Intel® DM3 Architecture Products on Linux

Configuration Guide

March 2005

05-1876-002

Publication Date: March 2005

Document Number: 05-1876-002

Intel Converged Communications, Inc.
1515 Route 10
Parsippany, NJ 07054

For **Technical Support**, visit the Intel Telecom Support Resources website at:
http://developer.intel.com/design/telecom/support

For **Products and Services Information**, visit the Intel Telecom Products website at:
http://www.intel.com/design/network/products/telecom

For **Sales Offices** and other contact information, visit the Where to Buy Intel Telecom Products page at:
http://www.intel.com/buy/wtb/wtb1028.htm

Contents

Figures

Tables

Revision History

This revision history summarizes the changes made in each published version of this document.

Document No.	Publication Date	Description of Revisions
05-1876-002	March 2005	Chapter 1, "Configuration Overview": : Added the major configuration steps of "Installing and Configuring Hot Swap Kit" and Pre-Configuring Empty Slots". Chapter 2, "SCD File Details": Added Table 2, "Channel Densities by Board and Media Load (non-Universal) - cPCI", on page 22 and updated Media Load information in these tables. Added Table 4, "Channel Densities for HDSI Boardschannel densities by media load:HDSI boards", on page 24 to support HDSI boards. Added Section 2.4, "PCD Files for DMN160TEC and DMT160TEC Boards", on page 27 Chapter 4, "Configuration Procedures": Revised Section 4.4.1, "Configuring Board Settings", on page 50 to include DMN160TEC and DMT160TEC information. Added Section 4.7, "Installing the HSK Software (hs_bld_kernel.sh)", on page 71, Section 4.8, "Configuring the HSK Configuration Files", on page 72 and Section 4.9, "Pre-configuring Empty Slots for CompactPCI Boards", on page 74.
05-1876-001	November 2004	Initial version of document.Much of the information contained in this document was previously published in the *DM3 Configuration File Reference*, document number 05-1272-005.

About This Publication

The following topics provide information about the *Intel® DM3 Architecture Products on Linux Configuration Guide*:

- Purpose
- Intended Audience
- How to Use This Publication
- Related Information

Purpose

This guide provides information about configuring Intel® NetStructure on DM3 Architecture PCI boards in a Linux environment. Configuration procedures are included, as well as a description of each configuration file and the associated configuration parameters.

Intended Audience

This information is intended for:

- Developers:
  - System, application, and technology developers
  - Toolkit vendors
  - VAR/system integrators
- System Operators:
  - System and network administrators
  - Support personnel (craftsperson)

How to Use This Publication

This information is organized as follows:

- Chapter 1, "Configuration Overview" describes the major configuration steps in the order in which they are performed and provides a brief overview of each aspect of configuring a system containing Intel® NetStructure on DM3 Architecture board.
- Chapter 2, "SCD File Details" provides additional detailed information about specific aspects of configuring a system that relate to the *pyramid.scd* (SCD) file.
- Chapter 3, "CONFIG File Details" provides additional detailed information about specific aspects of configuring a system that relate to the*.config* (CONFIG) files.

- Chapter 4, "Configuration Procedures" provides step-by-step procedures for configuring a system that uses Intel® NetStructure on DM3 Architecture board.
- Chapter 5, "SCD File Parameter Reference" describes each parameter associated with the SCD file. Included are a description, list of values, and configuration guidelines.
- Chapter 6, "CONFIG File Parameter Reference" describes each parameter associated with CONFIG files. Included are a description, list of values, and configuration guidelines.

Related Information

For additional information related to configuring an Intel® NetStructure product, see the following:

- For timely information that may affect configuration, see the Release Guide and Release Update. Be sure to check the Release Update for the system release you are using for any updates or corrections to this publication. Release Updates are available on the Telecom Support Resources Web site at *http://resource.intel.com/telecom/support/documentation/releases/index.htm*
- For information about installing the system software, see the systems software installation guide supplied with your release.
- For information about administrative functions relating to the Intel® NetStructure boards, see the systems administration guide supplied with your release.
- For information about administrative functions relating to the SNMP agent software, see the *SNMP Agent Software for Linux Operating Systems Administration Guide*.
- The Intel® Telecom Support Resources Web site at *http://developer.intel.com/design/telecom/support/* provides wide-ranging information in the form of technical notes, problem tracking, application notes, as well as other helpful documentation.
- *http://www.intel.com/design/network/products/telecom* (for product information)

Configuration Overview 1

The configuration overview describes the major configuration steps in the order in which they are performed. It also provides a brief overview of each aspect of configuring a system containing Intel® Dialogic® or Intel® NetStructure™ boards that are based on DM3 Architecture.

1.1 Major Configuration Steps

A board is configured by downloading firmware. Configuration files are used to identify which firmware components are downloaded and how these individual components are configured. The major steps used to configure an Intel® NetStructure system include:

1. Configuring the *pyramid.scd* (SCD) file parameters using the *DM3_cfg.sh* Utility
2. Configuring Simple Network Management Protocol (SNMP) agent software (optional)
3. Installing and configuring the Hot Swap Kit (HSK) Software (optional)
4. Pre-configuring empty slots (optional)
5. Assigning time slots when using a third party board as the clock master (optional)
6. Modifying the Feature Configuration Description (FCD) file (optional)
7. Transferring SNMP Agent Files to the Network Management Station
8. Configuring Global Call CDP File (optional)
9. Initializing the system
10. Reconfiguring the system (optional)

Detailed information about the board configuration procedures is provided in Chapter 4, “Configuration Procedures”.

1.2 The Configuration Process

Once the Intel® Dialogic® System Release is installed, you can configure your Intel® NetStructure boards by executing *config.sh*. This script file automatically invokes the DM3 Board Configuration utility (*DM3_cfg.sh*). This utility displays a series of screens and prompts you for configuration information that results in the creation of the *pyramid.scd* file. This ASCII text file contains the configuration parameters that you selected and is used by the downloader to initialize the system when the Intel® NetStructure boards are started. For detailed procedures, see Chapter 4, “Configuration Procedures”. An overview of the configuration process is as follows:

Configuring the SCD file parameters

The SCD (*pyramid.scd*) file, located in */usr/dialogic/cfg*, defines the physical parameters of a platform; the parameters in an SCD file are product and system specific. The file is used to configure the attributes of the telephony bus and the Intel® NetStructure board(s) including the assignment of PCD and FCD files to each board. SCD file parameters are set initially through the DM3 Board Configuration (*DM3_cfg.sh*) utility. This utility displays screens that allow you to modify SCD file parameters. For details about the SCD file, refer to Chapter 2, "SCD File Details". For details about the SCD file parameters, refer to Chapter 5, "SCD File Parameter Reference".

Configuring SNMP agent software
: SNMP agent software provides monitoring and administration of Intel® NetStructure boards using the Simple Network Management Protocol (SNMP). The SNMP Configuration Tool (*dlgcsnmpconf*) is invoked as part of the configuration script *config.sh*. This tool provides both automatic and manual methods of configuration SNMP communities and SNMP v1 trap destinations. In addition, the SNMP Agent MIB files must be installed on the network management station after the main configuration script (*config.sh*) has completed. For details about configuring the SNMP agent software, see Section 4.5, "Using the SNMP Agents Configuration Tool (dlgcsnmpconf)", on page 65.

Installing and configuring the HSK Software
: The Hot Swap Kit (HSK) software supports high availablity by providing Basic Hot Swap in out-of-box configurations, as well as providing redundant Single Board Computers (SBCc) in a High Availability Platform. After the HSK software is installed, the kernel is rebuilt. If multiple kernel versions are installed, the user can choose which kernel version to rebuild, as well as choosing to use the HSK kernel as the default boot kernel. For more information about configuring a high availabiltiy system, see the release guide , release update, and system admininstration guide applicable to your system release. For information about configuring the HSK software, see Section 4.7, "Installing the HSK Software (hs_bld_kernel.sh)", on page 71 and Section 4.8, "Configuring the HSK Configuration Files", on page 72.

Pre-configuring empty slots
: Slots can be pre-configured for boards that will be installed at some future time. By pre-configuring one or more vacant slots, a board or boards may be added after the system is running, without stopping the system. Slots are pre-configured by manually editing the SCD file and adding parameters for the future boards. For information about pre-configuring empty slots, see Section 4.9, "Pre-configuring Empty Slots for CompactPCI Boards", on page 74.

Assigning time slots when using a third-party board as the clock master
: Third-party boards and Intel® NetStructure boards must not transmit data on the same telephony bus time slots. Also, the third-party technology (clock master) must execute before Intel® NetStructure boards (slaves) in the startup sequence. To accommodate these requirements, adjustments must be made to the*.sctsbase* file and the *dlgsys.cfg* file. For information about assigning time slots when using a third-part board as the clock master, see Section 4.10, "Assigning Time Slots When Using a Third-Party Board as Clock Master", on page 75

Modifying the FCD file parameters
: It is sometimes necessary to adjust the parameters within the FCD file; this is done by editing the associated CONFIG file. For information about changing parameters in the FCD file, see Section 4.13, "Modifying the FCD File by Editing the CONFIG File", on page 78. The FCD file, and configuration file sets (*.pcd*, *.fcd*, and *.config* files) are located in */usr/dialogic/data*. For details about configuration file sets, refer to Section 2.2, "Configuration File Sets", on page 17. For details about CONFIG files, refer to Chapter 3, "CONFIG File Details".

Initializing the system

During system initialization, all required firmware for Intel® NetStructure products is downloaded and configured using the identified configuration files. For information about initializing the system, see Section 4.16, “Initializing the System”, on page 79.

Configuring CDP files (applicable when using Global Call protocols)

When using Global Call protocols, the protocols and country dependent parameters (*.cdp* file) must be configured. For information about CDP files, see Section 4.15, “Configuring Global Call CDP File”, on page 79.

Reconfiguring the system

If hardware is added or configuration parameters need to be changed, the system must be reconfigured. Parameter changes can be made by editing the appropriate file or re-running the appropriate utility. The system is then re-initialized using the *dlstop* and *dlstart* utilities. For additional information about reconfiguring the system, see Section 4.17, “Reconfiguring the System”, on page 80.

SCD File Details 2

This chapter provides background information about the SCD (*.scd*) file including directory location, formatting conventions, and a sample file. This chapter also includes information to help you set the parameters contained in the SCD file including the following:

2.1 SCD File Formatting Conventions

The SCD (*pyramid.scd)* file, located in */usr/dialogic/cfg*, is an ASCII file that contains board information required by the Intel® NetStructure board drivers and downloader. The SCD file is created when you run the *DM3_cfg.sh* utility. After your system is configured using the *DM3_cfg.sh* utility, SCD file parameters can be updated by manually editing the file. When manually editing the SCD file, use the following formatting conventions:

Parameters

All SCD file parameters have the following format:

```
<parameter name>    : <parameter value>
```

Sections

Configuration parameters are grouped into sections. The [TDMBus 0] section is used to configure the TDM bus, and the [Board #] sections are used to configure each board in the system. Each section begins with a section head enclosed in square braces. The parameters for the section immediately follow and are enclosed in curly braces. The format is as follows:

```
[<section-name section-id>]  {
   <parameter settings>
}
```

Comments

Comments can be added to the SCD file. If you use a semicolon (;) or pound sign (#) anywhere on a line, all text to the right of the character until the end of the line is treated as a comment (ignored). The format is as follows:

```
;  comment
```

For a list of SCD file parameters, see Chapter 5, “SCD File Parameter Reference”.

SCD File Example

```
; SCD File
; written on Fri May 31 15:13:59 EDT 2002
; written by DM3_cfg.sh 1.23
NumStreams      : 4000
NumBindHandles  : 4000
[TDMBus 0] {
TDMBusType     : H110H100
BusCR          : 8
Group1CR       : 8
Group2CR       : 8
Group3CR       : 8
Group4CR       : 8
PrimaryLines   : CT_A
}
[Board 2] {
PCDName         : ml1_qs_net5.pcd
FCDName         : ml1_qs_net5.fcd
PCMEncoding     : ALAW
SlotNumber      : 1
BusType         : PCI
LogFile         : board2.log
DisplayConfig   : YES
TimeToSendMsg   : 50
MasterStatus    : SLAVE
NetRef1         : YES
NetRef1From     : 3
NetRef1CR       : 1
}
[Board 3] {
PCDName         : ml1_qs_4ess.pcd
FCDName         : ml1_qs_4ess.fcd
PCMEncoding     : MULAW
SlotNumber      : 2
BusType         : PCI
LogFile         : board3.log
DisplayConfig   : YES
TimeToSendMsg   : 50
MasterStatus    : PRIMARY
DeriveClockFrom: NETREF_1
}
[Board 4] {
PCDName         : ipt_cas.pcd
FCDName         : ipt_cas.fcd
PCMEncoding     : MULAW
SlotNumber      : 3
BusType         : PCI
LogFile         : board4.log
DisplayConfig   : YES


TimeToSendMsg   : 50
MasterStatus    : SLAVE
 [NIC 1] {
  IPAddress          : 000.000.000.000
  SubnetMask         : FFFFFF00
  TargetName         : board4
  HostIPAddress      : 000.000.000.000
  HostName           : localhost.localdomain
  UserName           : nobody
  GatewayIPAddress   : 000.000.000.255
 }
}
[Board 6] {
PCDName               : ipt_evr_isdn_net5.pcd
```

```
FCDName               : ipt_evr_isdn_net5.fcd
PCMEncoding           : ALAW
SlotNumber            : 5
BusType               : PCI
LogFile               : board6.log
DisplayConfig         : YES
TimeToSendMsg         : 50
MasterStatus          : SLAVE
NetRef1Fallback       : YES
NetRef1FromFallback   : 1
 [NIC 1] {
  IPAddress          : 000.000.000.000
  SubnetMask         : FFFFFF00
  TargetName         : board6
  HostIPAddress      : 000.000.000.000
  HostName           : localhost.localdomain
  UserName           : nobody
  GatewayIPAddress   : 000.000.000.255
 }
}
```

2.2 Configuration File Sets

The **PCD File Name** (**PCDName)** and **FCD File Name** (**FCDName)** parameters contained in the SCD file define the board-level configuration files assigned to each board. These files are part of a **configuration file set**. The set of files associated with a specific configuration all have the same name; only the extensions (*.pcd*, *.fcd*, and *.config*) differ. A set of these files with the same name are used for a specific board type. The board type can include a single board or a group of similar boards. Depending on the board type and the protocol that the board will use, a specific PCD and FCD file are downloaded to that board as identified in the SCD file. If the FCD file needs to be modified, the CONFIG file in that same set is edited.

The files associated with configuration file sets include the following:

CONFIG File
: The CONFIG file (*.config*), located in */usr/dialogic/data*, contains the modifiable parameter settings used to configure board components. After parameters are modified in the CONFIG file, the *fcdgen* utility is used to generate a modified FCD file. For additional information about CONFIG files, see Chapter 3, "CONFIG File Details".

Feature Configuration Description (FCD) File
: The FCD file is not included with the system software, but is created and downloaded to a board when the associated PCD file and CONFIG file are downloaded to the board. The FCD file is also copied to the */usr/dialogic/data* directory.
An FCD file (*.fcd*) must be downloaded to each Intel® NetStructure board in the system. The purpose of the FCD file is to adjust the settings of the components that make up each product. For example, the FCD file may contain instructions to set certain country codes, or may send messages that configure the Telephony Service Provider (TSP) component to operate with a particular network protocol.

The FCD file defines a simple message form that the downloader parses and sends to a specific component. These parameters are sent to a component within an Intel® NetStructure message and can be thought of as configurable **features** of a component. The FCD file is created automatically from the associated CONFIG file during the board initialization process. For information about changing FCD file parameters, see Section 4.13, "Modifying the FCD File by Editing the CONFIG File", on page 78.

Note: The FCD file should not be edited directly. If parameters require modification, the changes are made by editing the associated CONFIG file. Also, an FCD file should not be copied from another directory to the */usr/dialogic/data* directory.

Note: HDSI boards use country-specific FCD and PCD files. Depending on the PCD/FCD files selected for an HDSI board, the PDM encoding method will be set to either A law or Mu law, based on the default value for that country. If this value is not the same as the TDM Bus Media Type parameter setting, the HDSI board will fail to download.

Product Configuration Description (PCD) File

A PCD file (*.pcd*), located in */usr/dialogic/data*, must be downloaded to each Intel® NetStructure board in the system. The purpose of the PCD file is to determine the software components your system will use. It defines the product by mapping download object files to specific processors, configuring the kernel for each processor, and setting the number of component instances to run on each processor.

Note: The PCD file should not be modified by the user.

As an example, the configuration file set for a DM/V960A-4T1-PCI board using the 4ess protocol includes the following files:

- *ml2_qs2_4ess.config*
- *ml2_qs2_4ess.fcd*
- *ml2_qs2_4ess.pcd*

2.3 Media Loads

Media loads are pre-defined sets of features supported by DM3 boards. A media load consists of a configuration file set (PCD, FCD, and CONFIG files) and an associated firmware load that are downloaded to each board. Universal media loads simultaneously support voice, fax, and conferencing resources.

2.3.1 Features Supported

The media loads, or feature sets, are numbered (for example, 1, 2, 9b) for identification purposes and apply to the following boards:

- Intel® NetStructure™ DM/V, DM/V-A, and DM/V-B Boards
- Intel® NetStructure™ High Density Station Interface Boards

2.3.1.1 Intel® NetStructure™ DM/V, DM/V-A, and DM/V-B Boards

Intel NetStructure DM/V, DM/V-A, and DM/V-B boards are supported by media loads 1 through 10 and universal media loads 1 through 3. Intel NetStructure DM/IP boards are supported by media loads 2 and 11. The features supported by each media load is as follows:

Media Load 1 – Basic Voice

- Provides play, record, digit generation, and digit detection
- All half duplex voice operations
- Supports the following coders:
  - 64 kbps and 48 kbps G.711 PCM VOX and WAV
  - 24 kbps & 32 kbps OKI ADPCM VOX and WAV
  - 64/88/128/176 kbps Linear PCM VOX and WAV
- Speed control on 8 kHz coders
- Volume control
- Cached prompts
- GTG/GTD
- Call progress analysis
- Transaction record

  Note: Density limitations may exist for transaction record. Check the specific media load for details. Also, see the following Web sites for additional information:
  http://resource.intel.com/telecom/support/tnotes/gentnote/dl_soft/tn253.htm
  http://resource.intel.com/telecom/support/tnotes/tnbyos/2000/tn031.htm
- All call control features when using a board with a network interface

Media Load 1b – Basic Voice Plus

- All Basic Voice features (see Media Load 1)
- Frequency Shift Keying (ADSI/2-way FSK/ETSI-FSK)

Media Load 2 – Enhanced Voice

- All Basic Voice features (see Media Load 1)
- Continuous speech processing (CSP) (Does not apply to DM/IP Series boards)
- Enhanced coders:
  - G.726 at 16 kbps, 24 kbps, 32 kbps, and 40 kbps
  - GSM (TIPHON* and Microsoft*)
  - IMA ADPCM
  - TrueSpeech
- Silence Compressed Record (G.711, OKI ADPCM, Linear 8kHz, and G.726)
- IP transcoders (Intel NetStructure DM/IP Series boards only)
  - G.711: 1 frame/packet at 10, 20, or 30 ms (A-law or mu-law)
  - G.723: 1, 2, or 3 frames/packet at 30 ms (silence compression with VAD and CNG)
  - G.729: 1, 2, 3, or 4 frames/packet at 10 ms (silence compression with VAD and CNG)

- GSM: 1, 2, or 3 frames/packet at 20 ms (silence compression with VAD and CNG)
- Frequency Shift Keying (ADSI/2-way FSK/ETSI-FSK) (DM/V-B boards only)

Media Load 2b – Enhanced Voice plus CSP Streaming to CT Bus

- All Enhanced Voice features (see Media Load 2)
- CSP streaming to CT Bus

Media Load 2c – Enhanced Echo Cancellation

- All Enhanced Voice features plus CSP Streaming to CT Bus (see Media Load 2b)
- Enhanced Echo Cancellation (in addition to standard 16 ms tap length, provides selectable tap lengths of 32 ms and 64 ms)

Media Load 5 – Fax

- All Enhanced Voice Features (see Media Load 2)
- V.17 Fax

Media Load 5b

- All Enhanced Voice features (see Media Load 2)
- All Fax Features (see Media Load 5)
- CSP CT Bus Streaming

Media Load 5bc – Enhanced Echo Cancellation

- All Enhanced Voice features (see Media Load 2)
- All Fax Features (see Media Load 5)
- CSP CT Bus Streaming
- Enhanced Echo Cancellation (see Media Load 2c)

Media Load 9b – Conferencing Only (Rich Conferencing)

- Traditional DCB conferencing plus echo cancellation:
  - No voice channels
  - Conferencing
  - Signal detection
  - Tone clamping
  - Tone generation
  - Echo cancellation (16 ms)

Media Load 9c – Conferencing Only (Basic Conferencing, No Tone Clamping, No Tone Generation, No Echo Cancellation)

Media Load 9d – Conferencing Only (Standard Conferencing - DCB)

- This media load provides the same support as Media Load 9B except the following:
  - No Echo Cancellation

Media Load 10 – Enhanced Voice Plus Conferencing

- All Enhanced Voice features (see Media Load 2)
- Rich Conferencing (Signal Detection, Tone Clamping, Tone Generation, 16 ms Echo Cancellation)

For a list of channel densities based on non-Universal media load configurations, refer to Table 1 and Table 2.

Universal Media Loads 1 through 3 – Refer to Table 3 for a list of channel densities based on Universal media loads.

Note: Some of the media loads listed in the following tables may be system release dependent. Check the Release Guide for your system release to determine the latest media loads.

Table 1. Channel Densities by Board and Media Load (non-Universal) - PCI

Board	Media Loads											
	Voice Only					Fax			Conferencing Only			Voice and Conferencing
	1	1b	2	2b	2c	5	5b	5bc	9b	9c	9d	10
DM/V480-4T1-PCI	48 V†											
DM/V600-4E1-PCI	60 V†											
DM/V960-4T1-PCI	96 V											
DM/V1200-4E1-PCI	120 V											
DM/V480A-2T1-PCI		48 V						48 V 12 F				48 V 60 C
DM/V600A-2E1-PCI		60 V						60 V 12 F				60 V 60 C
DM/V960A-4T1-PCI		96 V				96V 4F						
DM/V1200A-4E1-PCI		120 V	120 V									
DM/V2400A-PCI		240 V		120 V		120 V 12 F			120 C	240 C		60 V 60 C
DMV600BTEP												
DMV1200BTEP							120 V 12 F					120 V 54 C
DMV3600BP		360 V						120 V 24 F	160 C	576 C	270 C	60 V 60 C

† For the DM/V480-4T1-PCI board, this media load has full 4 span (96 channels) density of tone detection and generation. For the DM/V600-4E1-PCI board, this media load has full 4 span (120 channels) density of tone detection and generation.

Table 2. Channel Densities by Board and Media Load (non-Universal) - cPCI

Board	Media Loads											
	Voice Only					Fax			Conferencing Only			Voice and Conferenc-ing
	1	1b	2	2b	2c	5	5b	5bc	9b	9c	9d	10
DM/V480A-2T1-cPCI		48 V										48 V 60 C
DM/V600A-2E1-cPCI		60 V										60 V 60 C
DM/V960A-4T1-cPCI		96 V	96V									
DM/V1200A-4E1-cPCI		120 V	120 V									
DM/V2400A-cPCI		240 V			120 V	120 V 30 F			120 C	240 C		120 V 60 C
DMV600BTEC												
DMV1200BTEC								120 V 24 F				120 V 60C
DMV4800BC		480 V	240						288 C	704 C	352 C	

Table 3. Channel Densities by Board and Media Load (Universal)

Board	ML	Media Loads/Features Supported										
		Voice Only							Fax	Conferencing Only		
		Basic Voice	Trans action Record	Enhanced Voice	TrueSpeech	Enhanced Echo Cancellation‡	CSP Streaming to CT Bus	FSK	Fax	Conferencing	Conferencing - Tone	Conferencing - Echo Cancellation
DM/V960A-4T1-PCI	UL2	96		96					4	15	15	15
DM/V1200A-4E1-PCI	UL1	60							8	60	60	60
DMV600BTEP	UL1	60	60	60	60	60	60	60	16	60	60	60
DMV1200BTEP	UL1	120	120	120	120	120	120	120	12	30	30	
DMV1200BTEP	UL2	120	120					120	12	120	120	
DMV600BTEC	UL1	60	60	60	60	60	60	60	16	60	60	60
DMV1200BTEC	UL1	120	120	120	120	120	120	120	12	60	60	
DMV3600BP	UL1	120	120	120	120	120	120	120	12	30	30	
DMV3600BP	UL2	120	120					120	12	60	60	60
DMV3600BP	UL3	240	100					240	12	60	60	
DMV4800BC	UL1	160	160	160	160		160	160	16	160	160	
DMV4800BC	UL3	224	20					224	12	224	224	
DMV4800BC	UL4	160		160	160	160	160	160	15	90	90	90

Note: Features within a resource group (Headings marked as Voice Only, Fax, or Conferencing Only) are inclusive. Features across resource groups are additive. For example, on the DMV600BTEP board using UL1, there are 60 total voice resources, 16 fax resources, and 60 conferencing resources. This means that any combination of the listed voice resources (Voice Only subheadings marked as Basic Voice, Transaction Record, Enhanced Voice, TrueSpeech, and Enhanced Echo Cancellation, CSP Streaming to CT Bus, and FSK) can be used up to a total of 60. For example., 30 Basic Voice plus 10 Enhanced Voice plus 10 TrueSpeech plus 10 CSP Streaming to CT Bus. In addition to these various voice resources, the UL1 media load can use 16 fax resources and 60 conferencing resources (with Tone Clamping and Echo Cancellation) simultaneously.

‡ Default configuration is standard EC (16 ms). To set it to EEC tail length, change the CSP parameter 0x2c03 accordingly in the respective .config file. Conferencing EC, however, will always be 16 ms, regardless of the EEC parameter setting.

2.3.1.2 Intel® NetStructure™ High Density Station Interface Boards

Intel NetStructure High Density Station Interface (HDSI Series) boards only support Media Load 1. Refer to Table 4 for a list of channel densities.

Table 4. Channel Densities for HDSI Boardschannel densities by media load:HDSI boards

Board	Media Load	Stations†	Basic Voice	FSK
HDSI/480-PCI	1	48	48	48
HDSI/720-PCI	1	72	72	72
HDSI/960-PCI	1	96	96	96
HDSI/1200-PCI	1	120	96‡	120

† HDSI only supports fixed routing (voice device is permanently associated with the respective station).

‡ Tone only. Voice is not supported.

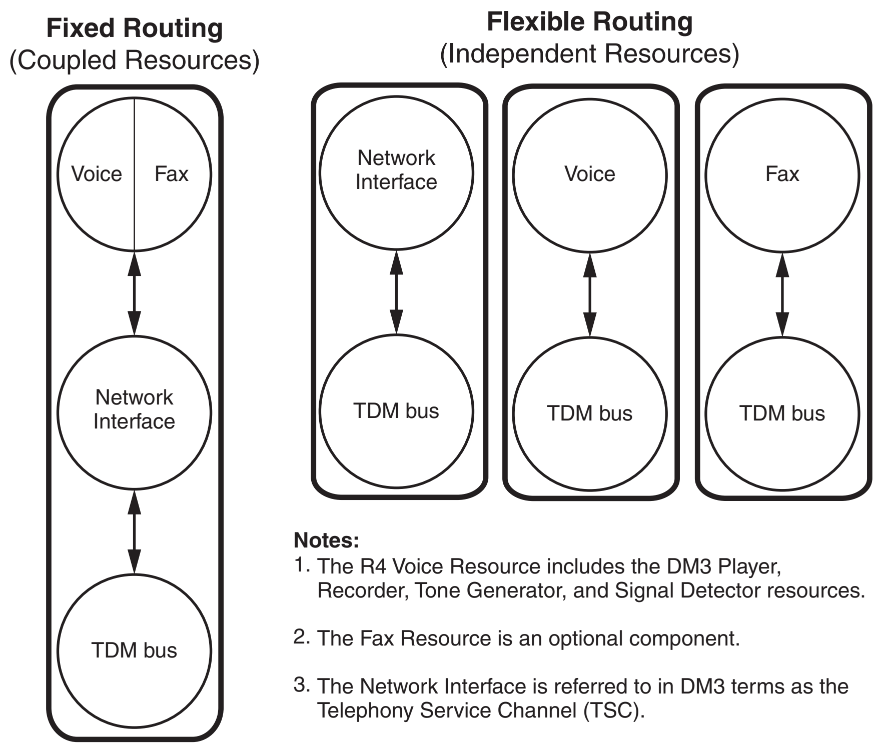
2.3.2 Flexible Routing Configurations

All media loads support flexible routing. With flexible routing, the resource devices (voice/fax) and network interface devices are independent, which allows exporting and sharing of the resources. All resources have access to the TDM bus. For example, on a DM/V960A-4T1-cPCI board, each voice resource channel device and each network interface device can be independently routed on the TDM bus.

These routing configurations are also referred to as cluster configurations. Flexible routing configuration uses **independent resources**. From a DM3 perspective, the **flexible routing cluster** allows more freedom by nature of its independent resources, as shown in Figure 1.

Note: Fixed routing is not supported on DM/V, DM/V-A, and DM/V-B boards.

Figure 1. Cluster Configurations for Fixed and Flexible Routing



2.3.3 Media Load Configuration File Sets

For most products, the file names of the configuration file set reflects the media load supported. If a media load number (ml*x*) is not present in the file name, no media load is supported for that configuration. The following section provides information about the media load configuration file sets for the boards that use media loads:

- Intel® NetStructure™ DM/V, DM/V-A, and DM/V-B Series Boards
- Intel® NetStructure™ High Density Station Interface Boards

2.3.3.1 Intel® NetStructure™ DM/V, DM/V-A, and DM/V-B Series Boards

Media load configuration files for DM/V Series, DM/V-A Series, and DM/V-B Series boards are identified by having an *mlx* or ul*x* prefix, where *x* represents the specific media load.

For DM/V, DM/V-A and DM/V-B boards, media loads support flexible routing and the configuration file sets are customized by feature set, as well as by protocol. For example, on a DM/V960-4T1 board using the T1 ISDN Net5 protocol, *ml2_qsa_net5.pcd* supports media load 2 and a flexible routing configuration.

For DM/V-B boards, individual trunks can be configured to use different line types and protocols. For instructions about configuring individual trunks on a DM/V board, see Section 4.4, "Using the DM3 Board Configuration Utility (DM3_cfg.sh)", on page 49.

2.3.3.2 Intel® NetStructure™ High Density Station Interface Boards

Configuration files for Intel NetStructure High Density Station Interface boards are prefaced with a country code. This code represents the country-specific protocol that is supported. For example, the HDSI CONFIG file for Austria E1 (code = at), supporting 48 channels is:

- *at_hdsi_48_play_rec.config*

Refer to Table 5 for a list of country codes for all supported countries.

Table 5. Intel NetStructure High Density Station Interface Country Codes

Code	Country
at	Austria
au	Australia
be	Belgium
ch	Switzerland
de	Germany
dk	Denmark
es	Spain
fr	France
gb	United Kingdom
hk	Hong Kong
ie	Ireland
it	Italy
jp	Japan
lu	Luxembourg
mx	Mexico
my	Malaysia
nl	Netherlands
no	Norway
nz	New Zealand
pt	Portugal
se	Sweden
sg	Singapore
us	United States
za	South Africa

2.4 PCD Files for DMN160TEC and DMT160TEC Boards

The DMN160TEC and DMT160TEC boards support a total of 16 trunks that can be configured in groups of four as either T-1 or E-1 interfaces. The trunks are grouped as follows:

- trunks 1 to 4
- trunks 5 to 8
- trunks 9 to 12
- trunks 13 to 16

In addition, each trunk within a group can be configured to use a different ISDN protocol supported by the line type (T-1 or E-1) assigned to the group. This is accomplished by creating a PCD file. The PCD file is configured from the DM3 Board Configuration (*DM3_cfg.sh*) utility's Specify the PCD File screen. When creating a PCD file, the utility also creates an associated CONFIG file. The associated FCD file is automatically created when the PCD file is downloaded to the board.

For instructions about configuring a PCD/FCD file, see Section 4.4.1, "Configuring Board Settings", on page 50.

2.5 CT Bus Clock Fallback

The system provides clock fallback to maintain timing in the event that the current clock source fails. The following sections provide reference information about the two types of clock fallback:

- Primary Clock Fallback
- Reference Master Fallback

2.5.1 Primary Clock Fallback

For the following discussion, refer to Figure 2, "Clock Fallback", on page 28 for an illustration of the CT Bus clocking concepts.

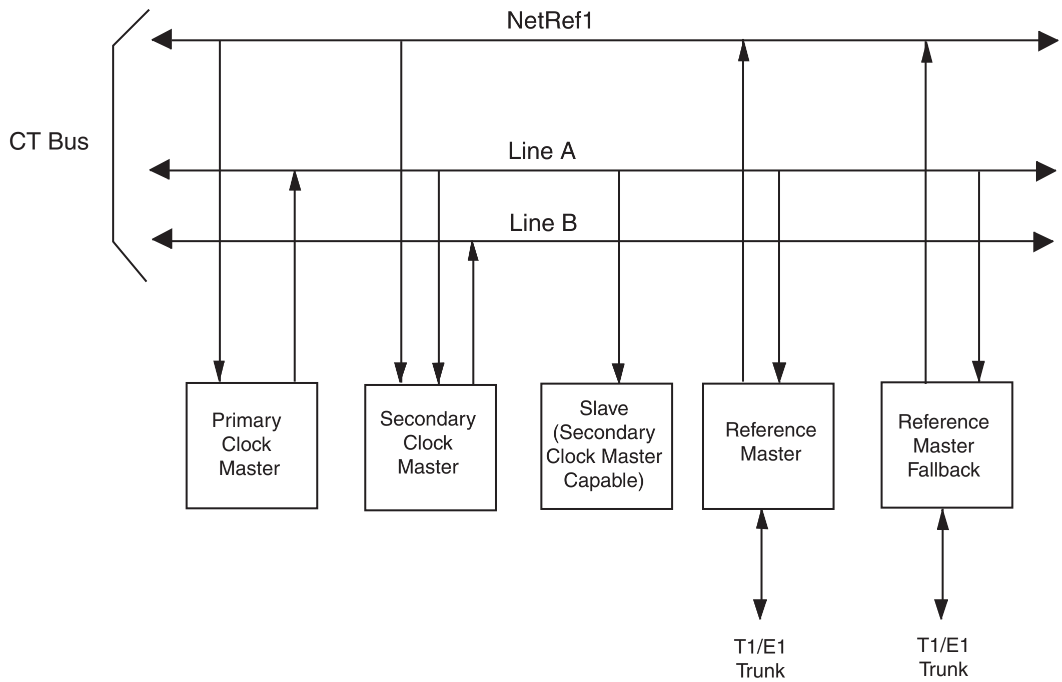
The Primary Clock Master is a device (board) that provides timing to all other devices attached to the bus. The Primary Clock Master drives bit and framing clocks for all of the other boards (slaves) in the system via CT bus Line A or Line B. This bus clocking is synchronized to either the board's internal oscillator or, preferably, to the NetRef1 line which provides a timing reference (8 kHz) derived from a T-1 or E-1 interface signal.

In addition, a Secondary Clock Master can be defined as a backup for the same purpose. This board, like the Primary Clock Master, is capable of driving the bit and framing clocks for all of the other boards in the system. The Secondary Clock Master uses whichever CT Bus line (A or B) is not defined for the Primary Master Clock. If the system senses a failure of the Primary Clock Master, the system will cause the clock source to fall back to the Secondary Clock Master. The Secondary Clock Master, like the primary, also provides clocking that is synchronized to either the board's internal oscillator or, preferably, to the NetRef1 line.

Both the Primary and Secondary Clock Masters are defined by the user. If the Primary Clock Master should fail and no Secondary Clock Master has been defined or has already failed, the system will lose clocking.

Note: For information about specific CT Bus clocking parameters, see Section 5.1.2, "TDM Bus Configuration", on page 88.

Figure 2. Clock Fallback



2.5.2 Reference Master Fallback

In addition to supporting clock master fallback, the system also provides for fallback in the event the board designated as the Reference Master fails. The failure could be caused by degradation of the board itself, or by a degradation of the T-1 or E-1 trunk from which the reference signal is derived. A second line on the same board or a line on a second board can be assigned as the Reference Master Fallback board (the user must specify the trunk to be used on the Reference Master Fallback board). In the event that the Network Reference board stops providing a reliable signal to drive the NetRef1 line, the system will switch to the Reference Master Fallback board for this purpose.

CONFIG File Details 3

This chapter provides background information about CONFIG (*.config*) files including directory location and formatting conventions. This chapter also includes information to help you set the parameters contained in the CONFIG file including the following:

3.1 CONFIG File Formatting Conventions

The CONFIG (*.config)* files, located in */usr/dialogic/data*, are ASCII files that contain component configuration information required by Intel® telecom boards. When manually editing the CONFIG file, use the following formatting conventions:

Parameters

Many CONFIG file parameters use the SetParm command to assign values. The format is `SetParm` followed by an equal sign, followed by the hexadecimal parameter number, followed by a comma, followed by the parameter value:

```
SetParm=parameter-number, parameter-value
```

Additional commands used to set parameters include:

- AddNFASInterface (see "GroupID (Group Identifier)", on page 116)
- transition, pulse, train, and sequence (see Section 3.5, "[CAS] Section", on page 34)
- Variant (see Section 3.6, "[CHP] Section", on page 43)
- defineBSet (see Section 3.7, "[TSC] Section", on page 45).

Sections

Configuration parameters are grouped into sections. In general, each section begins with a section name enclosed in square brackets. (The section names are listed and described in

Section 3.2, “CONFIG File Sections”, on page 30.) The parameters for the section immediately follow the section name.

```
[section-name]
```

Some sections group parameters that apply to a specific network interface (trunk) or channel (line). These section names are followed by a period (.) and the trunk number. For sections that group parameters like this, there is a separate section for each trunk.

```
[section-name.trunk-number]
```

Comments
: Comments can be added to the CONFIG file. If you use an exclamation point (!) anywhere on a line, all text to the right of the exclamation point until the end of the line is treated as a comment (ignored).

```
! comment
```

For a list of CONFIG file parameters, see Chapter 6, “CONFIG File Parameter Reference”.

3.2 CONFIG File Sections

CONFIG file parameters are grouped in sections based on the board components and subcomponents being configured. Modifiable CONFIG file sections include the following:

[0x44]
: Defines the companding method (along with the [TSC] **encoding** parameter 0x1209) for Intel® NetStructure™ DI Series Station Interface boards. This section only applies to DI Series boards.

[0x2b]
: Defines parameter used to enable streaming of echo cancellation data over the TDM bus in Continuous Speech Processing (CSP) applications. This section is only applicable in media load 2c CONFIG files.

[0x2c]
: Defines enhanced echo cancellation parameters used to set the tail length, or tap length, used in Continuous Speech Processing (CSP) applications. This section also defines the parameters associated with Silence Compressed Streaming (SCS).

[encoder]
: Defines parameters used during the encoding process that utilize the Automatic Gain Control (AGC) and Silence Compressed Record (SCR) algorithms. For details about setting algorithm parameters, see Section 3.3, “[Encoder] Section”, on page 31.

[recorder]
: Defines recording parameters used during the recording process including the enabling and disabling of AGC and SCR on a per board basis.

[0x39]
: Defines conferencing parameters applicable to all conferencing lines on a board.

[0x3b]
Defines conferencing parameters applicable to conferencing lines on a board. The [0x3b] parameters apply to all conferencing lines on the board. The [0x3b.x] parameters apply specifically to trunk x on the board.

[lineAdmin.x]
Defines line device parameters applicable to each line on a board.

