



# Infinity Series H.100 T1/Primary Rate ISDN Board

---

## TECHNICAL MANUAL

Documentation Revision 2.0: January 28, 2011

Copyright ©2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2011  
by American Tel-A-Systems, Inc.  
All rights reserved.  
257M025C

**Note:** This manual applies to the H.100 PCI board with the 386 processor , the H.100 PCI board with the ARM processor and the H.100 PCI Express board with the ARM processor. Because of the differences between the processors, there are minor differences in the firmware. These differences have been noted where appropriate.

Note that this manual refers to the revised T1 Board with speech resources running 1.0e or later firmware. As these boards have different switching and DSP hardware than earlier versions, some features described are not applicable to earlier boards.

## Contents

1.0	INTRODUCTION.....	1-1
1.1	Features and Capabilities.....	1-2
1.1.1	The Physical Interface .....	1-2
1.1.2	Signaling Protocols. ....	1-3
1.1.3	DSP Functions.....	1-4
1.1.4	The H.100 Bus.....	1-4
1.1.5	Clock Modes. ....	1-5
1.1.6	Message Passing. ....	1-5
1.1.7	Flash EAROM for Firmware.....	1-6
1.1.8	EEPROM for Configuration Information. ....	1-6
1.1.9	Analog Audio Ports.....	1-6
1.2	How to Use This Manual.....	1-7
2.0	QUICK START.....	2-1
3.0	INSTALLATION. ....	3-1
3.1	PCI Configuration. ....	3-2
3.2	Jumpers & Headers. ....	3-2
3.3	Connectors: P2, J101, P5 and P6. ....	3-3
3.4	Installation. ....	3-4
4.0	INITIALIZATION.....	4-1
4.1	PCI Initialization. ....	4-1
4.2	Initialization Commands. ....	4-2
4.3	Configuration Memory. ....	4-7
4.4	Backup & Restoration of Configuration Information. ....	4-8
5.0	COMMUNICATING WITH THE PC.....	5-1
5.1	Commands and Responses Protocol. ....	5-2
5.1.1	Sending Commands to the Board. ....	5-3
5.1.2	Reading Messages from the Board. ....	5-3
5.1.3	Reading Board Information. ....	5-4
5.2	Interrupts. ....	5-4
5.2.1	Interrupt Initialization. ....	5-5
5.2.2	Step-by-Step Summary. ....	5-5

5.3	Commands and Responses.....	5-6
5.3.1	Characteristics of Command Strings.....	5-6
5.3.2	Command Parameters.....	5-6
5.3.3	Commands from the PC to the H.100 T1/PRI Board. . .	5-7
	▶ Alarm Commands . . . . .	5-7
	▶ Audio Port Commands . . . . .	5-7
	▶ B-Channel Commands . . . . .	5-8
	▶ Layer 3 “D” Commands for NT Spans.....	5-9
	▶ Layer 3 “D” Commands for CI Spans.....	5-11
	▶ Interrupt Control Commands. . . . .	5-13
	▶ Conference Commands . . . . .	5-14
	▶ Layer 3 Message Commands . . . . .	5-14
	▶ MVIP Compatibility Commands. . . . .	5-14
	▶ Query Commands. . . . .	5-15
	▶ Reset Commands. . . . .	5-15
	▶ Setup Commands. . . . .	5-15
	▶ TEI Management Commands.....	5-17
	▶ Version Requests. . . . .	5-17
	▶ Download Commands.....	5-17
	▶ Diagnostics. . . . .	5-17
5.3.4	Responses from the H.100 T1/PRI ISDN Board. . . . .	5-18
	▶ Acknowledgments.....	5-18
	▶ Alarm Events. . . . .	5-18
	▶ Layer 3 “D” Responses for NT Spans.....	5-19
	▶ Layer 3 “D” Responses for CI Spans. . . . .	5-21
	▶ Error Messages. . . . .	5-24
	▶ Layer 3 Message Received Response. . . . .	5-24
	▶ Query Responses.....	5-24
	▶ B-Channel State Change Messages.....	5-25
	▶ Diagnostic Responses. . . . .	5-25
	▶ Maintenance Responses. . . . .	5-26

6.0	THE H.100 BUS & CLOCK MODES. . . . .	6-1
6.1	The H.100 Bus. . . . .	6-1
6.1.1	Legacy Bus Compatibility. . . . .	6-2
6.2	Clock Modes. . . . .	6-3
6.2.1	Slave Mode. . . . .	6-4
6.2.2	Primary Master Mode. . . . .	6-5
6.2.3	Secondary Master Mode. . . . .	6-6
6.2.4	Clock Fallback. . . . .	6-7
6.2.5	Clock Errors. . . . .	6-7
6.2.6	Clock Termination. . . . .	6-8
6.3	Configuration Information. . . . .	6-8
7.0	T1 FRAMING & CLOCKING. . . . .	7-1
7.1	DS1 Electrical Interface. . . . .	7-1
7.2	Configuring the Interfaces. . . . .	7-3
7.3	Alarms. . . . .	7-4
7.4	Line Interface Status Events. . . . .	7-6
7.5	Elastic Store Buffer Events. . . . .	7-6
7.6	Maintenance Functions. . . . .	7-7
7.6.1	D4 Loopback Codes. . . . .	7-7
7.6.2	ESF Bit Pattern Messages. . . . .	7-8
7.6.3	Loopback Control. . . . .	7-9
7.6.4	Performance Report Messages. . . . .	7-10
7.6.5	BERT Testing. . . . .	7-12
7.7	Robbed Bit Signaling. . . . .	7-16
7.8	Primary Rate ISDN. . . . .	7-17
7.9	Non-Facilities Associated Signaling. . . . .	7-17
8.0	ISDN LAYER 2 PROTOCOL. . . . .	8-1
8.1	Layer 1. . . . .	8-1
8.2	Layer 2. . . . .	8-2
8.3	Layer 2 States. . . . .	8-4
8.4	TEI Management. . . . .	8-5
8.5	Sending and Receiving Layer 3 Messages. . . . .	8-5
8.6	Transmit Queue Size. . . . .	8-6

9.0	USING “D” MESSAGES FOR LAYER 3. . . . .	9-1
9.1	Q.931 Messages. . . . .	9-1
9.2	“D” Command & Response Messages. . . . .	9-3
9.3	Information Elements. . . . .	9-5
9.3.1	Bearer Capability. . . . .	9-5
9.3.2	Call Reference. . . . .	9-6
9.3.3	Cause. . . . .	9-6
9.3.4	Address Numbers & Subaddresses. . . . .	9-8
9.3.5	Progress Indicator. . . . .	9-9
9.3.6	Notification Indicator. . . . .	9-10
9.3.7	Network Specific Facilities. . . . .	9-10
9.4	Default Directory Numbers. . . . .	9-11
9.5	Display Text. . . . .	9-12
9.6	Call Handling Procedures. . . . .	9-14
9.7.1	Call Establishment. . . . .	9-14
9.7.2	Call Clearing Procedures. . . . .	9-16
9.7	NT Call Handling Examples. . . . .	9-18
9.7.1	Placing an Outbound Call. . . . .	9-18
9.7.2	Receiving an Inbound Call. . . . .	9-19
9.8	CI Call Handling Examples. . . . .	9-20
9.8.1	Originating a Call. . . . .	9-20
9.8.2	A Terminating Call. . . . .	9-21
9.9	Call Processing Errors. . . . .	9-21
9.10	Layer 3 Timers. . . . .	9-22
9.11	Supplementary Services. . . . .	9-24
9.11.1	Facility Messages. . . . .	9-24
9.11.2	Name Identification Service. . . . .	9-26
9.11.3	Release Link Transfer. . . . .	9-27
9.11.4	Enhanced Explicit Call Transfer. . . . .	9-29
9.11.5	Call Deflection. . . . .	9-31
9.11.6	QSIG Call Diversion and Call Forwarding. . . . .	9-32
9.11.7	QSIG Call Transfer by Join. . . . .	9-34
9.11.8	QSIG Path Replacement. . . . .	9-35
9.11.9	QSIG Call Transfer by Rerouting. . . . .	9-36
9.11.10	QSIG Single Step Call Transfer. . . . .	9-37
9.11.11	QSIG Message Waiting Indication. . . . .	9-38
9.11.12	HOLD & RETrieve. . . . .	9-40
9.12	Layer 3 Maintenance Messages. . . . .	9-42
9.12.1	RESTART & RESTART ACKNOWLEDGE. . . . .	9-42
9.12.2	SERVICE & SERVICE ACKNOWLEDGE. . . . .	9-43

