of the protocol stack.
3. The destination address of the packet does not match either the LAN or WAN address, so the packet is routed to the gateway address. The packet is sent through the LAN port to the router, which forwards it to the correct address.

TCP/IP Reference Material

Suggested references for additional information about TCP/IP include:

Comer, D.E., *Internetworking with TCP/IP Volume I: Principles, Protocols and Architecture*, Prentice-Hall, Englewood Cliffs, New Jersey, 1991.

Comer, D. E. and Stevens, D. L., *Internetworking with TCP/IP Volume II: Implementation and Internals*, Prentice-Hall, Englewood Cliffs, New Jersey, 1991.

IP machines and networks that will be attached to the Internet must obtain registered addresses from the Internet Network Information Center (InterNIC). However, for networks with only a few IP machines, it is probably better to contact your local Internet Service Provider and let them handle the details.

INDEX

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

Z

3Com Corporation
5400 Bayfront Plaza
P.O. Box 58145
Santa Clara, CA
95052-8145

Part No. 1.024.1660-00