[NFAS] and [NFAS.x]
Non-Facility Associated Signaling (NFAS). Defines the Primary D channel and NFAS trunk parameters. The [NFAS] section defines the number of NFAS groups on a board. The [NFAS.x] sections define the parameters specific to each group. For details about setting the NFAS parameters, see Section 3.4, “[NFAS] Section”, on page 33.

[CAS]
Channel Associated Signaling (CAS). Defines the signaling types used by a CAS protocol and the [TSC] section assigns these signaling type to voice channels. For details about the different CAS signals, see Section 3.5, “[CAS] Section”, on page 34.

[CCS] and [CCS.x]
Common Channel Signaling (CCS). Defines common channel signaling parameters applicable to technologies such as ISDN. The [CCS] section defines board-based parameters and the [CCS.x] section defines the line-based parameters.

[CHP]
Channel Protocol (CHP). Defines the telephony communication protocol that is used on each network interface using the Variant Define *n* command. For details about setting [CHP] parameters using the Variant Define *n* command, see Section 3.6, “[CHP] Section”, on page 43.

[TSC]
Telephony Service Component (TSC). Defines sets of B channels and associated characteristics using the defineBSet command. For details about setting [TSC] parameters using the defineBSet command, see Section 3.7, “[TSC] Section”, on page 45.

3.3 [Encoder] Section

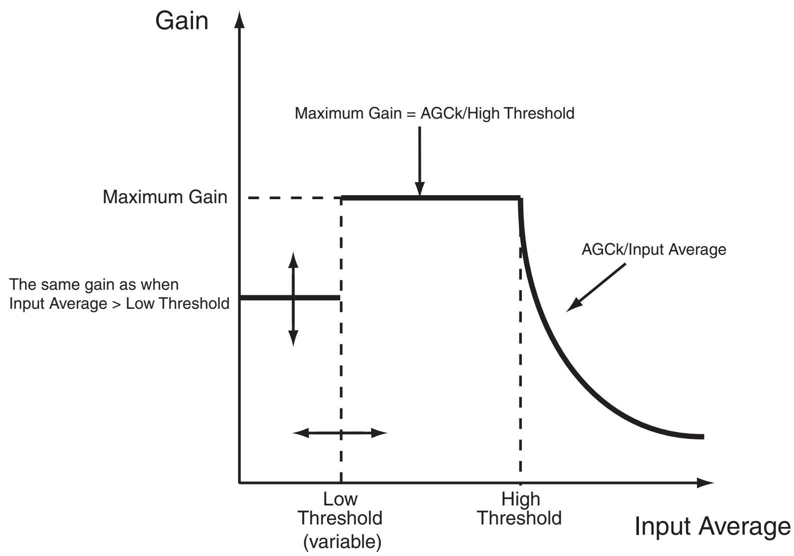
The encoder parameters are used to perform an encoding process on a media stream. Automatic Gain Control (AGC) and Silence Compressed Record (SCR) are two algorithms used as part of this encoding process.

The AGC is an algorithm for normalizing an input signal to a target record level. The target record level should be chosen to be the optimum level for an encoder and, at the same time, produce a suitable playback level for a listener.

The AGC algorithm is controlled by three parameters: **PrmAGCk**, **PrmAGCmax_gain**, and **PrmAGClow_threshold**. **PrmAGCk** is a target output level. **PrmAGCmax_gain** is the a limit on the possible maximum gain. The ratio, **PrmAGCk/PrmAGCmax_gain** gives the AGC High Threshold value. This is the threshold for which inputs above it produce output level at the **PrmAGCk** level and inputs with a level below it produce outputs which linearly decrease with the input level. The **PrmAGClow_threshold**, on the other hand, is an upper limit for a noise level estimate. That is, a signal with a level above the **PrmAGClow_threshold** is declared speech, independently of whether it is or not. Below the threshold, the AGC algorithm itself tries to discriminate between voiced and unvoiced signals.

Figure 3 is a graphical representation of the AGC gain relative to input average.

Figure 3. AGC Gain vs. Input Average



The SCR algorithm operates on 1 millisecond blocks of speech and uses a twofold approach to determine whether a sample is speech or silence. Two Probability of Speech values are calculated using a Zero Crossing algorithm and an Energy Detection algorithm. These values are combined to calculate a Combined Probability of Speech.

The Zero Crossing algorithm counts the number of times a sample block crosses a zero line, thus establishing a rough "average frequency" for the sample. If the count for the sample falls within a predetermined range, the sample is considered speech.

The Energy Detection algorithm allows user input at the component level of a background noise threshold range via the **SCR_LO_THR** and **SCR_HI_THR** parameters. Signals above the high threshold are declared speech and signals below the low threshold are declared silence.

SCR declares speech or silence for the current 1 millisecond sample based on the following:

- previous 1 millisecond sample declaration (speech or silence)
- Combined Probability of Speech in relation to the Speech Probability Threshold (**SCR_PR_SP**)
- Combined Probability of Speech in relation to the Silence Probability Threshold (**SCR_PR_SIL**)
- Trailing Silence (**SCR_T**) relative to Silence Duration

The logic is as follows:

Previous sample = Silence

If Combined Probability of Speech > Speech Probability Threshold
    then Declare Speech
    else Declare Silence

Previous sample = Speech

If Combined Probability of Speech > Silence Probability Threshold
    then Declare Speech
    else If Silence Duration < Trailing Silence
        then Declare Speech
        else Declare Silence

3.4 [NFAS] Section

Non-Facility-Associated Signaling (NFAS) uses a single ISDN PRI D channel to provide signaling and control for up to 10 ISDN PRI lines. Normally, on an ISDN PRI line, one D channel is used for signaling and 23 B channels (bearer channels) are used for transferring information. In an NFAS configuration, therefore, one D channel can support the signaling and control for up to 239 B channels. The trunk that provides the signaling is called the primary D channel. The trunks that use all 24 channels as B channels are called NFAS trunks.

Notes:
1. For a board containing multiple primary D channels, the maximum number of trunks supported by each NFAS group on that board is reduced. This is due to the additional message load on the board's CPU.
2. NFAS is supported on only the ISDN NI-2, 4ESS, 5ESS, and DMS protocols.
3. NFAS D channel backup (DCBU) is supported only on ISDN NI-2 protocol.
4. When NFAS is used, the **SignalingType** parameter in the [lineAdmin] section of the CONFIG file must be modified. For details about this parameter modification, see Section 6.10, "[lineAdmin.x] Parameters (Digital Voice)", on page 107.

The CONFIG file contains an [NFAS] section and multiple [NFAS.x] sections. The [NFAS] section defines the number of NFAS instances created, that is, defines the number of NFAS groups. For each NFAS group, there is an [NFAS.x] section in the CONFIG file. For example, if there are two NFAS groups defined in the [NFAS] section, there will be two [NFAS.x] sections, [NFAS.1] and [NFAS.2].

NFAS parameters are modified by editing the respective lines in the [NFAS] and [NFAS.x] sections of the CONFIG file. For example, to increase the number of NFAS groups per board from one to four, change the value of **NFAS_INSTANCE_MAP** (parameter = 0x3E02) from a value of 1 (one group per board) to a binary value of 1111 (four NFAS groups per board) represented by 0xF.

Following is an excerpt from the [NFAS] section of a CONFIG file that illustrates that part of the file before and after editing.

Before editing:

```
[NFAS]
SetParm=0x3e02,0x1 !INSTANCE MAP, default = 1 (1 group/board)
```

After editing:

```
[NFAS]
SetParm=0x3e02,0xf !INSTANCE MAP - 4 NFAS groups/board
```

3.5 [CAS] Section

Information about the [CAS] section of the CONFIG file is given in the following sections:

- CAS Signaling Parameters
- Transition Signal
- Pulse Signal
- Train Signal
- Sequence Signal

3.5.1 CAS Signaling Parameters

The Channel Associated Signaling (CAS) component is responsible for managing the generation and detection of digital line signaling functions required to manage voice channels. Each CAS instance corresponds to the CHP instance of the same voice channel.

The [CAS] section of the CONFIG file is a subcomponent of the [TSC] section. Commands in the [CAS] section define the signaling types used by a CAS protocol, and the [TSC] section assigns these signaling type to voice channels. For example, many CAS protocols use off-hook and wink signals, which can be defined in this section. For an explanation of the [TSC] section of the CONFIG file, see Section 3.7, “[TSC] Section”, on page 45.

Note: The CAS signaling parameters should only be modified by experienced users if the default settings do not match what the line carrier or PBX is sending or expecting for the line protocol configuration running on the card.

CAS parameters are defined using the following signal definition types:

Transition Signal
: This signaling state changes from the current signaling state to a new signaling state.

Pulse Signal
: This signaling state changes from the current signaling state to a new signaling state, and then reverts to the original signaling state.

Train Signal
: This signaling state alternates between two predefined signaling states in a regular defined pattern (series of pulses).

Sequence Signal
: This signaling state is defined by a set of train signals.

For information about specific CAS parameters, see the following sections:

- Section 6.14, "[CAS] Parameters for T1 E&M Signals", on page 117
- Section 6.15, "[CAS] Parameters for T1 Loop Start Signals", on page 119
- Section 6.16, "[CAS] Parameters for T1 Ground Start Signals", on page 124

3.5.2 Transition Signal

The transition command defines an ABCD-bit transition from one state to another. It is used to define the CAS transition signals required by a protocol. The transition command uses the following syntax:

```
transition = SignalId, PreVal, PostVal, PreTm, PostTm
```

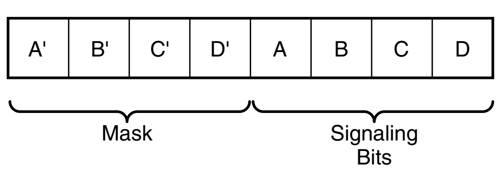
The transition signal definition includes the following values:

SignalId
: Unique identifier (parameter number) of the transition signal. The Channel Protocol (CHP) uses the SignalId to recognize the transition when it is received, and also to generate the transition when needed.

  Note: SignalId should not be modified by the user.

PreVal
: Defines the ABCD bit states on the line before the transition occurs. The four least significant bits represent the ABCD signaling bits (0 or 1). The four most significant bits represent a mask (A'B'C'D') that specifies if each corresponding signaling bit value counts. If a mask bit is set to 1, the corresponding signaling bit is counted. If a mask bit is set to 0, the corresponding signaling bit is ignored. See Figure 4.

Figure 4. Pre-transition ABCD Bit States



PostVal
: Defines the ABCD bit states on the line after the transition occurs. The format of this field is the same as the PreVal field.

PreTm
: Defines the minimum amount of time, in milliseconds, for the duration of the pre-transition interval.

PostTm
: Defines the minimum amount of time, in milliseconds, for the duration of the post-transition interval.

Transition Example

The following is an example of a transition command that defines a transition signal:

```
transition = 0xC15CA001, 0xF0,0xFF, 100, 300
```

In the example shown, the transition signal is defined as having the following values:

SignalId = 0xC15CA001
: Defines the CAS T1 E&M transition signal off-hook.

PreVal = 0xF0 (11110000)
: Defines the mask as having a hexadecimal value of F (1111) and the signaling bits as having a hexadecimal value of 0 (0000). Since all of the mask bits are 1, all of the signaling bits are significant. Thus, the A, B, C, and D bits all have a value of 0 before the transition.

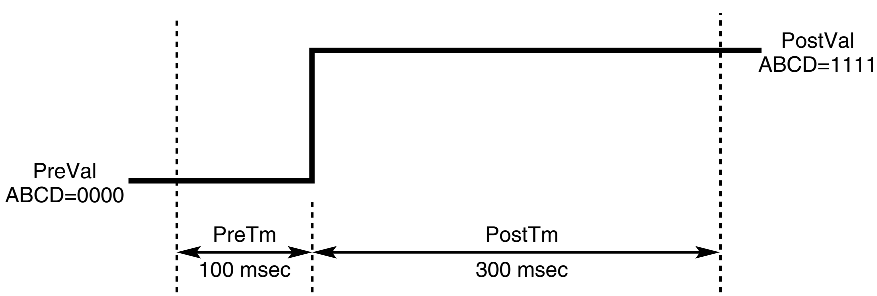
PostVal = 0xFF (11111111)
: Defines both the mask and the signaling bits as having a hexadecimal value of F (1111). Since all of the mask bits are 1, all of the signaling bits are significant. Thus, the A, B, C, and D bits all have a value of 1 after the transition.

PreTm = 100 ms
: Specifies that the PreTm signaling bits must be present for at least 100 ms before they transition to the PostVal signaling values.

PostTm = 300 ms
: Specifies that the PostVal signaling bits must be present for at least 300 ms before the signal is reported to the protocol (or if the signal is being sent, then the CAS subcomponent ensures that the PostVal signaling value is generated for at least 300 ms).

Figure 5 is a graphical representation of this signal definition.

Figure 5. Example of Off-hook Transition Signal (0xC15CA001)



3.5.3 Pulse Signal

The pulse command defines an ABCD-bit transition from one state to another, and then back to the original state. It is used to define the CAS pulse signals required by a protocol. The pulse command uses the following syntax:

```
pulse = SignalId, OffVal, OnVal, PreTm, MinTm, NomTm, MaxTm, PostTm
```

The pulse signal definition includes the following values:

SignalId
: Unique identifier (parameter number) of the pulse signal. The Channel Protocol (CHP) uses the SignalId to recognize the pulse when it is received, and also to generate the pulse when needed.

   Note: SignalId should not be modified by the user.

OffVal
: Defines the ABCD bit states on the line before the transition occurs. The four least significant bits represent the ABCD signaling bits (0 or 1). The four most significant bits represent a mask (A'B'C'D') that specifies if each corresponding signaling bit value counts. See Figure 4, “Pre-transition ABCD Bit States”, on page 35. If a mask bit is set to 1, the corresponding signaling bit is counted. If a mask bit is set to 0, the corresponding signaling bit is ignored.

OnVal
: Defines the ABCD bit states on the line during the pulse. The format of this field is the same as the OffVal field.

PreTm
: Defines the minimum time, in milliseconds, for the duration of the pre-pulse interval.

MinTm
: Defines the minimum time, in milliseconds, for the duration of the pulse interval.

NomTm
: Defines the nominal time, in milliseconds, for the duration of the pulse interval.

MaxTm
: Defines the maximum time, in milliseconds, for the duration of the pulse interval.

PostTm
: Defines the minimum time, in milliseconds, for the duration of the end-of-pulse interval.

Pulse Example

The following is an example of a pulse command that defines a pulse signal:

```
pulse = 0xC15CA011, 0xF0, 0xFF, 100, 220, 250, 280, 100
```

In the example shown, the pulse signal is defined as having the following values:

SignalId = 0xC15CA011
: Defines the CAS T1 E&M pulse signal Wink.

OffVal = 0xF0
: Defines the mask as having a hexadecimal value of F (1111) and the signaling bits as having a hexadecimal value of 0 (0000). Since all of the mask bits are 1, all of the signaling bits are significant. Thus, the A, B, C, and D bits all have a value of 0 before the transition from the OffVal to the OnVal, and after the transition from the OnVal to the OffVal.

OnVal = 0xFF
: Defines both the mask and the signaling bits as having a hexadecimal value of F (1111). Since all of the mask bits are 1, all of the signaling bits are significant. Thus, the A, B, C, and D bits all have a value of 1 after the transition to the OnVal.

PreTm = 100 ms
Specifies that the OffVal signaling bits must be present for at least 100 ms before they transition to the OnVal signaling values.

MinTm = 220 ms
Specifies that the OnVal signaling bits must be present for at least 220 ms before they transition to the OffVal signaling values.

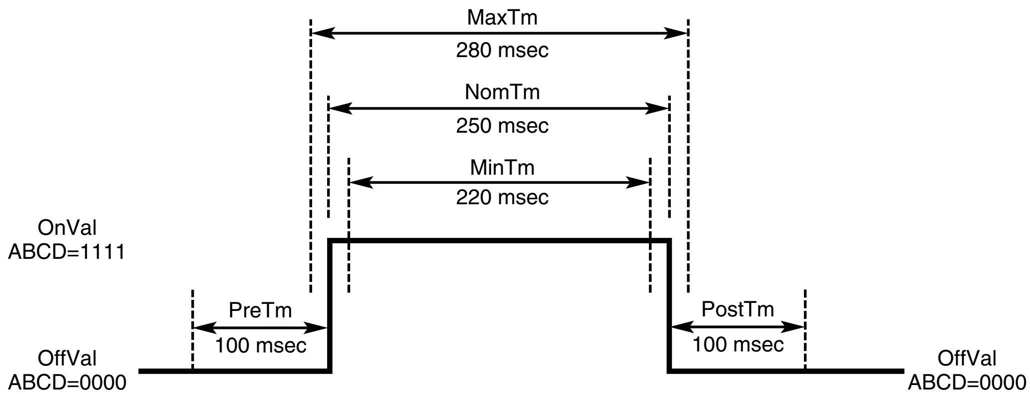
NomTm = 250 ms
Specifies that the OnVal signaling bits are generated for 250 ms before transitioning to the OffVal signaling values.

MaxTm = 280 ms
Specifies that the OnVal signaling bits must be present for no longer than 280 ms before they transition to the OffVal signaling values.

PostTm = 100 ms
Specifies that the OffVal signaling bits must be present for at least 100 ms before the signal is reported to the protocol (or if the signal is being sent, then the CAS component ensures that the OffVal signaling value is generated for at least 100 ms).

Figure 6 is a graphical representation of this signal definition.

Figure 6. Example of Wink Pulse Signal (0xC15CA011)



3.5.4 Train Signal

The train command defines a set of transitions from one signaling state to another in a predefined pattern (set of pulses). It is used to define CAS signals required by a protocol. The train command uses the following syntax:

```
train = SignalId, OffVal, OnVal, PulseTmlMin, PulseTmMax, PulseTmNom, preTm, interTmMin,
interTmMax, interTmNom, postTm, digitCount, pulseCount, label, pulseCount, label, ...
```

The train signal definition includes the following values:

SignalId
: Unique identifier (parameter number) of the train signal. The Channel Protocol (CHP) uses the SignalId to recognize the train when it is received, and also to generate the train when needed.

  Note: SignalId should not be modified by the user.

OffVal
: Defines the ABCD bit states on the line before the transition occurs. The four least significant bits represent the ABCD signaling bits (0 or 1). The four most significant bits represent a mask (A'B'C'D') that specifies if each corresponding bit value counts. See Figure 4, "Pre-transition ABCD Bit States", on page 35. If a mask bit is set to 1, the corresponding signaling bit is counted. If a mask bit is set to 0, the corresponding signaling bit is ignored.

OnVal
: Defines the ABCD bit states on the line during one pulse of the train. The format of this field is the same as the OffVal field.

PulseTmMin
: Defines the minimum time, in milliseconds, for the duration of the pulse interval.

PulseTmMax
: Defines the maximum time, in milliseconds, for the duration of the pulse interval.

PulseTmNom
: Defines the nominal time, in milliseconds, for the duration of the pulse interval.

preTm
: Defines the minimum time, in milliseconds, for the duration of the pre-train interval.

interTmMin
: Defines the minimum time, in milliseconds, for the duration of the inter-pulse interval.

interTmMax
: Defines the maximum time, in milliseconds, for the duration of the inter-pulse interval.

interTmNom
: Defines the nominal time, in milliseconds, for the duration of the inter-pulse interval.

postTm
: Defines the maximum time, in milliseconds, for the duration of the post-train interval.

digitCount
: Defines the number of digit definitions in the train. The pulse count for each digit (ASCII character) is defined by the label pairs following digitCount.

pulseCount
: Defines the number of train pulses that define the digit (ASCII character) identified by the label parameter.

label
: Defines the digit (ASCII character) associated with the corresponding pulseCount value.

Train Example

The following is an example of a train command that defines a train signal:

```
train = 0xC15CA032, 0xCC, 0xC4, 31, 33, 32, 600, 62, 66, 64, 20, 12, 10, 0, 1, 1, 2, 2, 3, 3, 4,
4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 11, #, 12, *
```

In the example shown, the train signal is defined as having the following values:

SignalId = 0xC15CA032
: Defines the CAS T1 loop start train signal parameter.

OffVal = 0xCC
: Defines both the mask and the signaling bits as having a hexadecimal value of C (1100). Since only mask bits A and B have a value of 1, only signaling bits A and B are significant. Thus, the A and B bits both have a value of 1 before the transition from the OffVal to the OnVal.

OnVal = 0xC4
: Defines the mask as having a hexadecimal value of C (1100) and the signaling bits as having a hexadecimal value of 4 (0100). Since mask bits A and B have a value of 1, signaling bits A and B are significant. Thus, the A bit has a value of 0 and the B bit has a value of 1 after the transition to the OnVal.

PulseTmMin = 31
: Specifies that the OnVal signaling bits must be present for at least 31 ms before they transition to the OffVal signaling values.

PulseTmMax = 33
: Specifies that the OnVal signaling bits must be present for no longer than 33 ms before they transition to the OffVal signaling values.

PulseTmNom = 32
: Specifies that the OnVal signaling bits must be present for 32 ms before they transition to the OffVal signaling values.

preTm = 600
: Specifies that the OffVal signaling bits must be present for 600 ms before the train signal begins.

interTmMin = 62
: Specifies that the OffVal signaling bits must be present for at least 62 ms before they transition to the OnVal signaling values.

interTmMax = 66
: Specifies that the OffVal signaling bits must be present for no longer than 66 ms before they transition to the OnVal signaling values.

interTmNom = 64
: Specifies that the OffVal signaling bits must be present for 64 ms before they transition to the OnVal signaling values.

postTm = 20
: Specifies that the OffVal signaling bits must be present for 20 ms before the signal is reported to the protocol (or if the signal is being sent, then the CAS component ensures that the OffVal value is generated for at least 20 ms).

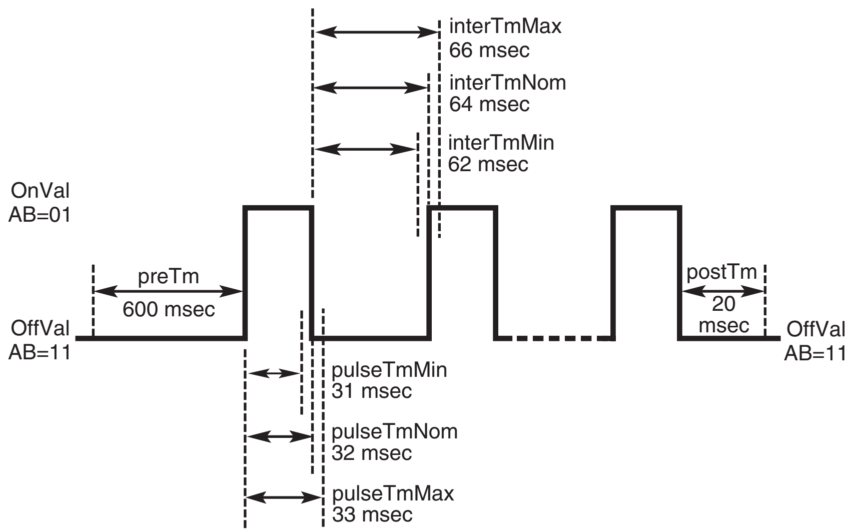
digitCount = 12
: Specifies that 12 digits/characters are defined by this train signal.

pulseCount, label pairs = 10,0 1,1 2,2 3,3 4,4 5,5 6,6 7,7 8,8 9,9 11,# 12,*

The first pair indicates that 10 pulses correspond to the digit "0", the next pair indicates that 1 pulse corresponds to the digit "1", and the last pair indicates that 12 pulses correspond to the ASCII character "*".

Figure 7 is a graphical representation of this signal definition.

Figure 7. Example of T1 Loop Start Train Signal (0xC15CA032)



3.5.5 Sequence Signal

The sequence command defines a set of train signals. It is used to define CAS signals required by a protocol. The sequence command uses the following syntax:

```
sequence = SignalId, TrainSigId, preTm, interTmMin, interTmMax, interTmNom, postTm
```

The sequence signal definition includes the following values:

SignalId

Unique identifier (parameter number) of the sequence signal. The Channel Protocol (CHP) uses the SignalId to recognize the sequence when it is received, and also to generate the sequence when needed.

Note: SignalId should not be modified by the user.

TrainSigId

Defines the train signal that the sequence signal uses.

preTm

Defines the minimum time, in milliseconds, for the duration of the pre-sequence interval.

interTmMin

Defines the minimum time, in milliseconds, for the duration of the inter-train interval.

interTmMax
Defines the maximum time, in milliseconds, for the duration of the inter-train interval.

interTmNom
Defines the nominal time, in milliseconds, for the duration of the inter-train interval.

postTm
Defines the minimal time, in milliseconds, for the duration of the post-sequence interval.

Sequence Example

The following is an example of a sequence command that defines a sequence signal:

```
sequence = 0xC15CA033, 0xC15CA032, 720, 640, 680, 660, 1600
```

In the example shown, the sequence signal is defined as having the following values:

SignalId = 0xC15CA033
Specifies the CAS T1 loop start sequence signal parameter.

TrainSigId = 0xC15CA032
Specifies the train signal definition that the sequence signal uses.

preTm = 720
Specifies that the OffVal signaling bits (as defined in the train definition) must be present for 720 ms before the sequence signal begins (that is, before the first train signal begins).

interTmMin = 640
Specifies that the OffVal signaling bits (as defined in the train definition) must be present for at least 640 ms between train signals.

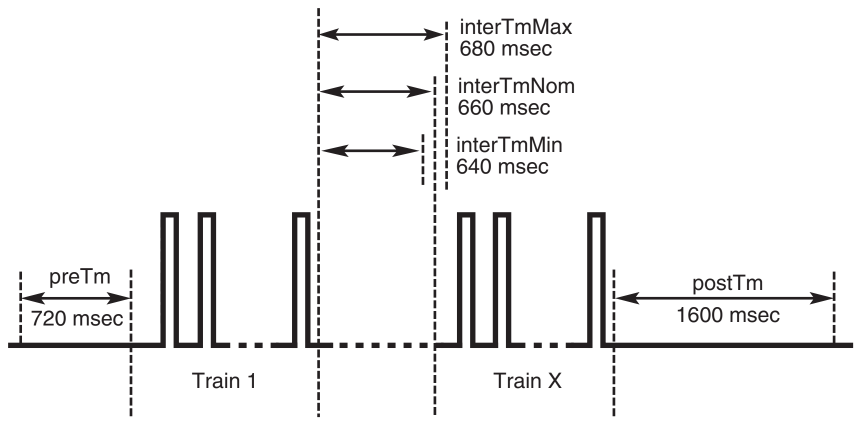
interTmMax = 680
Specifies that the OffVal signaling bits (as defined in the train definition) must be present for no longer than 680 ms between train signals.

interTmNom = 660
Specifies that the OffVal signaling bits (as defined in the train definition) must be present for 660 ms between train signals.

postTm = 1600
Specifies that the OffVal signaling bits (as defined in the train definition) must be present for 1600 ms before the signal is reported to the protocol (or if the signal is being sent, then the CAS component ensures that the OffVal value is generated for at least 1600 ms).

Figure 8 is a graphical representation of this signal definition.

Figure 8. Example of T1 Loop Start Sequence Signal (0xC15CA033)



3.6 [CHP] Section

The Channel Protocol (CHP) component implements the telephony communication protocol that is used on each network interface. There are different versions of this component for handling different signaling types as well as different protocol types on different B channels. There is one CHP instance created for each B channel in the system.

The [CHP] section of the CONFIG file is a subset of the [TSC] section. Protocol specific parameters, primarily in the form of variants, are defined in the [CHP] section. The selection of which of these protocol variants to use on which line (span) is determined in the [TSC] section. For more information on protocol variants selection, see Section 3.7, "[TSC] Section", on page 45.

A number of protocol variants are defined in the [CHP] section of the CONFIG file. Variants are defined by the Variant Define *n* command, where *n* is the variant identifier. The Variant Define *n* command defines variant "*n*" as all of the parameter definitions in the [CHP] section preceding the command.

Note: If a parameter is defined multiple times prior to the Variant Define *n* command, then only the last definition of the parameter is used for that variant.

Example

```
! T1 Protocol variant definitions
Variant VariantFormat          1      !T1 CAS format
Variant ProtocolType           1      !E&M=1,LS-FXS=2,GS-FXS=3
Variant Wink                   y
Variant Dial                   y
Variant DialFormat             1      !DTMF=1, MF, DP
Variant ANI                    0      !No=0, Pre, Post
Variant ANIFormat              1      !DTMF=1, MF, DP
Variant ANICount               0
Variant DNIS                   y
Variant DNISFormat             1      !DTMF=1, MF, DP
```

```
Variant DNISCount            0
Variant CallProgress         y
  .
  .
  .

! Define Protocol Variant 2 as T1 E&M Wink Start, DTMF Dial &
! DNIS + callProgress
Variant Define 2

! Note: Previous variant parms are kept, and the following
! commands replace the specified parameters

! Define Protocol Variant 5 as T1 Loop Start FXS, DTMF Dial
! & DNIS, callProgress, and DialTone detection.
Variant ProtocolType         2      !E&M=1,LS-FXS=2,GS-FXS=3
Variant Wink                 n
Variant Define 5

! Define Protocol Variant 4 as T1 E&M Wink Start, no tone
! and no callProgress
Variant ProtocolType         1      !E&M=1,LS-FXS=2,GS-FXS=3
Variant Wink                 y
Variant Dial                 n
Variant DNIS                 n
Variant CallProgress         n
Variant Define 4

! Define Protocol Variant 6 as Ground Start FXS
Variant ProtocolType         3      !E&M=1,LS-FXS=2,GS-FXS=3
Variant Wink                 n
Variant Dial                 y
Variant DNIS                 y
Variant CallProgress         y
Variant Define 6
```

From the [CHP] example, selecting protocol variant 2 would include all the parameter definitions from the beginning of the [CHP] section to the Variant Define 2 line. This would define the protocol as T1 E&M Wink start, DTMF dialing, DNIS digits, with call progress (ProtocolType = 1, Wink = y, Dial = y, DialFormat = 1, DNIS = y, CallProgress = y).

If, however, you were to select protocol variant 5, this would include all of the parameter definitions from the beginning of the [CHP] section to the Variant Define 5 line. In this case, the protocol type would change to LS-FSX (Loop start) and Wink Start would be disabled, but DTMF dialing, DNIS digits, and call progress would still be used (ProtocolType = 2, Wink = n).

If protocol variant 4 were selected, all of the parameter definitions from the beginning of the [CHP] section to the Variant Define 4 line would be included. Now, the protocol type would change back to E&M Wink start with no DTMF dialing, no DNIS digits, and no call progress (ProtocolType = 1, Wink = y, Dial = n, DNIS = n, CallProgress = n).

You may also create your own variant if none of the existing defined protocol variants match your need. For example, to create a new protocol variant in which you want to use E&M Immediate start (instead of Wink start) with no ANI/DNIS digits provided, you may add another Variant Define *n* after the Variant Define 2 statement. In this example, we can use n = 1 because this number has not yet been defined in the CONFIG file. That part of the [CHP] section would then become:

```
! Define Protocol Variant 2 as T1 E&M Wink Start, DTMF Dial & DNIS + callProgress
Variant Define 2
! Define Protocol Variant 1 as T1 E&M Immediate Start, DTMF Dial + callProgress
Variant Wink                    n
Variant DNIS                    n
Variant Define 1
```

By disabling DNIS in protocol variant 1, which follows protocol variant 2, we have also caused DNIS to be disabled in protocol variant 5. DNIS was originally enabled in protocol variant 5 because protocol variant 5 followed protocol variant 2 which defined it as enabled. We will now need to re-enable DNIS in protocol variant 5 as shown in the following example:

```
! Define Protocol Variant 5 as T1 Loop Start FXS, DTMF Dial
! & DNIS, callProgress, and DialTone detection.
! Add DNIS back to this protocol variant
Variant DNIS                    y
Variant ProtocolType 2  ! E&M=1, LS-FXS=2, GS-FXS=3
Variant Wink n
Variant Define 5
```

Although protocol variants are defined in the [CHP] section, protocol variants are assigned in the [TSC] section of the CONFIG file. Selecting a particular Variant Define n is accomplished by changing the values of the **Inbound** and **Outbound** parameters for a particular line. The **Inbound** and **Outbound** parameters are the sixth and seventh parameters respectively in the defineBSet command in the [TSC] section of the CONFIG file.

For information about the defineBSet command and setting TSC parameters, see Section 3.7, “[TSC] Section”, on page 45.

For information about each CHP parameter, see the following sections:

- Section 6.20, “[CHP] Parameters”, on page 136
- Section 6.21, “[CHP] T1 Protocol Variant Definitions”, on page 138
- Section 6.22, “[CHP] ISDN Protocol Variant Definitions”, on page 154

3.7 [TSC] Section

The [TSC] section of the CONFIG file defines a set of B channels and associated characteristics using the defineBSet command. The syntax of the defineBSet command is:

```
defineBSet = SetId, LineId, StartChan, NumChans, BaseProtocol, Inbound, OutBound, DChanDesc,
Admin, Width, BChanId, SlotId, Direction, Count, [BChanId, SlotId, Direction, Count,] 0
```

To change a [TSC] parameter, you change the value of the applicable defineBSet parameter in the CONFIG file. For example, to change the protocol variant from 2 to 4 for both inbound and outbound call processing on all 30 channels of line 2, you would change the value of the **Inbound** and **Outbound** parameters for line 2 (**SetId**=20) from 2 to 4. For information on defining protocol variants, see Section 3.6, “[CHP] Section”, on page 43.

Following is an excerpt from the [TSC] section of a CONFIG file that illustrates that part of the file before and after editing.

Before editing:

```
defineBSet=10,1,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=20,2,1,30, 0,2,2,1,20,1, 1,1,3,15, 16,17,3,15,0
```

After editing:

```
defineBSet=10,1,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=20,2,1,30, 0,4,4,1,20,1, 1,1,3,15, 16,17,3,15,0
```

For information about each TSC parameter, see Section 6.25, "[TSC] defineBSet Parameters", on page 163.

3.8 Configuring Trunks for Clear Channel Signaling

For DM/V600BTEP, DM/V600TEC, DM/V1200TEP, DM/V1200TEC, DMN160TEC and DMT160TEC boards, trunks can be configured for clear channel signaling on a trunk by trunk basis. This allows the mixing of ISDN trunks and clear channel trunks on the same board. For T1 trunks in a clear channel configuration, time slot 24 can be used as a bearer channel, in addition to the other 23 bearer channels. For E1 trunks in a clear channel configuration, time slot 16 can be used as a bearer channel, in addition to the other 30 bearer channels.

Refer to Section 4.4.1, "Configuring Board Settings", on page 50 for information about configuring trunks on these boards by creating a PCD file with multiple trunk protocols.

Configuration Procedures 4

This chapter provides detailed procedures for configuring a system. This also includes assumptions, prerequisites, and the order in which to perform the procedures. The configuration procedures are divided into the following categories:

4.1 Assumptions and Prerequisites

The following assumptions and prerequisites exist regarding the configuration procedures:

- All required Intel® System Release software, including prerequisites, have been installed according to the procedures in the software installation guide supplied with the system release.
- The Intel® System Release was installed in the default directory */usr/dialogic*. Command instructions, directories paths and environment variable are shown relative to the default subdirectory.
- If a third-party board is being used as the primary clock master, it is assumed that third-party technology can use a range of time slots starting at 0.

- If SNMP agent software is installed, it is assumed all prerequisites have been met as outlined in the software installation guide and SNMP agent software administration guide supplied with your system release.
- The following GNU General Public License (GPL) tools have been installed: ld960, nm960, ldnew, ldarm, and nmarm. These files are required by PDKManager and the source files can be downloaded from the GNU ftp site *ftp://ftp.gnu.org/gnu/binutils/binutils-2.13.tar.gz*. For more information about the GNU Binutils, see *http://www.gnu.org/directory/binutils.html*.

4.2 Order of Procedures

Except for Modifying the FCD File by Editing the CONFIG File, configuration procedures should be performed in the order presented below. FCD file parameter modifications can be made any time prior to starting the system.

Procedures that are required when initially configuring any system are noted as such. The additional procedures may be required depending on your system.

1. Starting the Board Configuration Utility (config.sh) (**required**)
2. Using the DM3 Board Configuration Utility (DM3_cfg.sh) (**required**)
3. Using the SNMP Agents Configuration Tool (dlgcsnmpconf)
4. Completing the Board Configuration Utility (config.sh) (**required**)
5. Installing the HSK Software (hs_bld_kernel.sh)
6. Configuring the HSK Configuration Files
7. Pre-configuring Empty Slots for CompactPCI Boards
8. Assigning Time Slots When Using a Third-Party Board as Clock Master
9. Setting the ClockDaemonMode
10. Modifying the FCD File by Editing the CONFIG File
11. Transferring SNMP Agent Files to the Network Management Station
12. Configuring Global Call CDP File
13. Initializing the System (**required**)
14. Reconfiguring the System

4.3 Starting the Board Configuration Utility (config.sh)

If you want to keep a record of all configuration prompts and responses, use the Linux *script* utility prior to starting the board configuration procedure. You can then see the *script* output file for information such as configuration parameter selections. For information about using the *script* utility, see the Red Hat Linux documentation.

The following procedure explains the initial steps of board configuration. After these initial steps, further instructions depend on the system you are configuring.

1. Enter the following command to start the board configuration script:

   ```
   ./config.sh
   ```

   The system displays the following messages:

   ```
                   Intel(R) Dialogic(R) System Release 6.1 for Linux
                               CONFIGURATION

   This is the configuration tool for Intel(R) Dialogic(R) System Release 6.1 PCI
   for Linux.
   You will be asked to supply information for configuring the software.

   Would you like to configure SNMP on this system (y/n, default=n) ?
   ```

2. If you installed the SNMP agent software, type `y`; otherwise type `n`.

   Note: Do not enter y to configure SNMP if you have not installed the SNMP agent software. If you do this, the configuration procedure is aborted and you will be prompted to run the installation script (install.sh) again so you can install the SNMP agent software.

   If you enter y to configure SNMP, the SNMP Agents Configuration Tool is automatically invoked when board configuration is complete.

   After you respond to the prompt for configuring SNMP, the system displays the following messages:

   ```
   Copying driver files.....
   Drivers will now be loaded...

   Starting DM3-specific configuration...

   Press ENTER to continue...
   ```

3. Press Enter. At this time, *config.sh* invokes DM3_cfg.

Proceed with Section 4.4, “Using the DM3 Board Configuration Utility (DM3_cfg.sh)”, on page 49.

4.4 Using the DM3 Board Configuration Utility (DM3_cfg.sh)

The output of the DM3 Board Configuration (*DM3_cfg.sh*) utility is the *pyramid.scd* (SCD) file. The *DM3_cfg.sh* part of the configuration begins with the DM3 Board Configuration - Main Screen. Online help is available for all parameters accessible through the DM3 Board Configuration screens. To access the online help, type a question mark (?) instead of a value from any parameter screen. The procedure for configuring parameters using the DM3 Board Configuration utility includes the following:

- Configuring Board Settings

- Configuring the TDM Bus

Note: If the *DM3_cfg.sh* utility is run a second time (for example, to reconfigure a system after it is initialized), a backup copy of the existing SCD file is created. A new SCD file is created and all previously configured parameter settings are removed from the file and replaced by the system default values. The file name of the backup SCD file is in the format:

pyramid.scd_<year><day of year><hour><minute>

For example, a copy saved on June 25 (176th day of the year), 2005 at 4:58 p.m. would be named *pyramid.scd_20051761658*.

4.4.1 Configuring Board Settings

The DM3 Board Configuration - Main Screen lists all boards that are installed. For example:

```
                .......................................
                 DM3 Board Configuration - Main Screen
                .......................................
   This is a summary of the current board configuration
-------------------------------------------------------------
PCI    Board                   Logical      PCD
Slot  Enabled    Model           ID       File Name
0       Yes      DM/V1200-4E1    1        NOT_SET
1       Yes      DM/V960-4T1     2        NOT_SET
6       Yes      DMV1200BTEC     3        NOT_SET
13      Yes      DMN160TEC       14       NOT_SET
-------------------------------------------------------------
T        -     TDM BUS SETTINGS  -        H100
-------------------------------------------------------------
.     You must configure or disable each board shown.        .
.     When a board is configured a valid PCD file name       .
.     is displayed in the PCD File Name column.              .
-------------------------------------------------------------
Enter the slot number of the board to configure,
T to change TDM Bus Settings,S to save and quit,
or Q to quit without saving:
```

Proceed as follows:

1. Enter the slot number of the board to configure. (Slot numbers are displayed on the left of the Main Screen.) For example, to select the DM/V1200-4E1, you would, in this case, enter the number `0`.