10.0	CONTROLLING THE B-CHANNELS.....	10-1
10.1	Overview of the Command Structure. ....	10-1
10.2	Legacy Computer Telephony Busses.....	10-2
10.2.1	SCbus Compatibility. ....	10-3
10.2.2	SCbus Timeslot Assignment. ....	10-3
10.2.3	MVIP Compatibility. ....	10-4
10.2.4	MVIP Compatibility Commands.....	10-5
10.3	Configuring the Board. ....	10-9
10.4	Using the “C” Commands. ....	10-11
10.4.1	Making a Connection.....	10-11
10.4.2	Call Progress Tones.....	10-12
10.4.3	Sending DTMF Tones. ....	10-13
10.4.4	Sending MF Tones.....	10-13
10.4.5	Detecting DTMF Tones.....	10-14
10.4.6	Robbed Bit Signaling.....	10-14
10.4.7	Address Signaling Protocols. ....	10-16
10.4.8	Recording Alert Tone .....	10-17
10.4.9	An Example of an Inbound Call .....	10-18
10.4.10	An Example of an Outbound Call.....	10-19
10.4.11	An Example of Detecting DTMF. ....	10-19
10.5	Voice Resources.....	10-20
10.6	Analog Audio Ports. ....	10-20
10.7	Conferencing. ....	10-20
10.7.1	Controlling a Conference.....	10-23
10.7.2	DTMF Detection and Clamping. ....	10-24
10.7.3	Conferencing Using the MK Command. ....	10-24
11.0	DIAGNOSTICS & ERROR MESSAGES. ....	11-1
11.1	Diagnostic Commands.....	11-1
11.2	Error & Event Messages. ....	11-2
11.3	Diagnostic Tests.....	11-6
APPENDIXES:		
A.	Environmental Specifications. ....	A-1
B..	Custom Tones. ....	B-1

## Infinity Series H.100 T1/Primary Rate ISDN Board Technical Manual

Copyright © American Tel-A-Systems, Inc., January 2011

Printed in U.S.A. All rights reserved.

This document and the information herein is proprietary to American Tel-A-Systems, Inc. It is provided and accepted in confidence only for use in the installation, operation, repair and maintenance of Amtelco equipment by the original owner. It also may be used for evaluation purposes if submitted with the prospect of sale of equipment.

This document is not transferable. No part of this document may be reproduced in whole or in part, by any means, including chemical, electronic, digital, xerographic, facsimile, recording, or other, without the express written permission of American Tel-A-Systems, Inc.

The following statement is in lieu of a trademark symbol with every occurrence of trademarked names: trademarked names are used in this document only in an editorial fashion, and to the benefit of the trademark owner with no intention of infringement of the trademark. "MVIP", "H-MVIP", "MVIP-90", and "MVIP-95" are registered trademarks of GO-MVIP. "SCSA" and "SCbus" are registered trademarks of the Dialogic Corporation. "CT bus" and "ECTF" are registered trademarks of the Enterprise Computer Telephony Forum

American Tel-A-System, Inc.  
800-356-9148

• 4800 Curtin Drive • McFarland, WI 53558 •  
• 4145 North Service Road, Suite 200 • Burlington, Ontario L7L 6A3 •  
• 257M025C •

*The H.100 T1/Primary Rate ISDN Board*



## **FCC Part 15 Requirements**

**WARNING:** This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

## **FCC Part 68 Registration**

This equipment complies with Part 68 of the FCC rules and requirements adopted by the ACTA. On the upper left corner of the front of this equipment is a label that contains, among other information, a product identifier in the format US:AAAEQ##TXXX. If requested, this number must be provided to the telephone company.

This equipment is registered with the FCC under Part 68 as a component device for use with any generic PC Type computer or compatible. In order for FCC registration of this product to be retained, all other products used in conjunction with this product to provide your telephony function must also be FCC Part 68 registered for use with these hosts. If any of these components are not registered, then you are required to seek FCC Part 68 registration of the assembled equipment prior to connection to the telephone network. Part 68 registration specifies that you are required to maintain the approval and as such become responsible for the following:

- any component device added to your equipment, whether it bears component registration or not, will require that a Part 68 compliance evaluation is done and possibly that you have testing performed and make a modification filing to the FCC before that new component can be used;
- any modification/update made by a manufacturer to any component device within your equipment, will require that a Part 68 compliance evaluation is done and possibly that you have testing performed and make a modification filing to the FCC before the new component can be used;
- if you continue to assemble additional quantities of this compound equipment, you are required to comply with the FCC's Continuing Compliance requirements.

The Digital I/F FIC code for this equipment is 02IS5.

The Service Order code for this equipment is 6.oP.

The network Interface Jack for this equipment is an RJ49C.

*The H.100 T1/Primary Rate ISDN Board*

• *viii* •

If this equipment, an XDS T1/PRI Board, causes harm to the network, the telephone company will notify you in advance that temporary discontinuance of service may be required. But if advanced notice isn't practical, the telephone company will notify the customer as soon as possible. Also, you will be advised of your right to file a complaint with the FCC if you believe it is necessary.

The telephone company may make changes in its facilities, equipment, operations, or procedures that could affect the operation of the equipment. If this happens, the telephone company will provide advance notice in order for you to make necessary modifications to maintain uninterrupted service.

Changes to the ISDN protocols offered by the telephone company may require changes to the setup parameters of the board. The board may cease functioning until such changes are made.

If trouble is experienced with this equipment, an XDS T1/PRI Board, for warranty or repair information, please contact:

American Tel-A-System, Inc.  
800-356-9148  
4800 Curtin Drive  
McFarland, WI 53558

If the equipment is causing harm to the telephone network, the telephone company may request that you disconnect the equipment until the problem is resolved.

There are no user serviceable components on the board. All repairs should be accomplished by returning the board to Amtelco with a description of the problem.

Connection to party lines is subject to state tariffs. Contact the state public utility commission, public service commission or corporation commission for information.

If your home has specially wired alarm equipment connected to the telephone line, ensure that the installation of this XDS T1/PRI Board does not disable your alarm equipment. If you have questions about what will disable alarm equipment, consult your telephone company or a qualified installer.

**WARNING:** This device contains Electrostatic Sensitive Devices. Proper care should be taken when handling this device to avoid damage from static discharges.

*The H.100 T1/Primary Rate ISDN Board*

## **Canadian Customers**

CP-01, Issue 8, Part 1

### Section 14.1

**Notice:** “The industry Canada label identifies certified equipment. This certification means that the equipment meets certain telecommunications network protective, operational and safety requirements as prescribed in the appropriate Terminal Equipment Technical Requirements document(s). The Department does not guarantee the equipment will operate to the user’s satisfaction.

Before installing this equipment, users should ensure that it is permissible to be connected to the facilities of the local telecommunications company. The equipment must also be installed using an acceptable method of connection. The customer should be aware that compliance with the above conditions may not prevent degradation of service in some situations.