   The Modify Board Settings screen is then displayed. This screen contains the current board settings for the board selected. For example:

```
                          .........................
                           Modify Board Settings
                          .........................
      These are the current settings for the board selected.
-------------------------------------------------------------------
(read only)..........Physical Slot: 0
(read only).............Model: DM/V1200-4E1
-------------------------------------------------------------------
. A PCD file must be selected. The corresponding FCD file will    .
. be automatically selected. Other parameters can be modified,    .
. or the default values shown will be used. To view default       .
. values for NIC Configuration, select that item.                 .
-------------------------------------------------------------------
1) ................Logical ID: 1
2) ............Board Enabled: Yes
3) ............PCD File Name: NOT_SET
4) ............FCD File Name: NOT_SET
5) .............PCM Encoding: NOT_SET
```

```
6) ........TDM Clock Function: SLAVE
7) .........NIC Configuration  (IPT w/onboard NIC only)
-------------------------------------------------------------------
Enter the number of the parameter to modify, S to save and return to
the Main Screen, or C to cancel and return to the Main Screen.
```

For information about each parameter, refer to Chapter 5, "SCD File Parameter Reference".

Note: Number 7, NIC Configuration, only applies to Intel® NetStructure DM/IP boards.

2. Type the number `1` and press Enter if you want to change the default value for the board **Logical ID** parameter. The Set Board Logical ID screen is then displayed.

```
.................................
.     Set Board Logical ID      .
.................................
_________________________________
The board logical ID must be a
unique positive integer from 1-16
_________________________________
Currently: 1
_________________________________
Enter a new value or press ? for help or
press Enter to return to the previous screen:
```

3. To change the **Logical ID**, type the **Logical ID** number and press Enter. To accept the current value, press Enter. The Modify Board Settings screen is then displayed with the new value for **Logical ID**.

   Note: The logical ID number must be unique to each board.

4. Type the number `2` and press Enter to disable (or enable) the board. The Enable or Disable Board screen is then displayed.

```
.............................................................
.                  Enable or Disable Board                  .
.............................................................
_____________________________________________________________
Choose 1 if you want to enable this board.
Selecting 2 will disable the board and comment out the board
parameters in the System Configuration Description (SCD) file.
_____________________________________________________________
1 - Yes, I want to enable this board
2 - No, I want to disable this board
_____________________________________________________________
Currently: Yes
_____________________________________________________________
Enter a new value or press ? for help or
press Enter to return to the previous screen:
```

5. If the board is currently enabled and you wish to disable it, type the number 2 and press Enter. If the board is currently disabled and you wish to enable it, type the number 1 and press Enter. The Modify Board Settings screen is then displayed with the new value for **Board Enabled**.

   Note: If this board is currently enabled and is serving as a clock master or a clock reference source for the system, you will not be allowed to disable the board from this screen.

6. Type the number `3` and press Enter to select a PCD file.

   Note: To create a PCD file for a DM/V-B, DMN160TEC, or DMT160TEC board, skip to Step 8. For all other boards, proceed with Step 7.

   Note: A value for the **PCD File Name** parameter must be selected, because the default value of NOT_SET cannot be used.

For details about configuration file sets, see Section 2.2, "Configuration File Sets", on page 17.

7. The Specify the PCD File screen is then displayed:

```
..................................................................
.                     Specify the PCD File                       .
..................................................................
__________________________________________________________________
A Product Configuration Description (PCD) file must be selected
in order to configure your board. The corresponding Feature
Configuration Description (FCD) file will be automatically selected.
For features available in specific PCD files, see the Release Guide.
The following PCD files are valid for your board:
__________________________________________________________________
1 - ml1_qs_dass2.pcd
2 - ml1_qs_dpnss.pcd
3 - qs_isdn_net5.pcd
4 - ml1_qs_net5.pcd
5 - qs_isdn_qsige1.pcd
6 - ml1_qs_qsige1.pcd
7 - qs_r2mf.pcd
8 - ml1_qs_r2mf.pcd
9 - ml1_qs_ts16.pcd
Currently: NOT_SET
__________________________________________________________________


You have several options:
------------------------------------------------------------------
# - Enter the number of the PCD you want to select from the list above to use
one of the supplied single protocol files or your own previously created file.
d - Create new PCD files with multiple protocols for the trunks.
? - Help
Enter - Return to the previous screen
```

7a. Enter the number of the PCD file you want to select for the board, then press Enter.

The Modify Board Settings screen is then displayed with the new value for **PCD File Name** and **FCD File Name**.

```
                     ........................
                      Modify Board Settings
                     ........................
      These are the current settings for the board selected.
-----------------------------------------------------------------
(read only)..........Physical Slot: 0
(read only)............Model: DM/V1200-4E1
-----------------------------------------------------------------
. A PCD file must be selected. The corresponding FCD file will    .
. be automatically selected. Other parameters can be modified,    .
. or the default values shown will be used. To view default       .
. values for NIC Configuration, select that item.                 .
-----------------------------------------------------------------
1) ...............Logical ID: 1
2) ............Board Enabled: Yes
3) ............PCD File Name: ml1_qs_net5.pcd
4) ............FCD File Name: ml1_qs_net5.fcd
5) .............PCM Encoding: NOT_SET
6) ........TDM Clock Function: SLAVE
7) .........NIC Configuration   (IPT w/onboard NIC only)
-----------------------------------------------------------------
Enter the number of the parameter to modify, S to save and return to
the Main Screen, or C to cancel and return to the Main Screen.
```

Note: The **FCD File Name** parameter will default to a file with the same name as the **PCD File Name** parameter, but with a *.fcd* extension. The **FCD File Name** parameter cannot be set by typing the number 4 from the Modify Board Settings screen.

7b. Skip to Step 9.

To configure a DM/V-B board **that has network interfaces (for example, a DM/V1200BTEP)**, a DMN160TEC board, or a DMT160TEC board proceed as follows:

8. Type the number 3 and press Enter to create a PCD file.

   For a DM/V-B board, a Specify the PCD File screen similar to the following is then displayed:

```
..................................................................
.                    Specify the PCD File                       .
..................................................................
__________________________________________________________________
A Product Configuration Description (PCD) file must be selected
in order to configure your board. The corresponding Feature
Configuration Description (FCD) file will be automatically selected.
For features available in specific PCD files, see the Release Guide.
The following PCD files are valid for your board:
__________________________________________________________________
1 - qsb_default.pcd
Currently: NOT_SET
__________________________________________________________________


You have several options:
------------------------------------------------------------------
# - Enter the number of the PCD you want to select from the list above to use
one of the supplied single protocol files or your own previously created file.
d - Create new PCD files with multiple protocols for the trunks.
? - Help
Enter - Return to the previous screen
```

   If you are creating a PCD file for a DM/V-B board and select option d, addtional information similar to the following is displayed provided that a PCD file is currently selected:

```
Current board configuration for qsb_default.pcd is:
        Media Load = UL1
        Trunk 1 = NI2(T1)
        Trunk 2 = NI2(T1)
        Trunk 3 = NI2(T1)
        Trunk 3 = NI2(T1)

Please press ‘q’ to exit from trunk configuration or any key to continue....
```

   For a DMN160TEC or DNMT160TEC board, a Specify the PCD File screen similar to the following is then displayed:

```
..................................................................
.                    Specify the PCD File                       .
..................................................................
__________________________________________________________________
A Product Configuration Description (PCD) file must be selected
in order to configure your board. The corresponding Feature
Configuration Description (FCD) file will be automatically selected.
For features available in specific PCD files, see the Release Guide.
The following PCD files are valid for your board:
__________________________________________________________________
1 - 16xt_default.pcd
2 - gtone_16xt_16_4ess.pcd
3 - gtone_16xt_16_5ess.pcd
4 - gtone_16xt_16_cas.pcd
5 - gtone_16xt_16_dms.pcd
6 - gtone_16xt_16_e1cc.pcd
7 - gtone_16xt_16_isdne1cc.pcd|
8 - gtone_16xt_16_net5.pcd
9 - gtone_16xt_16_ni2.pcd
10 - gtone_16xt_16_ntt.pcd
11 - gtone_16xt_16_qsige1.pcd
12 - gtone_16xt_16_qsigt1.pcd
13 - gtone_16xt_16_r2mf.pcd
14 - gtone_16xt_16_t1cc.pcd
15 - gtone_16xt_8_4ess_8_isdnt1cc.pcd
16 - gtone_16xt_8_isdnt1cc_8_net5.pcd
17 - gtone_16xt_8_net5_8_isdne1cc.pcd
Currently: NOT_SET
__________________________________________________________________

You have several options :
```

```
------------------------
# - Enter the number of the PCD file you want to select from the list above to use
one of the supplied single protocol files or your own previoulsy created file
d - Create new PCD files with multiple protocols for the trunks.
? - Help
Enter - Return to the previous screen
```

8a. Select option d to create a PCD file, then press Enter. For a DM/V-B board, the following screen is then displayed:

Note: If only one Media Load is available for this board, or you are creating a PCD file for a DMN160TEC or DMT160TEC board, the following Supported Media Loads screen will not be displayed.

```
To create a configuration, you must select from the media load and protocol
options that are available for the board. Start by selecting a media load from
the following list of media loads supported for the board.

Supported Media Loads:
-------------------------------------------------------------------
Index         Media Load

1              UL1
2              ML5B
3              ML10

# - Enter the Index number for the media load you want to use.
q - Exit from configuring trunks.
```

8b. For a DM/V-B board, type the index number of the media load you wish to use and press Enter to select a media load.

Depending on the board type, one of the following protocol selection screens will be displayed.

For a DM/V-B board, the following screen will be displayed:

```
To create configuration, select trunk protocols from the following list. Refer to a
protocol by its Protocol ID. All the selected protocols have the same Group number.
Protocols from mixed groups are not supported. Enter protocols for all trunks.

Number of trunks for this board = 4

                      Supported Protocols and Their Groups :
                      ------------------------------------
Protocol ID   Group   Protocol Name    Protocol ID   Group   Protocol Name
1             1       4ESS(T1)         8             1       NET5(E1)
2             1       5ESS(T1)         9             2       CAS(T1)
3             1       NTT(T1)          10            2       T1CC(T1)
4             1       NI2(T1)          11            3       R2MF(E1)
5             1       DMS(T1)          12            3       E1CC(E1)
6             1       QSIGT1(T1)       13            4       DPNSS(E1)
7             1       QSIGE1(E1)       14            4       DASS2(E1)

<Trunk Number>:<Protocol ID> - Enter the trunk number(s) for the board and the
protocol(s) you want to select for each trunk in the following format.

<Trunk Number>:<Protcol ID>,<Trunk Number>:<Protcol ID> etc.

For example for a board with 4 trunks enter the data as  1:2,2:3,3:3,4:1

For consecutive trunks with the same protocol, enter the trunk number as  <Start
Trunk Number-End Trunk Number>.

For example for board with 4 trunks, enter the data as ; 1-4:3  or  1-2:1,3-4:2

q - Exit from configuring trunks.
```

For a DMN160TEC board, the following screen will be displayed:

```
To create configuration, select trunk protocols from the following list. Refer to a
protocol by its Protocol ID. All the selected protocols have the same Group number.
Protocols from mixed groups are not supported. Enter protocols for all trunks.

Number of trunks for this board = 16

                      Supported Protocols and Their Groups :
                      -------------------------------------
Protocol ID   Group   Protocol Name    Protocol ID   Group   Protocol Name
1             1       4ESS(T1)         8             1       QSIGE1(E1)
2             1       5ESS(T1)         9             1       NET5(E1)
3             1       NTT(T1)          10            1       ISDNE1CC(E1)
4             1       NI2(T1)          11            2       T1CC(T1)
5             1       DMS(T1)          12            3       E1CC(E1)
6             1       QSIGT1(T1)       13            4       DPNSS(E1)
7             1       ISDNT1CC(T1)     14            4       DASS2(E1)

<Trunk Number>:<Protocol ID> - Enter the trunk number(s) for the board and the
protocol(s) you want to select for each trunk in the following format.

<Trunk Number>:<Protcol ID>,<Trunk Number>:<Protcol ID> etc.

For example for a board with 4 trunks enter the data as  1:2,2:3,3:3,4:1

For consecutive trunks with the same protocol, enter the trunk number as  <Start
Trunk Number-End Trunk Number>.

For example for board with 4 trunks, enter the data as ; 1-4:3  or  1-2:1,3-4:2

q - Exit from configuring trunks.
```

For a DMT160TEC, the following screen will be displayed:

```
To create configuration, select trunk protocols from the following list. Refer to a
protocol by its Protocol ID. All the selected protocols have the same Group number.
Protocols from mixed groups are not supported. Enter protocols for all trunks.

Number of trunks for this board = 16

                      Supported Protocols and Their Groups :
                      -------------------------------------
Protocol ID   Group   Protocol Name    Protocol ID   Group   Protocol Name
1              1      4ESS(T1)         9             1       NET5(E1)
2              1      5ESS(T1)         10            1       ISDNE1CC(E1)
3              1      NTT(T1)          11            2       T1CC(T1)
4              1      NI2(T1)          12            3       E1CC(E1)
5              1      DMS(T1)          13            4       DPNSS(E1)
6              1      QSIGT1(T1)       14            4       DASS2(E1)
7              1      ISDNT1CC(T1)     15            2       CAS(T1)
8              1      QSIGE1(E1)       16            3       R2MF(E1)

<Trunk Number>:<Protocol ID> - Enter the trunk number(s) for the board and the
protocol(s) you want to select for each trunk in the following format.

<Trunk Number>:<Protcol ID>,<Trunk Number>:<Protcol ID> etc.

For example for a board with 4 trunks enter the data as  1:2,2:3,3:3,4:1

For consecutive trunks with the same protocol, enter the trunk number as  <Start
Trunk Number-End Trunk Number>.

For example for board with 4 trunks, enter the data as ; 1-4:3  or  1-2:1,3-4:2

q - Exit from configuring trunks.
```

8c. Select trunk protocols for all of the trunks.

For example, to create a PCD file for a DM/V-B board with Trunk 1 and Trunk 2 using the 4ESS protocol, and Trunk 3 and Trunk 4 using the 5ESS protocol, enter the following:

1-2:1,3-4:2

The following screen will be displayed:

```
The custom gull_qsb_2_4ess_2_5ess PCD, FCD and CONFIG files successfully generated.

Press any key to continue...
```

Note: Not all protocol groups can be combined on a single board:

Group 1 cannot be combined with any other group.

Group 4 cannot be combined with any other group.

Groups 2 and 3 can be combined with each other on the same board, but neither can be combined with group 1 or 4.

Note: For DMN160TEC and DMT160TEC boards, each set of four trunks (1-4, 5-8, 9-12, and 13-16) must be assigned the same Protocol ID. You can assign a different Protocol ID to each set of four trunks, but they must all be from the same Group number. For example, if you assign the 4ESS (T1) protocol from Group 1 to the first set of four trunks, you can only assign protocol or Clear Channel values from Group 1, such as ISDNE1CC(E1), to the other sets of trunks. You can mix T1 and E1 protocols on the same board as long as they are from the same Group.

Note: When combining T1 and E1 protocols on the same DM/V-B, DMN160TEC, or DMT160TEC board, refer to the PCM Encoding (PCMEncoding) parameter in Chapter 5, "SCD File Parameter Reference" for information about A-law to Mu-law conversion.

8d. Press any key. The DM3 Board Configuration - Main Screen will be displayed with the PCD and FCD file names appearing.

```
  .........................
                     Modify Board Settings
                   .........................
      These are the current settings for the board selected.
-------------------------------------------------------------------
(read only)..........Physical Slot: 6
(read only)............Model: DM/V1200BTEC
-------------------------------------------------------------------
. A PCD file must be selected. The corresponding FCD file will    .
. be automatically selected. Other parameters can be modified,    .
. or the default values shown will be used. To view default       .
. values for NIC Configuration, select that item.                 .
-------------------------------------------------------------------
1) ................Logical ID: 3
2) ............Board Enabled: Yes
3) ............PCD File Name: gull_qsb_2_4ess_2_5ess.pcd
4) ............FCD File Name: gull_qsb_2_4ess_2_5ess.fcd
5) .............PCM Encoding: NOT_SET
6) ........TDM Clock Function: SLAVE
7) .........NIC Configuration  (IPT w/onboard NIC only)
-------------------------------------------------------------------
Enter the number of the parameter to modify, S to save and return to
the Main Screen, or C to cancel and return to the Main Screen.
```

8e. When you are finished with the board settings for this board, enter `S` to save and return to the Main Screen (or `C` to cancel and return to the Main Screen). The current board configuration is displayed on the Main Screen again, and you can:

- Enter another slot number to configure another board. The current board settings for that board are displayed as shown in Step 1. Follow this procedure from Step 1 to Step 13 for each board to be configured.

- Enter `T` to configure the TDM Bus Settings. Refer to Section 4.4.2, “Configuring the TDM Bus”, on page 59.
- Enter `S` to save the current configuration and quit.
- Enter `Q` to quit without saving. (The previously saved or default parameter values will be used.)

If you quit the DM3 Board Configuration (*DM3_cfg.sh*) utility by entering `S` or `Q`, processing returns to the *config.sh* utility and proceeds as follows:

- If SNMP agent software is installed, *config.sh* invokes the SNMP Agents Configuration Tool (*dlgcsnmpconf*). Refer to Section 4.5, “Using the SNMP Agents Configuration Tool (dlgcsnmpconf)”, on page 65.
- If you are satisfied with all of the board configuration information, refer to Section 4.6, “Completing the Board Configuration Utility (config.sh)”, on page 69.

9. Type the number `5` and press Enter to select the PCM encoding method. The Set PCM Encoding Method screen is displayed.

```
...............................................................
.                  Set PCM Encoding Method                    .
...............................................................
_______________________________________________________________
Use the default value unless NOT_SET is displayed.
NOT_SET indicates that the PCM encoding cannot be determined.
This usually occurs because the board does not have a telephony
network front end (e.g., resource-only board).
Select one of the following:
_______________________________________________________________
1 - MULAW (standard for T1)
2 - ALAW (standard for E1)
_______________________________________________________________
Currently: ALAW
_______________________________________________________________
Enter a new value or press ? for help or
press Enter to return to the previous screen:
```

10. Type the number `1` and press Enter to set the **PCM Encoding** parameter to MULAW, or type the number `2` and press Enter to set the **PCM Encoding** parameter to ALAW. The Modify Board Settings screen is displayed with the new value for the **PCM Encoding** parameter.

    Note: For DM/V-B boards (excluding DM/V-B resource-only boards), DMN160TEC, and DMT160TEC boards, you can set their network interfaces to either T-1 or E-1 in the same system, regardless of the CT Bus PCM encoding method. (Network interfaces are configured based on the media load specified for downloading that board.)

    For example, if the PCM encoding method on the CT Bus is set to A-law, a DM/V-B board that has some or all of its network (front end) interfaces configured for T-1, will automatically convert the A-law data sent to and received from the CT Bus to Mu-law for transmitting and receiving on the T-1 configured front ends. The board will always transmit to and receive from each front end using the PCM encoding method determined by the network interface setting.

    For all other boards, the CT Bus PCM encoding method selected by the user must match the PCM Encoding of the network interface or the board will not start properly.

```
                      ........................
                       Modify Board Settings
                      ........................
      These are the current settings for the board selected.
-------------------------------------------------------------------
(read only)..........Physical Slot: 0
(read only).............Model: DM/V1200-4E1
-------------------------------------------------------------------
. A PCD file must be selected. The corresponding FCD file will    .
. be automatically selected. Other parameters can be modified,    .
. or the default values shown will be used. To view default       .
. values for NIC Configuration, select that item.                 .
-------------------------------------------------------------------
1) ................Logical ID: 1
2) ............Board Enabled: Yes
3) ............PCD File Name: ml1_qs_net5.pcd
4) ............FCD File Name: ml1_qs_net5.fcd
5) .............PCM Encoding: ALAW
6) ........TDM Clock Function: SLAVE
7) .........NIC Configuration   (IPT w/onboard NIC only)
-------------------------------------------------------------------
Enter the number of the parameter to modify, S to save and return to
the Main Screen, or C to cancel and return to the Main Screen.
```

11. Type the number `6` and press Enter if you want to select this board as the Primary Clock Master or Secondary Clock Master. (The default is slave.) The Set TDM Bus Master Status screen is displayed.

```
......................................................................
.                  Set TDM Bus Master Status                          .
......................................................................
______________________________________________________________________
Select one of the following:
______________________________________________________________________
1 - PRIMARY (This board will act as Clock Master)
2 - SECONDARY (CT Bus Only, backup Clock Master if the Primary fails)
3 - SLAVE (always gets Clock from another board)
______________________________________________________________________
Currently: SLAVE
______________________________________________________________________
Enter a new value or press ? for help or
press Enter to return to the previous screen:
```

12. Type the number `1 to 3` that corresponds to the TDM clocking function you wish to select for this board, then press Enter to return to the Modify Board Settings screen.

    Note: Only one board can be specified as Primary Clock Master, and only one board can be specified as Secondary Clock Master.

    Note: For information about selecting the Reference Master (that is, the board that provides the NetRef signal), see Section 4.4.2, "Configuring the TDM Bus", on page 59.

    For example, to set the board as the Primary Clock Master, type `1` and press Enter. The Modify Board Settings screen is displayed with the new value for the **TDM Clock Function** parameter.

```
                      ........................
                       Modify Board Settings
                      ........................
      These are the current settings for the board selected.
-------------------------------------------------------------------
(read only)..........Physical Slot: 0
(read only).............Model: DM/V1200-4E1
-------------------------------------------------------------------
. A PCD file must be selected. The corresponding FCD file will    .
. be automatically selected. Other parameters can be modified,    .
. or the default values shown will be used. To view default       .
. values for NIC Configuration, select that item.                 .
-------------------------------------------------------------------
1) ................Logical ID: 1
```

```
2) .............Board Enabled: Yes
3) .............PCD File Name: ml1_qs_net5.pcd
4) .............FCD File Name: ml1_qs_net5.fcd
5) ..............PCM Encoding: ALAW
6) ........TDM Clock Function: PRIMARY
7) .........NIC Configuration   (IPT w/onboard NIC only)
-------------------------------------------------------------------
Enter the number of the parameter to modify, S to save and return to
the Main Screen, or C to cancel and return to the Main Screen.
```

13. When you are finished with the board settings for this board, enter `S` to save and return to the Main Screen (or `C` to cancel and return to the Main Screen). The current board configuration is displayed on the Main Screen again, and you can:
    - Enter another slot number to configure another board. The current board settings for that board are displayed as shown in Step 1. Follow this procedure from Step 1 to Step 13 for each board to be configured.
    - Enter `T` to configure the TDM Bus Settings. Refer to Section 4.4.2, "Configuring the TDM Bus", on page 59.
    - Enter `S` to save the current configuration and quit.
    - Enter `Q` to quit without saving. (The previously saved or default parameter values will be used.)

If you quit the DM3 Board Configuration (*DM3_cfg.sh*) utility by entering `S` or `Q`, processing returns to the *config.sh* utility and proceeds as follows:

- If SNMP agent software is installed, *config.sh* invokes the SNMP Agents Configuration Tool (*dlgcsnmpconf*). Refer to Section 4.5, "Using the SNMP Agents Configuration Tool (dlgcsnmpconf)", on page 65.
- If you are satisfied with all of the board configuration information, refer to Section 4.6, "Completing the Board Configuration Utility (config.sh)", on page 69.

The first time you configure a DM/V-B Series board, you can select different configuration files than the default files assigned by the system.

4.4.2 Configuring the TDM Bus

When you enter the letter T from the Main Screen to configure the TDM bus, the Modify TDM Bus Configuration screen is displayed:

```
              ..............................
               Modify TDM Bus Configuration
              ..............................
These are the current TDM related settings for the system.
Set a clock master on the Modify Board page for that board.
-------------------------------------------------------------------
(read only)..........Clock Master: (Slot 0) DM/V1200-4E1
(read only)......Bus Clock Rate: 8
-------------------------------------------------------------------
1) ......................TDM Bus Type: H100
2) ..............Primary Clock Source: OSC
3) .........Primary CT Bus Clock Line: CT_A
4) ......................NetRef Board: (Slot NA) NA
5) .........NetRef Clock Source Trunk: NA
6) ............NetRef1 Fallback Board: (Slot NA) NA
```

```
7) ...NR1 Fallback Clock Source Trunk: NA
-------------------------------------------------------------------
Enter the number of the parameter to modify.
S to Save and return to the Main Screen, or
C to Cancel and return to the Main Screen.
```

Note: The **Bus Clock Rate** is displayed on the TDM Bus Configuration screen as a read-only parameter. You do not have to set this parameter; its value depends on the **TDM Bus Type**. When the **TDM Bus Type** is H100 or H110, the **Bus Clock Rate** is automatically set to 8 MHz.

Note: In a mixed system using both DM3 and Springware architecture boards, either all DM3 or all Springware boards must be used for configuring the system clocking.

For information about each TDM bus parameter, refer to Section 5.1.2, "TDM Bus Configuration", on page 88. For additional reference information about TDM Bus concepts, especially fallback clocking, see Section 2.5, "CT Bus Clock Fallback", on page 27.

Proceed as follows:

1. Type the number `1` and press Enter to configure the TDM Bus Type parameter. The Set TDM Bus Type screen is then displayed.

```
................................................
.               Set TDM Bus Type               .
................................................
________________________________________________
This value is used for the Primary Clock Master.
Select one of the following:
________________________________________________
1 - H100
2 - H110
________________________________________________
Currently: H100
________________________________________________
Enter a new value or press ? for help or
press Enter to return to the previous screen:
```

2. Type the number that corresponds to the TDM Bus Type you wish to select. For example, type `2` to select H.110 for the TDM Bus Type.. Then press Enter to return to the Modify TDM Bus Configuration screen.

   The TDM Bus Configuration screen is displayed with the value selected for the TDM Bus Type.

```
              ..............................
               Modify TDM Bus Configuration
              ..............................
These are the current settings for the PRIMARY clock master.
The same settings will be used for all boards in the system.
-------------------------------------------------------------------
(read only).........Clock Master: (Slot 0) DM/V1200-4E1
(read only)......Bus Clock Rate: 8
-------------------------------------------------------------------
1) .....................TDM Bus Type: H100
2) .............Primary Clock Source: OSC
3) .........Primary CT Bus Clock Line: CT_A
4) .....................NetRef Board: (Slot NA) NA
5) .........NetRef Clock Source Trunk: NA
6) ...........NetRef1 Fallback Board: (Slot NA) NA
7) ...NR1 Fallback Clock Source Trunk: NA
-------------------------------------------------------------------
Enter the number of the parameter to modify.
S to Save and return to the Main Screen, or
C to Cancel and return to the Main Screen.
```

3. Type the number `2` and press Enter to configure the Primary Clock Source parameter. The Set Clock Source screen is then displayed.

```
.........................................
.           Set Clock Source            .
.........................................
_________________________________________
Select one of the following:
_________________________________________
O - Oscillator (board's internal clock)
1-4 - Trunk 1-4
N1 - NETREF 1
N2 - NETREF 2 (H110 only)
_________________________________________
Currently: OSC
_________________________________________
Enter a new value or press ? for help or
press Enter to return to the previous screen:N1
```

4. Type the letter `O` to select the internal oscillator, type a trunk number to select a trunk, or type `N1` to select NetRef1.

   Note: For DMN160TEC and DMT160TEC boards, the allowable trunk range is from 1 to 16.

   Then, press Enter to return to the Modify TDM Bus Configuration screen. For example, to select NetRef1 as the clock source, type `N1` and press Enter. The TDM Bus Configuration screen is displayed with the new value for Primary Clock Source.

```
              ..............................
               Modify TDM Bus Configuration
              ..............................
These are the current settings for the PRIMARY clock master.
The same settings will be used for all boards in the system.
-------------------------------------------------------------------
(read only)..........Clock Master: (Slot 0) DMV1200-4E1
(read only)......Bus Clock Rate: 8
-------------------------------------------------------------------
1) ......................TDM Bus Type: H100
2) .............Primary Clock Source: NETREF_1
3) ........Primary CT Bus Clock Line: CT_A
4) ......................NetRef Board: (Slot NA) NA
5) ........NetRef Clock Source Trunk: NA
6) ...........NetRef1 Fallback Board: (Slot NA) NA
7) ...NR1 Fallback Clock Source Trunk: NA
-------------------------------------------------------------------
Enter the number of the parameter to modify.
S to Save and return to the Main Screen, or
C to Cancel and return to the Main Screen.
```

5. Type the number `3` and press Enter to configure the Primary CT Bus Clock Line parameter. The Set Primary CT Bus Clock Line screen is then displayed.

```
.............................................................
.            Set Primary CT Bus Clock Line                  .
.............................................................
_____________________________________________________________
Specify whether the primary clock will use Line A or Line B.
Select one of the following:
_____________________________________________________________
1 - CT_A
2 - CT_B
_____________________________________________________________
Currently: CT_A
_____________________________________________________________
Enter a new value or press ? for help or
press Enter to return to the previous screen:
```

6. Type the number 1 or 2 that corresponds to the Primary CT Bus Clock Line you wish to select, then press Enter to return to the Modify TDM Bus Configuration screen. For example, to retain the default value of CT_A, just press Enter. The TDM Bus Configuration screen is then displayed with the unchanged value of CT_A as the Primary CT Bus Clock Line.

```
                  .............................
                   Modify TDM Bus Configuration
                  .............................
These are the current settings for the PRIMARY clock master.
The same settings will be used for all boards in the system.
-------------------------------------------------------------------
(read only)..........Clock Master: (Slot 0) DM/V1200-4E1
(read only)........Bus Clock Rate: 8
-------------------------------------------------------------------
1) .....................TDM Bus Type: H100
2) .............Primary Clock Source: NETREF_1
3) .........Primary CT Bus Clock Line: CT_A
4) ......................NetRef Board: (Slot NA) NA
5) .........NetRef Clock Source Trunk: NA
6) ...........NetRef1 Fallback Board: (Slot NA) NA
7) ...NR1 Fallback Clock Source Trunk: NA
-------------------------------------------------------------------
Enter the number of the parameter to modify.
S to Save and return to the Main Screen, or
C to Cancel and return to the Main Screen.
```

7. Type the number 4 and press Enter to configure the NetRef Board parameter. (The NetRef Board parameter is the Reference Master, that is, the board that provides the NetRef signal.) The Select NetRef Board screen is then displayed.

```
                  ......................................
                           Select NetRef Board
                  ......................................
______________________________________________________________
PCI     Logical      Clock           Model
Slot    ID           Status
0       1            PRIMARY        DM/V1200-4E1
1       2            SLAVE          DM/V960-4T1
6       3            SLAVE          DMV1200BTEP
13      14           SLAVE          DMN160TEC
______________________________________________________________
.  Select the Slot number of one of the boards shown to       .
.  provide the NetRef Signal. This will only be used if       .
.  the PRIMARY clock master uses a NETREF clock source.       .
______________________________________________________________
```

8. Type the slot number of the board that is to provide the reference signal to the CT Bus NetRef1 line.

   Note: This parameter is only applicable if you selected NetRef1 as the Primary Clock Source.

   For example, type 1 to select the board in slot 1. The Modify TDM Bus Configuration screen is displayed with the new value for the **NetRef Board** parameter.

```
                  .............................
                   Modify TDM Bus Configuration
                  .............................
These are the current TDM related settings for the system.
Set a clock master on the Modify Board page for that board.
-------------------------------------------------------------------
(read only)..........Clock Master: (Slot 0) DM/V1200-4E1
(read only)......Bus Clock Rate: 8
-------------------------------------------------------------------
1) .....................TDM Bus Type: H100
2) .............Primary Clock Source: NETREF_1
3) .........Primary CT Bus Clock Line: CT_A
4) ......................NetRef Board: (Slot 1) DM/V960-4T1
5) .........NetRef Clock Source Trunk: 1
```

```
6) ...........NetRef1 Fallback Board: (Slot NOT_SET) NOT_SET
7) ...NR1 Fallback Clock Source Trunk: NOT_SET
-------------------------------------------------------------------
Enter the number of the parameter to modify.
S to Save and return to the Main Screen, or
C to Cancel and return to the Main Screen.
```

9. Type the number 5 and press Enter to select the NetRef clock source trunk. The Set NetRef Clock Source screen is then displayed.

```
.........................................
.        Set NetRef Clock Source        .
.........................................
_________________________________________
Valid only if PRIMARY clock master is set
to NetRef and a NETREF board has been
selected.
Select a value in this range:
_________________________________________
1 - Trunk 1
through
4 - Trunk 4
_________________________________________
Enter a new value or press ? for help or
press Enter to return to the previous screen:
```

Note: For DMN160TEC and DMT160TEC boards, the allowable trunk range is from 1 to 16.

10. Type the number of the trunk that is to provide the clock source for the reference signal and press Enter. For example, to select trunk 2, type the number 2 and press Enter. The TDM Bus Configuration screen is then displayed with the new value for the **NetRef Clock Source Trunk** parameter.

```
              ..............................
               Modify TDM Bus Configuration
              ..............................
These are the current settings for the PRIMARY clock master.
The same settings will be used for all boards in the system.
-------------------------------------------------------------------
(read only)..........Clock Master: (Slot 0) DM/V1200-4E1
(read only)......Bus Clock Rate: 8
-------------------------------------------------------------------
1) ......................TDM Bus Type: H100
2) .............Primary Clock Source: NETREF_1
3) .........Primary CT Bus Clock Line: CT_A
4) ......................NetRef Board: (Slot 1) DM/V960-4T1
5) .........NetRef Clock Source Trunk: 2
6) ...........NetRef1 Fallback Board: (Slot NOT_SET) NOT_SET
7) ...NR1 Fallback Clock Source Trunk: NOT_SET
-------------------------------------------------------------------
Enter the number of the parameter to modify.
S to Save and return to the Main Screen, or
C to Cancel and return to the Main Screen.
```

11. Type the number 6 and press Enter to select a NetRef fallback board. The Select NetRef1 Fallback screen is displayed.

```
           .....................................
              Select NetRef1 Fallback Board
           .....................................
__________________________________________________________
PCI     Logical      Clock           Model
Slot    ID           Status
0       1            PRIMARY        DM/V1200-4E1
6       3            SLAVE          DMV1200BTEP
13      14           SLAVE          DMN160TEC
```

```
_____________________________________________________
.  Select the Slot number of one of the boards shown to  .
.  act as the NetRef Fallback. This will only be used if .
.  the PRIMARY clock master uses a NETREF clock source.  .
.  If no boards are shown it means there are none that   .
.  are appropriate to act as the Fallback board.         .
_____________________________________________________
```

12. Type the slot number of the board that is to provide the reference fallback signal to the CT Bus NetRef1 line.

    Note: This parameter is only applicable if you selected NetRef1 as the Primary Clock Source.

    For example, type `6` to select the board in slot 6. The TDM Bus Configuration screen is displayed with the new value for the **NetRef1Fallback Board** parameter.

```
       .............................
              Modify TDM Bus Configuration
              .............................
These are the current settings for the PRIMARY clock master.
The same settings will be used for all boards in the system.
-------------------------------------------------------------------
(read only)..........Clock Master: (Slot 0) DM/V1200-4E1
(read only)......Bus Clock Rate: 8
-------------------------------------------------------------------
1) ......................TDM Bus Type: H100
2) .............Primary Clock Source: NETREF_1
3) .........Primary CT Bus Clock Line: CT_A
4) ......................NetRef Board: (Slot 1) DM/V960-4T1
5) .........NetRef Clock Source Trunk: 2
6) ...........NetRef1 Fallback Board: (Slot 6) DMV1200BTEP
7) ...NR1 Fallback Clock Source Trunk: 1
-------------------------------------------------------------------
Enter the number of the parameter to modify.
S to Save and return to the Main Screen, or
C to Cancel and return to the Main Screen.
```

13. Type the number `7` and press Enter to select a NetRef fallback clock source trunk. The Set NetRef Fallback Clock Source screen is then displayed.

```
.........................................
.    Set NetRef Fallback Clock Source   .
.........................................
_________________________________________
Valid only if PRIMARY clock master is set
to NetRef and a NETREF Fallback board has
been selected.
Select a value in this range:
_________________________________________
1 - Trunk 1
through
4 - Trunk 4
_________________________________________
Enter a new value or press ? for help or
press Enter to return to the previous screen:
```

    Note: For DMN160TEC and DMT160TEC boards, the allowable trunk range is from 1 to 16.

14. Type the number of the trunk that is to provide the fallback clock source for the reference signal, then press Enter to return to the TDM Bus Configuration screen.

    For example, type `3` to select trunk 3, then press Enter. The TDM Bus Configuration screen is displayed with the new value for the **NR1Fallback Clock Source Trunk** parameter.

```
        ..............................
                Modify TDM Bus Configuration
               ..............................
These are the current settings for the PRIMARY clock master.
The same settings will be used for all boards in the system.
-------------------------------------------------------------------
(read only)..........Clock Master: (Slot 0) DM/V1200-4E1
(read only)......Bus Clock Rate: 8
-------------------------------------------------------------------
1) .....................TDM Bus Type: H100
2) .............Primary Clock Source: NETREF_1
3) ........Primary CT Bus Clock Line: CT_A
4) .....................NetRef Board: (Slot 1) DM/V960-4T1
5) ........NetRef Clock Source Trunk: 2
6) ...........NetRef1 Fallback Board: (Slot 6) DMV1200BTEP
7) ...NR1 Fallback Clock Source Trunk: 3
-------------------------------------------------------------------
Enter the number of the parameter to modify.
S to Save and return to the Main Screen, or
C to Cancel and return to the Main Screen.
```

15. When you're finished with the TDM Bus configuration, enter `S` to save and return to the Main screen (or `C` to cancel and return to the Main Screen). The current board configuration is displayed on the Main Screen again.

```
        .......................................
             DM3 Board Configuration - Main Screen
            .......................................
   This is a summary of the current DM3 board configuration
-------------------------------------------------------------
PCI    Board                    Logical      PCD
Slot  Enabled    Model            ID      File Name
0       Yes      DM/V1200-4E1     1       ml1_qs_net5.pcd
1       Yes      DM/V960-4T1      2       ml1_qs_net5.pcd
6       Yes      DMV1200BTEP      3       gul1_qsb_2_4ess_2_5ess.pcd
13      Yes      DMN160TEC        14      gdti16_4_4ess_4_5ess_4_ni2_4_qsige1.pcd
-------------------------------------------------------------
T        -     TDM BUS SETTINGS  -       H100
-------------------------------------------------------------
.     You must configure or disable each board shown.       .
.     When a board is configured a valid PCD file name      .
.     is displayed in the PCD File Name column.             .
-------------------------------------------------------------
Enter the slot number of the board to configure,
T to change TDM Bus Settings,S to save and quit,
or Q to quit without saving:
```

16. From the Main Screen, type `S` to save and quit the *DM3_cfg.sh* utility (or `Q` to quit without saving). Processing returns to the *config.sh* utility and proceeds as follows:
    - If SNMP agent software is installed, *config.sh* invokes the SNMP Agents Configuration Tool (*dlgcsnmpconf*). Refer to Section 4.5, "Using the SNMP Agents Configuration Tool (dlgcsnmpconf)", on page 65.
    - Otherwise, refer to Section 4.6, "Completing the Board Configuration Utility (config.sh)", on page 69.

4.5 Using the SNMP Agents Configuration Tool (dlgcsnmpconf)

If you installed SNMP agent software, once the DM3 Board Configuration utility (*DM3_cfg.sh*) is complete, the SNMP Agents Configuration Tool (*dlgcsnmpconf*) is automatically invoked. The following message is displayed:

```
SNMP configuration...
```

The procedure for configuring the SNMP agent software includes the following:

- Configuring the Community String
- Configuring Trap Destinations (Sinks)

4.5.1 Configuring the Community String

SNMP v1 uses community strings to provide simple access control for management information base (MIB) objects. If a management software tool uses an Intel® Dialogic SNMP MIB, it must use the identical community strings that the SNMP agent software is configured to use. Communities can be created in two ways:

- Configuring SNMP Communities Automatically
- Configuring SNMP Communities Manually

Both the automatic and manual configuration methods provide an opportunity to configure trap destinations.

Note: If the automatic configuration process is used, the *dialogic* community is created. This community grants external management stations **read-only** access to the Intel® Dialogic MIB. However, if the external management station requires **write** access to writable SNMP objects in the Intel® Dialogic MIB, then use the instructions in Section 4.5.1.2, "Configuring SNMP Communities Manually", on page 67 to create a community string that grants external managers read-write access.

4.5.1.1 Configuring SNMP Communities Automatically

The automatic configuration method creates the *admin* community, giving it both read and write access on the local host. The automatic method configures the Net-SNMP agent by creating the read-write *admin* community and the *dialogic* community, which is set to read-only for all external managers.

The SNMP agent software part of the configuration begins with the following messages:

```
SNMP configuration...
Dialogic SNMP Agents Configuration Tool
(C)2000-2001 Intel Corp.

You may choose to manually configure all communities and trap
sinks(destinations), or you may select an automatic configuration.  If the
automatic configuration is chosen, this tool will create the required
'admin' community and prompt you to enter trap sinks.  Selecting the manual
configuration allows you to easily create custom communities and configure
trap sinks.  If the 'admin' community does not exist yet, it may be created
the same way as other communities using the manual configuration.  Note, the
'admin' community MUST be assigned read-write priviledges or else abnormal
behavior will occur when the Intel Dialogic SNMP Agents are loaded.

Would you like to proceed with automatic configuration? (no will select
manual configuration) (y)es or (n)o?
```

Proceed as follows:

1. Type y for automatic configuration.

2. You are asked for confirmation; type the letter `y` again.

   The SNMP Agents Configuration Tool creates and configures the *admin* and *dialogic* communities.

   Configuration continues with the following prompt:

   ```
   Configure trap sink(destination) (y)es or (n)o?
   ```

Continue with the instructions in Section 4.5.2, “Configuring Trap Destinations (Sinks)”, on page 68.

4.5.1.2 Configuring SNMP Communities Manually

The manual method allows the user to enter communities; it does not create any communities automatically. If the manual configuration method is used, then the user is responsible for creating the *admin* community with read and write privileges. If the *admin* community already exists in the Net-SNMP configuration, the SNMP Agents Configuration Tool indicates that the community is detected and does not require configuration.

The SNMP agent software part of the configuration begins with the following messages:

```
SNMP configuration...
Dialogic SNMP Agents Configuration Tool
(C)2000-2001 Intel Corp.

You may choose to manually configure all communities and trap
sinks(destinations), or you may select an automatic configuration.  If the
automatic configuration is chosen, this tool will create the required
'admin' community and prompt you to enter trap sinks.  Selecting the manual
configuration allows you to easily create custom communities and configure
trap sinks.  If the 'admin' community does not exist yet, it may be created
the same way as other communities using the manual configuration.  Note, the
'admin' community MUST be assigned read-write priviledges or else abnormal
behavior will occur when the Intel Dialogic SNMP Agents are loaded.

Would you like to proceed with automatic configuration? (no will select
manual configuration) (y)es or (n)o?
```

Proceed as follows:

1. Type the letter `n` for manual configuration.

   The following prompt is displayed:

   ```
   Configure communities (access control)? (y)es or (n)o?
   ```

2. Type the letter `y` to create and configure communities. (Typing the letter `n` skips community configuration and proceeds to the trap destination configuration prompt shown in Step 7 of this procedure.)

   If you type the letter `y`, you are prompted for the community name:

   ```
   Enter community name (leave blank to cancel):
   ```

3. If the *admin* community has not been created yet, either manually or by the automatic configuration method, then type `admin` as the community name.

   You are prompted to enter the access privileges for the community:

```
Make this community read-write? (y)es or (n)o?
```

4. Type the letter `y` if the community will allow write requests, or n if the community will allow only read requests.

   Note: For the *admin* community, access **must** be read-write.

   The next prompt asks if external managers will be allowed to use this community to access the Intel® Dialogic MIB:

```
Allow external managers access with this community? (y)es or (n)o?
```

5. Type the letter `y` to grant access to the Intel® Dialogic MIB using this community, or the letter `n` to grant only the local host access to the MIB using this community.

   Note: For the *admin* community, local access only is recommended.

   The following prompt asks you to confirm the community configuration:

```
Prepared to add "rwcommunity yourcommunityname localhost" to config file.
Proceed (y)es or (n)o?
```

   where *yourcommmunityname* is the community name that you entered.

6. Type the letter `y` to write the community to the configuration file (*/usr/share/snmp/snmpd.conf*).

   The SNMP Agents Configuration Tool then allows you to configure additional communities:

```
Add another community? (y)es or (n)o?
```

7. Type the letter `y` to add another community or n to continue with trap destination configuration.

   If you type the letter `y`, the prompt shown in Step 2 is repeated, allowing you to configure another community.

   If you type the letter `n`, the following prompt is displayed:

```
Configure trap sink(destination) (y)es or (n)o?
```

Continue with the instructions in Section 4.5.2, "Configuring Trap Destinations (Sinks)", on page 68.

4.5.2 Configuring Trap Destinations (Sinks)

Trap destinations are machines that are configured to receive SNMP v1 traps from managed nodes. Trap destinations are also called **trap sinks**. The SNMP Agents Configuration Tool allows as many trap sinks as required by the user. If a trap sink is not reachable by the managed node, the configuration tool displays a warning message and allows the user to back out of the configuration.

After starting the SNMP Agents Configuration Tool and using either the automatic or manual method to configure communities, configuration continues with the following prompt:

```
Configure trap sink(destination) (y)es or (n)o?
```

Proceed as follows:

1. Type the letter `y` to configure a trap destination or the letter `n` to exit the configuration tool.

   If you type the letter `y`, the following prompt is displayed:

   ```
   Type host name to be trap sink:
   ```

2. Type the name of the management station that is configured to receive traps. The following prompt is displayed:

   ```
   Allow agent to send SNMPv1 traps to 'hostname' (y)es or (n)o?
   ```

   where *hostname* is the name of the management station that you entered.

3. Type the letter `y` to add the specified host as a trap destination.

   The prompts are repeated, allowing you to configure additional trap destination(s). When done, type n to exit the configuration tool and write the configuration changes to */usr/share/snmp/snmpd.conf*. A backup of the original configuration file is created as */usr/share/snmp/snmpd.conf.backup*.

For additional information about the SNMP agent software, see the *SNMP Agent Software for Linux Administration Guide*.

Proceed with Section 4.6, “Completing the Board Configuration Utility (config.sh)”, on page 69.

4.6 Completing the Board Configuration Utility (config.sh)

When *config.sh* is complete, the following messages are displayed:

```
Configuration is complete.

Before using the software, you must ensure that the Intel(R) Dialogic(R)
environment variables are set using one of the following actions:
         (a) Logout and login
         (b) At the shell prompt execute:     . /etc/profile.d/ct_intel.sh
         (c) Reboot system

The Intel(R) Dialogic(R) system services will automatically
start every time the system is rebooted.

NOTE: To start and stop system services manually, use the dlstop and
    dlstart scripts found in /usr/dialogic/bin
```

Once *config.sh* is complete, the SCD parameter settings should be verified and any additional manual changes to the file should be addressed.

1. To verify the configuration, check the SCD file that was generated (in *dialogic/cfg*). For SCD file formatting conventions, see Chapter 2, “SCD File Details”.

   For example:

```
NumStreams       : 4000             ; size of driver stream table
NumBindHandles   : 4000             ; size of driver bind table
[TDMBus 0] {
TDMBusType       : H110
    .
    . (other TDM bus parameters go here)
    .
}
[Board 1] {
PCDName     : ml1_qs_net5.pcd
FCDName     : ml1_qs_net5.fcd
   .
   . (other board parameters go here)
   .
}

[Board 2] {
PCDName     : ipvs_evr_2isdn_ntt_311c.pcd
FCDName     : ipvs_evr_2isdn_ntt_311c.fcd
   .
   . (other board parameters go here)
   .
   [NIC 1] {
      .
      . (NIC parameters go here)
      .
   }
   .
}
```

Continue with any additional configuration procedures that are applicable to your system:

- If you want to install and configure the HSK software, see Section 4.7, "Installing the HSK Software (hs_bld_kernel.sh)", on page 71 and Section 4.8, "Configuring the HSK Configuration Files", on page 72.
- If you want to pre-configure empty slots, see Section 4.9, "Pre-configuring Empty Slots for CompactPCI Boards", on page 74.
- If you are using a third-party board as the clock master, see Section 4.10, "Assigning Time Slots When Using a Third-Party Board as Clock Master", on page 75.
- If you wish to adjust the FSK transmit and receive signal levels, see Section 4.12, "Setting the FSK Transmit and Receive Signal Levels", on page 77.
- If you need to change parameter settings in an FCD file, see Section 4.13, "Modifying the FCD File by Editing the CONFIG File", on page 78.
- If you have SNMP agent software installed, see Section 4.14, "Transferring SNMP Agent Files to the Network Management Station", on page 79.
- If you have the Global Call Protocol Package installed, see Section 4.15, "Configuring Global Call CDP File", on page 79.

When you are satisfied with all configuration information, proceed with Section 4.16, "Initializing the System", on page 79.

4.7 Installing the HSK Software (hs_bld_kernel.sh)

The Hot Swap Kit (HSK) software installation and configuration is started by manually invoking the *hs_bld_kernel.sh* utility.

Note: Prior to starting the *hs_bld_kernel.sh* utility, if you want to keep a record of all configuration prompts and responses for later reference, use the Linux script utility. For information about using the script utility, see the Red Hat Linux documentation.

Proceed as follows:

1. You may optionally back up the existing Linux Red Hat kernel as shown in the following example:

   ```
   cd/usr/src
   tar cvzf linux-2.4.21-27.EL.tgz linux-2.4.21-27.EL
   ```

2. Go to the redistributable-runtime subdirectory of the Intel Dialogic software directory for this release and type the following to start the HSK software installation:

   ```
   ./hs_bld_kernel.sh
   ```

   The utility displays the following:

   ```
   Please select the number corresponding to the model of your chassis
   or abort this script by typing <Control-C>.
   1   zt5082c
   2   zt5083
   3   zt5084
   4   zt5085
   5   zt5087
   6   MPCHC5091

   Note:  The Hot Swap Kit is not supported on the zt5090 chassis.

   Selection ==>
   ```

Note: The only chassis/single board computer (SBC) combinations supported in this release are:

zt5085/zt5524

mpchc5091/zt5524

If you are using a different chassis/SBC combination from one of the above, the utility will display an error message and abort the installation.

3. Type the number (`4` or `6`) that corresponds to the Intel® NetStructure™ chassis you are using.

   The utility displays the following:

   ```
   Installing the Hot Swap Kit source RPM ... DONE

   Applying the HSK kernel source code patches ... DONE

   Now, the new kernel and module source files will be compiled
   and a new kernel linked. Depending upon the speed of your
   system, this process may take an hour or more.

   To compile and link the new kernel and modules, press <Enter>
   or abort the script by typing <Control-C>.
   ```

4. Press Enter to continue (or Ctrl-C to abort the script).

Note: Rebuilding the kernel may take an hour or more depending on your system.

The utility displays the following outputs from the rebuilding process:

```
Cleaning the kernel source tree using 'make mrproper' ... DONE

Generating a new kernel configuration
with Hot Swap Enabled ... DONE
---enabled

CPCI Hot Swap PICMG 2.1/2.12 Support (CONFIG_HA_PCI_HOT_SWAP)[N/y//?]
```

5. Type y and press Enter to continue.

```
Special Hot Swap Support (none,ZT5523-ZT5524,ADLINK_6860)[none]
```

6. Type ZT5523-ZT5524 and press Enter.

```
Generating dependencies using 'make dep'
...

The installation, build and configuration of the
Hot Swap Kit is complete.
You must reboot your system in order to run the new kernel and modules.
```

Note: You must update the configuration files before rebooting the system.

The utility completes by displaying the name of the boot manager entries for the HSK kernel.

For a Uniprocessor (UP) system, the following is displayed:

```
If you did not choose to boot the
Hot Swap Kit
kernel by default, you must manually select the appropriate boot entry
based upon the boot manager which is used by your system:

     Boot Manager      Menu Entry
     ------------      ----------
         grub          Red Hat Enterprise Linux AS (2.4.21-27.EL-hsk
         lilo          linux-2.4.21-27.EL-hsk
```

For a symmetric multiprocessor (SMP) system the following is displayed:

```
If you did not choose to boot the
Hot Swap Kit
kernel by default, you must manually select the appropriate boot entry
based upon the boot manager which is used by your system:

     Boot Manager      Menu Entry
     ------------      ----------
         grub          Red Hat Enterprise Linux AS (2.4.21-27.ELsmp-hsk
         lilo          linux-2.4.21-27.ELsmp-hsk
```

4.8 Configuring the HSK Configuration Files

After the HSK software installation has completed, you may need to edit certain HSK configuration files.

Proceed as follows:

1. For the mpchc5091/zt5524 combination you will need to edit the grub loader (/etc/grub.conf) as shown in bold below.

```
#grub.conf generated by anaconda
#
#Note that you do not have to rerun grub after making the changes to this file
#NOTICE: You have a /boot partition. This means that
#    all kernel and initrd paths are relative to /boot/, eg.
#    root (hd0,0)
#    kernel/vmlinuz-version ro root=/dev/hda2
#    initrd/initrd-version.img
#boot=/dev/hdad
default=1
timeout=10
splashimage=(hd0,0)/grub/splash.xmp.gz
title Red Hat Enterprise Linux AS (2.4.21-27.EL)
      root(hd0,0)
      kernel/vmlinuz-2.4.21-27.EL ro root=LABEL=/ hdc=ide-scsi
      initrd/initrd-2.4.21-27.EL.img
title Red Hat Enterprise Linux AS (2.4.21-27.EL-hsk)
      root(hd0,0)
      kernel/vmlinuz-2.4.21-27.EL-hsk ro root=LABEL=/ hdc=ide-scsi pcibr=(20&1):80:a:80:80
      initrd/initrd-2.4.21-27.EL-hsk.img
```

Note: If you are using the lilo boot manager (not supported in this release), you will need to update the */etc/lilo.conf* file for the HSK entry.

For the zt5085 chassis, enter the following:

```
image=/boot/vmlinuz-2.4.21-27.El[smp]-hsk
        label=linux
        initrd=/boot/initrd-2.4.21-27.El[smp]-hsk.img
        read-only
        append="root=LABEL=/ pcibr=(20&1):12:6:80:80,(40&1):12:8:80:80,(2040&1):10:6:60:60"
```

For the mpchc5091 chassis, enter the following:

```
image=/boot/vmlinuz-2.4.21-27.El[smp]-hsk
        label=linux
        initrd=/boot/initrd-2.4.21-27.El[smp]-hsk.img
        read-only
        append="root=LABEL=/ hdc=ide-scsi pcibr=(20&1):80:a:80:80"
```

After editing the *lilo.conf* file, use the following command to have the changes take effect:

```
/sbin/lilo
```

2. Determine if the */etc/hssd_slots.conf* file is the correct file for your chassis/SBC configuration.

   For a zt5085 chassis with a zt5524 SBC, use the default file.There is no need to change the file.

   For an mpchc5091 chassis, enter the following:

   ```
   cd/etc/
   cp hssd_slots.conf.mpchc5091ac-zt5524 hssd_slots.conf
   ```

   to replace the default file with the *hssd_slots.conf.mpchc5091ac-zt5524* file.

3. If you are using the mpchc5091/5524 configuration, you will need to edit the */sbin/hssd* file by uncommenting the appropriate entry for the mpchc5091 chassis by deleting the # and commenting all other entries as follows:

   ```
   #ZT5524 on ZT5085
   #insmod -f$HSSD_DRIVER_NAME alt=alt_zt554x,alt_cpv8540,alt_i21554 poll_time=500
   #parent_list="20&1","2040&1"
   #pcibr=(20&1):12:6:80:80,(40&1):12:8:80:80,(2040&1):10:6:60:60
   ```

```
#ZT5524 on MPCHC5091AC
#pcibr=(20&1):12:6:80:80
insmod -f$HSSD_DRIVER_NAME alt=alt_zt554x,alt_cpv8540,alt_i21554 poll_time=500
parent_list="20&1"
```

4. Reboot your system with the HSK kernel.

5. After rebooting, rerun config.sh to ensure that the Intel Dialogic driver is copied properly.

   If you do not want to reconfigure Intel NetStructure boards after the driver is copied, you can exit the config.sh utility.

Continue with any additional configuration procedures that are applicable to your system:

- If you want to pre-configure empty slots, see Section 4.9, "Pre-configuring Empty Slots for CompactPCI Boards", on page 74.
- If you are using a third-party board as the clock master, see Section 4.10, "Assigning Time Slots When Using a Third-Party Board as Clock Master", on page 75.
- If you need to change parameter settings in an FCD file, see Section 4.13, "Modifying the FCD File by Editing the CONFIG File", on page 78.
- If you have SNMP agent software installed, see Section 4.14, "Transferring SNMP Agent Files to the Network Management Station", on page 79

When you are satisfied with all configuration information, proceed with Section 4.16, "Initializing the System", on page 79.

4.9 Pre-configuring Empty Slots for CompactPCI Boards

Slots that do not currently have boards installed can be pre-configured for CompactPCPI boards that will be installed at some future time. By pre-configuring one or more vacant slots, a board or boards may be added after the system is running, without stopping the system (hot insertion).

Note: If, after preconfiguring an empty slot or slots, the system is stopped and then restarted without inserting a board in a preconfigured slot, the system will fail.

1. Using a text editor such as the *vi* editor, open the *pyramid.scd* file which is located in the */usr/dialogic/cfg* directory.

2. Configure the following parameters for each board that is to be added at a later time:
   - PCDName
   - FCDName
   - PCMEncoding
   - Slot Number ([Board *x*])
   - BusType
   - TimeToSendMessage
   - MasterStatus

> *Note:* You cannot pre-configure a board as a bus master.

For example, to pre-configure vacant slot 5 for a board that will use the *ml1_qs_4ess.pcd* file, you would add the following lines, shown in bold, to the existing *pyramid.scd* file:

```
[Board 5] {
PCDName: ml1_qs_4ess.pcd
FCDName: ml1_qs_4ess.fcd
PCMEncoding: MULAW
SlotNumber: 5
BusType: PCI
TimeToSendMsg: 50
MasterStatus: SLAVE

    .
    .
    .

}
```

3. Save and close the modified *pyramid.scd* file.

Continue with any additional configuration procedures that are applicable to your system:

- If you are using a third-party board as the clock master, see Section 4.10, "Assigning Time Slots When Using a Third-Party Board as Clock Master", on page 75.
- If you need to change parameter settings in an FCD file, see Section 4.13, "Modifying the FCD File by Editing the CONFIG File", on page 78.
- If you have SNMP agent software installed, see Section 4.14, "Transferring SNMP Agent Files to the Network Management Station", on page 79.

When you are satisfied with all configuration information, proceed with Section 4.16, "Initializing the System", on page 79.

4.10 Assigning Time Slots When Using a Third-Party Board as Clock Master

Third-party boards and Intel® NetStructure boards must not transmit data on the same telephony bus time slots. Also, the third-party technology (clock master) must execute before Intel® NetStructure boards (slaves) in the startup sequence. Transmit time slots for Intel® NetStructure boards are assigned during initialization as specified in the */usr/dialogic/cfg/.sctsbase* file. Use the following procedure to modify the time slot for Intel® NetStructure boards to be a value greater than 0 (a value greater than the number of time slots required for the third-party board); then, the third-party board can use time slots in the beginning of the time slot range.

Also, you will need to set the clocking daemon to PASSIVE, thereby setting all Intel® NetStructure boards to slaves and setting the clock daemon to not perform clock fallback. For information about setting the clocking daemon, see Section 4.11, "Setting the ClockDaemonMode", on page 76.

Notes: ***1.*** It is assumed that the third-party technology can use a range of time slots starting at 0.

2. The third-party board must be configured as both the primary clock master and the reference master on the TDM bus.

Proceed as follows:

1. From the command prompt, go to the */usr/dialogic/cfg* directory and locate the *.sctsbase* file.

2. Using a text editor (for example, *vi*), open the *.sctsbase* file.

3. Add a line to */usr/dialogic/cfg/.sctsbase* that will cause the starting time slot for Intel® NetStructure boards to be a value greater than 0. For example, if the third-party board uses time slots 0 through 1023 (1024 time slots), write the value “1024” to */usr/dialogic/cfg/.sctsbase* by adding this number as the first line in the file:

   ```
   1024
   ```

4. Save and close the */usr/dialogic/cfg/.sctsbase* file. Intel® NetStructure boards will now use time slots above those specified in the *.sctsbase* file.

Continue with Section 4.11, “Setting the ClockDaemonMode”, on page 76

4.11 Setting the ClockDaemonMode

The ClockDaemonMode specifies whether or not the clock daemon supplied with the system software performs clock fallback. The ClockDaemonMode is modified by editing the dlgsys.cfg file.

1. From the command prompt, go to the `/usr/dialogic/cfg` directory and locate the *dlgsys.cfg* file.
2. Using a text editor (for example, vi), open the *dlgsys.cfg* file. Initially, the last line of the file reads:

   ClockDaemonMode :ACTIVE

The ClockDaemonMode values supported are:

- ACTIVE [default]: Clocking daemon performs clock fallback. An Intel NetStructure board is selected as the clock master.
- PASSIVE: Clocking daemon does not perform clock fallback and sets all Intel NetStructure boards to slaves. Select PASSIVE when using a third-party board as a clock master.
- DISABLED: Clocking daemon does not control the TDM bus. Select DISABLED when using another application to control the TDM bus.

3. If you wish to change the ClockDaemonMode to a value other than the default value of ACTIVE, edit this line in the *dlgsys.cfg* file. For example, to change the mode to PASSIVE, you would edit the line to read:

   ClockDaemonMode :PASSIVE

4. Save and close the *dlgsys.cfg* file.

For further information about the clock daemon and clock fallback, see the *OA&M API for Linux Library Reference* and the *OA&M API for Linux Programming Guide*.

Continue with any additional configuration procedures that are applicable to your system:

- If you wish to adjust the FSK transmit and receive signal levels, see Section 4.12, “Setting the FSK Transmit and Receive Signal Levels”, on page 77.
- If you need to change parameter settings in an FCD file, see Section 4.13, “Modifying the FCD File by Editing the CONFIG File”, on page 78.
- If you have SNMP agent software installed, see Section 4.14, “Transferring SNMP Agent Files to the Network Management Station”, on page 79.
- If you have the Global Call Protocol Package installed, see Section 4.15, “Configuring Global Call CDP File”, on page 79.

When you are satisfied with all configuration information, proceed with Section 4.16, “Initializing the System”, on page 79.

4.12 Setting the FSK Transmit and Receive Signal Levels

Two-way Frequency Shift Keying (FSK) and ETSI FSK allow the exchange of small amounts of data between a telephone and the server using FSK as the transport layer. The two-way FSK functionality allows products to transmit and receive half-duplex FSK Bell 202 1200 bps data over the Public Switch Telephone Network (PSTN). ETSI FSK functionality is based on the specification ETSI 201 912.

The Transmit and Receive Signal Level parameters allow you to adjust the signal level of both the transmit and receive FSK signal levels.

FSK Transmit Signal Level

To set the signal level of the FSK transmit signal to other than the default value of -14 dbm, you will need to edit the applicable CONFIG file. For example, to set the FSK transmit signal level to a value of -20dbm, you need to add a new section [0x2a] at the end of the CONFIG file and include the FSK Transmit Signal Level parameter in that section as follows:

```
[0x2a]
SetParm=0x2a04,-20          !FM_ParmFSKTxSignalLevel
```

The value can range from -50 to -5dbm.

FSK Receive Signal Level

To set the signal level of the FSK receive signal to other than the default value of -46dbm, you also need to edit the CONFIG file by adding the FSK Receive Signal parameter to the new [0x2a] section. For example, to set the receive signal level to a value of -15dbm, add the line shown in bold to the new section you created for the FSK Transmit Signal Level parameter:

```
[0x2a]
SetParm=0x2a04,-20          !FM_ParmFSKTxSignalLevel
SetParm=0x2a00,-15          !FM_ParmFSKRxSignalLevel
```

The value can range from -60 to -5dbm.

After editing the CONFIG file, you will need to generate a new FCD file. Refer to Section 4.13, "Modifying the FCD File by Editing the CONFIG File", on page 78.

4.13 Modifying the FCD File by Editing the CONFIG File

If the default settings in the FCD files are not appropriate for your configuration, you can modify the settings in the FCD file by editing the CONFIG file. Modifications can be made at any time prior to starting the system. Once the CONFIG file parameters are modified and the CONFIG file is saved, the changes are automatically made to the FCD file after downloading.

Note: If you want to preserve the default parameter values contained in the CONFIG file, make a backup copy of the file prior to editing it.

To edit the CONFIG file:

1. From the command prompt, go to the */usr/dialogic/data* directory and locate the CONFIG file.

2. Using a text editor (for example, *vi*), open the CONFIG file that corresponds to the FCD file you want to modify. By default, the CONFIG file will have the same file name as the FCD file, but with a *.config* extension.

3. Edit the CONFIG file as necessary.

   For a detailed description of the CONFIG file sections and formatting conventions, see Chapter 3, "CONFIG File Details". For details about CONFIG file parameters, see Chapter 6, "CONFIG File Parameter Reference".

4. Save and close the CONFIG file.

Note: The modified FCD file is automatically created when the PCD file and modified CONFIG file are downloaded to the board.

Continue with any additional configuration procedures that are applicable to your system:

- If you have SNMP agent software installed, see Section 4.14, "Transferring SNMP Agent Files to the Network Management Station", on page 79.
- If you have the Global Call Protocol Package installed, see Section 4.15, "Configuring Global Call CDP File", on page 79.
- Otherwise, proceed with Section 4.16, "Initializing the System", on page 79.

4.14 Transferring SNMP Agent Files to the Network Management Station

The SNMP agent MIB files must be installed on the network management station. In addition, the *SNMP Agent Software Administrative Guide* should be made available at the network management station. The files can be sent from the managed node to the network management station using any method you are familiar with, such as e-mail or FTP.

1. Copy the SNMP agent MIBs to a directory accessible to your network management application. The MIBs are stored in */usr/dialogic/cfg* when the SNMP agent software is installed on the managed node. For more information, consult the network management application installation instructions.

2. Copy the *SNMP Agent Software Administrative Guide* to the network management station. If the Intel® Dialogic documentation was installed on the managed node, documentation files (*.html* and *.pdf* versions) are stored in */usr/dialogic/docs*.

Continue with any additional configuration procedures that are applicable to your system:

- If you have the Global Call Protocol Package installed, see Section 4.15, “Configuring Global Call CDP File”, on page 79.
- Otherwise, proceed with Section 4.16, “Initializing the System”, on page 79.

4.15 Configuring Global Call CDP File

If you are using the Global Call Protocol Package, the following configuration procedures are applicable:

- Configuring the country dependent parameters (CDP) file
- Downloading the protocol and CDP file

The protocol package is included with the system software or can be found at:

http://resource.intel.com/telecom/support/releases/protocols/index.htm

For detailed procedural information, see the *Global Call Country Dependent Parameters (CDP) Configuration Guide*.

4.16 Initializing the System

The new configuration settings will not take effect until the system is rebooted. Rebooting the system initializes all the Intel® NetStructure products in the system and also starts the SNMP agent software. Before rebooting, make sure you perform all of the necessary configuration procedures.

To initialize the system for the first time, proceed as follows:

1. Shut down the system and restart it. Rebooting the system initializes all the Intel® NetStructure products in the system.

2. Upon startup, check the screen or system log file for startup messages. You should see:

   ```
   Parsing SCD file /usr/dialogic/cfg/pyramid.scd succeeded
   ```

   followed by messages for individual boards. For each board, look for a message that says:

   ```
   Configuring and downloading board succeeded boardNumber=n
   ```

   followed by a list of boards that were detected and then:

   ```
   n Dialogic Boards Successfully Installed
   ```

3. Once the system is started, the startup script will exit. To verify that the startup script has completed, enter the command:

   ```
   ps -ef | grep S90ct_intel | grep -v grep
   ```

   If you can enter this command, the startup script has completed.

4. To display information about boards that are present in the system and recognized by the device driver, enter the command:

   ```
   listboards -b0
   ```

   The Listboards utility displays information about each board. For more information about the Listboards utility, see the *System Release Administration Guide*.

5. After starting the system for the first time, you may want to use some of the demos provided by the system software to verify that your system is operating properly. See the *Intel Dialogic System Release 6.1 for Linux Release Notes* for information about the demos included in this release.

If you have problems, see the Troubleshooting section of the *System Release Administration Guide*. Problems on initial startup are typically caused by errors in your configuration settings.

Once the system is initialized for the first time, you do not need to reboot the system to implement additional configuration changes. The system can be re-initialized using the dlstart and dlstop utilities. For a detailed procedure, refer to Section 4.17, “Reconfiguring the System”, on page 80.

4.17 Reconfiguring the System

Once the system is initialized for the first time, the system must be stopped and restarted in order to make any additional configuration changes. You only have to reboot the system for the initial system startup. To restart (re-initialize) the system, you stop and then restart the system using the dlstop and dlstart utilities.

1. Before you stop the system, the application must be stopped and the application must ensure that all channels have been closed.

2. To stop the system, enter the command:

    ```
    /usr/dialogic/bin/dlstop
    ```

    The messages displayed depend on the boards installed and may include the following:

    ```
    [TEELOGGER]: Starting logging of admin script
    Shutting down Telephony Fault Detectors
    Stopping DM3 Boards
    REGVOX: Deleting DM3 Devices ...
    REGVOX: Delete DM3 Devices Done.
    Shutting down hot swap monitor
    Stopping timeslot doler
    Shutting down DeviceMapper Server
    [TEELOGGER]: Finishing logging of admin script
    Shutting down CORBA Name Server
    Shutting down CORBA Event Server
    ```

3. Modify SCD file parameters, if necessary.

    3a. From the command prompt, go to the */usr/dialogic/data* directory to locate the SCD file.

    3b. Using a text editor (for example, *vi*), open the file.

    3c. Edit the parameters. For details about the SCD file including formatting conventions, refer to Chapter 2, "SCD File Details". For details about each parameter, refer to Chapter 5, "SCD File Parameter Reference".

    3d. Save and close the SCD file.

    Note: An alternative procedure is to rerun the *DM3_cfg.sh* utility. If the utility is rerun, a backup copy of the existing SCD file is created. A new SCD file is created and all previously configured parameter settings are removed from the file and replaced by the system default values. The file name of the backup SCD file is in the format:

    pyramid.scd_<year><day of year><hour><minute>

    For example, a copy saved on June 25 (176th day of the year), 2004 at 4:58 p.m. would be named *pyramid.scd_20041761658*.

4. Modify CONFIG/FCD file parameters, if necessary. For a detailed procedure, refer to Section 4.13, "Modifying the FCD File by Editing the CONFIG File", on page 78.

    For details about the CONFIG file including formatting conventions and parameter sections, refer to Chapter 3, "CONFIG File Details". For details about each parameter, refer to Chapter 6, "CONFIG File Parameter Reference".

5. Reconfigure SNMP agent software, if applicable.

    5a. Run the SNMP Agents Configuration Tool (*dlgcsnmpconf*), located in */usr/dialogic/lib/snmp*, to reconfigure SNMP communities and trap destinations. For a detailed procedure, see Section 4.5, "Using the SNMP Agents Configuration Tool (dlgcsnmpconf)", on page 65.

    5b. Once *dlgcsnmpconf* is re-run, restart the SNMP agent and apply changes by typing:

    ```
    /etc/init.d/dlgcsnmpd restart
    ```

6. To start the system, enter the command:

    ```
    /usr/dialogic/bin/dlstart
    ```

    Startup should only be performed when the system is stopped, that is, after a dlstop command.

    The messages displayed depend on the products installed and may include the following:

    ```
    Starting ORB Event Server:
    Starting Orbacus4 nameserv:
    Starting error logger
    [TEELOGGER]: Starting logging of admin script
    Starting DeviceMapper
    [TBPARMS]:Initializing System Variables for TDMBus
    [TBPARMS]:Base timeslot: 0
    [TBPARMS]:TDMBus Variables initialized
    Starting timeslot doler
    Starting DM3 Boards
    DM3 driver already loaded
    Creating device /dev/mercd
    Starting DM3 download phase
    downloader Version 2.47 Prod 0.02 Build: 00
    Copyright (c) Intel Corporation 1997-2002
      (Using: Host Library version: 6.10  Build: 1)
    Using data files from /usr/dialogic/data
    ```

    followed by screens of hardware configuration information and verification, and completing with the following:

    ```
    REGVOX: Adding DM3 Devices ...
    REGVOX: Adding DM3 Devices Done
    CHEETAHSTART: Removing temp files.
    CHEETAHSTART: Building shared memory.
    CHEETAHSTART: Done.
    Starting fault detection services
    Starting clocking daemon
    [TEELOGGER]: Finishing logging of admin script
    ```

For detailed procedures about other administrative tasks, see the *System Release for Linux Administration Guide* and the *SNMP Agent Software for Linux Administration Guide*.

SCD File Parameter Reference 5

This chapter provides the details about each SCD file parameter. This include a parameter description, range of values, and guidelines. The SCD parameters are divided into the following categories:

5.1 DM3 Board Configuration Utility (DM3_cfg.sh) Parameters

This section includes all the SCD file parameters identified in the DM3 Board Configuration utility (*DM3_cfg.sh*). The DM3 Board Configuration parameters are divided into the following categories:

- Board Settings Parameters
- TDM Bus Configuration

Notes: ***1.*** Parameter names are identified differently in the DM3 Board Configuration utility as opposed to how they are identified in the SCD file. In this section, parameter names are documented in the format:

DM3 Board Configuration Parameter Name (SCDFileParameterName)

2. There are some parameters that exist within the DM3 Board Configuration utility, but are not named directly in the SCD file. For these parameters, only the DM3 Board Configuration utility parameter name is used. For example:

- **Logical Id** exists in the DM3 Board Configuration utility and is identified in the SCD file as [board #].
- **Model** and **Board Enabled** only exist in the DM3 Board Configuration utility.

5.1.1 Board Settings Parameters

This section includes all the SCD file parameters viewable through the Modify Board Settings screen. These parameters (except for Model) are in each board section of the SCD file. Additional board section parameters are set through the Modify TDM Bus Configuration screen.

The Modify Board Settings parameters include:

- PCI Slot (SlotNumber)
- Model
- Logical ID
- Board Enabled

- PCD File Name (PCDName)
- FCD File Name (FCDName)
- PCM Encoding (PCMEncoding)
- TDM Clock Function (MasterStatus)

PCI Slot (SlotNumber)

Description: The **PCI Slot** (**SlotNumber**) parameter defines the physical location of the PCI or Compact PCI board in the chassis.

Note: This parameter is viewable (read-only) in the DM3 Board Configuration (*DM3_cfg.sh*) utility, but is not modifiable. The parameter is viewable from both the Modify Board Settings and Modify Network Interface Connector Configuration screens.

Values: The default value for this parameter is the value returned by the Listboards utility.

Guidelines: The *DM3_cfg.sh* utility assigns this value and it cannot be changed from the DM3 Board Configuration screens. If the board is physically moved to a new slot, or if the board's device ID selector is changed, manually edit the SCD file to reflect the change. Otherwise, do **not** change the default value of this parameter. For more information about the Listboards utility, refer to the *System Release for Linux Administration Guide*.

Model

Description: The **Model** parameter defines the model number of the board selected. It exists within the DM3 Board Configuration (*DM3_cfg.sh*) utility to allow you to easily identify the boards in a multi-board system.

Note: This parameter does not exist in the SCD file. It exists only in the DM3 Board Configuration utility and is viewable (read-only), but is not modifiable. It is viewable from both the Modify Board Settings and Modify Network Interface Connector Configuration screens.

Values: Valid model number of a board as identified by the system.

Guidelines: The default value **cannot** be changed.

Logical ID

Description: The **Logical ID** parameter is a user-assigned ID used by the drivers to identify a board. Within the SCD file, the **Logical ID** is assigned as part of the section head for a particular board. The *DM3_cfg.sh* utility assigns a default value of **PCI Slot** + 1.

Note: **Logical Id** exists in the DM3 Board Configuration utility and is identified in the SCD file as [board #].

Values: 1 to 16 (unique number for each board in the system)

Guidelines: **Logical ID** helps you keep track of each board in the system. It is frequently easiest to assign **Logical ID** to a value that matchs the physical location of the board, although this is not a requirement. Giving the first board in the system a **Logical ID** of 1, and sequentially numbering the rest from left to right, makes it easy to identify the board without having to check the configuration file.

Note: If you want to change the board's Logical ID after configuring the system for the first time, you must stop the system and unload the drivers prior to implementing the change.

Since some chassis have slot numbers beginning with 0 rather than 1, and a **Logical ID** of 0 is not a valid value, the default value for **Logical ID** is the value of **PCI Slot** + 1. So if the default **Logical ID** for your boards begin with 2 and you would prefer 1, change the default values.

Board Enabled

Description: The **Board Enabled** parameter specifies whether the system should download firmware to activate the board when the system is started. This parameter is not in the SCD file. It exists within the *DM3_cfg.sh* configuration utility to allow you to easily create an SCD that will only download a few boards in a multi-board system. If you disable a board, its parameters are written as comment lines in the SCD file.

Values:

- Yes [default]: Enable the board
- No: Disable the board

Guidelines: Enter Yes if you want to download this board. Enter No if you want to temporarily suspend the use of a board in your system. Entering No will comment out the board parameters in the SCD file.

PCD File Name (PCDName)

Description: The **PCD File Name** (**PCDName**) parameter specifies the name of the product configuration description (PCD) file. The PCD file lists object files and maps them to specific processors, configures the kernel for each processor, and sets the number of component instances to run on each processor. The default value is NOT_SET.

For the DMN160TEC and DMT160TEC boards, a composite PCD file can be created and assigned to the board. The composite PCD file is a customized combination of loads (PCD files) based on the line type (T-1 or E-1) and protocol required for each trunk. A different line-type/protocol combination can be used for each trunk on the board.

Note: The *DM3_cfg.sh* utility uses the value of the **PCD File Name** (**PCDName**) parameter to set the value of the **FCD File Name** (**FCDName**) parameter.

Values:

- A valid PCD file name (all boards)
- A valid composite PCD file name (DMN160TEC boards)

Values: A valid PCD file name

Note: You **cannot** use the default value of NOT_SET for this parameter. The value of the **PCD File Name** (**PCDName**) parameter must be changed to a valid file name.

Guidelines: For a DMN160TEC or DMT160TEC board, if a single line type and protocol will be used on all board trunks, select one of the PCD files supplied with the system release. A list of applicable files is displayed on the Specify a PCD File screen.

For a DMN160TEC or DMT160TEC board, if different line types and protocols are required for each trunk or group of trunks on the board, you will need to create a PCD file. For details about creating a PCD file, see Section 4.4.1, “Configuring Board Settings”, on page 50.

For all other boards, select the PCD file according to the feature set (media load) and protocol that this board will use. For example, select *ml1_qs_net5.pcd* for media load 1 features and NET5 protocol.

Guidelines: Select the PCD file according to the feature set (media load) and protocol that this board will use. For example, select *ml1_qs_net5.pcd* for media load 1 features and NET5 protocol.