Repairs of certified equipment should be coordinated by a representative designated by the supplier. Any repairs or alterations made by the user to this equipment, or equipment malfunctions, may give the telecommunications company cause to request the user to disconnect the equipment.

Users should ensure for their own protection that the electrical ground connections of the power utility, telephone lines and internal metallic water pipe system, if present, are connected together. This precaution may be particularly important in rural areas.

**CAUTION:** Users should not attempt to make such connections themselves, but should contact the appropriate electric inspection authority, or electrician, as appropriate.

• x •

## **Product Safety**

The ISDN cord(s) must remain disconnected from the telecommunications system until the card has been installed within a host which provides the necessary protection of the operator.

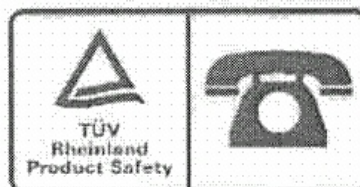
If it is subsequently desired to open the host equipment for any reason, the ISDN cord(s) must be disconnected prior to effecting access to any internal parts which may carry telecommunications network voltages.

This board is not intended to be connected directly to the PSTN network. Connection must be made by way of an approved CSU interface device. It is the responsibility of the CSU to provide primary high voltage protection.

*The H.100 T1/Primary Rate ISDN Board*

## European Approvals

### CE Approval



This apparatus is approved by CTR4 for connection to an ISDN using ISDN primary access as specified in this section under the conditions set out in this document.

This apparatus, when operated as ISDN terminal equipment has been designed for operation on EURO ISDN S1 primary access connections for point to point connections, S1 PABX connections complying with EURO ISDN and for EURO ISDN S1 point to point connections. It supports most of the services of the EURO ISDN standard (ETSI DSS1).

### EN55022 EMC declaration

This is a class B product. In a domestic environment, this product may cause radio interference in which case the user may be required to take adequate measures.

No changes or modifications to the T1/PRI card are allowed without explicit written permission from American Tel-A-Systems, Inc., as these could void the end user's authority to operate the device.

## Declaration of Conformity

**Model Number:** 257L130 XDS H.100 Four Port T1 Board w/o voice  
257L119 XDS H.100 Four Port T1 Board with voice  
257L131 XDS H.100 Eight Port T1 Board w/o voice  
257L120 XDS H.100 Eight Port T1 Board with voice  
257L132 XDS H.100 Four Port E1 Board w/o voice  
257L121 XDS H.100 Four Port E1 Board with voice  
257L133 XDS H.100 Eight Port E1 Board w/o voice  
257L122 XDS H.100 Eight Port E1 Board with voice

**Standards to which the conformity is declared:**

EN55024, EN55022 Class A, EN60950,  
FCC Part 15 Class A, ICES-003-Class A,  
TBR4, TBR12, and TBR13

**The undersigned declares that the equipment specified above:**

- conforms to the above Standards,
- is in conformity to all the essential requirements of Directive 1999/5/EC. .

**Manufacturer:** Amtelco

**Company name:** American Tel-A-Systems Inc.  
DBA - Amtelco

**Address:** 4800 Curtin Drive  
McFarland, Wisconsin 53558 USA

**Signature:**



**Printed Name:** Paul N. Henning

**Position:** Director of Research and Development

**Date:** 15 March 2007

*The H.100 T1/Primary Rate ISDN Board*

# 1.0 Introduction

The Infinity Series H.100 T1/Primary Rate Interface ISDN Board is designed to provide either four or eight T1/Primary Rate ISDN (Integrated Services Digital Network) interfaces connected to the H.100 bus on a board with the PCI bus form factor. The board is also equipped with DSP resources to provide tone generation and detection.

There is a version of the board in both the PCI and PCI Express form factor. This manual covers both versions. For the purposes of this manual, the PCI and PCI Express busses will be referred to as the PCI bus except where it is important to differentiate between them. This manual is also applicable to an earlier version of the board with an Intel 386 processor.

When operating in the T1 mode, each interface or span provides 24 channels. Robbed bit signaling support is provided for loop start, ground start and E&M formats. Address signaling protocols such as immediate or wink start are also supported. Each interface may operate in either a CI (customer interface) or NT (network or central office) mode.

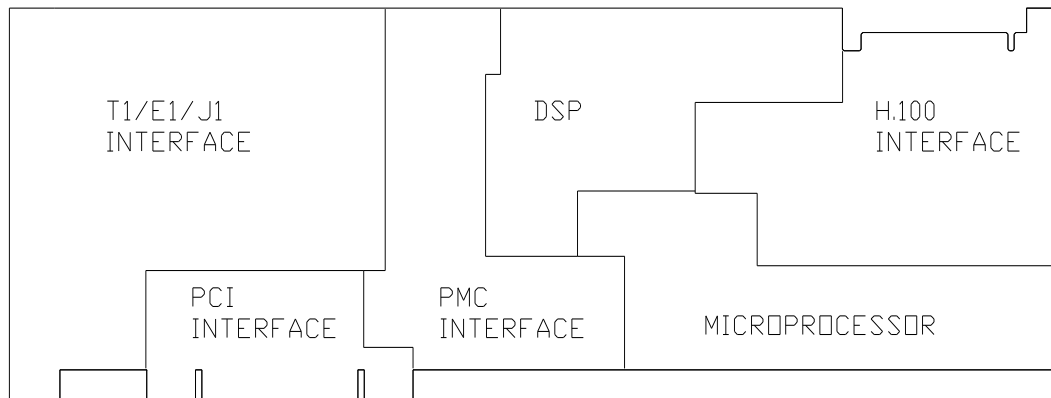
When operating in Primary Rate ISDN mode each interface provides a D channel for call control signaling and 23 64 kbps. B channels for either speech or circuit switched data. Each interface can be configured as either a CI (Customer Interface) or NT (Network Termination) interface. The board provides complete support for the ISDN Layer 1 and Layer 2 protocols, as well as optional support for the Layer 3 protocols as defined in Q.931.

The H.100 bus was devised by the Enterprise Computer Telephony Forum (ECTF) to provide a single telecom bus for the entire industry.

*The H.100 T1/Primary Rate ISDN Board*

It is intended for add-in boards using the PCI form factor. A wide variety of boards are available from a number of different vendors. The H.100 bus also has compatibility modes that allow for connection to legacy computer telephony busses such as the MVIP-90 bus and SCbus.

The board is equipped with a processor that can be used to control the lower level functions of the board. The host PC controls the board using messages passed through dual-ported RAM. The board shares a common message passing and control scheme with other Infinity Series H.100 boards. This scheme is also compatible with legacy XDS boards for the MVIP-90 bus and SCbus.



*Figure 1: The H.100 T1/Primary Rate ISDN Board Functional Areas*

## 1.1 Features and Capabilities

This section presents an overview of the features and capabilities of the Infinity Series H.100 T1/Primary Rate Interface ISDN Board.

### 1.1.1 The Physical Interface

Four or eight independent interfaces are provided on the board. Each interface on the board provides a complete T1 or Primary Rate ISDN

*The H.100 T1/Primary Rate ISDN Board*



interface. This interface can be configured under software control as either a piece of customer premise equipment to interface to a central office or PBX, or as a network termination to interface to customer premise equipment such as a channel bank or a PBX. Layer 1 support is provided by the board to handle all the details of framing and clocking in all of the common framing formats.

Each interface provides 24 channels when operating as a T1 interface or 23 B-channels when operating as a Primary Rate ISDN interface. These channels can be switched to the H.100 bus, to each other, or to one of the DSP resources. In addition, when operating as an ISDN interface, there is a 64 kbps D channel which is used for signaling.