FCD File Name (FCDName)

Description: The **FCD File Name** (**FCDName**) parameter specifies the name of the feature configuration description (FCD) file. The FCD file adjusts the settings of the components that make up each product. For example, an FCD file may contain instructions to set certain country-specific codes or configure network interface protocols.The default value is the same file name as the PCD file name, but with an *.fcd* extension.

Values: A valid FCD file name

Guidelines: When using the *DM3_cfg.sh* utility, the **PCD File Name** (**PCDName**) parameter must be set first, because this determines the default FCD file name. The **FCD File Name** parameter will default to a file with the same name as the **PCD File Name** parameter, but with a *.fcd* extension. The default value is appropriate for most configurations, although there are cases where different FCD files can be used for the same PCD file. If you do not want to use the default, you have to change the FCD file name by manually editing the SCD file.

For information about manually editing the SCD file, see Section 2.1, “SCD File Formatting Conventions”, on page 15.

PCM Encoding (PCMEncoding)

Description: The **PCM Encoding** parameter specifies the pulse code modulation (PCM) encoding method as either ALAW or MULAW.

Note: NOT_SET is the default if the appropriate setting cannot be determined (for example, for resource-only boards), and this **must** be changed to either ALAW or MULAW.

Values:

- ALAW (default for E-1 boards)
- MULAW (default for T-1 boards)

Guidelines: The default value is set to whatever is appropriate for the board and is based on the board’s network interface: ALAW for E-1 boards or MULAW for T-1 boards. All boards connected via a telephony bus cable (CT Bus) must use the same encoding method. Some boards (such as resource-only boards) have no network interface and some boards are capable of multiple encoding methods; in these cases the default value is NOT_SET, and you must make sure that you select the correct value.

Note: For DM/V-B boards (excluding DM/V-B resource-only boards), DMN160TEC, and DMT160TEC boards, you can set their network interfaces set to either T-1 or E-1 in the same system, regardless of the CT Bus PCM encoding method. (Network interfaces are configured based on the media load specified for downloading that board.

For example, if the PCM encoding method on the CT Bus is set to A-law, a DM/V-B board that has some or all of its network (front end) interfaces configured for T-1, will automatically convert the A-law data sent to and received from the CT Bus to Mu-law for transmitting and receiving on the T-1 configured front ends. The board will always transmit to and receive from each front end using the PCM encoding method determined by the network interface setting.

For all other boards, the CT Bus PCM encoding method selected by the user must match the PCM Encoding of the network interface or the board will not start properly.

Note: HDSI boards use country-specific PCD and FCD files. Depending on the PCD/FCD files selected for an HDSI board, the PCM encoding method will be set to either A-law or Mu-law, based on the default value for that country. If this value is not the same as the TDM bus value, the HDSI board will fail to download.

TDM Clock Function (MasterStatus)

Description: The **TDM Clock Function** (**MasterStatus**) parameter specifies whether a board is the Primary Clock Master, Secondary Clock Master, or a slave. A clock master is one of the boards in a system that is designated to provide reference timing for all boards attached to the bus. This board must derive timing from a network reference (Reference Master) which ultimately derives clock from a T-1 or E-1 line (for example, NetRef1), or else must derive timing from a digital network interface or, as a last alternative, from its own internal oscillator.

The CT Bus has two types of clock masters: **Primary** Clock Master and **Secondary** Clock Master. The Secondary Clock Master becomes the clock master if the Primary Clock Master fails or is removed from the system.

Note: This parameter is ignored when performing a single board stop and start (stopbrd and startbrd utilities) and takes effect only when `dlstart` starts the System Service. For information about the stopbrd and startbrd utilities, refer to the *System Release for Linux Administration Guide*.

Values:

- PRIMARY: Only one board can be assigned PRIMARY clock master.
- SECONDARY: Only one board can be assigned SECONDARY clock master. This value is valid only in CT Bus applications.
- SLAVE [default]: All other boards are slaves (that is, all boards that are neither PRIMARY or SECONDARY clock masters).

Guidelines: Since the default value for the **TDM Clock Function** (**MasterStatus**) parameter is SLAVE when using the *DM3_cfg.sh* utility, you must change the default for boards that are either the Primary Clock Master or the Secondary Clock Master. Only one board can be the Primary Clock Master, and one board can be the Secondary Clock Master.

Note: The configuration in the SCD file is the preferred configuration during system startup. During single board start (`stopbrd` and `startbrd`) the setting for the **TDM Clock Function** (**MasterStatus**) parameter is ignored since the state of the TDM bus may have changed due to fallback occurring. The clocking daemon will add the board based on the current bus configuration (most likely as SLAVE). If the system is re-started (`dlstop` and `dlstart`) with two PRIMARY's in the SCD file, then an error will occur since two clock masters are defined.

H.100 boards require certain signals that can only be provided by another H.100 board and routed over the H.100 cable. Make sure that the board that you intend to use as the clock master can do the following:

- Derive timing from a network reference or directly from a digital network interface such as a T-1 or E-1 line
- Provide both H.100 core signals and compatible bus signals

For additional reference information about the CT Bus, see Section 2.5, “CT Bus Clock Fallback”, on page 27.

If you want to use a third-party board as the clock master, instructions are given in Section 4.10, “Assigning Time Slots When Using a Third-Party Board as Clock Master”, on page 75.

5.1.2 TDM Bus Configuration

This section includes all the SCD file TDM bus parameters. Some of these parameters are in the TDMBus section of the file and some are in board sections of the file. Most of the parameters are viewable through the TDM Bus Configuration screen. These parameters include:

- Bus Clock Rate (BusCR)
- TDM Bus Type (TDMBusType)
- Group1CR
- Group2CR
- Group3CR
- Group4CR
- Primary Clock Source (DeriveClockFrom)
- Primary CT Bus Clock Line (PrimaryLines)
- NetRef Board (NetRef1)
- NetRef Clock Source Trunk (NetRef1From)
- NetRef1 Fallback Board (NetRef1Fallback)
- NR1 Fallback Clock Source Trunk (NetRef1FromFallback)
- NetRef1CR

Bus Clock Rate (BusCR)

Description: The **BusClock Rate (BusCR)** parameter specifies the TDM bus clock rate. When **TDMBusType** is set to H100 or H110 (CT Bus), the default value is 8.

Values:

- 8 (MHz) [default for CT Bus]
- 4 (MHz) [default for SCbus] (SCbus is not supported.)

Guidelines: Do **not** change the default value. This parameter is viewable (read-only) in the *DM3_cfg.sh* utility, but is accessible by manually editing the SCD file.

TDM Bus Type (TDMBusType)

Description: The **TDM Bus Type** parameter specifies the bus mode for the TDM bus as either H100 or H110.

Values:

- H100 [default]: CT Bus on PCI
- H110: CT Bus on CompactPCI
- SCBUS: SCbus on PCI (This value is not supported.)

Guidelines: Set this parameter to H110 for CompactPCI boards. Set this parameter to H100 for PCI boards.

Group1CR

Description: The **Group1CR** parameter specifies the clock rates for Group 1.

Values: 8 Mhz [default]

Guidelines: Do **not** change the default value. This parameter is not viewable in the *DM3_cfg.sh* utility. It is only viewable by manually editing the SCD file.

Group2CR

Description: The **Group2CR** parameter specifies the clock rates for Group 2.

Values: 8 MHz [default]

Guidelines: Do **not** change the default value. This parameter is not viewable in the *DM3_cfg.sh* utility. It is only viewable by manually editing the SCD file.

Group3CR

Description: The **Group3CR** parameter specifies the clock rates for Group 3.

Values: 8 MHz [default]

Guidelines: Do **not** change the default value. This parameter is not viewable in the *DM3_cfg.sh* utility. It is only viewable by manually editing the SCD file.

Group4CR

Description: The **Group4CR** parameter specifies the clock rates for Group 4.

Values: 8 MHz [default]

Guidelines: Do **not** change the default value. This parameter is not viewable in the *DM3_cfg.sh* utility. It is only viewable by manually editing the SCD file.

Primary Clock Source (DeriveClockFrom)

Description: The **Primary Clock Source** (**DeriveClockFrom**) parameter specifies the clock source used to drive the Primary or Secondary Clock Master.

Values:

- OSC [default]: Derive clocking from the board's internal oscillator
- 1 to 16: Derive clocking from specified trunk 1 to 16 (DMN160TEC and DMT160TEC only)
- 1 to 4: Derive clocking from specified trunk 1 to 4 (all other DM3 hardware)
- N1: Derive clocking from NetRef1 (Preferred clock source)
- N2: Derive clocking from NetRef2 (This value is not supported.)

Guidelines: This parameter is in the board section of the SCD file and only has to be set for the board that is the Primary Clock Master. If a Secondary Clock Master is configured, the same clock source will also be assigned to the Secondary Clock Master. The value NA (not applicable) will be displayed for all other boards.

The default clock source is the internal oscillator of the Primary Clock Master board. However, clocking should be derived from a network reference or directly from a digital network interface if available, not from a board's internal oscillator. The internal oscillator should be used as the clock source only for internal testing purposes.

Note: When configuring a board with front-end capability as resource only, the system will not detect this and might select this board as a clock master. In this event, you must manually configure another board in the system as the clock master.

Note: When the **NetRef Board (NetRef1)** parameter specifies the same board as that chosen to be the Primary Clock Master, do not select a local trunk (Trunk 1 - 4) on this board as the Primary Clock Source. Instead, select N1 - NetRef 1 as the source for driving the Primary Clock Master.

Primary CT Bus Clock Line (PrimaryLines)

Description: The **Primary CT Bus Clock Line** (**PrimaryLines**) parameter specifies whether the Primary Line is Line A or Line B. Line A and Line B are the two lines that the CT Bus sets aside for clock synchronization. Either Line A or Line B can be assigned as the Primary Line; the remaining line is assigned as the Secondary Line. The Primary Line is driven by the Primary Clock Master, and the Secondary Line is driven by the Secondary Clock Master.

Values:

- CT_A [default]: Line A
- CT_B: Line B

Guidelines: This parameter is in the TDMBus section of the SCD file and is assigned to the Primary Clock Master.

NetRef Board (NetRef1)

Description: The **NetRef Board** (**NetRef1**) parameter specifies the board providing the NETREF signal.

Values: A valid **PCI Slot** (**SlotNumber**) number

Guidelines: This parameter is only set if the Primary Clock Master uses a NETREF clock source, that is, if the **Primary Clock Source** parameter is set to N1 (NETREF1). In addition, this parameter (**NetRef1**) is included in the board section of the SCD file for the board that provides the NETREF signal.

Note: Although the DM3 Board Configuration screen displays the slot number and model number of the NETREF board, the SCD file identifies the board by setting the **NetRef1** parameter to a value of Yes.

NetRef Clock Source Trunk (NetRef1From)

Description: The **NetRef Clock Source Trunk** (**NetRef1From**) parameter specifies the network interface source of the NETREF signal.

Values:

- 1 to 16: DMN160TEC and DMT160TEC
- 1 to 4: All other DM3 hardware

Values: 1 to 4

Guidelines: This parameter is only set if the Primary Clock Master uses a NETREF clock source, that is, if the **Primary Clock Source** parameter is set to N1 (NETREF1). In addition, this parameter is included in the board section of the SCD file for the board that provides the NETREF signal.

NetRef1 Fallback Board (NetRef1Fallback)

Description: The **NetRef1 Fallback Board** (**NetRef1Fallback**) parameter specifies the fallback board for the NETREF signal. The fallback board provides the NETREF signal only if the NetRef board (**NetRef1**) fails.

Values: A valid **PCI Slot** (**SlotNumber**) number

Guidelines: This parameter is only set if the Primary Clock Master uses a NETREF clock source, that is, if the **Primary Clock Source** parameter is set to N1 (NETREF1). In addition, this parameter (**NetRef1Fallback**) is included in the board section of the SCD file for the board that provides the fallback NETREF signal.

Note: Although the DM3 Board Configuration screen displays the slot number and model number of the NETREF board, the SCD file identifies the board by setting the **NetRef1Fallback** parameter to a value of Yes.

NR1 Fallback Clock Source Trunk (NetRef1FromFallback)

Description: The **NR1 Fallback Clock Source Trunk** (**NetRef1FromFallback**) parameter specifies the fallback network interface source of the NETREF signal. The fallback board (trunk) provides the NETREF signal only if the NetRef board (**NetRef1**) fails.

Values: 1 to 4

Values:

- 1 to 16: DMN160TEC and DMT160TEC
- 1 to 4: All other DM3 hardware

Guidelines: This parameter is only set if the primary clock master uses a NETREF clock source, that is, if the **Primary Clock Source** parameter is set to N1 (NETREF1). In addition, this parameter (**NetRef1FromFallback**) is included in the board section of the SCD file for the board that provides the fallback NETREF signal.

NetRef1CR

Description: The **NetRef1CR** parameter specifies the rate of the outbound speed at which the NETREF should be generated on the NETREF bus line. This parameter is required only for the board that has **NetRef1** set to Yes. For all other boards, this parameter is omitted from the SCD file.

Note: This parameter is not modifiable through the DM3 Board Configuration utility (*DM3_cfg.sh*).

Description:

Values:

- 1[default]: 8 kHz (this is the only value that is currently supported)
- 2: 1.536 MHz (not currently supported)
- 3: 1.544 MHz (not currently supported)
- 4: 2.048 MHz (not currently supported)

Guidelines: Do **not** change the default value.This parameter is included in the board section of the SCD file for the board that provides the NETREF signal.

5.2 Additional Parameters in the SCD File

The following parameters cannot be edited using the DM3 Board Configuration screens. The *DM3_cfg.sh* utility automatically includes these parameters in the SCD file. Except for the **LogFile** parameter, the default values for these parameters should not be changed by the user.

- NumStreams
- NumBindHandles
- BusType
- DisplayConfig
- LogFile

- TimeToSendMsg

NumStreams

Description: The **NumStreams** parameter defines the number of streams supported by the driver. The driver supports a maximum of 1000 streams per board up to four boards. The default value is 4000.

Values: 0 to 4000

Guidelines: Do **not** change the default value. An incorrect settings can cause download or run-time problems. This driver parameter appears at the beginning of the SCD file (not in a section) before the board-level parameters.

NumBindHandles

Description: The **NumBindHandles** parameter defines the number of message queue handles supported by the driver. The default value is 4000.

Values: 0 to 4000

Guidelines: Do **not** change the default value. An incorrect setting can cause download or run-time problems. This driver parameter appears at the beginning of the SCD file (not in a section) before the board-level parameters.

BusType

Description: The **BusType** parameter specifies the board's bus type, that is, the system bus that the board is plugged into in the system.

Note: The default value of PCI is used for CompactPCI boards.

Values: PCI [default]

Guidelines: Do **not** change the default value. This parameter appears in each board section of the SCD file.

DisplayConfig

Description: The **DisplayConfig** parameter indicates whether or not configuration information is displayed during download.

Values: Yes [default]

Guidelines: Do **not** change the default value. This parameter appears in each board section of the SCD file.

LogFile

Description: The **LogFile** parameter defines the name of the log file created for each Intel® Dialogic board (for example, *board1.log*).

Values: board<*logical ID*>.log [default]

Guidelines: This parameter appears in each board section of the SCD file. Omit this parameter to disable logging.

TimeToSendMsg

Description: The **TimeToSendMsg** parameter specifies the minimum time to send a message.

Values: 50 milliseconds [default]

Guidelines: Do **not** change the default value. This parameter appears in each board section of the SCD file.

CONFIG File Parameter Reference 6

This chapter lists and describes the parameters contained in the CONFIG files. Parameters are listed in the same order as they appear in the CONFIG files and they are grouped according to the CONFIG file sections. Within the CONFIG files, the parameters are grouped in the following sections.

Note: Not all parameters are included in each CONFIG file, as this depends on the board supported by that particular file. CONFIG file parameters that **should not be modified** by the user are omitted from this document. Exceptions are made for parameters that, although they should not be modified by the user, are needed in understanding a particular set of parameters (for example, the

[TSC] `defineBSet` **Width** parameter). For these exceptions, the parameter description states that the value should not be modified by the user.

6.1 [0x44] Parameters

Companding

Number: 0x4401

Description: The **Companding** parameter controls the encoding of audio for music played when placing a station on hold.

Note: This parameter only applies to DI Series Station Interface boards.

Values:

- 0 [default]: mu-law
- 1: A-law

Guidelines: The **Companding** parameter must be set to the same encoding method as set in the Encoding parameter 0x1209. For details about the Encoding parameter, refer to Section 6.24, "[TSC] Parameters", on page 162.

HDSI boards use country-specific PCD and FCD files. Depending on the PCD/FCD files selected for an HDSI board, the PCM encoding method will be set to either A-law or Mu-law, based on the default value for that country. If this value is not the same as the TDM bus value, the HDSI board will fail to download.

To change the PCM encoding method for the HDSI board from the default value, you will need to edit the Companding parameter in the associated Config file and then restart the system. For additional information about modifying FCD file parameters, see Section 4.13, "Modifying the FCD File by Editing the CONFIG File", on page 78.

6.2 [0x2a] Parameters

No parameters are currently listed in the [0x2a] section of the CONFIG file, but the FSK Transmit and Receive Signal Level parameters are assigned to this section by editing the CONFIG file. For information about adding these parameters to the CONFIG file, see Section 4.12, "Setting the FSK Transmit and Receive Signal Levels", on page 77.

6.3 [0x2b] Parameters

The [0x2b] section of the CONFIG file is used to enable streaming of echo cancellation data over the TDM bus in Continuous Speech Processing (CSP) applications. The [0x2b] section is only applicable to media load CONFIG files that include the **EC Streaming to TDM Bus** parameter. For additional information, refer to Section 2.3, "Media Loads", on page 18

EC Streaming to TDM Bus

Number: 0x2b12

Description: The **EC Streaming to TDM Bus** parameter specifies whether to enable streaming of echo-cancellation data over the time division multiplexing (TDM) bus in Continuous Speech Processing (CSP) applications. This parameter is set on a per channel basis.

Values:

- 0x01: Enable
- 0x02 [default]: Disable

Guidelines: The **EC Streaming to TDM Bus** parameter is only used with media load CONFIG files that support this parameter. For more information about media loads and board densities, see Section 2.3, "Media Loads", on page 18.

6.4 [0x2c] Parameters

The [0x2c] section of the CONFIG file defines enhanced echo cancellation parameters used to set the tail length or tap length, or other Continuous Speech Processing (CSP) related parameters such as Silence Compressed Streaming (SCS). The [0x2c] section includes the following parameters:

- EC Mode
- EC Tail Length

EC Mode

Number: 0x2c1f

Description: The **EC Mode** parameter is used with the **EC Tail Length** parameter to define the echo cancellation tail length, or tap length, used in Continuous Speech Processing (CSP) applications. This parameter is set on a per board basis.

Values:

- 0x02 [default]: For tap length of 24, 32, or 64 ms

Guidelines: The **EC Mode** parameter is used with media loads that support Enhanced Echo Cancellation. For more information about media loads and board densities, see Section 2.3, "Media Loads", on page 18.

When the **EC Tail Length** parameter is set to 0xC0 (24 ms), 0x100 (32 ms), or 0x200 (64 ms), set the **EC Mode** parameter to 0x02.

EC Tail Length

Number: 0x2c03

Description: The **EC Tail Length** parameter specifies the echo cancellation tail length, or tap length, used in Continuous Speech Processing (CSP) applications. This parameter is set on a per board basis and is used in conjunction with the **EC Mode** parameter.

Values:

- 0x80: 16 ms (default for DM/V-A boards)
- 0xC0: 24 ms
- 0x100: 32 ms
- 0x200: 64 ms (default for DM/V-B boards)

Guidelines: The **EC Tail Length** parameter is used with media load 2c CONFIG files only. For more information about media loads and board densities, see Section 2.3, "Media Loads", on page 18.

SCS_PR_SP (SCS Speech Probability Threshold)

Number: 0x2c14

Description: The **SCS_PR_SP** parameter is used for silence compressed streaming (SCS) in continuous speech processing (CSP) applications. It sets the threshold that the calculated probability of speech value is compared to in declaring speech. If the probability of speech is greater than this parameter, speech is declared.

The **SCS_PR_SP** parameter primarily affects the sensitivity of detecting speech at the leading time of a speech period versus false speech detection. This parameter is defined as:

(Probability Value) * 223

Note: Multiplying by 223 converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

For example, for a speech probability threshold of 0.58, **SCS_PR_SP** would have a value of:

0.58 * 223 = 4865393

Values:

- 4194304 to 223 (0.50 to 1.00 probability)
- 6710886 [default]

Guidelines: This parameter is not included in the configuration file. If you want to modify the parameter value, you must add this parameter manually in the configuration file in the [0x2c] section.

SCS_PR_SIL (SCS Silence Probability Threshold)

Number: 0x2c15

Description: This parameter is used for silence compressed streaming (SCS) in continuous speech processing (CSP) applications. It sets the threshold that the calculated probability of speech value is compared to in declaring silence. If the probability of speech is less than this parameter, silence is declared.

The **SCS_PR_SIL** parameter primarily affects the sensitivity of detecting silence at the trailing time of a speech period versus false silence detection. **SCS_PR_SIL** is defined as:

(Probability Value) * 223

Note: Multiplying by 223 converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

For example, for a silence probability threshold of 0.39, **SCS_PR_SIL** would have a value of:

0.39 * 223 = 3271557

Values:

- 0 to 4194304 (0.0 to 0.50 probability)
- 2097152 [default]

SCS_LO_THR (SCS Low Background Noise Threshold)

Number: 0x2c16

Description: The **SCS_LO_THR** parameter is used for silence compressed streaming (SCS) in continuous speech processing (CSP) applications. It controls the low threshold for background noise estimation, and along with the **SCS_HI_THR** parameter, forms a range of loudness. Any signal below this threshold is declared silence.

SCS_LO_THR is defined as:

10(dB value/20) * 223

Note: Multiplying by 223 converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

For example, for a low noise threshold of -78 dB, **SCS_LO_THR** would have a value of:

10(-78/20) * 223 = 1057

Values:

- 839 to 83887 (-80 dB to -40 dB)
- 11286 [default]

Guidelines: Increasing the low background noise threshold increases the probability of losing speech and decreases the probability of recording noise.

SCS_HI_THR (SCS High Background Noise Threshold)

Number: 0x2c17

Description: The **SCS_HI_THR** parameter is used for silence compressed streaming (SCS) in continuous speech processing (CSP) applications. It controls the high threshold for background noise estimation and, along with **SCS_LO_THR** parameter, forms a range of loudness. Any signal above this threshold is declared speech.

SCS_HI_THR is defined as:

10(dB value/20) * 223

Note: Multiplying by 223 converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

For example, for a high noise threshold of -20 dB, **SCS_HI_THR** would have a value of:

10(-20/20) * 223 = 838861

Values:

- 83887 to 223+ (-40 dB to 0 dB)
- 838861 [default]

Guidelines: Reducing the high background noise threshold increases the probability of streaming noise and decreases the probability of losing speech.

SCS_T (SCS Trailing Silence)

Number: 0x2c24

Description: The **SCS_T** parameter is used for silence compressed streaming (SCS) in continuous speech processing (CSP) applications. It defines the duration of silence allowed after the end of a speech block before silence compression begins.

Values:

- 100 to 1000 milliseconds (Values must be entered in 10 millisecond increments. For example, 200, 210, 220 milliseconds)
- 200 milliseconds [default]

Guidelines: If this value is set too low, words and sentences will run together. If it is set too high, silence compression efficiency will be reduced, resulting in larger files.

SCS_Initial_data (SCS duration of initial stream data)

Number: 0x2c25

Description: The **SCS_Initial_data** parameter is used for silence compressed streaming (SCS) in continuous speech processing (CSP) applications. It controls the initial data to be streamed to the application regardless of the presence of silence or non-silence on the line. This initial data can be used by the application to qualify background noise and line conditions.

Values: 0 [default] to 2000 milliseconds (Values must be entered in 10 millisecond increments. For example, 200, 210, 220 milliseconds)

6.5 [encoder] Parameters

The [encoder] section of the CONFIG file includes the following parameters:

- PrmAGCk (AGC K Constant)
- PrmAGClow_threshold (AGC Noise Level Lower Threshold)
- PrmAGCmax_gain (AGC Maximum Gain)
- SCR_T (SCR Trailing Silence)
- SCR_PR_SP (SCR Speech Probability Threshold)
- SCR_PR_SIL (SCR Silence Probability Threshold)
- SCR_LO_THR (SCR Low Background Noise Threshold)
- SCR_HI_THR (SCR High Background Noise Threshold)

PrmAGCk (AGC K Constant)

Number: 0x401

Description: The **PrmAGCk** parameter is the target output level to the TDM bus divided by 32 (to limit K to the range 0 to -1). Note that K is the average level for the output.

PrmAGCk is defined as: K * 2^{23}

K is defined as ($10^{(\text{output level in dB})/20}$)/32. Multiplying by 2^{23} converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards. Therefore, K = 0.006529 corresponds to -13.6 dB average, since 0.006529 = ($10^{(-13.6/20)}$)/32. Note that -13.6 dB average would result in -6.6 dBm level of the analog output signal.

Values: 0x2061 to 0xD5F1 (-30.0 dB to -13.6 dB output levels)

Guidelines: It is recommended that the value be set in the range of -30.0 dB to -13.6 dB. Higher values (-50.0 dB for example) may result in strong peak to average compression if it is enabled or just severe clipping if peak to average compression is disabled.

Here is a sample calculation to obtain a hexadecimal value of **PrmAGCk** for an output level of -19.6 dB:

($10^{(-19.6/20)}$)/32 * 2^{23} = 0x006B39

PrmAGClow_threshold (AGC Noise Level Lower Threshold)

Number: 0x405

Description: The **PrmAGClow_threshold** parameter defines the lower threshold for noise level estimates. Any signal above this threshold will be considered speech. Thus, this threshold should be set quite high in order to let the AGC algorithm determine when there are voiced and unvoiced periods. The parameter is given in terms of the average level.

PrmAGClow_threshold is defined as: $10^{(\text{output level in dB})/20} * 2^{23}$. Multiplying by 2^{23} converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

Note: The AGC high threshold is determined by the ratio of the **PrmAGCk** value over the **PrmAGCmax_gain** value.

Values: 0x20C5 to 0x3000 (-60 dB to -25 dB)

Guidelines: It is recommended that the value be set in the range of -60 dB to -40 dB.

Here is a sample calculation to get a hexadecimal value of **PrmAGClow_threshold** for a noise threshold level of -50 dB_{avg}:

$10^{(-50/20)} * 2^{23} = 0x679F$

PrmAGCmax_gain (AGC Maximum Gain)

Number: 0x408

Description: The **PrmAGCmax_gain** parameter defines the maximum gain divided by 32. This parameter controls the maximum possible gain applied by the AGC algorithm. It also implies the High Threshold Level above which all the inputs produce the target output levels and below which produce the levels linearly decreasing with their input level.

PrmAGCmax_gain is defined as: $((10^{((\text{maximum gain in dB})/20)})/32) * 2^{23}$. Multiplying by 2^{23} converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

Values: 0x040000 to 0x7E7DB9 (0 dB to 30 dB)

Guidelines: It is recommended that the value be set in the range of 0 dB to 30 dB.

Here is a sample calculation to obtain a hexadecimal value of **PrmAGCmax_gain** for a maximum gain of 21 dB:

$((10^{(21/20)})/32) * 2^{23}$

SCR_T (SCR Trailing Silence)

Number: 0x415

Description: The **SCR_T** parameter is used for Silence Compressed Recording and defines the duration of silence allowed following speech before SCR begins (trailing silence).

Values: 100 to 1000 (milliseconds)

Guidelines: If it is set too low, words and sentences will run together; if it is set too high, SCR efficiency will be reduced, resulting in larger files.

SCR_PR_SP (SCR Speech Probability Threshold)

Number: 0x417

Description: The **SCR_PR_SP** parameter is used for Silence Compressed Recording and sets the threshold that the calculated probability of speech value is compared to in declaring speech. If the probability of speech is greater than this parameter, speech is declared.

The **SCR_PR_SP** parameter primarily affects the sensitivity of detecting speech at the leading time of a speech period versus false speech detection.

SCR_PR_SP is defined as: (Probability Value) * 2^{23}. Multiplying by 2^{23} converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

Values: 4194304 to 2^{23} (0.50 to 1.00 probability)

Guidelines: For example, for a speech probability threshold of 0.58, **SCR_PR_SP** would have a value of:

$0.58 * 2^{23} = 4865392$

SCR_PR_SIL (SCR Silence Probability Threshold)

Number: 0x418

Description: The **SCR_PR_SIL** parameter is used for Silence Compressed Recording and sets the threshold that the calculated probability of speech value is compared to in declaring silence. If the probability of speech is less than this parameter, silence is declared.

The **SCR_PR_SIL** parameter primarily affects the sensitivity of detecting silence at the trailing time of a speech period versus false silence detection.

SCR_PR_SIL is defined as: (Probability Value) * 2^{23}. Multiplying by 2^{23} converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

Values: 0 to 4194304 (0.0 to 0.50 probability)

Guidelines: For example, for a silence probability threshold of 0.39, **SCR_PR_SIL** would have a value of:

$0.39 * 2^{23} = 3271557$

SCR_LO_THR (SCR Low Background Noise Threshold)

Number: 0x419

Description: The **SCR_LO_THR** parameter is used for Silence Compressed Recording and controls the low threshold for background noise estimation and, along with **SCR_HI_THR** parameter, forms a range of loudness. Any signal below this threshold is declared silence. Increasing this threshold increases the probability of losing speech and decreases the probability of recording noise.

SCR_LO_THR is defined as: $10^{(\text{dB value}/20)} * 2^{23}$. Multiplying by 2^{23} converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

Values: 839 to 83887 (-80 dB to -40 dB)

Guidelines: For example, for a low noise threshold of -78 dB, **SCR_LO_THR** would have a value of:

$10^{(-78/20)} * 2^{23} = 1056$

SCR_HI_THR (SCR High Background Noise Threshold)

Number: 0x41A

Description: The **SCR_HI_THR** parameter is used for Silence Compressed Recording and controls the high threshold for background noise estimation and, along with **SCR_LO_THR** parameter, forms a range of loudness. Any signal above this threshold is declared speech. Reducing this threshold increases the probability of recording noise and decreases the probability of losing speech.

SCR_HI_THR is defined as: $10^{(\text{dB value}/20)} * 2^{23}$. Multiplying by 2^{23} converts the value into a linear 24-bit value that accommodates the 24-bit DSPs used on DM3 boards.

Values: 83887 to 2^{23}+ (-40 dB to 0 dB)

Guidelines: For example, for a high noise threshold of -20 dB, **SCR_HI_THR** would have a value of:

$10^{(-20/20)} * 2^{23} = 838860$

6.6 [recorder] Parameters

The [recorder] section of the CONFIG file includes the following parameters:

- Duration (Record Duration)
- BufferTruncate (Buffer Truncate)
- BeepSignalID (Pre-Recording Beep)
- AGCOnOff (AGC Flag)
- SCR (SCR Flag)

Duration (Record Duration)

Number: 0x200

Description: The **Duration** parameter specifies the maximum duration (in milliseconds) for which to record. The maximum duration time is 72 hours (259,200,000 ms).

Values: 0 to 259,200,000 (milliseconds)

Guidelines: The time specified must be divisible by 4.

BufferTruncate (Buffer Truncate)

Number: 0x202

Description: The **BufferTruncate** parameter specifies the amount of data (in milliseconds) to truncate from the record buffer at the end of a recording.

Values: 0 to 4000 (milliseconds)

Guidelines: The suggested range is about 50 to 150 ms (varies with coder).

BeepSignalID (Pre-Recording Beep)

Number: 0x203

Description: The **BeepSignalID** parameter is the signal identifier of the beep tone preceding the recording.

Values:

- 0x21: 444 Hz tone for 400 ms
- 0x22: 1000 Hz tone for 400 ms

AGCOnOff (AGC Flag)

Number: 0x205

Description: The **AGCOnOff** parameter enables or disables Automatic Gain Control (AGC) on a per board basis. These settings can be changed for individual channels using API calls.

Values:

- 0: Disable AGC
- 1: Enable AGC

SCR (SCR Flag)

Number: 0x209

Description: The **SCR** parameter enables or disables Silence Compressed Recording (SCR) on a per board basis. These settings can be changed for individual channels using API calls.

Values:

- 9: Enable SCR
- 10: Disable SCR

6.7 [0x39] Parameters

The [0x39] section of the CONFIG file includes a single parameter.

ToneClamping (Tone Clamping)

Number: 0x3925

Description: The **ToneClamping** parameter is used to disable conference tone clamping. Tone clamping reduces the amount of DTMF tones heard in a conference.

Note: To also disable the conference notification tone on Dialogic Integrated Series boards (or any Intel® Dialogic® Voice board), modifications to the **NotificationTone** parameter are also required. For details, see "NotificationTone (Conferencing Notification Tone)", on page 107.

Values: 0

Guidelines: Tone clamping is enabled by default on Dialogic Integrated Series boards. To disable tone clamping, the **ToneClamping** parameter must be added to the [0x39] section of the CONFIG file (using the `SetParm` format) and set to a value of 0. To re-enable tone clamping, the **ToneClamping** parameter must be removed from the [0x39] section of the CONFIG file.

6.8 [0x3b] Parameters

The [0x3b] section of the CONFIG file includes a single parameter.

ActiveTalkerNotifyInterval (Active Talker Notification Interval)

Number: 0x3b02

Description: The **ActiveTalkerNotifyInterval** parameter is the periodic duration at which a message listing the active talkers in a conference is sent to the host.

Values represent 10 millisecond units. For example, to send a notification message once per second, the **ActiveTalkerNotifyInterval** parameter should be set to a value of 100 (1 second = 1000 ms / 10 = 100).

Values: 0 to 1000 (10 ms units)

Guidelines: If a low value is used, it can affect system performance due to the more frequent updating of the status (which results in a high quantity of internal notification messages). If a high value is used, it will result in less frequent updating of status, but the non-silence energy of a conferee may not be reported if it occurs between notification updates. For example, if the notification interval is set to 2 seconds and a conferee only says "yes" or "no" quickly in between notifications, that vocalization by the conferee will not be reported.

6.9 [0x3b.x] Parameters

The [0x3b.x] section of the CONFIG file includes a single parameter.

NotificationTone (Conferencing Notification Tone)

Number: 0x3b06

Description: The **NotificationTone** parameter is used to disable the conferencing notification tone on DM/V-A, DM/V-B, and Dialogic Integrated (DI) Series boards. The conferencing notification tone is generated to alert conferees when a party enters or exits a conference.

Note: To also disable tone clamping on Dialogic Integrated Series boards, modifications to the **ToneClamping** parameter are required. For details, see "ToneClamping (Tone Clamping)", on page 106.

Values: 0

Guidelines: This tone is enabled by default and the **NotificationTone** parameter must be added to the CONFIG file to disable the tone. To re-enable the tone, the **NotificationTone** parameter must be removed from the [0x3b.*x*] sections of the CONFIG file.

To disable the conferencing notification tone, the **NotificationTone** parameter must be added to the [0x3b.*x*] sections of the CONFIG file and set to a value of 0. Each [0x3b.*x*] section applies to a specific conferencing line. That is, [0x3b.1] applies to conferencing line 1, [0x3b.2] applies to conferencing line 2, and so on.

For DM/V-A and DM/V-B boards, there are 30 conferencing lines, and therefore, 30 [0x3b.*x*] sections in the CONFIG file: [0x3b.1] through [0x3b.30].

For the DI/0408-LS-A Dialogic Integrated Series board, there are three conferencing lines, and therefore, three [0x3b.*x*] sections in the CONFIG file: [0x3b.1] through [0x3b.3].

For the DI/SI-16, DI/SI-24. and DI/SI-32 Dialogic Integrated Series boards, there are five conferencing lines, and therefore, five [0x3b.*x*] sections in the CONFIG file: [0x3b.1] through [0x3b.5].

6.10 [lineAdmin.x] Parameters (Digital Voice)

For digital voice boards, the line administration parameters are associated with an individual T1 or E1 trunk. The parameters defined in the [lineAdmin.x] section are associated with line x. For example, parameters in the [lineAdmin.3] section of the CONFIG file are associated with line 3. Digital voice line administration parameters include:

- LineType (Line Type)
- SignalingType (Signaling Type)
- Coding (Coding)
- ZeroCodeSuppression (Zero Code Suppression)
- FramingAlgorithm (CRC Checking)
- LOSDeclaredTime (LOS Declared Time)
- LOSClearedTime (LOS Cleared Time)
- REDCFADecay (RED CFA Decay)
- REDCFADeclareTime (RED CFA Declare Time)
- REDCFAClearedTime (RED CFA Cleared Time)

- YellowCFADeclareTime (Yellow CFA Declare Time)
- YellowCFAClearTime (Yellow CFA Clear Time)
- RAICRCCFADeclareTime (RAI CRC CFA Declare Time)
- RAICRCCFAClearTime (RAI CRC CFA Clear Time)

LineType (Line Type)

Number: 0x1601

Description: The **LineType** parameter defines the physical line type (T1 or E1) and the framing format (for example, D4 or ESF). Framing formats include:

D4 framing (D4)
For T1 lines, in D4 framing, 12 frames of 193 bits each (2,316 bits total) constitute a superframe. This framing format supports AB signaling.

Extended superframe (ESF)
For T1 lines, in ESF framing, 24 frames of 193 bits each (4,632 bits total) constitute an extended superframe. This framing format supports ABCD signaling.

CEPT E1
For E1 lines, uses CEPT E1 framing.

Cyclic redundancy check 4 (CRC-4) multi-frame
For E1 lines, this provides for CRC error detection. In this framing format, E1 lines have an extra framing that can coexist with the standard framing and the time slot 16 signaling framing. This extra framing is used to compute and check CRC-4 on incoming lines, to detect remote CRC-4 alarms, and to notify the remote line of CRC-4 errors. When CRC-4 framing is enabled, all CRC-related statistics will be collected and reported, and the RAI_CRC_CFA alarm will be detected and reported.

Analog
Analog is the framing used for Dialogic Integrated Series boards.

Values:

- 0: T1 D4 (dsx1_D4)
- 1: T1 ESF (dsx1_ESF)
- 2: E1 CEPT E1 (dsx1_E1)
- 3: E1 CRC 4 multi-frame (dsx1_E1_CRC)
- 4: analog

SignalingType (Signaling Type)

Number: 0x1602

Description: The **SignalingType** parameter defines the signaling type to be used by the T1 or E1 line. Signaling types include:

Channel associated signaling (CAS)
In CAS, the signaling for each channel is directly associated with that channel. T1 robbed-bit signaling is an example of CAS.

Common channel signaling (CCS)
In CCS, a common channel carries the signaling for all of the channels on that T1 or E1 line. ISDN is an example of CCS, where the D channel is used to carry the signaling for all of the B channels.

Clear channel signaling (Clear)
In this type, none of the channels on the T1 or E1 line are used for signaling purposes. Clear channel signaling is the ability to access telephony channels in the system and configure them to a user defined call control protocol, or to simply leave the lines 'clear'. The resources should have access to the telephony bus for media routing purposes, as well as signal detection, signal generation, and tone generation capabilities, if desired.

Note: In a clear channel configuration, the CT Bus does not preserve frames, so any in-band signaling is lost. That is, T1 CAS robbed bit signaling cannot be performed on a line configured to use clear channel signaling.

Values:

- 4: CAS
- 5: CCS
- 6: Clear

Guidelines: When using Non-Facility-Associated Signaling (NFAS), Signaling Type is dependent on whether the T1 or E1 line is a primary, standby (DCBU), or NFAS ISDN trunk. The primary trunk must be set to CCS and the standby and NFAS trunks must be set to Clear.

Note: NFAS is supported on only the ISDN NI-2, 4ESS, 5ESS and DMS protocols, and NFAS D channel backup (DCBU) is supported only on ISDN NI-2 protocol.

For additional parameters that need to be modified for NFAS, see Section 6.12, "[NFAS] Parameters", on page 114

Coding (Coding)

Number: 0x1603

Description: The **Coding** parameter defines the coding scheme to be used by a digital line type. Coding schemes include:

Modified alternate mark inversion (B8ZS)
This is a modified AMI code that only applies to T1 lines and is used to preserve one's density on the line. Whenever eight consecutive zeros occur on the line, they are replaced by an 8-bit string that violates the bipolar signaling. If the preceding pulse was positive, the polarity of the substituted eight bits is 000+-0-+. If the preceding pulse was negative, the polarity of the substituted eight bits is 000-+0+-.

Alternate mark inversion (AMI)
This is a bipolar signal conveying binary digits in which each successive 1 (mark) is of the opposite polarity. If the previous mark was a positive pulse, then the next mark will be a negative pulse. Spaces have an amplitude of zero (no pulse).

High density bipolar three zero (HDB3)
High density bipolar three zero is a modified AMI code that only applies to E1 and is used to preserve one's density on the line. Whenever four consecutive zeros appear, the four-zeros group is replaced with an HDB3 code. This could be either of two HDB3 codes, depending on

whether there was an odd or even number of ones since the last bipolar violation. If an odd number of ones occurred, the substituted four bits are 000V, where V represents a bipolar violation. If an even number of ones occurred, the substituted four bits are P00V, where P represents a parity bit and V represents a bipolar violation.

Values:

- 7: B8ZS
- 8: AMI
- 9: HDB3

ZeroCodeSuppression (Zero Code Suppression)

Number: 0x1604

Description: The **ZeroCodeSuppression** parameter is an algorithm used by T1 lines that inserts a 1 bit into a stream to prevent the transmission of eight or more consecutive 0 bits, which could produce timing errors. Instead, this algorithm maintains a minimum one's density to reduce timing errors.

Values:

- 10: Bell - Bell zero code suppression (Jam Bit 7)
- 11: GTE - GTE zero code suppression (Jam Bit 8, except in signaling frames when Jam Bit 7 is used if the signaling bit is 0)
- 12: DDS - Digital Data Service zero code suppression (data byte is replaced with 10011000)
- 13: None - No zero code suppression is used.

Guidelines: The **ZeroCodeSuppression** parameter is used when AMI line-coding is used, that is, when the **Coding** parameter is set to AMI. Since AMI does not perform zero code suppression, the **ZeroCodeSuppression** parameter ensures there are no long strings of consecutive zeros on the line.

If the **Coding** parameter is set to B8ZS or HDB3 (for E1), then zero code suppression is performed by the line-coding and the **ZeroCodeSuppression** parameter is ignored.

FramingAlgorithm (CRC Checking)

Number: 0x1624

Description: A T-1 front end can run two different framing algorithms when configured as extended superframe (ESF): a default algorithm and an alternate CRC-6 checking algorithm. The CRC-6 checking algorithm allows the circuit to confirm the CRC-6 bits in the received multiframe, as a guard against mimic framing patterns, before forcing a new frame alignment. The CRC Checking parameter allows you to enable the CRC-6 checking algorithm.

Values:

- 0: Default algorithm
- 1: Alternate CRC-6 checking algorithm

Guidelines: This parameter only applies to T-1 trunks whose Line Type parameter (0x1601) is set to 1 (dsx1_ESF). For all other Line Types, this parameter is invalid.

LOSDeclaredTime (LOS Declared Time)

Number: 0x160c

Description: The **LOSDeclaredTime** parameter defines the number of milliseconds for which no signal is detected at the input port before a loss of signal (LOS) or carrier-failure alarm (CFA) can be declared.

Values: 0 to 2500 (milliseconds)

LOSClearedTime (LOS Cleared Time)

Number: 0x160d

Description: The **LOSClearedTime** parameter defines the number of milliseconds for which a signal must be detected at the input port before a declared LOS or CFA can be cleared.

Values: 0 to 2500 (milliseconds)

REDCFADecay (RED CFA Decay)

Number: 0x1609

Description: The **REDCFADecay** parameter is the denominator of the fraction used to calculate the decay slope in the integration process when RED CFA condition has not been declared and LOS or LOF is intermittent.

Values: 4 to 15 (1/4 to 1/15)

REDCFADeclareTime (RED CFA Declare Time)

Number: 0x160a

Description: The **REDCFADeclareTime** parameter defines the number of milliseconds that a red alarm condition must be received at the input port before a RED CFA condition can be declared.

Values: 0 to 2500 (milliseconds)

REDCFAClearedTime (RED CFA Cleared Time)

Number: 0x160b

Description: The **REDCFAClearedTime** parameter defines the number of milliseconds that a normal signal must be received at the input port before a declared RED CFA condition can be cleared.

Values: 1000 to 15000 (milliseconds)

YellowCFADeclareTime (Yellow CFA Declare Time)

Number: 0x160e

Description: The **YellowCFADeclareTime** parameter defines the number of milliseconds for which a Remote Alarm Indication (RAI) signal is detected at the input port before a yellow CFA condition can be declared.

Values: 0 to 2500 (milliseconds)

YellowCFAClearTime (Yellow CFA Clear Time)

Number: 0x160f

Description: The **YellowCFAClearTime** parameter defines the number of milliseconds for which a RAI signal is not detected at the input port before a declared yellow CFA condition can be cleared.

Values: 0 to 2500 (milliseconds)

RAICRCCFADeclareTime (RAI CRC CFA Declare Time)

Number: 0x1610

Description: The **RAICRCCFADeclareTime** parameter defines the number of seconds for which a RAI signal and CRC Error is detected at the input port before a RAI CRC CFA can be declared.

Values: 0 to 450 (milliseconds)

RAICRCCFAClearTime (RAI CRC CFA Clear Time)

Number: 0x1611

Description: The **RAICRCCFAClearTime** parameter defines the number of seconds for which a RAI signal and Remote CRC Error is not detected at the input port before a declared RAI CRC CFA can be cleared.

Values: 0 to 450 (milliseconds)

InitialBitPattern (Initial CAS Signaling Bit Pattern)

Number: 0x1625

Description: The **InitialBitPattern** parameter defines the values of the CAS ABCD signaling bits that are transmitted for all channels on the specified line at the time the firmware is downloaded and intialized.

Values: 0x0 to 0xf, where the hexadecimal value represents the binary ABCD bit values. For example, 0xd defines the ABCD bit pattern as 1101.

Guidelines: For a T1 line, the default is 0x0. For an E1 line, the default is 0xd.

6.11 [LineAdmin.x] Parameters (Analog Voice)

For DM3 analog voice boards, the line administration parameters consist of a collection of audio filter coefficients.

Audio Filter Coefficients

Number: 0x1630 - 0x1638

Description: The **Audio Filter Coefficients** parameter defines the filter coefficients programmed into the Quad Subscriber Line Audio-Processing Circuit (QSLAC) device on the board. These filters optimize the analog circuitry to match specific loading impedances, receive and transmit gain, and provide equalization of the transmit and receive paths of the loop start trunks on a DM3 analog voice board.

The filters are grouped by termination impedances – either 600 ohms or a predetermined Complex impedance. The filter coefficients are further grouped in each [LineAdmin.x] section as being associated with a group of four analog loop start trunks. For example, parameters included in the [LineAdmin.3] section of the CONFIG file are associated with trunks 9 through 12. All of the parameters defined in a [LineAdmin.x] section must be defined for only one impedance value (for example, 600 ohm or Complex). The filter coefficient parameters in a particular [LineAdmin.x] section must not be altered or individually defined.

For example, to define the termination impedance as Complex for ports 1 through 4 on a DMV160LP board, you must first comment out all of the lines in the [LineAdmin.1] section for 600 ohms in the CONFIG file by inserting an exclamation symbol (!) at the beginning of the [LineAdmin.1] line and at the beginning of each SetVParm line as shown in the following:

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!  QSLAC Coefficients Line Admin 1 for loop start trunk interfaces 1-4,
!  US Coefficients for input impedance of; 600 ohms
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

![LineAdmin.1]  ! ComponentName=LineAdmin  Std_ComponentType=0x16

!SetVParm(0x1630,0x200800) = 0xAB, 0x60 ! DMV160-LP LSI US-600ohm QSLAC GX Filter

!SetVParm(0x1631,0x200802) = 0x4A, 0xE0 ! DMV160-LP LSI US-600ohm QSLAC GR Filter

!SetVParm(0x1632,0x207004) = 0x23, 0x25, 0x42, 0x9F, 0x42, 0xAD, 0x3A, 0xAE, 0x72, 0xD6, 0xAC,
0xF2, 0xB6, 0x6E, 0x01 ! DMV160-LP LSI US-600ohm QSLAC Z Filter

!SetVParm(0x1633,0x206813) = 0xF8, 0xFB, 0x87, 0xD8, 0xFC, 0x87, 0xA8, 0x7A, 0x87, 0xA8, 0x7A,
0x87, 0xA8, 0x70 ! DMV160-LP LSI US-600ohm QSLAC B1 Filter

!SetVParm(0x1634,0x200821) = 0xA5, 0xD0 ! DMV160-LP LSI US-600ohm QSLAC B2 Filter

!SetVParm(0x1635,0x205823) = 0x01, 0x11, 0x01, 0x90, 0x01, 0x90, 0x01, 0x90, 0x01, 0x90, 0x01,
0x90 ! DMV160-LP LSI US-600ohm QSLAC X Filter

!SetVParm(0x1636,0x20682F) = 0xDC, 0x01, 0xA2, 0x60, 0x2C, 0xCD, 0x2A, 0x97, 0x23, 0x26, 0xBB,
0x2E, 0x23, 0xC7 ! DMV160-LP LSI US-600ohm QSLAC R Filter

!SetVParm(0x1637,0x20003D) = 0x80 ! DMV160-LP LSI US-600ohm QSLAC AISN
```

```
!SetVParm(0x1638,0x20003E) = 0x7F ! DMV160-LP LSI US-600ohm QSLAC OperFunc
```

and then uncomment all of the lines in the [LineAdmin.1] section for Complex impedance by deleting the exclamation symbol (!) from the beginning of the [LineAdmin.1] line and the beginning of each SetVParm line as shown in the following:

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!  QSLAC Coefficients Line Admin 1 for loop start trunk interfaces 1-4,
!  US Coefficients for recommended compromise complex impedance,
!  300 ohms + [1K ohms // 220 nF], ref: EIA/TIA 464B
!  Select when line conditions are as follow: non-conditioned lines, long distance from CO
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

[LineAdmin.1]  ! ComponentName=LineAdmin  Std_ComponentType=0x16

SetVParm(0x1630,0x200800) = 0x23, 0x60 ! DMV160-LP LSI US-complex QSLAC GX Filter

SetVParm(0x1631,0x200802) = 0x3B, 0xE0 ! DMV160-LP LSI US-complex QSLAC GR Filter

SetVParm(0x1632,0x207004) = 0xB3, 0xAB, 0x2D, 0xB9, 0xAA, 0x3B, 0xA3, 0xB3, 0xC3, 0x3B, 0x37,
0xCA, 0xB5, 0x6E, 0x01 ! DMV160-LP LSI US-complex QSLAC Z Filter

SetVParm(0x1633,0x206813) = 0xA8, 0x7B, 0xA5, 0xB2, 0xD3, 0xCB, 0x2A, 0xBA, 0xAC, 0x45, 0xD2,
0xAD, 0x3E, 0xE0 ! DMV160-LP LSI US-complex QSLAC B1 Filter

SetVParm(0x1634,0x200821) = 0xAA, 0x31 ! DMV160-LP LSI US-complex QSLAC B2 Filter

SetVParm(0x1635,0x205823) = 0xAB, 0xA0, 0x2B, 0x33, 0x2D, 0xA3, 0x2A, 0x24, 0xAA, 0x2D, 0xA2,
0x25 ! DMV160-LP LSI US-complex QSLAC X Filter

SetVParm(0x1636,0x20682F) = 0xB3, 0xD0, 0xDA, 0x10, 0x2D, 0xA9, 0xAC, 0x5B, 0xFA, 0xAE, 0x72,
0x2C, 0xAA, 0x3E ! DMV160-LP LSI US-complex QSLAC R Filter

SetVParm(0x1637,0x20003D) = 0x80 ! DMV160-LP LSI US-complex QSLAC AISN

SetVParm(0x1638,0x20003E) = 0x7F ! DMV160-LP LSI US-complex QSLAC OperFunc
```

Values:

- 600: 600 ohms
- Complex: Complex

Guidelines: Select 600 ohms to match a 600 ohm impedance and select Complex to match a complex impedance.

6.12 [NFAS] Parameters

Non-Facility-Associated Signaling (NFAS) uses a single ISDN PRI D channel to provide signaling and control for multiple ISDN PRI lines. When using NFAS, modifications also need to be made to other sections of the CONFIG file. For details, see the following:

- "SignalingType (Signaling Type)", on page 108.
- Section 6.13, "[NFAS.x] Parameters", on page 115.

There is only one NFAS component level parameter.

NFAS_INSTANCE_MAP (NFAS Instance Map)

Number: 0x3E02

Description: The **NFAS_INSTANCE_MAP** parameter defines the number of NFAS groups or NFAS instances created on a particular board. One NFAS group is created for each primary D channel on the board.

Values:

- 0x0: 0 (0000)
- 0x1: 1 (0001)
- 0x3: 2 (0011)
- 0x7: 3 (0111)
- 0xF: 4 (1111)

Guidelines: The **NFAS_INSTANCE_MAP** parameter value is a hexadecimal bitmap that represents the number of NFAS groups that are needed. The bitmap's least significant bit correlates to the first NFAS instance, the next least significant bit corresponds to the second NFAS instance, and so on. So, starting with the least significant bit and working towards the most significant bit, set each bit's value to 1 for each NFAS instance needed. For example, to create three NFAS groups, set the value of the **NFAS_INSTANCE_MAP** parameter to 0x07 (0111).

6.13 [NFAS.x] Parameters

Non-Facility-Associated Signaling (NFAS) uses a single ISDN PRI D channel to provide signaling and control for multiple ISDN PRI lines. For each group defined by the **NFAS_INSTANCE_MAP** parameter, there will be an [NFAS.x] section in the CONFIG file. For example, [NFAS.1] corresponds to the NFAS instance for the first group, [NFAS.2] corresponds to the NFAS instance for the second group, and so on.

When using NFAS, modifications also need to be made to other sections of the CONFIG file. For details, see the following parameters:

- "NFAS_INSTANCE_MAP (NFAS Instance Map)", on page 115.
- "SignalingType (Signaling Type)", on page 108

NFAS instance level parameters include:

- GroupID (Group Identifier)
- NFAS_PrimaryIntID (Primary Instance Identifier)
- NFAS_Standby_IntID (Standby Instance Identifier)

GroupID (Group Identifier)

Number: 0x3E00

Description: The **GroupID** parameter is defined for each NFAS group created. This parameter defines the NFAS group including the trunks that are assigned to it.

Values: 1 to 4

Guidelines: When setting this parameter, the trunks assigned to the group must also be defined. For each group, multiple trunks are identified and added in recurring sets of triplets, using the following command:

```
AddNFASInterface(x)= a,b,c,  a',b',c', ...
```

Where:

x = GroupID
NFAS group into which the interface needs to be added. For [NFAS.x], this would be "x".

a = InterfaceID
Unique number for this interface assigned by the user. A maximum of 10 interfaces can be assigned to a single group.

b = BoardNumber
Logical board number (as defined by the **Logical ID** parameter) on which the trunk being assigned to the InterfaceID resides.

c = InstanceNumber
Instance number of the trunk that is being assigned to the InterfaceID. Trunks are numbered sequentially based on their physical location on the boards, from top to bottom.

For example, to add all four trunks on board 2 and the first two trunks on board 3 to the fourth NFAS group, enter the following to the [NFAS.4] section in the CONFIG file:

```
[NFAS.4]
AddNFASInterface(4)=0,2,1, 1,2,2, 2,2,3, 3,2,4, 4,3,1, 5,3,2
SetParm=0x3E04,0
```

NFAS_PrimaryIntID (Primary Instance Identifier)

Number: 0x3E04

Description: The **NFAS_PrimaryIntID** parameter defines the primary D channel used by the NFAS group and is set for every [NFAS.x] group that is created.

Values: 0 to 9 (valid **InterfaceID** value)

Guidelines: The parameter is set to one of the [NFAS.x] InterfaceIDs defined by the **GroupID** parameter's `AddNFASInterface` command. For details, see "GroupID (Group Identifier)", on page 116.

For example, to define the primary D channel for NFAS group 4 to be the second trunk on board 3, enter the following to the [NFAS.4] section in the CONFIG file:

```
[NFAS.4]
AddNFASInterface(4)=0,2,1, 1,2,2, 2,2,3, 3,2,4, 4,3,1, 5,3,2
SetParm=0x3e04,5
```

NFAS_Standby_IntID (Standby Instance Identifier)

Number: 0x3E05

Description: The **NFAS_Standby_IntID** parameter defines the standby, or backup, D channel used by the NFAS group. This parameter is set for every [NFAS.x] group that implements D channel backup (DCBU).

Note: DCBU is supported on only DM/V, DM/N, DM/T, and DMN160TEC boards using ISDN NI-2.

Values: 0 to 9 (valid **InterfaceID** value)

Guidelines: The parameter is set to one of the [NFAS.x] InterfaceIDs defined by the **GroupID** parameter's `AddNFASInterface` command. For details about the `AddNFASInterface` command, see "GroupID (Group Identifier)", on page 116.

In the example:

```
[NFAS.4]
AddNFASInterface(4)=0,2,1, 1,2,2, 2,2,3, 3,2,4, 4,3,1, 5,3,2
SetParm=0x3e04,5
```

to define the first trunk on board 2 the standby D channel for the fourth NFAS group, add parameter 0x3e05 to the [NFAS.4] section of the CONFIG file and set it to a value of 0:

```
[NFAS.4]
AddNFASInterface(4)=0,2,1, 1,2,2, 2,2,3, 3,2,4, 4,3,1, 5,3,2
SetParm=0x3e04,5
SetParm=0x3e05,0
```

6.14 [CAS] Parameters for T1 E&M Signals

The basis for the T1 E&M wink protocol is Channel Associated Signaling (CAS). The CAS component is responsible for the generation and detection of CAS signals on the phone network interface. The CAS T-1 E&M wink signals are defined in this section of the CONFIG file and assigned as variants in the [CHP] T-1 Protocol Variant Definition section of the CONFIG file. For details, see Section 6.21, "[CHP] T1 Protocol Variant Definitions", on page 138.

Note: The CAS signaling parameters should only be modified by experienced users if the default settings do not match what the line carrier or PBX is sending or expecting for the line protocol configuration running on the card.

CAS T1 E&M wink parameters include:

- Offhook (E&M Off-hook Signal)
- Onhook (E&M On-hook Signal)
- FlashOnhook (E&M Flash On-hook Signal)
- Wink (E&M Wink Signal)
- Flash (E&M Flash Signal)

Offhook (E&M Off-hook Signal)

Number: 0xC15CA001

Description: The **Offhook** parameter defines the transition signal from an on-hook state to an off-hook state. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xF0 (11110000)
- PostVal: 0xFF (11111111)
- PreTm: 100 ms
- PostTm: 300 ms

Onhook (E&M On-hook Signal)

Number: 0xC15CA002

Description: The **Onhook** parameter defines the transition signal from an off-hook state to an on-hook state. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xFF (11111111)
- PostVal: 0xF0 (11110000)
- PreTm: 300 ms
- PostTm: 100 ms

FlashOnhook (E&M Flash On-hook Signal)

Number: 0xC15CA003

Description: The **FlashOnhook** parameter defines the transition signal from an off-hook state to a flash on-hook state during a blind transfer. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xFF (11111111)
- PostVal: 0xF0 (11110000)
- PreTm: 300 ms
- PostTm: 1000 ms

Wink (E&M Wink Signal)

Number: 0xC15CA011

Description: The **Wink** parameter defines a pulse signal for the purposes of protocol handshaking and is typically used as an acknowledgment signal to the line carrier or PBX. It is most often used to acknowledge signaling bit changes detected from the carrier or to signal the start or end of digit collection. The signal transitions from OffVal to OnVal and back to OffVal. For detailed information about pulse signals and their associated values, see Section 3.5.3, "Pulse Signal", on page 36.

Values:

- OffVal: 0xF0 (11110000)
- OnVal: 0xFF (11111111)
- PreTm: 100 ms
- MinTm: 210 ms
- NomTm: 250 ms
- MaxTm: 280 ms
- PostTm: 100 ms

Flash (E&M Flash Signal)

Number: 0xC15CA012

Description: The **Flash** parameter defines a pulse signal for the purposes of requesting special processing. This signal is typically sent to transfer a call to another phone or channel while the call is connected and in progress. The signal goes from OffVal to OnVal and back to OffVal. For detailed information about pulse signals and their associated values, see Section 3.5.3, "Pulse Signal", on page 36.

Values:

- OffVal: 0xFF (11111111)
- OnVal: 0xF0 (11110000)
- PreTm: 100 ms
- MinTm: 210 ms
- NomTm: 250 ms
- MaxTm: 280 ms
- PostTm: 100 ms

6.15 [CAS] Parameters for T1 Loop Start Signals

The basis for the T1 loop start protocol is Channel Associated Signaling (CAS). The CAS component is responsible for the generation and detection of CAS signals on the phone network interface. The CAS T-1 loop start signals are defined in this section of the CONFIG file and assigned as variants in the [CHP] T-1 Protocol Variant Definition section of the CONFIG file. For details, see Section 6.21, "[CHP] T1 Protocol Variant Definitions", on page 138.

Note: The CAS signaling parameters should only be modified by experienced users if the default settings do not match what the line carrier or PBX is sending or expecting for the line protocol configuration running on the card.

CAS T1 loop start parameters include:

- PBX_Open (Loop Start PBX Open Signal)
- PBX_Close (Loop Start PBX Close Signal)

- Net_Answer (Loop Start Net Answer Signal)
- Net_Drop (Loop Start Net Drop Signal)
- Net_Abandon (Loop Start Net Abandon Signal)
- Net_RingOn (Loop Start Net Ring On Signal)
- Net_RingOff (Loop Start Net Ring Off Signal)
- PBX_FlashOpen (Loop Start PBX Flash Open Signal)
- Net_FlashDrop (Loop Start Net Flash Drop Signal)
- PBX_Flash (Loop Start PBX Flash Signal)
- Loop Start Train Definition
- Loop Start Sequence Definition

PBX_Open (Loop Start PBX Open Signal)

Number: 0xC15CA021

Description: The **PBX_Open** parameter defines the transition signal sent to drop a call. In an analog environment, the station goes from off-hook to on-hook. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xFF (11111111)
- PostVal: 0xF5 (11110101)
- PreTm: 100 ms
- PostTm: 100 ms

PBX_Close (Loop Start PBX Close Signal)

Number: 0xC15CA022

Description: The **PBX_Close** parameter defines the transition signal sent to make an outbound call, or to answer an incoming call. In an analog environment, the station goes from on-hook to off-hook. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xF5 (11110101)
- PostVal: 0xFF (11111111)
- PreTm: 100 ms
- PostTm: 100 ms

Net_Answer (Loop Start Net Answer Signal)

Number: 0xC15CA023

Description: The **Net_Answer** parameter defines the transition signal that, when received, indicates that the network has answered an outbound call. In an analog environment, when the station goes off-hook, the network answers with loop current. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

Values:

- PreVal: 0xF5 (11110101)
- PostVal: 0xF0 (11110000)
- PreTm: 100 ms
- PostTm: 100 ms

Net_Drop (Loop Start Net Drop Signal)

Number: 0xC15CA024

Description: The **Net_Drop** parameter defines the transition signal that, when received, indicates that the network has dropped the call. In an analog environment, with the station off-hook, the network hangs up by dropping loop current. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

Values:

- PreVal: 0xA0 (10100000)
- PostVal: 0xAA (10101010)
- PreTm: 100 ms
- PostTm: 100 ms

Net_Abandon (Loop Start Net Abandon Signal)

Number: 0xC15CA025

Description: The **Net_Abandon** parameter defines the transition signal that, when received, indicates the network has dropped an offered call, that is, the network stops ringing the line. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

Values:

- PreVal: 0xF0 (11110000)
- PostVal: 0xF5 (11110101)
- PreTm: 1300 ms
- PostTm: 4500 ms

Net_RingOn (Loop Start Net Ring On Signal)

Number: 0xC15CA026

Description: The **Net_RingOn** parameter defines the transition signal that, when received, indicates that the network is ringing the line, and an inbound call is offered. In an analog environment, the station is on-hook and the network rings the station. This is the leading edge of the ring. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

Values:

- PreVal: 0xF5 (11110101)
- PostVal: 0xF0 (11110000)
- PreTm: 3900 ms
- PostTm: 50 ms

Net_RingOff (Loop Start Net Ring Off Signal)

Number: 0xC15CA027

Description: The **Net_RingOff** parameter defines the transition signal that, when received, indicates that the network is still offering an inbound call, but has stopped ringing the line. In an analog environment, the station is on-hook and the network pauses between rings. This is the trailing edge of the ring. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

Values:

- PreVal: 0xF0 (11110000)
- PostVal: 0xF5 (11110101)
- PreTm: 1300 ms
- PostTm: 0 ms

PBX_FlashOpen (Loop Start PBX Flash Open Signal)

Number: 0xC15CA029

Description: When flash hook transfer is enabled, the **PBX_FlashOpen** parameter defines the transition signal that is sent to drop a call. In an analog environment, the station drops the call with a flash hook. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

Values:

- PreVal: 0xFF (11111111)
- PostVal: 0xF5 (11110101)
- PreTm: 100 ms
- PostTm: 1000 ms

Net_FlashDrop (Loop Start Net Flash Drop Signal)

Number: 0xC15CA02A

Description: When flash hook transfer is enabled, the **Net_FlashDrop** parameter defines the transition signal that, when received, indicates that the network has dropped the call. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

Values:

- PreVal: 0xA0 (10100000)
- PostVal: 0xAA (10101010)
- PreTm: 100 ms
- PostTm: 1000 ms

PBX_Flash (Loop Start PBX Flash Signal)

Number: 0xC15CA031

Description: The **PBX_Flash** parameter defines the pulse signal used to initiate a blind transfer while the call is connected. For detailed information about pulse signals and their associated values, see Section 3.5.3, “Pulse Signal”, on page 36.

Values:

- OffVal: 0xFF (11111111)
- OnVal: 0xF5 (11110101)
- PreTm: 100 ms
- MinTm: 210 ms
- NomTm: 250 ms
- MaxTm: 280 ms
- PostTm: 100 ms

Loop Start Train Definition

Number: 0xC15CA032

Description: The Loop Start Train Definition defines a set of transitions from one signaling state to another in a predefined pattern (set of pulses). This parameter is used to define CAS signals required by a protocol. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- OffVal: 0xCC (11001100)
- OnVal: 0xC4 (11000100)
- pulseTmMin: 32
- pulseTmMax: 32
- pulseTmNom: 32
- preTm: 600
- interTmMin: 64
- interTmMax: 64
- interTmNom: 64
- postTm: 20
- digitCount: 12
- pulseCount: 10,0
- label: 1,1,2,2,3,3,4,4,5,5,6,6,7,7,8,8,9,9,11,#,12,*

Loop Start Sequence Definition

Number: 0xC15CA033

Description: The Loop Start Sequence Definition defines a set of trains for use with HDSI and Dialogic Integrated Series boards. This parameter is used to define CAS signals required by a protocol. For detailed information about sequence signals and their associated values, see Section 3.5.5, "Sequence Signal", on page 41.

Values:

- TrainSigId: 0xC15CA03
- preTm: 2
- interTmMin: 720
- interTmMax: 660
- interTmNom: 660
- postTm: 1600

6.16 [CAS] Parameters for T1 Ground Start Signals

The basis for the T1 ground start protocol is Channel Associated Signaling (CAS). The CAS component is responsible for the generation and detection of CAS signals on the phone network interface. The CAS T-1 ground start signals are defined in this section of the CONFIG file and assigned as variants in the [CHP] T-1 Protocol Variant Definition section of the CONFIG file. For details, see Section 6.21, "[CHP] T1 Protocol Variant Definitions", on page 138.

Note: The CAS signaling parameters should only be modified by experienced users if the default settings do not match what the line carrier or PBX is sending or expecting for the line protocol configuration running on the card.

CAS T1 ground start parameters include:

- PBX_Ground (Ground Start PBX Ground Signal)
- PBX_Answer (Ground Start PBX Answer Signal)
- PBX_Release (Ground Start PBX Release Signal)
- PBX_Drop (Ground Start PBX Drop Signal)
- Net_Ground (Ground Start Net Ground Signal)
- Net_Drop (Ground Start Net Drop Signal)
- Ring_On (Ground Start Ring On Signal)
- Ring_Off (Ground Start Ring Off Signal)
- PBX_FlashDrop (Ground Start PBX Flash Drop Signal)
- Net_FlashDrop (Ground Start Net Flash Drop Signal)
- Net_Answer (Ground Start Net Answer Signal)
- PBX_Flash (Ground Start PBX Flash Signal)

PBX_Ground (Ground Start PBX Ground Signal)

Number: 0xC15CA041

Description: The **PBX_Ground** parameter defines the transition signal sent by the station to make an outbound call, or to answer an incoming call (off-hook). From the station side, it is the GS-FXS transmitting a generic seize, and from the corresponding office or network side, it is the GS-FXO receiving a generic seize. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xF5 (11110101)
- PostVal: 0xF0 (11110000)
- PreTm: 100 ms
- PostTm: 100 ms

PBX_Answer (Ground Start PBX Answer Signal)

Number: 0xC15CA042

Description: The **PBX_Answer** parameter defines the transition signal sent by the station (GS-FXS) when an inbound call is answered. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xA5 (10100101)
- PostVal: 0xFF (11111111)
- PreTm: 100 ms
- PostTm: 100 ms

PBX_Release (Ground Start PBX Release Signal)

Number: 0xC15CA043

Description: The **PBX_Release** parameter defines the transition signal sent by the station (GS-FXS) to release an outbound call. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xF0 (11110000)
- PostVal: 0xF5 (11110101)
- PreTm: 50 ms
- PostTm: 50 ms

PBX_Drop (Ground Start PBX Drop Signal)

Number: 0xC15CA044

Description: The **PBX_Drop** parameter defines the transition signal used by the station (GS-FXS) to know that the network (GS-FXO) has dropped the call. The network generates a GS_Net_Drop signal. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xFF (11111111)
- PostVal: 0xF5 (11110101)
- PreTm: 50 ms
- PostTm: 100 ms

Net_Ground (Ground Start Net Ground Signal)

Number: 0xC15CA045

Description: The **Net_Ground** parameter defines the transition signal sent by the network (GS-FXO) to make an outbound call, or to answer an incoming call. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xAA (10101010)
- PostVal: 0xA0 (10100000)
- PreTm: 100 ms
- PostTm: 50 ms

Net_Drop (Ground Start Net Drop Signal)

Number: 0xC15CA046

Description: The **Net_Drop** parameter defines the transition signal that, when received by the station (GS-FXS), indicates that the network has dropped the call. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

Values:

- PreVal: 0xA0 (10100000)
- PostVal: 0xAA (10101010)
- PreTm: 100 ms
- PostTm: 50 ms

Ring_On (Ground Start Ring On Signal)

Number: 0xC15CA047

Description: The **Ring_On** parameter defines the transition signal generated by the network (GS-FXO) to ring the line, and indicate an inbound call is offered to the station (GS-FXS). This is the leading edge of the ring. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

Values:

- PreVal: 0xF5 (11110101)
- PostVal: 0xF0 (11110000)
- PreTm: 3900 ms
- PostTm: 50 ms

Ring_Off (Ground Start Ring Off Signal)

Number: 0xC15CA048

Description: The **Ring_Off** parameter defines the transition signal generated by the network (GS-FXO) to stop the “ring” on a line when a call is offered to the station (GS-FSO). This is the trailing edge of the ring. For detailed information about transition signals and their associated values, see Section 3.5.2, “Transition Signal”, on page 35.

This signal indicates that the network (GS-FXO) is “ringing” the line, and an inbound call is offered to the station (GS-FXS). This is the trailing edge of the ring.

Values:

- PreVal: 0xF0 (11110000)
- PostVal: 0xF5 (11110101)
- PreTm: 1300 ms
- PostTm: 0 ms

PBX_FlashDrop (Ground Start PBX Flash Drop Signal)

Number: 0xC15CA049

Description: When flash hook transfer is enabled, the **PBX_FlashDrop** parameter defines the transition signal that, when received, indicates that the PBX has dropped the call. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xFF (11111111)
- PostVal: 0xF5 (11110101)
- PreTm: 50 ms
- PostTm: 1000 ms

Net_FlashDrop (Ground Start Net Flash Drop Signal)

Number: 0xC15CA04A

Description: When flash hook transfer is enabled, the **Net_FlashDrop** parameter defines the transition signal that, when received, indicates that the network has dropped the call. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xA0 (10100000)
- PostVal: 0xAA (10101010)
- PreTm: 100 ms
- PostTm: 1000 ms

Net_Answer (Ground Start Net Answer Signal)

Number: 0xC15CA04B

Description: The **Net_Answer** parameter defines the transition signal that, when received, indicates that the network has answered an outbound call. For detailed information about transition signals and their associated values, see Section 3.5.2, "Transition Signal", on page 35.

Values:

- PreVal: 0xAA (10101010)
- PostVal: 0xA0 (10100000)
- PreTm: 100 ms
- PostTm: 50 ms

PBX_Flash (Ground Start PBX Flash Signal)

Number: 0xC15CA051

Description: The **PBX_Flash** parameter defines the pulse signal used by the station to initiate a blind transfer while the call is connected. For detailed information about pulse signals and their associated values, see Section 3.5.3, "Pulse Signal", on page 36.

Values:

- OffVal: 0xFF (11111111)
- OnVal: 0xF5 (11110101)
- PreTm: 100 ms
- MinTm: 210 ms
- NomTm: 250 ms
- MaxTm: 280 ms
- PostTm: 100 ms

6.17 [CAS] User-defined CAS and Tone Signal Parameters

The CAS component is responsible for the generation and detection of CAS signals on the phone network interface. The CAS user-defined and tone signals are defined in this section of the CONFIG file and assigned as variants in the [CHP] section of the CONFIG file. For details, see Section 6.21, "[CHP] T1 Protocol Variant Definitions", on page 138.

Note: The CAS signaling parameters should only be modified by experienced users if the default settings do not match what the line carrier or PBX is sending or expecting for the line protocol configuration running on the card.

Hookflash (Hook Flash)

Number: 0x9201

Description: The **Hookflash** parameter defines the pulse signal used to define hookflash times for HDSI and Dialogic Integrated Series boards. For detailed information about pulse signals and their associated values, see Section 3.5.3, "Pulse Signal", on page 36.

Note: Only modifications to the MinTm and MaxTm values are supported.

Values:

For HDSI Boards:
- OffVal: 0xFF (11111111)
- OnVal: 0xF5 (11110101)
- PreTm: 100 ms
- MinTm: Time (milliseconds)
- NomTm: 650 ms
- MaxTm: Time (milliseconds)
- PostTm: 100 ms

For Dialogic Integrated Series Boards:
- OffVal: 0xCC (11001100)
- OnVal: 0xC4 (11110100)
- PreTm: 100 ms
- MinTm: Time (milliseconds)
- NomTm: 250 ms
- MaxTm: Time (milliseconds)
- PostTm: 100 ms

6.18 [CAS] User-defined Signals for Selectable Rings Parameters

The CAS component is responsible for the generation and detection of CAS signals on the phone network interface. The CAS user-defined signals for selectable rings are defined in this section of the CONFIG file and assigned as variants in the [CHP] section of the CONFIG file. For details, see Section 6.21, "[CHP] T1 Protocol Variant Definitions", on page 138.

Note: The CAS signaling parameters should only be modified by experienced users if the default settings do not match what the line carrier or PBX is sending or expecting for the line protocol configuration running on the card.

The CAS user-defined signals for selectable rings parameters include:

- Net_RingOn (Ring Cadence On-time)
- Net_RingOff (Ring Cadence Off-time)

Net_RingOn (Ring Cadence On-time)

Number: 0x9110

Description: The **Net_RingOn** parameter is one of two pulse signal used to define the ring cadence for HDSI and Dialogic Integrated Series boards. **Net_RingOn** defines the on-time signal and **Net_RingOff** defines the off-time signal. For detailed information about pulse signals and their associated values, see Section 3.5.3, “Pulse Signal”, on page 36.

The value used for the MinTm, NomTm, and MaxTm (the same time must be used for all three values) is the total on-time duration of the pulse. If MinTm, NomTm, and MaxTm are set to 2000, then the total on-time duration of the pulse is 2000 ms. The PostTm value is not included in the total on-time duration since this value defines part of the off-time duration.

Note: Only modifications to the MinTm, NomTm, and MaxTm values are supported. When modifying these values, the same time must be used for all three values.

Values:

- OffVal: 0xA4 (10101010)
- OnVal: 0xAA (10100100)
- PreTm: 0 ms
- MinTm: 2000 ms
- NomTm: 2000 ms
- MaxTm: 2000 ms
- PostTm: 50 ms

Net_RingOff (Ring Cadence Off-time)

Number: 0x9111

Description: The **Net_RingOff** parameter is one of two pulse signal used to define the ring cadence for HDSI and Dialogic Integrated Series boards. **Net_RingOn** defines the on-time signal and **Net_RingOff** defines the off-time signal. For detailed information about pulse signals and their associated values, see Section 3.5.3, “Pulse Signal”, on page 36.

The total off-time duration of the pulse includes the **Net_RingOn** PostTm duration, the **NetRingOff** Pre-Pulse duration, and the **NetRingOff** on-time duration (defined by MinTm, NomTm, or MaxTm since all three times must be set to the same value). If the **Net_RingOn**

PostTm is set to 50 ms, the **Net_RingOff** PreTm duration is set to 50 ms, and the **Net_RingOff** on-time duration is set to 3900 ms, the total off-time duration is 4000 ms.

Note: Only modifications to the MinTm, NomTm, and MaxTm values are supported. When modifying these values, the same time must be used for all three values.

Values:

- OffVal: 0xA4 (10100100)
- OnVal: 0xA4 (10100100)
- PreTm: 50 ms
- MinTm: 3900 ms
- NomTm: 3900 ms
- MaxTm: 3900 ms
- PostTm: 0 ms

6.19 [CCS] Parameters

Common Channel Signaling (CCS) supports ISDN PRI out-of-band signaling utilizing the Q.931 signaling protocol for messaging. The parameters in the [CCS] and [CSS.x] sections of the CONFIG file define the number of CCS component instances created and configure the parameters associated with each CCS instance.

The CCS parameters include:

- INSTANCE_MAP (Instance Map)
- CCS_TMR_302 (Q.931 Timer 302)
- CCS_TMR_303 (Q.931 Timer 303)
- CCS_TMR_304 (Q.931 Timer 304)
- CCS_TMR_305 (Q.931 Timer 305)
- CCS_TMR_308 (Q.931 Timer 308)
- CCS_TMR_310 (Q.931 Timer 310)
- CCS_TMR_313 (Q.931 Timer 313)
- CCS_TEI_RETRY (TEI Retry Timer)
- CCS_TEI_STABILITY (TEI Stability Timer)
- SYMMETRICAL_LINK (Symmetrical Command Response Protocol)
- CCS_PROTOCOL_MODE (ISDN Protocol Mode)
- CCS_SWITCH_TYPE (Switch Type)
- L2_TRACE (Layer 2 Access Flag)

INSTANCE_MAP (Instance Map)

Number: 0x05

Description: The **INSTANCE_MAP** parameter is a bitmap that defines the number of CCS instances created. A CCS instance is created for each network interface that supports common channel signaling. The bitmap's least significant bit corresponds to the CCS instance associated with the first network interface on the board. The next least significant bit corresponds to the CCS instance associated with the second network interface on the board, and so on. If the bit associated with a network interface has a value of 1, then a CCS instance is created for that network interface. For example, a value of 0x5 (0101) means that CCS instances 1 and 3 are created allowing for common channel signaling on network interfaces 1 and 3.