### **1.1.2 Signaling Protocols**

The Infinity Series H.100 T1/PRI Board is capable of supporting robbed bit signaling in the T1 mode or ISDN signaling in the PRI mode.

Robbed bit signaling uses two or four bits stolen from each channel in each superframe. This limits the bandwidth in each channel to 56 kbps. Robbed bit formats are available that emulate a number of common analog line types. These types are loop start, ground start, and E&M. Either the customer or network side of such lines can be emulated. In addition to hook status and ringing, various address signaling protocols are also supported such as immediate and wink start.

ISDN uses the LAPD protocol on the D channel for call control. The Layer 2 protocol defines the mechanism used for the exchange of messages between terminal equipment and the network termination. The board manages the details of this protocol while providing access to the contents of Layer 3 messages.

Call control is handled by the exchange of Layer 3 messages as defined by the Q.931 standard. The H.100 T1/PRI Board allows the application to directly read and write the raw binary information field of these

*The H.100 T1/Primary Rate ISDN Board*

messages. Optionally, a higher level interface is provided that masks some of the details of these messages, making for a simplified applications interface. This is done through the use of “D” messages.

### **1.1.3 DSP Functions**

The H.100 T1/PRI ISDN Board is equipped with DSP’s that perform a variety of functions. DTMF and Energy detectors are available for each B channel. DTMF generators are available for each B channel for signaling purposes. Call Progress tones are also available, with dial-tone, busy, reorder, and audible ringback being provided as well as silence and a 1004 Hz. calibration tone.

Boards that are configured for four spans may also be provisioned for conferencing. This feature is not available on the eight span configuration.

### **1.1.4 The H.100 Bus**

The H.100 bus is a digital bus for transporting PCM (Pulse Code Modulation) signals between telephony boards. It was created by the ECTF to provide a common bus structure for future development that would end the “bus wars” between the various legacy busses such as the SCbus and the MVIP bus.

PCM is a standard method of digitizing phone signals. It involves encoding each channel at an 8 kHz rate using eight bits. The signals from multiple channels are then combined into a frame. On the H.100 bus, each frame consists of 128 channels or timeslots. The bit rate of the H.100 bus is 8.192 MHZ. Thirty-two wires, also called streams, each carrying 128 timeslots, are combined to form the bus, and provide a total of 4096 timeslots. Two timeslots are required for a full conversation, one for each talker. For compatibility purposes with legacy busses, the first sixteen streams can also run at either 2.048 or 4.096 MHZ. with 32

or 64 timeslots respectively.

In addition to the streams, a number of other signals necessary to maintain synchronization between all the boards in the system are carried on the bus. These signals provide the clocking and framing information. Redundant clocks are provided to aid in recovery if the primary clock should fail. For interoperation with the SCbus, MVIP bus, or H-MVIP bus a number of compatibility clock signals are also defined.

The H.100 bus consists of a 68 conductor ribbon cable that is used to interconnect the boards in the system. This cable connects to a header at the upper right hand edge on each board.

### **1.1.5 Clock Modes**

The H.100 T1/PRI Board can operate in a variety of clock modes. Modes are available so that the master clock can either be derived from the H.100 bus, one of the Interfaces, or be provided by the H.100 T1/PRI Board. The clock redundancy and clock fallback functions of the H.100 bus are also supported so that the H.100 T1/PRI Board can be set to provide a clock to the H.100 bus if the master clock on that bus should fail.

### **1.1.6 Message Passing**

The board occupies 8K of memory space on the host PC. This 8K may reside anywhere within the PC's address space. As a PCI board, the address and interrupt of the board is assigned at boot time. The message passing scheme used by the Infinity Series H.100 T1/PRI Board is identical to that of the other Infinity Series H.100 boards, allowing for the easy combination of a variety of Infinity Series H.100 boards in a single system.

The message passing scheme and message syntax of Infinity Series H.100 boards is similar to that of the older XDS series of MVIP and SCbus boards. At the driver and API level, support is provided for both series of boards so that the H.100 boards may interoperate with legacy boards using a common interface.

### **1.1.7 Flash EAROM for Firmware**

The firmware for both the main processors and for the DSP's is contained in Flash EAROM. This allows for easy upgrades of the firmware on the board in the field without requiring time consuming downloads every time a system boots. Once reprogrammed, the new firmware is retained even when the power is removed. The original, factory programmed firmware is also retained on board and can be accessed by installing a jumper.

### **1.1.8 EEPROM for Configuration Information**

ISDN interfaces can require a substantial amount of information to be programmed into the system. These includes items such as the signaling protocols and DNs (Directory Numbers) associated with each interface as well as board configuration information such as the type of span (CI or NT) and the protocol level supported. To reduce the burden on the application, the board has an EEPROM capable of providing non-volatile storage for this information. This allows the board to automatically configure itself upon a restart.

### **1.1.9 Analog Audio Ports**

The H.100 T1/PRI Board may be equipped with two analog audio ports. These ports may be used to connect up to two line level analog audio sources to H.100 bus timeslots for functions such as providing music on hold.

## 1.2 How to Use This Manual

The first five sections in this manual are organized in the order you should read and use them to get started with your H.100 T1/PRI ISDN Board. We recommend that you begin with these three steps.

1. Follow the instructions in section 2.0 (Quick Start) and 3.0 (Installation). These sections will tell you if your board is operating correctly within your system. You don't need to be familiar with the board's command set to complete this step.
2. Read section 4.0 (Initialization) to initialize the board within your system. Your application must perform these initialization procedures whenever you power-up your PC in order for the board to communicate with the PC.
3. Read section 5.0 (Communications with the PC) for an overview of how to communicate with the H.100 T1/PRI ISDN Board. Section 5.0 includes a summary of the commands for constructing your application and details concerning system interrupts.

Before you can actually build your application, read section 6.0 (The H.100 Bus and Clock Modes), 7.0 (T1 Framing & Clocking), 8.0 (ISDN Layer 2 Protocol), 9.0 (Using "D" Messages for Layer 3) and 10.0 (Controlling the B-Channels). These sections explain, with practical examples, how the H.100 T1/PRI ISDN Board operates and how to use the command set to achieve the desired results.

Section 11.0 explains diagnostic and error messages that may occur.

The Appendix contains information on power requirements and interfacing that will be helpful when installing your H.100 T1/PRI ISDN Board.

this page intentionally left blank

## 2.0 Quick Start

This section describes the first steps you should perform to determine if your Infinity Series H.100 T1/Primary Rate ISDN Board is communicating correctly with your PC system. You can perform this quick check without securing the board to the PC chassis or connecting any cables.

The exact procedure will vary depending on which operating system you are running. For each operating system, drivers are required to interface to the boards. The drivers supplied by Amtelco have tests built into them to verify communications with the boards. These drivers also come supplied with utility programs that allow the developer to test communications with the board. Please consult the appropriate documentation for the driver and operating system you are using.

### Quick Start Procedure

1. Make sure the PC power is off, then insert the board into a PCI slot.
2. Turn on your PC.
3. If the Amtelco driver is not already installed, install it now, following the instructions supplied with the driver.
4. Most Amtelco drivers will display a list of boards that are installed (see the documentation for the particular driver that you are using). If the H.100 T1/Primary Rate ISDN Board is listed, skip to step 6.
5. If the board is not listed, there may be a problem with the

*The H.100 T1/Primary Rate ISDN Board*

board not being seated correctly in the motherboard. There may also be a problem with a memory or interrupt conflict. Power down the PC and check that the board is properly seated in the connector and repeat steps 1-4. If this does not remedy the problem, try removing any other computer telephony boards in the system. If your PC is unable to find the board, consult the number at the end of this section.

6. Run the program “xdsutil” supplied with the driver. Send the message “IN” to the H.100 T1/Primary Rate ISDN Board. The board should respond with the message “IA”.
7. Send the message “VC” to the board. Verify that the Receive Message reads: VCxxxxvvvvPTN8 (where xxxxxvvvv is a variable indicating the firmware version). The four span version will read VCxxxxvvvvPTN4.
8. If the Communications screen shows the correct command responses, your H.100 T1/Primary Rate ISDN Board is communicating with the PC. You may now power down the computer and attach the necessary cables (see section 3.4)

For technical assistance, call Amtelco at 1-608-838-4194 ext.168.



### 3.0 Installation

This section describes how to install your Infinity Series H.100 T1/Primary Rate ISDN Board into your PC and how to use the jumpers, headers, and connectors. Before you begin the installation procedure, be sure to test the board as described in section 2.0 (Quick Start).

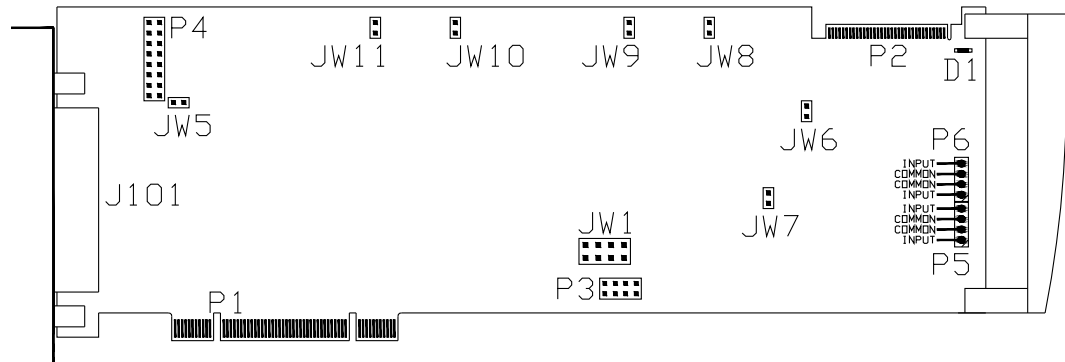


Figure 2: Location of Jumpers, Headers, and Connectors for boards with the 386 processor

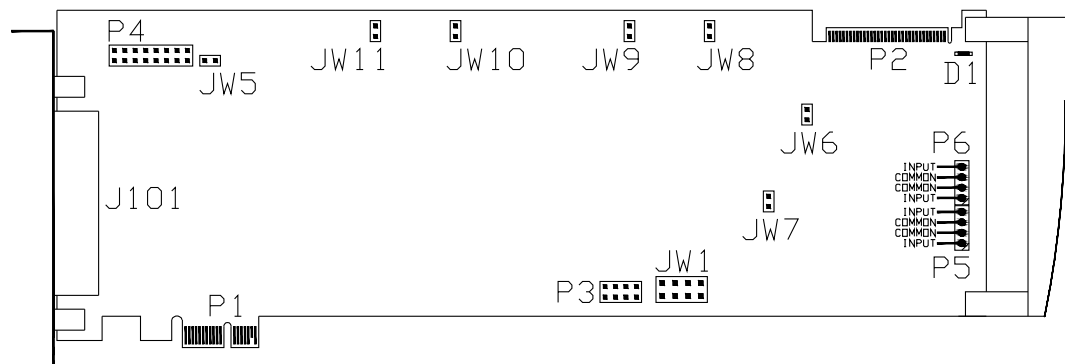


Figure 2a: Location of Jumpers, Headers, and Connectors for boards with the ARM Processor

The H.100 T1/Primary Rate ISDN Board

## **3.1 PCI Configuration**

As Infinity Series boards conform to the PCI standards, there are no switches to set to configure the H.100 T1/PRI Board's memory address, I/O addresses, or interrupt. The PC's BIOS will automatically configure the board at boot time to avoid conflicts with other boards in the system.

## **3.2 Jumpers & Headers**

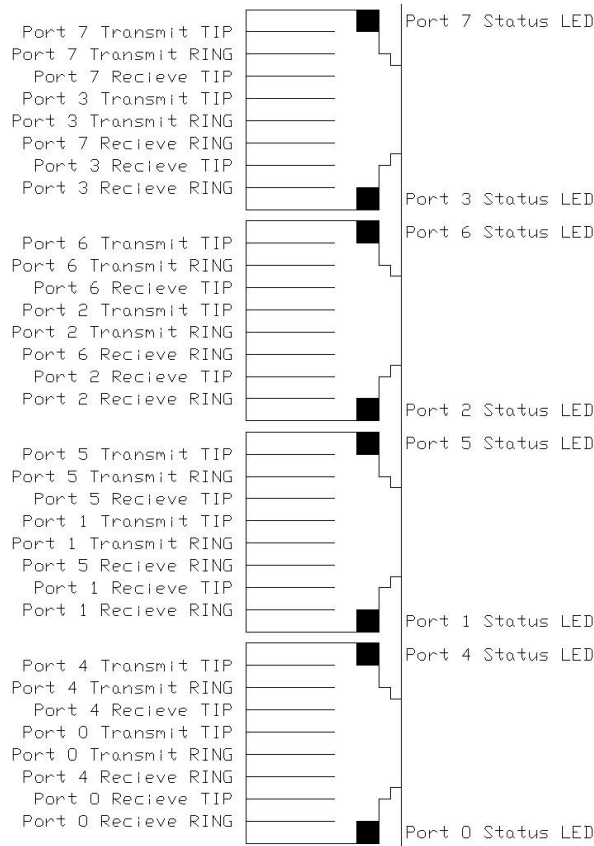
The following is a complete list of all jumpers for the H.100 T1/PRI Board:

- JW1-1**      Firmware Select. If firmware has been downloaded to the board, this jumper selects whether the downloaded firmware or the factory default firmware is used. When this jumper is installed, the factory default firmware is executed whenever the board is reset. When the jumper is not installed, the downloaded firmware will be executed after a reset if it is present. If no downloaded firmware is present, the factory default firmware is executed after reset.
  
- JW1-2**      Undefined, reserved for future use
  
- JW1-3**      Undefined, reserved for future use.
  
- JW1-4**      If installed, the firmware emulates the local stream usage of earlier boards in the "MO" command.
  
- JW5**        This jumper is used for factory testing and should not have jumpers installed.

- JW6-11** DSP Firmware Select. There is one jumper for each of the six DSPs which may be present on the board. If a jumper is installed the factory DSP firmware for that DSP is executed after reset. Otherwise, the downloaded firmware is executed if present. See JW1-1.
- P3** Diagnostic port. Never install jumpers here.
- P4** This header is used for programming internal logic and should never be jumpered.

### **3.3 Connectors: P2, P5, P6, and J101**

- P2** H.100 bus. The H.100 bus connector (P2) is a standard H.100 bus header. Use an H.100 bus ribbon cable to connect the H.100 T1/PRI Board to other H.100 boards within the same PC chassis.
- P5, P6** Analog Audio Ports. These connectors are used to connect an external audio source to analog audio port 0 (P5) and analog audio port 1 (P6). Audio is connected to pin 1, pin 4 or both, while pin 2 and 3 should be connected to the audio ground wire.
- J101** Interface Connections. This connector is a quad RJ-45 type connector. Each connector contains two T1 or Primary Rate ISDN ports or interfaces. On eight port boards, the first jack contains ports 0 & 4, the second jack 1 & 5, etc. Each jack also contains two status LEDs, one for each interface. See Figure 3.



*Figure 3: J101 Pinout and Status LED Locations*

### 3.4 Installation

To install the H.100 T1/PRI Board in your system:

1. Do not connect the board to the PSTN. Follow the procedures described in section 2.0 to verify the operation of the board.
2. If the quick check is successful, turn off the PC power and remove the board from the chassis.
3. Install any necessary board jumpers. See section 3.2 for jumper configurations.