Values: 0 to 0xffff

CCS_TMR_302 (Q.931 Timer 302)

Number: 0x14

Description: The **CCS_TMR_302** parameter is an ISDN Layer 3 timer. For exact timer definitions, refer to the Q.931 specification and the switch specifications.

Note: This parameter only applies to E1 boards.

Values:

- 0: Use the default value for the switch (15000 ms)
- -1: Disable the timer (has the same effect as setting the timer value to 0)
- n > 1: Timer value (milliseconds)

CCS_TMR_303 (Q.931 Timer 303)

Number: 0x0b

Description: The **CCS_TMR_303** parameter is an ISDN Layer 3 timer. For exact timer definitions, refer to the Q.931 specification and the switch specifications.

Values:

- 0: Use the default value for the switch (4000 ms)
- -1: Disable the timer (has the same effect as setting the timer value to 0)
- n > 1: Timer value (milliseconds)

CCS_TMR_304 (Q.931 Timer 304)

Number: 0x0c

Description: The **CCS_TMR_304** parameter is an ISDN Layer 3 timer. For exact timer definitions, refer to the Q.931 specification and the switch specifications.

Note: This parameter only applies to E1 boards.

Values:

- 0: Use the default value for the switch (30000 ms)
- -1: Disable the timer (has the same effect as setting the timer value to 0)
- n > 1: Timer value (milliseconds)

CCS_TMR_305 (Q.931 Timer 305)

Number: 0x0d

Description: The **CCS_TMR_305** parameter is an ISDN Layer 3 timer. For exact timer definitions, refer to the Q.931 specification and the switch specifications.

Values:

- 0: Use the default value for the switch (4000 ms for T1, 30000 ms for E1)
- -1: Disable the timer (has the same effect as setting the timer value to 0)
- n > 1: Timer value (milliseconds)

CCS_TMR_308 (Q.931 Timer 308)

Number: 0x0e

Description: The **CCS_TMR_308** parameter is an ISDN Layer 3 timer. For exact timer definitions, refer to the Q.931 specification and the switch specifications.

Values:

- 0: Use the default value for the switch (4000 ms)
- -1: Disable the timer (has the same effect as setting the timer value to 0)
- n > 1: Timer value (milliseconds)

CCS_TMR_310 (Q.931 Timer 310)

Number: 0x0f

Description: The **CCS_TMR_310** parameter is an ISDN Layer 3 timer. For exact timer definitions, refer to the Q.931 specification and the switch specifications.

Values:

- 0: Use the default value for the switch (10000 ms)
- -1: Disable the timer (has the same effect as setting the timer value to 0)
- n > 1: Timer value (milliseconds)

CCS_TMR_313 (Q.931 Timer 313)

Number: 0x10

Description: The **CCS_TMR_313** parameter is an ISDN Layer 3 timer. For exact timer definitions, refer to the Q.931 specification and the switch specifications.

Values:

- 0: Use the default value for the switch (4000 ms)
- -1: Disable the timer (has the same effect as setting the timer value to 0)
- n > 1: Timer value (milliseconds)

CCS_TEI_RETRY (TEI Retry Timer)

Number: 0x15

Description: The **CCS_TEI_RETRY** parameter defines the maximum amount of time that the data link remains in state 4 (TEI_ASSIGNED) before transitioning to state 5 (TEI_WAIT_ESTABLISH).

Values: Time (milliseconds)

CCS_TEI_STABILITY (TEI Stability Timer)

Number: 0x16

Description: The **CCS_TEI_STABILITY** parameter defines the minimum transition time between data link state 4 (TEI_ASSIGNED) and data link state 5 (TEI_WAIT_ESTABLISH).

Values: 0 to 100,000 (milliseconds)

SYMMETRICAL_LINK (Symmetrical Command Response Protocol)

Number: 0x13

Description: The **SYMMETRICAL_LINK** parameter enables or disables symmetrical data link operations.

Values:

- 0: Disable symmetrical data link operations
- 1: Enable symmetrical data link operations

CCS_PROTOCOL_MODE (ISDN Protocol Mode)

Number: 0x17

Description: The **CCS_PROTOCOL_MODE** parameter sets the network user-side protocol. User-side protocol is also known as TE (terminal emulation) protocol and Network-side protocol is also known as NT (network termination) protocol. This parameter also can be used to configure Q.SIG Master/Slave.

Note: Master/Slave mode pertains to Q.SIG protocols only.

Note: With the exception of the Q.SIG protocol (where the User-side and Network-side protocols are symmetrical), using the CSS_PROTOCOL_MODE parameter to configure a Network-side protocol is supported for back-to-back testing purposes only. The Network-side firmware is not fully qualified for operation in a deployment environment.

Values:

- 0: User or Slave Mode (Q.SIG)
- 1: Network or Master Mode (Q.SIG)

CCS_SWITCH_TYPE (Switch Type)

Number: 0x07

Description: The **CCS_SWITCH_TYPE** parameter defines the network switch type.

Values:

- 1: 4ess
- 2: 5ess
- 3: dms
- 4: ntt
- 6: dass2
- 7: net5
- 10: qsige1
- 11: qsigt1
- 12: ni2
- 13: dpnss

L2_TRACE (Layer 2 Access Flag)

Number: 0x09

Description: The **L2_TRACE** parameter is the ISDN Layer 2 access flag. When Layer 2 (Data Link layer) access is disabled, ISDN Link Access Protocol for the D channel (LAPD) functionality is obtained by accessing ISDN Call Control and Layer 3 (Network layer). When Layer 2 access is enabled, call control is no longer supported for the channels on this line and ISDN LAPD functionality is obtained by accessing Layer 2 directly.

Values:

- 0: Disable Layer 2 access
- 1: Enable Layer 2 access

6.20 [CHP] Parameters

The Channel Protocol (CHP) component implements the telephony communication protocol that is used on the network interface. The CHP component parameters include:

- R4Compatibility (R4 Compatibility Flag)
- InitialChanState (Initial Channel State)
- DisableBlock (Disable Block)

R4Compatibility (R4 Compatibility Flag)

Number: 0x1310

Description: The **R4Compatibility** parameter enables or disables R4 (Global Call) compatibility features. This parameter also enables retrieval of DNIS and ANI information in the offered call state.

Values:

- 0: Disable R4 compatibility [default]
- 1: Enable R4 compatibility
- 2: Disable R4 compatibility

InitialChanState (Initial Channel State)

Number: 0x1311

Description: The **InitialChanState** parameter defines the initial B channel state (CHP channel state) at the end of system initialization. The initial state of the ISDN B channel is either InService or OutOfService. Once the board is initialized, this initial state will be set on all channels of the board until a user application is invoked and explicitly modifies the state of the channel.

Values:

- 1: InService
- 2: OutOfService

Guidelines: This parameter must be set to OutOfService for ISDN protocols.

DisableBlock (Disable Block)

Number: 0x1312

Description: The **DisableBlock** parameter defines whether or not a blocking pattern (message) is sent on a channel when the channel is in the OutofService state. When **DisableBlock** is disabled, no pattern is sent (the switch will not present calls to the B channel).

When **DisableBlock** is enabled and a channel is in the InService state (**InitialChanState**=1), the protocol will send a non-blocking pattern on the channel (the switch will present calls to the B channel). When **DisableBlock** is enabled and a channel is in the OutofService state (**InitialChanState**=2), the protocol will send a blocking pattern on the channel (the switch will present calls to the B channel but these calls will be abandoned by the switch since the application will not respond to the call).

Values:

- 0: Disable blocking
- 1: Enable blocking

6.21 [CHP] T1 Protocol Variant Definitions

The CHP parameters define the line configurations that will be used by each network interface on the Intel telecom board. Within the [CHP] section of the CONFIG file, different possible T1 protocol variants (protocol configuration settings) can be implemented and are defined using the `Variant Define n` command. The `Variant Define n` is later set for each network channel in the [TSC] defineBSet section of the CONFIG file.

Note: Not all of the following variants exist in all CONFIG files. For PDK parameters in particular, refer to the *Global Call County Dependent Parameters (CDP) Configuration Guide*.

For a detailed description of the `Variant Define n` command, see Section 3.6, "[CHP] Section", on page 43. For a detailed description of the `defineBSet` command, also see Section 3.7, "[TSC] Section", on page 45.

Although T-1 signals (E&M wink, loop start and ground start) are assigned as part of the T-1 protocol variant definitions, these signals are defined in the [CAS] section of the CONFIG file. For detailed information about specific T-1 signal definitions, see the following:

- Section 6.14, "[CAS] Parameters for T1 E&M Signals", on page 117
- Section 6.15, "[CAS] Parameters for T1 Loop Start Signals", on page 119
- Section 6.16, "[CAS] Parameters for T1 Ground Start Signals", on page 124

Unless otherwise noted, T-1 Protocol Variant parameters apply to all signal types: E&M wink, loop start, and ground start. Parameters that only apply to a specific signal type are prefaced with the applicable signal type. For example, the parameter **EM_offhook** only applies to E&M wink signaling and the parameter **ProtocolType** applies to E&M wink, loop start, and ground start signaling.

The T-1 Protocol Variant parameters include:

- ProtocolType (Protocol Type)
- Wink (Wink Flag)
- StartTimeout (Start Timeout)
- FarEndAnswer (Far End Answer)
- AnswerTimeout (Answer Timeout)
- ReconnectTimeout (Reconnect Timeout)
- DisconnectTimeout (Disconnect Timeout)
- InterCallDelay (Inter-call Delay)
- Dial (Outbound Dialing Flag)
- DialFormat (Dial Digits Format)
- ANI (ANI Flag)
- ANIFormat (ANI Digit Format)
- ANICount (ANI Digit Count)
- DNIS (DNIS Flag)
- DNISFormat (DNIS Digit Format)

- DNISCount (DNIS Digit Count)
- PreDigitTimeout (Pre-digit Timeout)
- InterDigitTimeout (Inter-digit Timeout)
- CallProgress (Call Progress Detection)
- CaRingingSet (Ringing Signal)
- CaBusySet (Busy Signal)
- CaSitSet (SIT Signal)
- CaFaxSet (Fax Signal)
- CaPvdId (Voice Detection Signal)
- CaPamdId (Answering Machine Signal)
- CaSignalTimeout (Signal Timeout)
- CaAnswerTimeout (Answer Timeout)
- CaPvdTimeout (Voice Detection Timeout)
- DialToneId (Dial Tone Signal)
- CaDialTimeout (Dial Timeout)
- BlindTransfer (Blind Transfer)
- BtDialToneId (DTD Signal)
- BtStartTimeout (DTD Timeout)
- BtAddressDef (Address Definition)
- BtOrigFormat (Originator Address Digits)
- BtDestFormat (Destination Address Digits)
- BtCancelDigitsFormat (Cancel Digits Format)
- BtCancelFlashCount (Cancel Flash Count)
- BtCancelInterFlashDuration (Cancel Inter-flash Duration)
- BtCancelDigits (Cancel Digits)
- BtDestSuffix (Destination Suffix Digits)
- BtDestPrefix (Destination Prefix Digits)
- BtOrigPrefix (Originator Prefix Digits)
- BtOrigSuffix (Originator Suffix Digits)
- PolarityDetection (Polarity Flag)
- EM_Offhook (E&M Off-hook Signal)
- EM_Onhook (E&M On-hook Signal)
- EM_FlashOnhook (E&M Flash On-hook Signal)
- EM_Wink (E&M Wink Signal)
- EM_Flash (E&M Flash Signal)
- LS_PBX_Open (Loop Start PBX Open)
- LS_PBX_FlashOpen (Loop Start PBX Flash Open Signal)
- LS_PBX_Close (Loop Start PBX Close Signal)

- LS_PBX_Flash (Loop Start PBX Flash Signal)
- LS_Net_Answer (Loop Start Net Answer Signal)
- LS_Net_Drop (Loop Start Net Drop Signal)
- LS_Net_FlashDrop (Loop Start Net Flash Drop Signal)
- LS_Net_Abandon (Loop Start Net Abandon)
- LS_Net_RingOn (Loop Start Net Ring On Signal)
- LS_Net_RingOff (Loop Start Net Ring Off)
- GS_PBX_Ground (Ground Start PBX Ground Signal)
- GS_PBX_Answer (Ground Start PBX Answer Signal)
- GS_PBX_Release (Ground Start PBX Release Signal)
- GS_PBX_Drop (Ground Start PBX Drop Signal)
- GS_PBX_FlashDrop (Ground Start PBX Flash Drop Signal)
- GS_PBX_Flash (Ground Start PBX Flash Signal)
- GS_Net_Ground (Ground Start Net Ground Signal)
- GS_Net_Drop (Ground Start Net Drop Signal)
- GS_Net_FlashDrop (Ground Start Flash Drop Signal)
- GS_Net_RingOn (Ground Start Net Ring On Signal)
- GS_Net_RingOff (Ground Start Net Ring Off Signal)

ProtocolType (Protocol Type)

Description: The **ProtocolType** parameter defines the type of T1 protocol used on a channel.

Note: The **ProtocolType** parameter is also used when defining ISDN protocol variants.

Values:

- 1: E&M wink
- 2: Loop start-FXS
- 3: Ground start-FXS
- 4: Loop start-FXO (This value is not supported.)
- 5: Ground start-FXO (Supported by HDSI and Dialogic Integrated Series boards only.)

Wink (Wink Flag)

Description: The **Wink** parameter enables wink detection for outbound calls and wink generation for inbound calls. This parameter is enabled only when using T-1 E&M wink protocols. For all other protocols, this parameter is disabled.

If wink detection (generation) is enabled, up to three winks are supported in a sequence. Examples of CONFIG file settings for **ANI**, **Wink**, **DNIS**, and **R4Compatibility** parameters for 1, 2, and 3 wink sequences are as follows:

- 1 Wink Example: Seize (Wink) DNIS (Answer) ANI

```
Variant ANI    2    ! No=0, Pre=1, Post=2
Variant Wink   1    ! 1 wink sequence
Variant DNIS   y    ! Enable DNIS collection
SetParm=0x1310,1    ! R4 Compatibility Flag
                    ! 0=default, 1=enable, 2=disable
```

In the 1 Wink example, ANI information is collected after the call is answered, there is one wink in the sequence, DNIS information is collected, and the R4 Compatibility Flag is turned on.

- 2 Wink Example: Seize (Wink) ANI (DNIS) (Wink) Answer

```
Variant ANI    1    ! No=0, Pre=1, Post=2
Variant Wink   2    ! 1 wink sequence
Variant DNIS   y    ! Enable DNIS collection
SetParm=0x1310,1    ! R4 Compatibility Flag
                    ! 0=default, 1=enable, 2=disable
```

In the 2 Wink example, ANI information is collected before the call is answered, there are two winks in the sequence, DNIS information is collected, and the R4 Compatibility Flag is turned on.

- 3 Wink Example: Seize (Wink) DNIS (Wink) ANI (Wink) Answer

```
Variant ANI    1    ! No=0, Pre=1, Post=2
Variant Wink   3    ! 1 wink sequence
Variant DNIS   y    ! Enable DNIS collection
SetParm=0x1310,2    ! R4 Compatibility Flag
                    ! 0=default, 1=enable, 2=disable
```

In the 3 Wink example, ANI information is collected before the call is answered, there are three winks in the sequence, DNIS information is collected, and the R4 Compatibility Flag is turned off.

Values:

- 0 or n: Disable wink
- 1 or y: Enable 2 winks in the sequence
- 2: Enable 1 wink in the sequence
- 3: Enable 3 winks in the sequence

StartTimeout (Start Timeout)

Description: The **StartTimeout** parameter is defined differently depending on the T-1 protocol used. Depending on the value of the **ProtocolType** parameter, the **StartTimeout** parameter is defined as follows:

E&M wink
: The amount of time that a call originator will wait for a wink once it has seized the line for an outbound call. The outbound call will fail if this time is exceeded.

Ground start
: The amount of time that a call originator will wait for the dial tone signal once it has seized the line for an outbound call. The outbound call will fail if this time is exceeded.

Loop start
: The call originator will wait the specified amount of time after seizing the line for an outbound call, and then proceed to sending digits (that is, can be used to for dial tone before dialing).

Values: n > 0 (milliseconds)

FarEndAnswer (Far End Answer)

Description: The **FarEndAnswer** parameter defines whether the network will provide far end (remote) answer signaling. This parameter is enabled when using T1 loop start protocols only. For all other T-1 protocols, this parameter is disabled.

Values:

- y: Enable far end answer support
- n: Disable far end answer support

AnswerTimeout (Answer Timeout)

Description: The **AnswerTimeout** parameter defines the maximum amount of time allowed to answer a call once a remote party has sent its last wink. If the time is exceeded, the call fails. Otherwise, the timer is reset once the call is answered. This parameter is enabled only when using T1 E&M wink protocols. For all other T1 protocols, this parameter is disabled.

Values: n > 0 (milliseconds)

ReconnectTimeout (Reconnect Timeout)

Description: The **ReconnectTimeout** parameter defines the maximum amount of time allowed between a local disconnect (*MsgDropCall*) and a reconnect (*MSgReconnectCall*).

Values: n > 0 (milliseconds)

DisconnectTimeout (Disconnect Timeout)

Description: The **DisconnectTimeout** parameter defines the time before a remote drop is considered to be a disconnect. If *MsgDropCall* is followed by a *MsgReconnectCall* within this time period, then the call will be reconnected and remain in the connected state.

Note: The **DisconnectTimeout** parameter is also used when defining ISDN protocol variants.

Values: n > 0 (milliseconds)

Guidelines: For DNM160TEC and DMT160TEC boards, when there are 16 or more trunks in one NFAS group and calls are made on every channel at a very fast rate, the firmware may start missing calls. In this case, increase the value of the **DisconnectTimeout** parameter to at least 3000 milliseconds.

InterCallDelay (Inter-call Delay)

Description: The **InterCallDelay** parameter defines the minimum amount of time between outbound calls. This is the time the firmware will wait after a call is dropped and before making another call from the same channel.

Note: The **InterCallDelay** parameter is also used when defining ISDN Protocol variants, including E-1.

Values: n > 0 (milliseconds)

Dial (Outbound Dialing Flag)

Description: The **Dial** parameter enables or disables outbound dialing.

Values:

- y: Enable outbound dialing
- n: Disable outbound dialing

DialFormat (Dial Digits Format)

Description: The **DialFormat** parameter defines the format of the dial digits.

Values:

- 1: Dual Tone Multi-Frequency (DTMF)
- 2: Multi-Frequency (MF)
- 3: Dial Pulse (DP)

ANI (ANI Flag)

Description: The **ANI** parameter enables or disables the collection of ANI data. When the parameter is enabled, it also defines when the data is collected, before or after the call is answered.

Values:

- 0: Disable ANI data collection
- 1: Enable ANI data collection - pre-answer
- 2: Enable ANI data collection - post-answer

ANIFormat (ANI Digit Format)

Description: The **ANIFormat** parameter defines the format of the ANI digits.

Values:

- 1: Dual Tone Multi-Frequency (DTFM)
- 2: Multi-Frequency (MF)
- 3: Dial Pulse (DP)
- 4: Frequency Shift Keying, United States (FSK_US)
- 5: Frequency Shift Keying, Japan (FSK_JP)
- 6: Frequency Shift Keying, United Kingdom (FSK_UK - This value is not supported.)

ANICount (ANI Digit Count)

Description: The **ANICount** parameter defines the number of ANI digits to collect from the incoming call. If the parameter is set to 0, then digit collection will stop after the time-out period set by the **PreDigitTimeout** and **InterDigitTimeout** parameters defined in the [CHP] `Variant Define n` section of the CONFIG file. For details see:

- Section 3.6, "[CHP] Section", on page 43
- "InterDigitTimeout (Inter-digit Timeout)", on page 145

- “PreDigitTimeout (Pre-digit Timeout)”, on page 144

Values:

- 0: Collect all the digits provided
- n: Number of digits to collect

DNIS (DNIS Flag)

Description: The **DNIS** parameter enables or disables DNIS data collection for inbound calls.

Values:

- y: Enable DNIS collection
- n: Disable DNIS collection

DNISFormat (DNIS Digit Format)

Description: The **DNISFormat** parameter defines the format of the DNIS digits.

Values:

- 1: Dual Tone Multi-Frequency (DTMF)
- 2: Multi-Frequency (MF)
- 3: Dial Pulse (DP)

DNISCount (DNIS Digit Count)

Description: The **DNISCount** parameter defines the number of DNIS digits to collect from the incoming call. If the parameter is set to 0, then digit collection will stop after the time-out period set by the **PreDigitTimeout** and **InterDigitTimeout** parameters defined in the [CHP] `Variant Define n` section of the CONFIG file. For details see:

- Section 3.6, “[CHP] Section”, on page 43
- “InterDigitTimeout (Inter-digit Timeout)”, on page 145
- “PreDigitTimeout (Pre-digit Timeout)”, on page 144

Values:

- 0: Collect all the digits provided
- n: Number of digits to collect

PreDigitTimeout (Pre-digit Timeout)

Description: The **PreDigitTimeout** parameter defines the maximum amount of time that the protocol will wait to receive digits once a call has been initiated. E&M wink-start protocols start this time from the end of the wink.

Values: n > 0 (the value must be a multiple of 10 ms)

InterDigitTimeout (Inter-digit Timeout)

Description: The **InterDigitTimeout** parameter defines the maximum amount of time between digits. If a digit is not followed by another within this time limit, then digit collection is terminated.

Values: n > 0 (the value must be a multiple of 10 ms)

CallProgress (Call Progress Detection)

Description: The **CallProgress** parameter enables or disables call progress detection for call setup on outbound calls.

Note: The **CallProgress** parameter is also used when defining ISDN protocol variants.

Values:

- y: Enable call progress detection
- n: Disable call progress detection

CaRingingSet (Ringing Signal)

Description: The **CaRingingSet** parameter defines the signal set used to detect ringing for call progress analysis. The **CaRingingSet** parameter is also used when defining ISDN protocol variants.

Notes: *1.* Modification of the **CaRingingSet** parameter is not supported.

2. The **CaRingingSet** parameter is also used when defining ISDN protocol variants.

Values: 0x024940

CaBusySet (Busy Signal)

Description: The **CaBusySet** parameter defines the signal set used to detect busy for call progress analysis.

Notes: *1.* Modification of the **CaBusySet** parameter is not supported.

2. The **CaBusySet** parameter is also used when defining ISDN protocol variants.

Values: 0x004DE0

CaSitSet (SIT Signal)

Description: The **CaSitSet** parameter defines the signal set used to detect Standard Information Tones (SIT) for call progress analysis.

Notes: *1.* Modification of the **CaSitSet** parameter is not supported.

2. The **CaSitSet** parameter is also used when defining ISDN protocol variants.

Values: 0x02F240

CaFaxSet (Fax Signal)

Description: The **CaFaxSet** parameter defines the signal set used to detect fax tones for call progress analysis.

Notes: *1.* Modification of the **CaFaxSet** parameter is not supported.

2. The **CaFaxSet** parameter is also used when defining ISDN protocol variants.

Values: 0x014B80

CaPvdId (Voice Detection Signal)

Description: The **CaPvdId** parameter defines the signal to use for Positive Voice Detection (PVD) for call progress analysis.

Notes: *1.* Modification of the **CaPvdId** parameter is not supported.

2. The **CaPvdId** parameter is also used when defining ISDN protocol variants.

Values: 0x01F4C1

CaPamdId (Answering Machine Signal)

Description: The **CaPamdId** parameter defines the signal to use for Positive Answering Machine Detection (PAMD) for call progress analysis.

Notes: *1.* Modification of the **CaPamdId** parameter is not supported.

2. The **CaPamdId** parameter is also used when defining ISDN protocol variants.

Values: 0x01A041

CaSignalTimeout (Signal Timeout)

Description: The **CaSignalTimeout** parameter defines the maximum amount of time to wait to detect a call progress tone from one of the call analysis signal sets. For T1 loop start and ground start protocols, if this time is exceeded, then the outbound call will fail with the reason being NoAnswer.

Note: The **CaSignalTimeout** parameter is also used when defining ISDN protocol variants.

Values: n > 0 (the value must be a multiple of 10 ms)

CaAnswerTimeout (Answer Timeout)

Description: The **CaAnswerTimeout** parameter defines the maximum amount of time (in milliseconds) that call analysis will wait for ringback to stop. This is equivalent to the number of rings. For T1 loop start and ground start protocols, if this time is exceeded, then the outbound call will fail with the reason being NoAnswer.

Note: The **CaAnswerTimeout** parameter is also used when defining ISDN protocol variants.

Values: n > 0 (the value must be a multiple of 10 ms)

CaPvdTimeout (Voice Detection Timeout)

Description: The **CaPvdTimeout** parameter defines the maximum amount of time that call analysis will wait to detect positive answering machine detection (PAMD) or positive voice detection (PVD) once ringback has ceased. If this time is exceeded, then connection type will be reported as "Unknown", that is, not fax, PAMD, or PVD.

Note: The **CaPvdTimeout** parameter is also used when defining ISDN protocol variants.

Values: n > 0 (the value is expressed in multiples of 10 milliseconds. For example, a value of 200 equals 2000 milliseconds, or 2 seconds)

- For analog boards: default = 800
- For digital boards: default = 400

DialToneId (Dial Tone Signal)

Description: The **DialToneId** parameter defines the signal to use for dial tone detection. This parameter is used for loop start protocols. If this is set to 0 (Null), then dial tone detection is disabled.

Values:

- 0: Disable dial tone detection
- 0x00A261: Enable dial tone detection

CaDialTimeout (Dial Timeout)

Description: The **CaDialTimeout** parameter defines the maximum amount of time that call analysis will wait to detect a tone, for example, busy, SIT tones, and ring back.

Values: n > 0 (the value must be a multiple of 10 ms)

BlindTransfer (Blind Transfer)

Description: The **BlindTransfer** parameter enables or disables blind transfer.

Values:

- 0: Disable blind transfer
- 1: Enable blind transfer

BtDialToneId (DTD Signal)

Description: The **BtDialToneId** parameter works together with the **DialToneId** parameter and defines the signal used for dial tone detection in a blind transfer. If **BtDialToneId** is 0, then the protocol will wait for a time period (**BtStartTimeout**) before sending digits after generating a flash.

Values:

- 0: Pause (the pause time is equal to the value of **BtStartTimeout**)
- n: Signal parameter number

BtStartTimeout (DTD Timeout)

Description: The **BtStartTimeout** parameter is used only when the **BtDialToneId** parameter is set to a value of 0 (zero). This parameter defines the maximum amount of time that the protocol will wait for detecting dial tone after a flash has been generated. Once the **BtStartTimeout** value has been reached, a transfer failure will occur with the reason being ProtocolError, and the call will return to the connected state.

If the **BtDialToneId** parameter is set to 0 (zero), **BtStartTimeout** is the time period that the protocol will wait after a flash has been generated before sending digits

Values: n > 0 (milliseconds)

BtAddressDef (Address Definition)

Description: The **BtAddressDef** parameter defines what addresses will be sent on a blind transfer, and the order in which they will be sent. Addresses are analogous to phone numbers dialed (destination = DNIS), or dialing from (origination = ANI).

Values:

- 1: None
- 2: Destination
- 3: Origination
- 4: Destination, Origination
- 5: Origination, Destination

BtOrigFormat (Originator Address Digits)

Description: The **BtOrigFormat** parameter defines the format of the originator address digits in a blind transfer. The address is analogous to ANI (caller ID) information.

Values:

- 1: Dual Tone Multi-Frequency (DTMF)
- 2: Multi-Frequency (MF)
- 3: Dial Pulse (DP)

BtDestFormat (Destination Address Digits)

Description: The **BtDestFormat** parameter defines the format of destination address digits in a blind transfer. The address is analogous to DNIS information.

Values:

- 1: Dual Tone Multi-Frequency (DTMF)
- 2: Multi-Frequency (MF)
- 3: Dial Pulse (DP)

BtCancelDigitsFormat (Cancel Digits Format)

Description: The **BtCancelDigitsFormat** parameter defines the format of Cancel digits in a blind transfer.

Values:

- 1: Dual Tone Multi-Frequency (DTMF)
- 2: Multi-Frequency (MF)
- 3: Dial Pulse (DP)

BtCancelFlashCount (Cancel Flash Count)

Description: The **BtCancelFlashCount** parameter defines the number of flashes to be sent to cancel or abort a transfer.

Values: n > 0

BtCancelInterFlashDuration (Cancel Inter-flash Duration)

Description: The **BtCancelInterFlashDuration** parameter defines the time between flashes for canceling or aborting a transferred call.

Values: n > 0 (milliseconds)

BtCancelDigits (Cancel Digits)

Description: The **BtCancelDigits** parameter defines the digits to dial after the flash sequence to cancel or abort a call transfer. For example, *69.

Values: Digits

BtDestSuffix (Destination Suffix Digits)

Description: The **BtDestSuffix** parameter defines the digits to be dialed immediately following the destination address (dialed number/DNIS) in a blind transfer. For example, *34.

Values: Digits

BtDestPrefix (Destination Prefix Digits)

Description: The **BtDestPrefix** parameter defines the digits to be dialed immediately before the destination address (dialed number/DNIS) in a blind transfer. For example, *54

Values: Digits

BtOrigPrefix (Originator Prefix Digits)

Description: The **BtOrigPrefix** parameter defines the digits to be dialed immediately before the originator address (calling number/ANI) in a blind transfer. For example, *94.

Values: Digits

BtOrigSuffix (Originator Suffix Digits)

Description: The **BtOrigSuffix** parameter defines the digits to be dialed immediately following the originator address (calling number/ANI) in a blind transfer. For example, *64.

Values: Digits

PolarityDetection (Polarity Flag)

Description: The **PolarityDetection** parameter defines whether the CO reverses battery polarity as the first step before sending the call ring. In Japan, the CO (Nippon Telephone and Telegraph, NTT) reverses the loop polarity prior to sending a call. When **PolarityDetection** parameter is enabled, the polarity reversal sent from the CO in NTT is used when detecting the incoming call.

Values:

- 0: Disable polarity reversal (normal polarity)
- 1: Enable polarity reversal (reverse polarity)

EM_Offhook (E&M Off-hook Signal)

Description: The **EM_Offhook** parameter defines the T-1 E&M wink off-hook CAS transition signal id. This signal is sent to make an outbound call, or to answer an incoming call. Receiving this signal indicates that the network is offering a call.

Values: 0xC15CA001

EM_Onhook (E&M On-hook Signal)

Description: The **EM_Onhook** parameter defines the T-1 E&M wink on-hook CAS transition signal id. This signal is sent to drop a call.

Values: 0xC15CA002

EM_FlashOnhook (E&M Flash On-hook Signal)

Description: The **EM_FlashOnhook** parameter defines the T-1 E&M wink on-hook CAS transition signal used when flash hook transfer (blind transfer) is enabled. It defines the transition signal from an off-hook state to a flash on-hook state during a blind transfer. This signal is sent to drop a call.

Values: 0xC15CA003

EM_Wink (E&M Wink Signal)

Description: The **EM_Wink** parameter defines the T-1 E&M wink CAS pulse signal ID. This signal is used as part of the inbound call setup for wink-start E&M protocols. The signal is sent to tell the far end to proceed with the call in response to EM_Offhook.

Values: 0xC15CA011

EM_Flash (E&M Flash Signal)

Description: The **EM_Flash** parameter defines the T-1 E&M wink flash CAS pulse signal ID. This signal is used to initiate a blind transfer while the call is connected.

Values: 0xC15CA012

LS_PBX_Open (Loop Start PBX Open)

Description: The **LS_PBX_Open** parameter defines the T-1 loop start PBX Open CAS transition signal ID. This signal is sent to drop a call.

Values: 0xC15CA021

LS_PBX_FlashOpen (Loop Start PBX Flash Open Signal)

Description: The **LS_PBX_FlashOpen** parameter defines the T-1 loop start PBX Open CAS transition signal to use when flash hook transfer is enabled. This signal is sent to drop a call.

Values: 0xC15CA029

LS_PBX_Close (Loop Start PBX Close Signal)

Description: The **LS_PBX_Close** parameter defines the T-1 loop start PBX Close CAS transition signal ID. This signal is sent to make an outbound call, or to answer an incoming call.

Values: 0xC15CA022

LS_PBX_Flash (Loop Start PBX Flash Signal)

Description: The **LS_PBX_Flash** parameter defines the T-1 loop start PBX flash CAS pulse signal ID. This signal is used to initiate a blind transfer while the call is connected.

Values: 0xC15CA031

LS_Net_Answer (Loop Start Net Answer Signal)

Description: The **LS_Net_Answer** parameter defines the T-1 loop start Net Answer CAS transition signal ID. Receiving this signal indicates that the network has answered an outbound call.

Values: 0xC15CA023

LS_Net_Drop (Loop Start Net Drop Signal)

Description: The **LS_Net_Drop** parameter defines the T-1 loop start Net Drop CAS transition signal ID. Receiving this signal indicates that the network has dropped the call.

Values: 0xC15CA024

LS_Net_FlashDrop (Loop Start Net Flash Drop Signal)

Description: The **LS_Net_FlashDrop** parameter defines the T-1 loop start Net Drop CAS transition signal to use when flash hook transfer is enabled. Receiving this signal indicates that the network has dropped the call.

Values: 0xC15CA02A

LS_Net_Abandon (Loop Start Net Abandon)

Description: The **LS_Net_Abandon** parameter defines the T-1 loop start Net Abandon CAS transition signal ID. Receiving this signal indicates that the network has dropped an offered call, that is, ringing has stopped.

Values: 0xC15CA025

LS_Net_RingOn (Loop Start Net Ring On Signal)

Description: The **LS_Net_RingOn** parameter defines the T-1 loop start Net Ring On CAS transition signal ID. Receiving this signal indicates that the network is ringing the line, and an inbound call is offered. This is the leading edge of the ring.

Values: 0xC15CA026

LS_Net_RingOff (Loop Start Net Ring Off)

Description: The **LS_Net_RingOff** parameter defines the T-1 loop start Net Ring Off CAS transition signal ID. Receiving this signal indicates that the network is "ringing" the line, and an inbound call is offered. This is the trailing edge of the ring.

Values: 0xC15CA027

GS_PBX_Ground (Ground Start PBX Ground Signal)

Description: The **GS_PBX_Ground** parameter defines the T-1 ground start PBX Ground CAS transition signal ID. This signal is sent by the station to make an outbound call, or to answer an incoming call (off-hook). From the station side, it is the GS-FXS transmitting a generic seize, and from the corresponding office or network side, it is the GS-FXO receiving a generic seize.

Values: 0xC15CA041

GS_PBX_Answer (Ground Start PBX Answer Signal)

Description: The **GS_PBX_Answer** parameter defines the T-1 ground start PBX Answer CAS transition signal ID. This signal is sent by the station (GS-FXS) when an inbound call is answered.

Values: 0xC15CA042

GS_PBX_Release (Ground Start PBX Release Signal)

Description: The **GS_PBX_Release** parameter defines the T-1 ground start PBX Release CAS transition signal ID. This signal is sent by the station (GS-FXS) to release an outbound call.

Values: 0xC15CA043

GS_PBX_Drop (Ground Start PBX Drop Signal)

Description: The **GS_PBX_Drop** parameter defines the T-1 ground start PBX Drop CAS transition signal ID. This signal is used by the station (GS-FXS) to know that the network (GS-FXO) has dropped the call. The network generates a GS_Net_Drop signal.

Values: 0xC15CA044

GS_PBX_FlashDrop (Ground Start PBX Flash Drop Signal)

Description: The **GS_PBX_FlashDrop** parameter defines the T-1 ground start PBX Drop CAS transition signal ID. This signal is sent to drop a call when flash hook transfer is enabled.

Values: 0xC15CA049

GS_PBX_Flash (Ground Start PBX Flash Signal)

Description: The **GS_PBX_Flash** parameter defines the T-1 ground start PBX flash CAS pulse signal ID. This signal is used by the station to initiate a blind transfer while the call is connected.

Values: 0xC15CA051

GS_Net_Ground (Ground Start Net Ground Signal)

Description: The **GS_Net_Ground** parameter defines the T-1 ground start Net Ground CAS transition signal ID. Receiving this signal indicates that the network has answered an outbound call, or that the network is offering an inbound call.

Values: 0xC15CA045

GS_Net_Drop (Ground Start Net Drop Signal)

Description: The **GS_Net_Drop** parameter defines the T-1 loop start Net Drop CAS transition signal ID. Receiving this signal indicates that the network has dropped the call.

Values: 0xC15CA046

GS_Net_FlashDrop (Ground Start Flash Drop Signal)

Description: The **GS_Net_FlashDrop** parameter defines the T-1 ground start Net Drop CAS transition signal to use when flash hook transfer is enabled. Receiving this signal indicates that the network has dropped the call.

Values: 0xC15CA04A

GS_Net_RingOn (Ground Start Net Ring On Signal)

Description: The **GS_Net_RingOn** parameter defines the T-1 ground start Net Ring On CAS transition signal ID. Receiving this signal indicates that the network (GS-FXO) is “ringing” the line, and an inbound call is offered to the station (GS-FXS). This is the leading edge of the ring.

Values: 0xC15CA047

GS_Net_RingOff (Ground Start Net Ring Off Signal)

Description: The **GS_Net_RingOff** parameter defines the T-1 ground start Net Ring Off CAS transition signal ID. The signal generated by the network (GS-FXO) to stop the “ring” on a line when a call is offered to the station (GS_FSO). This is the trailing edge of the ring.

Values: 0xC15CA048

6.22 [CHP] ISDN Protocol Variant Definitions

The CHP parameters define line configurations. Within the [CHP] section of the CONFIG file, ISDN protocol variants are defined using the `Variant Define n` command. For a detailed description of the `Variant Define n` command, see Section 3.6, “[CHP] Section”, on page 43.

The ISDN protocol variant parameters include:

- ProtocolType (Protocol Type)
- InterCallDelay (Inter-call Delay)
- DisconnectTimeout (Disconnect Timeout)
- Layer1Protocol (Layer 1 Protocol)
- InfoTransferRate (Information Transfer Rate)
- InfoTransferCap (Information Transfer Cap)
- CalledNumberType (Called Number Type)
- CalledNumberPlan (Called Number Plan)
- CalledNumberCount (Called Number Count)
- CallingNumberType (Calling Number Type)
- CallingNumberPlan (Calling Number Plan)
- CallingNumberPresentation (Calling Number Presentation)
- CallingNumberScreening (Calling Number Screening)
- CallingNumberCount (Calling Number Count)
- CallProgress (Call Progress)
- CaHdgLoHiGl (Hello Edge/Low Glitch/High Glitch)
- CaAnsdglPSV (Answer Deglitcher/PAMD Speed Value)
- CaHdgLoHiGl (Hello Edge/Low Glitch/High Glitch)
- CaBusySet (Busy Signal)
- CaSitSet (SIT Signal)

- CaFaxSet (Fax Signal)
- CaPvdId (Voice Detection Signal)
- CaPamdId (Answering Machine Signal)
- CaSignalTimeout (Signal Timeout)
- CaAnswerTimeout (Answer Timeout)
- CaPvdTimeout (Voice Detection Timeout)

ProtocolType (Protocol Type)

Description: The **ProtocolType** parameter defines the type of ISDN protocol used on a channel. The value of the parameter is dependent on the firmware being downloaded and the CONFIG files used. For example, when downloading the *ml2_qsa_4ess.config* file, **ProtocolType** should be set to a value of 1.

Note: The **ProtocolType** parameter is also used when defining T-1 protocol variants.

Values:

- 1: 4ESS and NI-2
- 2: 5ESS
- 3: DMS100 and DMS 250
- 4: NTT
- 7: NET5 and QSIG

InterCallDelay (Inter-call Delay)

Description: The **InterCallDelay** parameter defines the minimum amount of time between outbound calls.

Note: The **InterCallDelay** parameter is also used when defining T-1 protocol variants.

Values: n > 0 (milliseconds)

DisconnectTimeout (Disconnect Timeout)

Description: The **DisconnectTimeout** parameter defines the time delay between proceeding and alert/connect. The call will transition to idle after this time period (sooner if ClearConf is received).