*The H.100 T1/Primary Rate ISDN Board*

4. Reinsert the board into the chassis. Seat it properly in a PCI or PCI Express slot as appropriate in the PC chassis and tighten the screw in the back of the board to secure it. Do not connect the board to the PSTN.
5. Connect the H.100 cable to P1.
6. Reinstall the PC cover. Connect the PC to the mains supply using a socket-outlet with protective earthing connection and connect any additional protective earthing used.
7. Connect the telephone cable(s) to J1. The telephone cable terminates in an RJ-45 male connector.

If it is subsequently desired to open the host equipment chassis for any reason, the PSTN cable must be detached prior to effecting access to any internal parts which may carry telecommunications network voltages.

this page intentionally left blank

## 4.0 Initialization

This section describes the procedures necessary to initialize the system and enable the PC to communicate with the Infinity Series H.100 T1/Primary Rate ISDN Board. XDS drivers will implement some of these procedures.

### 4.1 PCI Initialization

The system BIOS is responsible for recognizing PCI boards and mapping them into the I/O and memory spaces as required. It is also responsible for assigning interrupts to the board. This is done through a set of on board registers which contain information specifying the memory, I/O, and interrupt needs of the board. A set of BIOS functions exist for accessing this information. A detailed description of these functions can be found in the *PCI BIOS Specification* published by the PCI SIG, the PCI Special Interest Group.

Normally, the drivers supplied by Amtelco will take care of the process of finding Infinity Series boards and establishing communications. The information in the rest of this subsection is for background only.

The configuration registers of every PCI board contain a vendor ID and device ID code. These codes are unique to each board vendor. All Infinity Series H.100 boards have the same vendor and device IDs. The vendor ID is 14E3h. For the PCI versions of the board the device ID is 0101h, while for the PCI Express version the device ID is 0301h. A BIOS function exists that will find each instance of a particular vendor and device ID, and which returns with a bus and device number. The bus and device number is then used in functions to read the configuration registers.

*The H.100 T1/Primary Rate ISDN Board*

The configuration registers contain information on the base address of the memory and I/O assigned to the board by the BIOS. A PCI board may have up to six different base addresses. On Infinity Series H.100 boards, the first two base addresses are used by the PCI bus interface logic. The third base address which is contained in registers 18-1Bh contains the memory location of the dual-ported memory that is used to pass messages. The interrupt information is contained in register 3Ch. The information in these configuration registers can be used by a driver to address the board.

## 4.2 Initialization Commands

The H.100 T1/PRI Board is initialized by sending a sequence of command messages to the board. The process of sending messages is described in detail in Section 5.0, but normally it is accomplished either with a low-level driver XMT command or the API function **xds\_msg\_send**. Response messages are read using the low-level driver RCV command or the API function **xds\_message\_receive**.

To enable communications with the H.100 T1/PRI Board, an **IN** command message should be sent to the board. The board will respond with an **IA** message.

The board may be reset using the command message **RA**. The board will respond with an **RA** message.

Your application can now configure the H.100 T1/Primary Rate ISDN Board using these commands

<u>Command</u>	<u>Purpose</u>
<b>SCmsabb(c)</b>	Sets the clock mode for the board. The parameter <i>m</i> is the clock-mode. The parameter <i>s</i> is the clock sub-mode. The parameters <i>a</i> , <i>bb</i> , and <i>c</i> are used to

*The H.100 T1/Primary Rate ISDN Board*



specify additional clock control information such as compatibility modes, clock rates, local network, and CT\_NETREF settings. The default mode on power-up or restart is mode 0. See section 6.0 of this manual for details of clock mode arguments.

**SBabcd**

This command is used to define the clock rate for the lower 16 streams for compatibility with the SCbus or MVIP bus. The parameters a, b, c, d are used to set the rate for streams 0-3, 4-7, 8-11, and 12-15 respectively. The default value is 8.192 MHZ. The possible settings are:

- 0 - 2.048 MHZ., 32 timeslots per stream
- 1 - 4.096 MHZ., 64 timeslots per stream
- 2 - 8.192 MHZ., 128 timeslots per stream

**SEx**

Sets the encoding mode for the board. The parameter x can be either M for Mu-Law as used in North America and Japan, or A for A-Law as used in Europe and Asia. The default value is for Mu-Law.

**SFddfzbs**

Sets the framing parameters for each span. The span number is specified in dd. The f parameter specifies the framing, "D" for D4, or "E" for ESF. The z parameter specifies the zero suppression mode, "A" for AMI, or "B" for B8ZS. The b parameter specifies the line build out. Valid values are:

- 0 - DSX-1 (0-133ft.)/0dB CSU
- 1 - DSX-1 (133-266ft.)
- 2 - DSX-1 (266-399 ft.)
- 3 - DSX-1 (399 to 533 ft.)
- 4 - DSX-1 (533 to 655 ft.)

- 5 - -7.5 dB CSU
- 6 - -15 dB CSU
- 7 - -22.5 dB CSU

The *s* parameter specifies the signaling mode for the port, “N” for no signaling (used with NFAS), “R” for robbed-bit signaling, and “P” for Primary Rate ISDN.

**SPxxttd(pn)**

Sets the robbed-bit line protocol for each channel. The channel is specified by *xx*. The *t* parameter specifies the line type, “E” for E&M, “G” for ground start, “L” for loop start, or “N” for none. The *d* parameter specifies the direction, “S” for the FXS or CPE side, and “O” for the FXO or CO side. If the line type is E&M, an additional address digit protocol may be specified with the *p* parameter. A value of “I” specifies immediate start and “W” specifies wink start. The *n* parameter specifies the number of address digits to be expected on incoming calls. For MF signaling the *n* parameter should be set to “M.”

**ST(xx...xx)**

Set the span type for each interface on the board. Span types can be “N” for NT for a network termination or CO interface, “C” for CI or Customer Interface, or “U” for undefined for unused ports. An “\*” may be used to indicate no change. The N type is used when emulating a central office switch. The C type is used when interfacing to the central office or when the board is acting as customer premise equipment. A span type parameter must be included for each of the interfaces on the board.

**S**Tab****

Controls termination. Parameters a and b control termination for the H.100 and MVIP bus respectively. When set to E, termination is enabled and when set to D, termination is disabled. Boards on the end of the H.100 cable should have termination enabled. When operating in MVIP compatibility mode, the MVIP termination should be enabled when the following condition exists:

For systems with five or fewer MVIP Bus connections and less than 90 pF load on the clock lines, it is adequate to place the circuit board that is the master clock source at one end of the cable and provide termination on the circuit board which is physically at the other end of the cable.

On systems with more than five MVIP connections or more than 90 pF load on the clock lines, both ends of the cable should be electrically terminated with the 1000 Ohm/ 1000 pF termination. No other boards should terminate these lines.

**S**Xstt****

This command is used to set the base timeslot on the SCbus when reserving timeslots to transmit on. The parameters s and tt are hexadecimal numbers setting the lowest timeslot of the block of timeslots reserved for the board. This command should only be used when operating in the SCbus mode.

If the optional Layer 3 support is to be used, it will be necessary to set several additional parameters for each port or B channel. The following commands are used for that purpose:

**S**L(II...II)****

This command is used to select the protocol level supported by the board. The choices are:

- 2 - Layer 2 support only
- 3 - Layer 3 support (generic NI-1)

- A - Avaya support
- D - DMS-100 NI-1 support
- N - NI-2 support
- S - Siemens CoreNet support
- T - Telcordia Generic Requirements

If Layer 2 support is selected, it is the responsibility of the application to compose and interpret the Q.931 messages using the auxiliary mailboxes. Layer 3 support is provided through the “D” messages. (See section 8.0). The default is Layer 2 support. The protocol level parameter must be included for each of the ports on the board.