Note: The **DisconnectTimeout** parameter is also used when defining T-1 protocol variants.

Values: n > 0 (milliseconds)

Guidelines: For DNM160TEC and DMT160TEC boards, when there are 16 or more trunks in one NFAS group and calls are made on every channel at a very fast rate, the firmware may start missing calls. In this case, increase the value of the **DisconnectTimeout** parameter to at least 3000 milliseconds.

Layer1Protocol (Layer 1 Protocol)

Description: The **Layer1Protocol** parameter defines the User Layer 1 Protocol.

Values:

- 0x00: Protocol not present
- 0x01: CCITT
- 0x02: G.711 mu-law
- 0x03: G.711 A-law
- 0x04: G.721 ADPCM
- 0x05: G.721 kHz
- 0x06: 384 kHz Video
- 0x07: NS Rate Adaption
- 0x08: V120 Rate Adaption
- 0x09: X.31 HDLC

InfoTransferRate (Information Transfer Rate)

Description: The **InfoTransferRate** parameter defines the information transfer rate.

Values:

- 0x00: Rate undefined
- 0x10: 64 kbps
- 0x11: 128 kbps
- 0x13: 384 kbps
- 0x15: 1536 kbps
- 0x17: 1920 kbps
- 0x18: Multi-rate

InfoTransferCap (Information Transfer Cap)

Description: The **InfoTransferCap** parameter defines the information transfer capability.

Values:

- 0x00: Speech
- 0x08: Unrestricted digital
- 0x09: Restricted digital
- 0x10: 3 kHz
- 0x11: 7 kHz
- 0x18: Video

CalledNumberType (Called Number Type)

Description: The **CalledNumberType** parameter defines the type of outbound calls (Called Party Numbers).

Values:

- 0x00: Unknown
- 0x01: International
- 0x02: National
- 0x03: Network specific
- 0x04: Network subscriber
- 0x06: Network abbreviated

CalledNumberPlan (Called Number Plan)

Description: The **CalledNumberPlan** parameter defines the numbering plan to use for outbound calls (Called Party Numbers).

Values:

- 0x00: Unknown
- 0x01: ISDN
- 0x02: Telephony
- 0x03: Date X.121
- 0x04: Telex F.69
- 0x08: National standard
- 0x09: Private

CalledNumberCount (Called Number Count)

Description: The **CalledNumberCount** parameter defines the number of digits to collect from an incoming call.

Values:

- 0: Collect all the digits provided
- n: Number of digits to collect

CallingNumberType (Calling Number Type)

Description: The **CallingNumberType** parameter defines the type of outbound call (Calling Party Number).

Values:

- 0x00: Unknown
- 0x01: International
- 0x02: National
- 0x03: Network specific
- 0x04: Network subscriber
- 0x06: Network abbreviated

CallingNumberPlan (Calling Number Plan)

Description: The **CallingNumberPlan** parameter defines the numbering plan to use for outbound calls (Calling Party Numbers).

Values:

- 0x00: Unknown
- 0x01: ISDN
- 0x02: Telephony
- 0x03: Date X.121
- 0x04: Telex F.69
- 0x08: National standard
- 0x09: Private

CallingNumberPresentation (Calling Number Presentation)

Description: The **CallingNumberPresentation** parameter defines the presentation for calling number (outbound calls).

Values:

- 0x00: Allowed
- 0x01: Restricted
- 0x02: Not available

CallingNumberScreening (Calling Number Screening)

Description: The **CallingNumberScreening** parameter defines the screening for calling number (outbound calls).

Values:

- 0x00: User provided
- 0x01: Verified and passed
- 0x02: Verified and failed
- 0x03: Network provided

CallingNumberCount (Calling Number Count)

Description: The **CallingNumberCount** parameter defines the number of Calling Party Number digits to collect from incoming call.

Values:

- 0: Collect all the digits provided
- n: Number of digits to collect

CallProgress (Call Progress)

Description: The **CallProgress** parameter enables or disables call progress detection for call setup.

Note: The **CallProgress** parameter is also used when defining T-1 protocol variants.

Values:

- y: Enable call progress detection
- n: Disable call progress detection

CaHdgLoHiGl (Hello Edge/Low Glitch/High Glitch)

The **CaHdgLoHiGl** parameter combines three parameters into one. They include the Hello Edge, Low Glitch, and High Glitch parameters. The values for all three parameters are contained in the **CaHdgLoHiGl** parameter value, 0xFF020F13, where 02 is the default hexadecimal value (2 decimal) for the Hello Edge parameter, 0F is the default hexadecimal value (15 decimal) for the Low Glitch parameter, and 13 is the hexadecimal value (19 decimal) for the High Glitch parameter.

Description: The **Hello Edge** parameter defines the point at which a connect will be returned to the application.

Values:

- 1: Rising edge (immediately when a connect is detected)
- 2: Falling edge (after the end of the salutation0

Description: The **Low Glitch** parameter defines, in intervals of 10 milliseconds, the maximum silence period to ignore. This maximum silence period helps to eliminate spurious silence intervals.

Values: The default value is 15 decimal (150 milliseconds).

Description: The **High Glitch** parameter defines, in intervals of 10 milliseconds, the maximum nonsilence period to ignore. This maximum nonsilence period helps to eliminate spurious nonsilence intervals.

Values: The default value is 19 decimal (190 milliseconds).

CaAnsdglPSV (Answer Deglitcher/PAMD Speed Value)

The **CaAnsdglPSV** parameter combines two parameters into one. They include the Answer Deglitcher and PAMD Speed Value parameters. The values for both parameters are contained in the **CaAnsdglPSV** parameter value, 0xFFFFFF01, where 01 is the default hexadecimal value (1 decimal) for the PAMD Speed Value parameter and FF is the default hexadecimal value (-1 decimal) for the Answer Deglitcher parameter, which corresponds to disabling it. This parameter should only be enabled if you are concerned with measuring the length of the salutation.

Description: The **Answer Deglitcher** parameter defines the maximum silence period, in 10 millisecond intervals, allowed between words in a salutation.

Values: The default value is -1 (FFFF), for disabled.

Description: The **PAMD Speed Value** parameter defines the PAMD algorithm: PAMD_ACCU, PAMD_FULL, and PAMD_QUICK. PAMD_QUICK provides the fastest results based on the connect circumstances, but is the least accurate. PAMD_FULL performs hiss noise analysis to determine if this is an answer machine response, and then performs a full evaluation of the voice response if the hiss information is not sufficient to make the decision. PAMD_ACCU will not perform hiss noise analysis, since this is not required with today's digital answering systems, but will perform a full answer size voice response to achieve the most accurate result.

Values:

- 1 [default]: PAMD_ACCU
- 2: PAM_FULL
- 3: PAM_QUICK

CaRingingSet (Ringing Signal)

Description: The **CaRingingSet** parameter defines the signal set used to detect ringing for call progress analysis.

Note: The **CaRingingSet** parameter is also used when defining T-1 protocol variants.

Values: 0x024940

CaBusySet (Busy Signal)

Description: The **CaBusySet** parameter defines the signal set used to detect busy for call progress analysis.

Note: The **CaBusySet** parameter is also used when defining T-1 protocol variants.

Values: 0x004DE0

CaSitSet (SIT Signal)

Description: The **CaSiteSet** parameter defines the signal set used to detect Standard Information Tones (SIT) tones for call progress analysis.

Note: The **CaSiteSet** parameter is also used when defining T-1 protocol variants.

Values: 0x02F240

CaFaxSet (Fax Signal)

Description: The **CaFaxSet** parameter defines the signal set used to detect fax tones for call progress analysis.

Note: The **CaFaxSet** parameter is also used when defining T-1 protocol variants.

Values: 0x014B80

CaPvdId (Voice Detection Signal)

Description: The **CaPvdId** parameter defines the signal to use for positive voice detection in call progress analysis.

Note: The **CaPvdId** parameter is also used when defining T-1 protocol variants.

Values: 0x01F4C1

CaPamdId (Answering Machine Signal)

Description: The **CaPamdId** parameter defines the signal to use for positive answering machine detection in call progress analysis.

Note: The **CaPamdId** parameter is also used when defining T-1 protocol variants.

Values: 0x01A041

CaSignalTimeout (Signal Timeout)

Description: The **CaSignalTimeout** parameter defines the maximum amount of time to wait to detect a call progress tone from one of the call analysis signal sets. For T1 loop start and ground start protocols, if this time is exceeded, then the outbound call will fail with the reason being NoAnswer.

Note: The **CaSignalTimeout** parameter is also used when defining T-1 protocol variants.

Values: n > 0 (the value must be a multiple of 10 ms)

CaAnswerTimeout (Answer Timeout)

Description: The **CaAnswerTimeout** parameter defines the maximum amount of time that call analysis will wait for ringback to stop (equivalent to the number of rings). If this time is exceeded, then the outbound call will fail with the reason being NoAnswer.

Note: The **CaAnswerTimeout** parameter is also used when defining T-1 protocol variants.

Values: n > 0 (the value must be a multiple of 10 ms)

CaPvdTimeout (Voice Detection Timeout)

Description: The **CaPvdTimeout** parameter defines the maximum amount of time that call analysis will wait to detect positive answering machine detection (PAMD) or positive voice detection (PVD) once ringback has ceased. If this time is exceeded, then the call state will transition to “Connected” with the reason being Normal. If PAMD or PVD is detected within this time period, then the “Connected” reason will be PAMD or PVD respectively.

Note: The **CaPvdTimeout** parameter is also used when defining T-1 protocol variants.

Values: n > 0 (the value is expressed in multiples of 10 milliseconds. For example, a value of 200 equals 2000 milliseconds, or 2 seconds)

- For analog boards: default = 800
- For digital boards: default = 400

6.23 [CHP] Analog Voice Variant Definitions

The CHP analog voice variant parameters apply to the DM3 analog voice boards. The Tone_SigId4 variant is used to enable forced disconnect. The forced disconnect is a method used to signal the local side of a call that the remote side has disconnected. This is done via a tone that is generated by the central office or PBX on the local side. This tone can be a dial tone or a form of busy tone (regular busy or fast busy).

SigId4 (Signal ID 4)

Description: The SigId4 parameter is used to enable a forced disconnect by enabling disconnect tone supervision. To enable forced disconnect, comment out the value 0x0 by inserting an exclamation symbol (!) at the beginning of the line containing the 0x0 value in the CONFIG file and uncommenting the line containing the value 238113 by deleting the exclamation symbol from the beginning of the line. See the following example:

```
!Variant Tone_SigId4        0x0        ! Don't Enable Disconnect Tone Supervision
Variant Tone_SigId4        238113      ! Enable disconnect tone supervision.
```

Values:

- 0x0: Disable disconnect tone supervision
- 238113: Enable disconnect tone supervision

6.24 [TSC] Parameters

The parameter in the [TSC] section of the CONFIG file is associated with the B channel sets.

Encoding (Encoding Method)

Number: 0x1209

Description: The **Encoding** parameter defines the encoding method used on a line.

Values:

- 1: A-law
- 2: mu-law

6.25 [TSC] defineBSet Parameters

The parameters defined by the `defineBSet` command in the [TSC] section of the CONFIG file are associated with the B channel sets. The syntax of the `defineBSet` command is:

```
defineBSet = SetId, LineId, StartChan, NumChans, BaseProtocol, Inbound, OutBound, DChanDesc,
Admin, Width, BChanId, SlotId, Direction, Count, [BChanId, SlotId, Direction, Count,] 0
```

Note: The [TSC] defineBSet parameters do not have parameter numbers explicitly defined within the CONFIG file.

The defineBSet parameters include:

- SetId (Set Identifier)
- LineId (Line Identifier)
- StartChan (Start Channel)
- NumChans (Number of B Channels)
- BaseProtocol (Base Protocol)
- Inbound (Inbound Variant)
- Outbound (Outbound Variant)
- DChanDesc (D Channel Identifier)
- Admin (Admin)
- Width (Width)
- BChanId (B Channel Identifier)
- SlotId (Slot Identifier)
- Direction (Direction)
- Count (Count)

SetId (Set Identifier)

Description: The **SetId** parameter is an arbitrary identifier set by the user that identifies the B channel set in which the B channels are a member.

Values: Number

Guidelines: Each B channel set must have a unique identifier.

For example, for each line on a board, **SetId** can be set sequentially to a value that is a multiple of 10 as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,2,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,3,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,4,20,1, 1,1,3,23,0
```

LineId (Line Identifier)

Description: The **LineId** parameter defines the T1 or E1 line that carries all of the B channels in the set.

Values: 1 to 16

Guidelines: For example, on a board with four network interfaces, the value of **LineId** is set to 1 for line 1, 2 for line 2, and so on for each line as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,1,20,1, 1,1,3,23,0
```

StartChan (Start Channel)

Description: The **StartChan** parameter defines the first B channel in the set. This parameter is used in combination with the **NumChans** parameter to define a contiguous set of B channels.

Values: The value range depends on the technology, because the number of available B channels varies.

- 1 to 24: T1
- 1 to 30: E1
- 1 to 31: E1 clear channel

Guidelines: For example, on a T-1 line where 23 of the 24 channels are used as B channels, the value of **StartChan** is set to 1 as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,1,20,1, 1,1,3,23,0
```

NumChans (Number of B Channels)

Description: The **NumChans** parameter defines the total number of B channels in the set. This parameter is used in combination with the **StartChan** parameter to define a contiguous set of B channels.

Values: The range of values varies with technology because the number of time slots varies.

- 1 to 24: T1
- 1 to 30: E1
- 1 to 31: E1 clear channel

Guidelines: For example, on a T-1 line, a value of 1 for **StartChan** and a value of 23 for **NumChans** defines 23 B channels numbered from 1 to 23:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,2,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,3,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,4,20,1, 1,1,3,23,0
```

BaseProtocol (Base Protocol)

Description: The **BaseProtocol** parameter defines the base protocol on which the B channel set will run.

Values:

- 0: T-1 CAS, ISDN or Global Call protocols (where the default protocol is defined by the firmware) or clear channel
- 7: Circa Analog - Supports Circa L feature phones (Dialogic Integrated Series boards)

Guidelines: For T-1 CAS, ISDN, and Global Call protocols, each firmware load supports only only one base protocol, so this parameter will be set to 0 for these protocols. This parameter is also set to 0 for clear channel. Clear channel is the ability to access telephony channels in the system and configure them to a user defined call control protocol, or to simply leave the lines "clear". The resources should have access to the telephony bus for media routing purposes, as well as signal detection, signal generation, and tone generation capabilities, if desired.

For example, on T-1 ISDN lines, **BaseProtocol** is set to a value of 0 as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,1,20,1, 1,1,3,23,0
```

Inbound (Inbound Variant)

Description: The **Inbound** parameter selects one of the protocol type variant parameter sets defined in the [CHP] section of the CONFIG file to use for inbound calls. The protocol variant defines the type of protocol running on the set of B channels.

Values:

- 0: Clear channel (disable inbound calls)
- n: Variant identifier as defined in the [CHP] section of the CONFIG file

Guidelines: This parameter is set to 0 for clear channel. Clear channel is the ability to access telephony channels in the system and configure them to a user defined call control protocol, or to simply leave the lines "clear". The resources should have access to the telephony bus for media routing purposes, as well as signal detection, signal generation, and tone generation capabilities, if desired.

For example, on T-1 ISDN lines, **Inbound** is set to a value of 1 as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,1,20,1, 1,1,3,23,0
```

Outbound (Outbound Variant)

Description: The **Outbound** parameter selects one of the protocol type variant parameter sets defined in the [CHP] section of the CONFIG file to use for outbound calls. The protocol variant defines the type of protocol running on the set of B channels.

Values:

- 0: Clear channels (disable outbound calls)
- n: Variant identifier as defined in the [CHP] section of the CONFIG file

Guidelines: This parameter is set to 0 for clear channel (disable outbound calls). Clear channel is the ability to access telephony channels in the system and configure them to a user defined call control protocol, or to simply leave the lines “clear”. The resources should have access to the telephony bus for media routing purposes, as well as signal detection, signal generation, and tone generation capabilities, if desired.

For example, on T-1 ISDN lines, **Outbound** is set to a value of 1 as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,1,20,1, 1,1,3,23,0
```

DChanDesc (D Channel Identifier)

Description: The **DChanDesc** parameter is an ISDN parameter that identifies which trunk the D-channel resides for this B-set. This parameter is ignored for T-1 CAS, clear channel, and Global Call protocols.

Values: 1 to 16

Guidelines: For example, on a board with four T-1 ISDN lines, **DChanDesc** is set as follows:

```
defineBSet=10,1,1,24, 0,1,1,1,20,1, 1,1,3,24,0
defineBSet=20,2,1,24, 0,1,1,2,20,1, 1,1,3,24,0
defineBSet=30,3,1,24, 0,1,1,3,20,1, 1,1,3,24,0
defineBSet=40,4,1,24, 0,1,1,4,20,1, 1,1,3,24,0
```

Admin (Admin)

Description: The **Admin** parameter is an arbitrary 32-bit value set by the user that is exported to the TSC_AttrAdminGroup attribute of the TSC cluster for each B channel in the set. This attribute can be used to find and/or allocate TSC clusters.

Values: 0 to 0xFFFFFFFF

Guidelines: For example, on a T1 line, **Admin** is set to a value of 20 as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,1,20,1, 1,1,3,23,0
```

Width (Width)

Description: The **Width** parameter specifies the number of time slots used by each B channel. Currently, only one time slot per channel is used.

Note: This **Width** should not be modified by the user.

Values: 1

Guidelines: For example, on a T1 line, **Width** is set to a value of 1 as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,1,20,1, 1,1,3,23,0
```

BChanId (B Channel Identifier)

Description: The **BChanId** parameter defines the initial B channel in the set to which the TSC instance is associated. It is also the channel to which the initial time slot, defined by **SlotId**, will be mapped. B channels are then sequentially mapped to time slots for a count of **Count**.

Values: The range of values varies with technology because the number of time slots varies.

- 1 to 24: T1
- 1 to 31: E1

Guidelines: For example, on a T-1 board where the D channel is mapped to time slot 24 on all four lines, **BChanId** and **SlotId** are set to a value of 1 and **NumChans** is set to a value of 23. This defines 23 B channels numbered 1 to 23 mapped to time slots 1 to 23.

```
defineBSet=10,1,1,23,0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23,0,1,1,2,20,1, 1,1,3,23,0
defineBSet=30,3,1,23,0,1,1,3,20,1, 1,1,3,23,0
defineBSet=40,4,1,23,0,1,1,4,20,1, 1,1,3,23,0
```

For E-1 ISDN lines that usually contain a D channel mapped to time slot 16, the mapping of channels to time slots occurs in two sets of **BChanId**, **SlotId**, **Direction** and **Count** definitions. The first set of definitions maps time slots before the D channel and the second set maps time slots after the D channel.

For example, on an E-1 ISDN board with four network interfaces, where time slot 16 is used for signaling on all four lines, **BChanId** would be defined on each line as follows:

```
defineBSet=10,1,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=20,2,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=30,3,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=40,4,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
```

In this example, channels 1 to 15 are mapped to time slots 1 to 15 and channels 16 to 30 are mapped to time slots 17 to 31.

For E-1 clear channel lines where the time slot 16 is not used for signaling, additional `defineBSet` commands are added to clear channel 31. Both **StartChan** and **BChanId** are set to a value of 31, **NumChans** and **Count** are set to a value of 1, and **SlotId** is set to 16 as follows:

```
defineBSet=50,1,31,1, 0,0,0,1,21,1, 31,16,3,1,0
defineBSet=60,2,31,1, 0,0,0,1,21,1, 31,16,3,1,0
defineBSet=70,3,31,1, 0,0,0,1,21,1, 31,16,3,1,0
defineBSet=80,4,31,1, 0,0,0,1,21,1, 31,16,3,1,0
```

SlotId (Slot Identifier)

Description: The **SlotId** parameter defines the logical time slot the initial B channel, defined by **BChanId**, is using. B channels are then sequentially mapped to time slots for a count of **Count**.

Values: The range of values varies with technology because the number of time slots varies.

- 1 to 24: T1
- 1 to 31: E1 ISDN
- 1 to 31: E1 clear channel

Guidelines: For E-1 ISDN, the mapping of channels to time slots occurs in two sets of **BChanId**, **SlotId**, **Direction** and **Count** definitions. The first set of definitions maps the time slots before the D channel, and the second set maps the slots after the D channel.

For example, on an E-1 ISDN board with four network interfaces, where time slot 16 is used for signaling on all four lines, **SlotId** for all four lines would be as follows

```
defineBSet=10,1,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=20,2,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=30,3,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=40,4,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
```

For all lines in this example, channels 1 to 15 are sequentially mapped to time slots 1 to 15 and channels 16 to 30 are mapped to time slots 17 to 31.

For E-1 clear channel lines where time slot 16 is not used for signaling, additional `defineBSet` commands are added to clear channel 31 and to map time slot 16. Both **StartChan** and **BChanId** are set to a value of 31, **NumChans** and **Count** are set to a value of 1, and **SlotId** is set to 16 as follows:

```
defineBSet=50,1,31,1, 0,0,0,1,21,1, 31,16,3,1,0
defineBSet=60,2,31,1, 0,0,0,1,21,1, 31,16,3,1,0
defineBSet=70,3,31,1, 0,0,0,1,21,1, 31,16,3,1,0
defineBSet=80,4,31,1, 0,0,0,1,21,1, 31,16,3,1,0
```

Direction (Direction)

Description: The **Direction** parameter defines the direction in which the data can be sent: inbound, outbound, or both.

Values:

- 1: Inbound
- 2: Outbound
- 3: Both

Guidelines: For example, on an T-1 line where data is transferred both inbound and outbound, **Direction** is set to a value of 3 as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,2,20,1, 1,1,3,23,0
defineBSet=30,3,1,23, 0,1,1,3,20,1, 1,1,3,23,0
defineBSet=40,4,1,23, 0,1,1,4,20,1, 1,1,3,23,0
```

Count (Count)

Description: The **Count** parameter defines the number of time slots that are being mapped to B channels. This value is limited to the value of **NumChans** since only the number of channels that exist on a line can be mapped to a time slots.

Values: 1 to **NumChans**

Guidelines: For example, on a T-1 line containing two network interfaces, where time slot 24 is used as a D channel on both lines, the **Count** for both lines would be as follows:

```
defineBSet=10,1,1,23, 0,1,1,1,20,1, 1,1,3,23,0
defineBSet=20,2,1,23, 0,1,1,1,20,1, 1,1,3,23,0
```

For an E-1 line, **Count** is set to a value of 30 for lines that contain only B channels. For lines that contain a single D channel, the mapping of channels to time slots occurs in two sets of **BChanId**, **SlotId**, **Direction** and **Count** definitions. The first set of definitions maps the time slots before the D channel, and the second set maps the slots after the D channel. For example, on an E-1 board with four network interfaces, where time slot 16 is used for signaling on all four lines, the **Count** for all four lines would be as follows:

```
defineBSet=10,1,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=20,2,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=30,3,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
defineBSet=40,4,1,30, 0,1,1,1,20,1, 1,1,3,15, 16,17,3,15,0
```

For all lines in this example, channels 1 to 15 are mapped to time slots 1 to 15 and channels 16 to 30 are mapped to time slots 17 to 31.

For E-1 clear channel lines where the time slot 16 is not used for signaling, additional `defineBSet` commands are added to clear channel 31 and to map time slot 16. **Count** is set to a value of 1 (also the value of **NumChans**) as follows:

```
defineBSet=50,1,31,1, 0,0,0,1,21,1, 31,16,3,1,0
defineBSet=60,2,31,1, 0,0,0,1,21,1, 31,16,3,1,0
defineBSet=70,3,31,1, 0,0,0,1,21,1, 31,16,3,1,0
defineBSet=80,4,31,1, 0,0,0,1,21,1, 31,16,3,1,0
```

6.25.1 Gain Parameters

The Gain parameters define the transmit and receive gain for the DMV160LP Series (DMV160LP and DMV160LPHIZ), DI Series Station Interface (DI/408-LS-A-R2 and DI/SI32-R2), and the High Density Station Interface boards. The Gain parameters include:

- Tx Gain Min
- Tx Gain Max
- Tx Base Gain
- Tx Gain
- Rx Gain Min
- Rx Gain Max
- Rx Base Gain
- Rx Gain

Tx Gain Min

Number: 0x120F

Description: The **Tx Gain Min** parameter defines the minimum gain toward the station or trunk transmitting from the CT Bus.

Values: -31 dB (This is the default value for all boards except the HDSI series, which is -9 dB.)

Guidelines: For best results, the value should not be changed from the default.

Tx Gain Max

Number: 0x1210

Description: The **Tx Gain Max** parameter defines the maximum gain toward the station or trunk transmitting from the CT Bus.

Values: + 31 dB (This is the default value for all boards except the HDSI series, which is +3 dB.)

Guidelines: For best results, the value should not be changed from the default.

Tx Base Gain

Number: 0x120C

Values: +3 dB for DMV160LP Series and DI/0408-LS-A-R2 trunks; -3 dB for all DI/SI32-R2 variants and DI/0408-LS-A-R2 stations

Guidelines: The Tx Base Gain and Tx Gain parameters should be set to the same value.

Tx Gain

Number: 0x120D

Description: The **Tx Gain** parameter defines the gain toward the station or trunk transmitting from the CT Bus.

Values: +3 dB for DMV160LP Series and DI/0408-LS-A-R2 trunks; -3 dB for all DI/SI32-R2 variants and DI/0408-LS-A-R2 stations

Guidelines: The Tx Gain and Tx Base Gain parameters should be set to the same value.

Rx Gain Min

Number: 0x121B

Description: The **Rx Gain Min** parameter defines the minimum gain away from the station or trunk received by the CT Bus.

Values: -31 dB (This is the default value for all boards except the HDSI series, which is -9 dB.)

Guidelines: For best results, the value should not be changed from the default.

Rx Gain Max

Number: 0x121C

Description: The **Rx Gain Max** parameter defines the maximum gain away from the station or trunk received by the CT Bus.

Values: +31 dB (This is the default value for all boards except the HDSI series, which is +3 dB.)

Guidelines: For best results, the value should not be changed from the default.

Rx Base Gain

Number: 0x1218

Values: +3 dB for DMV160LP Series and DI/0408-LS-A-R2 trunks; -3 dB for all DI/SI32-R2 variants and DI/0408-LS-A-R2 stations

Guidelines: The Rx Base Gain and Rx Gain parameters should be set to the same value.

Rx Gain

Number: 0x1219

Description: The **Rx Gain** Parameter defines the gain away from the station or trunk received by the CT Bus.

Values: +3 dB for DMV160LP Series and DI/0408-LS-A-R2 trunks; -3 dB for all DI/SI32-R2 variants and DI/0408-LS-A-R2 stations

Guidelines: The Rx Gain and Rx Base Gain parameters should be set to the same value.

Glossary

0x1b: Section of the CONFIG files that defines parameters relating to data received from the network. The parameters associated with this section have parameter numbers that start with 0x1b and only apply to DM/IP technologies.

0x2b: Section of the CONFIG file that defines encoding for Audio of music played when placing station on hold.

0x2c: Section of the CONFIG file that defines echo cancellation parameters used in Continuous Speech Processing (CSP) applications. as well as parameters associated with Silence Compressed Streaming (SCS).

0x39: Section of the CONFIG file that defines conferencing parameters applicable to all conferencing lines on a board.

0x3b: Section of the CONFIG file that defines parameters relating to conferencing.

0x44: Section of the CONFIG file that defines the companding method for Intel® NetStructure™ DI Series Station Interface boards.

4ESS: A T1 protocol switch primarily used for switching digital voice, but it also supports ISDN protocols.

5ESS: A T1 protocol switch used for switching digital voice and data channels, and supports both basic rate and primary rate ISDN.

AGC: Automatic Gain Control is an encoding process that attempts to maintain a constant volume during voice recording.

alternate mark inversion: See AMI.

AMI: Alternate mark inversion is a form of bipolar signaling in which each successive mark is of the opposite polarity and spaces have zero amplitude.

Automatic Gain Control: See AGC.

base protocol: The protocol implemented by the CHP component. Protocol variants are derived from this base. Compare with *protocol variant*.

B channel: An ISDN bearer channel that carries voice, fax and compressed video.

CAS: Channel Associated Signaling is the component responsible for managing the generation and detection of digital line signaling functions required to manage voice channels. Channel Associated Signaling also applies to a signaling method in which the signaling for that channel is directly associated with the channel.

CCS: Common Channel Signaling is the component that applies to technologies such as ISDN that use common channel signaling. Common Channel Signaling also applies, in general, to a signaling method in which the signaling for a group of channels is carried on a separate (common) channel.

CDP: Country Dependent Parameters file defining parameters necessary for configuring products to different country requirements. This file has a *.cdp* extension.

CEPT: European Conference of Postal and Telecommunications Administrations. A group of European countries organized for the purpose of setting telecommunications standards in Europe.

CFA: Carrier-Failure Alarm.

CHP: Channel Protocol is the component responsible for implementing the telephony communication protocol that is used on each network interface.

clear channel: A signaling configuration where none of the line's bandwidth is used for signaling. Clear channel signaling is the ability to access telephony channels in the system and configure them to a user defined call control protocol, or to simply leave the lines 'clear'. The resources should have access to the telephony bus for media routing purposes, as well as signal detection, signal generation, and tone generation capabilities, if desired. NFAS is an example of clear channel signaling.

clock master: The device (board) that provides timing to all other devices attached to the TDM bus. The clock master drives bit and framing clocks for all of the other boards (slaves) in the system.

cluster: A collection of component instances that share specific TDM time slots on the network interface and which therefore operate on the same media stream data. The cluster concept in the Intel® Dialogic® architecture corresponds generally but not exactly to the concept of a "group" in S.100 or to a "channel" in conventional Dialogic architectural terminology. Component instances are bound to a particular cluster and its assigned time slots in an allocation operation.

CNG: Comfort Noise Generation.

CONFIG: A text-input configuration file containing component-specific parameters. This file has a *.config* extension and is used to create an FCD file.

configuration file: See CONFIG file.

configuration file set: A set of files associated with a specific board configuration. All the files in the set have the same name, but different extensions. The set includes the CONFIG, FCD, and PCD files.

Country Dependent Parameters: See CDP.

CRC: Cyclic Redundancy Check.

D channel: An ISDN channel that carries signaling information.

D4: A T1 protocol switch that supports T1 robbed bit signaling and provides D4 framing, but does not support ISDN protocols.

DM3: An architecture on which a whole set of Intel telecom products is built. The DM3 architecture is open, layered, and flexible, encompassing hardware as well as software components.

DMA: Direct memory access.

DMS: A T1 protocol switch (DMS-100) for primary rate ISDN applications.

DTD: Dial Tone Detection.

DTMF: Dual Tone Multi-Frequency. Touchtone dialing.

E&M: Two-way telephony signaling that uses an "E" (far end) lead and an "M" (near-end) lead. Signaling is accomplished by applying -48 volts DC to the leads.

encoder: The component responsible for performing an encoding process on a media stream.

FCD: Feature Configuration Description file that lists any non-default parameter settings that are necessary to configure a hardware/firmware product for a particular feature set. This file has a *.fcd* extension.

Feature Configuration Description: See FCD.

fixed routing: A routing configuration where the resource devices (voice/fax) and network interface devices are permanently coupled together in a fixed configuration. Only the network interface time slot device has access to the CT Bus

flash: While the phone is off-hook, quickly pressing and releasing the flash hook to signal the central office or PBX that you are requesting special processing, for example, call waiting.

flash hook: The plunger the phone's handset rests on while on-hook.

flexible routing: A routing configuration where the resource devices (voice/fax) and network interface devices are independent, which allows exporting and sharing of the resources. All resources have access to the CT Bus.

FRU: Field replaceable unit.

FXO: Foreign Exchange Office - a device at a central site that permits extending PBX services to remote sites. The FXO emulates a phone to the PBX.

FXS: Foreign Exchange Station - a device located remotely from a PBX that permits extending PBX services to remote sites. The FXS emulates a PBX to the remote phone.

ground start: A two-way, two-wire (tip and ring) signaling method similar to loop start in which the current flows in a circuit. Ground start is normally between a PBX and central office and seizure of the line is accomplished by momentarily grounding one of the circuit wires, usually the ring of the tip and ring circuit.

HDB3: A modified AMI signaling code that only applies to E1 and is used to preserve one's density on the line.

high density bipolar three zero: See HDB3.

in-band signaling: A signaling scheme where both the data and the signaling information for the data are carried over the same channels.

instance: A component instance is an addressable unit within the software architecture; it represents a single thread of control. The system resource management and messaging services operate at the instance level. A set of component instances that make up a resource instance communicate with one another using the system messaging services. A set of component instances is usually associated with a channel of call processing.

IPVS: IP Voice Streaming

ISDN: Integrated Services Digital Network. See primary rate ISDN.

LAPD: Link Access Protocol for the D channel.

Layer 1: Physical layer of the OSI model that address the physical aspects of network access.

Layer 2: Data Link layer of the OSI model that address data transfer and routing.

Layer 3: Network layer of the OSI model that addresses line communication procedures.

LCON: See LineAdmin.

LineAdmin: Line Administration component responsible for managing line devices.

LOF: Loss of frame.

LOS: Loss of signal.

loop start: A two-way, two-wire (tip and ring) signaling method in which the current used for signaling flows in a circuit (loop) between a telephone and PBX or a telephone and central office. Seizure of the line is accomplished by going off-hook which causes current to flow in a circuit (loop).

LOF: Loss of Frame.

media loads: Pre-defined, numbered sets of features supported by DM/V, DM/N, DM/T, DM/IP, and DM/F boards.

MF: Multi-Frequency

MLM: Load Module.

Net5: An E1 protocol switch. Net5 is a European ISDN primary rate switch.

NFAS: Non-Facility-Associated Signaling is a form of out-of-band signaling where a single ISDN primary rate D channel provides signaling and control for up to 10 ISDN primary rate lines.

NI-2: National ISDN-2. A U.S. standard software interface that can be installed on most switch types, providing maximum inter operability with ISDN lines.

NIC: Network interface card.

NTT: A T1 protocol switch (INS-Net 1500) that is used by Nippon Telephone and Telegraph (NTT) for primary rate ISDN.

on-hook: The signaling state that occurs when a handset is sitting on the phone (the phone's inactive state) and the flash hook is depressed. Compare with *off-hook*.

off-hook: The signaling state that occurs when the handset is removed from the phone and the flash hook is released. When a phone is taken off-hook it signals the central office or PBX that it needs attention, for example, to make a call or to answering an incoming call. Compare with *on-hook*.

OSI: Open Standards Interconnections. ISO-developed open standards-based framework for inter-system communications. The OSI model categorizes the communication process into seven layers. Layers 1 to 4 address network access and Layers 5 to 7 address messaging.

out-of-band signaling: A signaling scheme where the signaling is carried over channels separate from the channels carrying the data.

PAMD: Positive answering machine detection.

PBLM: Processor Boot Load Module.

PBX: Private Branch Exchange.

PCD: Product Configuration Description file that contains product or platform configuration description information. This file has a *.pcd* extension.

PCM: Pulse Code Modulation.

PDK: Protocol Developer Kit.

PDK Configuration Utility: A Graphical User Interface (GUI) application that enables you to generate an FCD file from a base CONFIG file and a number of protocol CDP files. This utility integrates Global Call protocols (CDP files) into the FCD file and all parameters are download during system initialization.

PDKManager: A command-line application that enables you to download and configure Global Call protocols on DM/V boards. These protocols are configured separately from the board-level parameters defined in the CONFIG file and are downloaded after system initialization.

PLM: Processor Load Module.

port: A logical entity that represents the point at which PCM data can flow into or out of a component instance or interface in a cluster. The port abstraction provides a high-level means of defining potential data flow paths within clusters and controlling the actual data flow using simple protocols. Ports are classified and designated in terms of data flow direction and the type of entity that provides the port.

primary D channel: the D channel that provides the signaling and control in an NFAS configuration.

primary rate ISDN: An application that uses a single channel to carry the signaling for all other channels on a line. On a T1 line, the application uses channels 1 through 23 (B channels) to carry data, digital voice, and compressed video. Channel 24 (D channel) carries the signaling for all 23 B channels. On an E-1 line, the application uses channels 1 through 15 and 17 through 31 (B channels) to carry data, digital voice, and compressed video. Channel 16 (D channel) carries the signaling for all 30 B channels.

Product Configuration Description: See PCD.

protocol variant: A version of the base protocol that has been customized by a set of parameters. This parameter set configures a CHP component to support a particular T-1 telephony protocol. Features such as wink start, DTMF DNIS and MF ANI are enabled and tuned by the parameters in a protocol variant. Compare with *base protocol*.

pulse: A temporary state change from the current signal state to a new signaling state, and then back to the original signaling state. Compare with *sequence*, *train* and *transition*.

PVD: Positive voice detection.

Q.931: Primary rate ISDN D channel signaling protocol standard. (ITU-T Recommendation I.451). The protocol defines the signaling packet, including message type and content, and allows for voice and data transfer on a single trunk.

QSIG: A T1 and E1 protocol switch. QSIG is an ISDN signaling and control protocol used for communications between two or more Private Integrated Network Exchange applications (PSS1). The signaling protocol for this standard is defined by Q.931.

R2MF: An E1 protocol switch. R2MF is an in-band common channel signaling protocol that uses channel 16 to convey the signaling for the 30 voice channels. This international signaling system is used mostly in Europe and Asia in non-ISDN applications to permit the transmission of numerical and other information relating to the called and calling subscriber lines.

RAI: Remote Alarm Indication.

Rate Adaption: Conversion of digital data into a different transfer speed (rate) and form.

recorder: The component responsible for a resource's message exchanges with the host, as well as media stream management and encoder component control functions.

red alarm: An alarm generated by the device at the receiving end of a T-1 or E-1 line to report a loss of signal or frame alignment (synchronization) in the signal being received (incoming data).

resource: A conceptual entity that provides a specific functionality to a host application. A resource contains a well defined interface or message set, which the host application utilizes when accessing the resource. Resource firmware consists of multiple components that run on top of the core platform software (which includes the platform-specific DM3 kernel and device driver). The Global Call resource is an example of such a resource, providing all of the features and functionality necessary for handling calls on the platform.

SCD: System Configuration Description file defining the physical parameters of a platform This file has a *.scd* extension.

SCR: Silence Compressed Record is an encoding process that compresses silence during voice recording.

SCS: Silence Compressed Streaming refers to the process of streaming audio energy to the host application with silence periods significantly reduced.

sequence: A set of train signals. Compare with *pulse, train* and *transition.*

Silence Compressed Record: See SCR.

Silence Compressed Streaming: See SCS

SIT: Standard Information Tones

slave: Device (board) that is not a clock master, but instead, derives its timing from the TDM bus.

System Configuration Description: See SCD.

TDM: Time division multiplexing.

TDM bus: The resource bus used to carry information between boards

TEI: Terminal Endpoint Identifier. TEI defines which device(s) attached to a BRI ISDN line is communicating with the CO.

time division multiplex: A multiplexing scheme in which a number of low speed digital signals are incorporated onto a high speed line in a byte-interleave pattern.

train: A set of transitions from one signaling state to another in a predefined pattern (set of pulses). Compare with *pulses, sequence* and *transition*.

transition: A permanent state change from the current signal state to a new signaling state. Compare with *pulse, sequence* and *train*.

TS16: An E1 protocol switch. TS16 is a type of clear channel signaling which allows time slot 16 to be used for data instead of signaling.

TSC: Telephony Service Component is the component responsible for managing the B channel sets.

VAD: Voice Activity Detection.

wink: A single pulse used sent from a phone, central office, or PBX as part of protocol hand-shaking.

yellow alarm: An alarm generated by the device at the receiving end of a T-1 or E-1 line and sent to the device at the transmitting (remote) end to signify that a red alarm condition exists at the receiving (local) end. The yellow alarm is sent to the transmitting device as long as the red alarm condition exists at the receiving end.

Numerical Index of Parameters

Index

Symbols

A

B

C

D

H

I

L

M

N

O

P

Q

T

U

V

W

Y

Z