**SSabcdeh**

This command is used to set optional Layer 3 behavior. Each option a-h can be set to either “Y” or “N” for yes or no. The default setting is no. The options are:

- a Disables the automatic sending of a CONNECT ACK message in response to a CONNECT message. The CONNECT ACK may be sent with a “DCxxA” command.
- b The calling party number in received SETUP messages is transmitted in a separate message of the form **D#xxC<calling#>**.
- c Selects between the NSAP and User-Specified formats for subaddress elements. NSAP is the default.
- d Disables Layer 3 timers if set to yes. This may be useful for debugging applications.

- e If set to yes, the last Layer 3 message received is placed in dual-ported memory at an offset of 1000h before being parsed.
- f-h Reserved for future use.

**SDxx(dn)** This command sets the default directory number for each B channel xx. For channels on NT spans, this is the number that will be used as the default called directory number for calls originating on the port. For channels on spans defined as a CI, this will be the number used as the calling party number. The directory number will be the calling number used for calls originating from the span. Directory numbers may be up to 15 digits in length.

### **4.3 Configuration Memory**

Much of the configuration information used to initialize the board is fixed in nature, such as the span types, framing parameters, line protocols and Directory Numbers. To simplify initialization of the board, the configuration can be stored in an onboard EEPROM. This information can be recalled upon a restart of the board eliminating the need to send this information to the board each time an application runs. To control the EEPROM, three commands are provided. These are:

**SMS** This command saves the current configuration including span types, framing parameters, the protocol level, and Directory Numbers.

**SML** This command will cause the configuration saved in the EEPROM to be loaded into the processor memory. It is not necessary to use this command on a restart as the information saved in the EEPROM will automatically be

loaded into the processor memory.

**SMC** This command will clear the EEPROM. If this command is used, the board will not read the EEPROM on a power up or restart and all required configuration information will have to be sent from the application.

If the board configuration is saved in the EEPROM, it will still be necessary to send the **IN** and **SC** messages to enable messages and set the clock mode.

## **4.4 Backup & Restoration of Configuration Information**

Because of the amount of configuration information on the board, a mechanism has been provided to allow for the transfer of this information through the dual-ported memory to a host application. This will facilitate moving configuration information from one board to another in case a board needs to be replaced. It will also serve as another level of configuration backup.

The EEPROM on the H.100 T1/PRI board is organized as 8K 8 bit bytes. As the entire dual-ported RAM is also 8K, the data being transferred must be segmented to fit in the dual ported RAM without interfering with the mailboxes and other data contained in dual ported memory. The segment size is 4K, thus requiring two segments to be read or written to transfer all the configuration information contained in the EEPROM. The data is transferred through a 4K buffer in the dual ported memory that has a beginning offset of 0. The flag used to control transfers is the same flag used for downloading .HEX or .SRE program files to the board and has an offset of 1EFFh.

The command to write data from the board to the dual-ported RAM is **SMWx** where x is the segment number. The process is to place a non-

zero value in the flag byte and then issue the command. When the flag is 0, the data in the buffer is valid and may be accessed by the host application.

The command to read data from the dual-ported RAM to the board is **SMRx** where x is the segment. The process is for the application to place a non-zero value in the flag byte, place the data in the buffer, and issue the command. When the flag is 0, the data in the buffer has been transferred to an internal buffer, and the transfer buffer is available for more data.

The read command places the configuration data in an intermediate buffer. Both segments must be read in before the data can be transferred to the final location. This is done with a command of the form **SMB**. Note that this places the data in active memory, but does not save the information to the EEPROM. To do this, the “SMS” command must be used after the data has been transferred. If this is not done, the configuration information will be lost if the board is powered down or restarted. The complete sequence would be:

- set the flag
- place the first 4K of data in dual-ported memory
- send SMR0
- when the flag is clear, set the flag
- place the second 4K of data in dual-ported memory
- send SMR1
- when the flag is clear, send SMB
- send SMS to save the configuration in EEPROM

this page intentionally left blank



## 5.0 Communicating with the PC

This section describes how the PC communicates with the Infinity Series H.100 T1/Primary Rate ISDN Board. It includes the definitions for the H.100 T1/PRI Board commands and responses along with a description of the mailboxes used for messaging.

The board is controlled by the host PC through a system of four mailboxes. The messages consist of short NUL-terminated ASCII strings, which are easy for the host software to compose and parse. The board is capable of buffering up to eight messages in either direction and can drive an interrupt line when it has a message for the host. Messages may not exceed 32 characters.

There are two main mailboxes, one for messages to the board and one for messages from the board, and two flags associated with them. A 00h in a flag byte indicates the mailbox is free, a non-zero value indicates that the mailbox is occupied. The mailboxes and their flags are contained in an 8K block of dual-ported memory at the following offsets:

receive mailbox	1F80h
transmit mailbox	1FC0h
transmit flag	1FFCh
receive flag	1FFEh

The board's base address is determined by reading PCI Configuration Space offset 18h. The 32-bit value at this location is the base address for the dual-ported memory on the board.

To send a message, the message is placed in the transmit mailbox and the transmit flag is set to 01h. To read a message, the message is

*The H.100 T1/Primary Rate ISDN Board*

removed from the receive mailbox and the receive flag is cleared to 00h. This will clear the interrupt hardware.

In addition to the two main mailboxes, there are two auxiliary mailboxes that are used for passing Layer 3 messages to and from the board. These mailboxes are only used in conjunction with the “LC” command and response messages in the main mailboxes. Each of these auxiliary mailboxes begins with two bytes indicating the length of the Layer 3 message (low order byte first) and 260 bytes for the body of the message (the maximum size of a information field for Q.931 messages). The mailbox for messages to the board has an offset of 1400h and the mailbox for messages from the board is at an offset of 1600h.

To send a Layer 3 message, the transmit flag for the main mailbox must be clear. The message and its length is first placed in the auxiliary mailbox. An “LC” command is then placed in the main mailbox, and finally, the transmit flag is set to 01h. It is important that both the Layer 3 message and the command be placed in the appropriate mailboxes before the transmit flag is set.

The presence of a Layer 3 message from the board is indicated by an “LC” message in the main mailbox. The Layer 3 message must be read before the receive flag is cleared. Failure to do so may result in the message being overwritten.

## **5.1 Commands and Responses Protocol**

This section describes the necessary step-by-step procedures for the PC to send a command to the board and to remove a response from the board.

### **5.1.1 Sending Commands to the Board**

The basic steps to sending a command to the H.100 T1/PRI ISDN Board are:

1. Build a command. Broadly speaking, a command is a string of ASCII characters with a NUL (00h) termination character.
2. Check the transmit flag. If the flag is 0, continue with the next step to put the command in memory. If the flag is not 0, wait until the flag is 0.
3. Insert the command in transmit mailbox memory beginning at the address of the transmit mailbox.
4. Write 01h to the transmit flag. This notifies the board that a message is waiting.

### **5.1.2 Reading Messages From the Board**

1. Check the receive flag. If the flag is 0, there is no message. If it is non-zero, a message is waiting. Continue with the next step to read the message.
2. Remove the message from memory, starting at the address of the receive mailbox. Messages are NUL terminated ASCII strings.
3. Write 0h to the receive flag.

### 5.1.3 Reading Board Information

A range of board information is included in memory so that it can be checked without sending a message:

<u>Type of Information</u>	<u>Offset Address</u>
Board ID	1F00-1F03
Firmware Version	1F04-1F07
Configuration Flags	1F08
Processor Type	1F0F
Number of transmit timeslots	1F10-1F11
Base timeslot	1F12-1F13
Clock mode settings	1F18-1F1B
Board configuration	1F1C-1F1E
Clock status bits	1F1F
ID Code (serial number)	1F30-1F3F

Note: The number of reserved transmit timeslots, and base timeslots are used only in the SCbus compatibility mode when reserving transmit timeslots.

The board stores its identity upon power up or a hardware restart. The phrase **Restart PTN (c) Amtelco 2010** appears in the receive mailbox (**PTN8 (c) Amtelco 2002** on 386 boards). The receive flag is not set and no interrupt is generated.

## 5.2 Interrupts

The H.100 T1/Primary Rate ISDN Board can generate an interrupt to the PC indicating that a message is available. The interrupt for PCI boards is assigned by the BIOS or Operating System at boot time. The assignment is dependent on which PCI slot the board is in. The interrupt line is usually shared by more than one device. If multiple Infinity Series boards are installed they may or may not all share the same

*The H.100 T1/Primary Rate ISDN Board*

interrupt line.

In order for an Infinity Series board to send interrupts to the PC, the PCI Interface circuit on the board must be programmed to enable interrupts. This is accomplished by setting bits 0 and 6 in the board's Interrupt Control/Status Register. This is a byte-wide register located at an offset of 4Ch from PCI Base Address 0. PCI Base Address 0 is contained in PCI Configuration Space register 10h. The Base address is a 32-bit value and is mapped into memory.

When an Infinity Series board sends a message, it generates a local interrupt to the PCI Interface circuit on the board. If the PCI Interface circuit has been programmed to generate interrupts to the PC, the local interrupt is passed through to the PC. When the PC receives an interrupt, its Interrupt Service Routine (ISR) should check the Infinity board's receive flag to see if a message is pending (i.e. the receive flag is non-zero). It should then process the message for the board and write a 0 to the board's receive flag.

### **5.2.1 Interrupt Initialization**

1. Clear the board's receive flag.
2. Read the PCI Base Address 0 from PCI Configuration Space offset 10h (this must be a 32-bit access).
3. Set bits 0 and 6 of PCI Base Address 0 + 4Ch. Do not modify any other bits in this register. This register is a byte-wide memory mapped register.

### **5.2.2 Step-by-Step Interrupt Processing Summary**

1. Check to see if the receive flag is non-zero.
2. Remove the message from the receive mailbox.

*The H.100 T1/Primary Rate ISDN Board*

3. Write 0h to the receive flag.
4. Re-enable the interrupt controller on the PC.

## **5.3 Commands and Responses**

This section gives a general overview of the H.100 T1/PRI ISDN Board commands and responses. The commands are grouped by function and then listed in alphabetical order by two-letter command. Refer to sections 7.0 through 10.0 for examples and explanations of how to use these commands.

### **5.3.1 Characteristics of Command Strings**

- ▶ All commands consist of null (00h) terminated ASCII strings.
- ▶ There are no spaces or other delimiters between parameters in the commands.
- ▶ All letters in command strings must be UPPERCASE unless otherwise noted.
- ▶ Lowercase monospaced letters (such as `xx`) in the following command references represent parameters within commands. Each letter represents one ASCII digit.
- ▶ Numeric parameters are always hexadecimal numbers.

### **5.3.2 Command Parameters**

The table below documents the common parameters for many of the commands listed in the next sections.

**Common Command Parameters**

<b>Parameter</b>	<b>Definition</b>	<b>Values</b>
xx	B-Channel number	00-F7h(8 span) 00-77h(4 span)
dd	D-Channel number, span	00-07h(8 span) 00-03h(4 span)
sstt	H.100 bus stream & timeslot number, ss = stream, tt = timeslot on stream	ss = 00-1Fh tt = 00-7Fh
aabb	H.100 bus stream & timeslot number, aa = stream, bb = timeslot on stream	aa = 00-1Fh bb = 00-7Fh
bsstt	MVIP-95 terminus, b = bus ss = stream, tt = timeslot	b = H, L ss = 00-1F tt = 00-7F

**5.3.3 Commands from the PC to the H.100 T1/PRI Board**

Note that sections 7.0-9.0 of this manual provide supplemental information for the commands and messages documented here.

**Alarm Commands**

- ABddc**                    Blue alarm event on dd, e - event, C - clear, S - set
- AYddc**                    Yellow alarm on dd, e - event, C - clear, S - set

**Analog Audio Port Commands**

- AD0x**                    Disable Audio Port x
- AE0xsstt**                Enable Audio Port x to stream ss timeslot tt
- AGxxg**                    Set gain for Audio Port x to g

**B-Channel Commands**

<b>CAxxsstt</b>	Set B-channel xx to listen to stream ss timeslot tt														
<b>CBxx</b>	Set channel xx to hold if not busy, else return SBxx														
<b>CCxxssttaabb(B)</b>	Connect B-channel xx to stream ss timeslot tt and from stream aa timeslot bb, B - opt. alerting tone														
<b>CDxx</b>	Disconnect B-channel xx														
<b>CFxxd</b>	Generate a hook flash on channel xx of duration d														
<b>CHxx</b>	Put B-channel xx on hold														
<b>CLxx</b>	Set B-channel xx to detect DTMF, play dial tone														
<b>CKxx(ds)</b>	Send MF dial string ds to B-channel xx, valid tones are 0-9, K, P, S, *, and #.														
<b>CLxxo</b>	Set B-channel xx to detect DTMF digits, option o F - turn detection off M - monitor mode, maintain existing connections Q - regular mode, suppress dial tone														
<b>CMxx</b>	Enable audio channel xx, no change to signaling bits														
<b>CPxxxy</b>	Give B-channel xx call progress tone y where y values are: <table border="0" style="margin-left: 2em;"> <tr> <td>North American tones</td> <td>ETSI tones</td> </tr> <tr> <td>0 - dial tone</td> <td>6 - ETSI dial tone</td> </tr> <tr> <td>1 - reorder</td> <td>7 - ETSI congestion tone</td> </tr> <tr> <td>2 - busy</td> <td>8 - ETSI busy tone</td> </tr> <tr> <td>3 - audible ringback</td> <td>9 - ETSI ringback</td> </tr> <tr> <td>4 - Digital Milliwatt</td> <td>A - U.K. ringback</td> </tr> <tr> <td>5 - silence</td> <td>B - Japanese ringback</td> </tr> </table>	North American tones	ETSI tones	0 - dial tone	6 - ETSI dial tone	1 - reorder	7 - ETSI congestion tone	2 - busy	8 - ETSI busy tone	3 - audible ringback	9 - ETSI ringback	4 - Digital Milliwatt	A - U.K. ringback	5 - silence	B - Japanese ringback
North American tones	ETSI tones														
0 - dial tone	6 - ETSI dial tone														
1 - reorder	7 - ETSI congestion tone														
2 - busy	8 - ETSI busy tone														
3 - audible ringback	9 - ETSI ringback														
4 - Digital Milliwatt	A - U.K. ringback														
5 - silence	B - Japanese ringback														
<b>CRxx</b>	Generate ringing on channel xx (robbed bit)														
<b>CSxxsstt</b>	Play call progress tone to stream ss timeslot tt, xx values 00-0Bh correspond to progress tones 0-B														
<b>CSxxD</b>	Disconnect call progress tone xx from H.100 bus														
<b>CTxx(ds)</b>	Send dial string ds to B-channel xx, valid tones in string are: 0-9, *, #, A-D, U - upper tone (941 Hz.), L - lower tone, (697 Hz.), X - short pause, P - pause														



**CVxxffflffllnnffrr** Generate a custom tone on B-channel xx,  
fff - freq. 1st & 2nd tone, ll - level 1st & 2nd tone  
nn - on duration, ff - off duration, rr - repetitions

**CXxxsstt(B)** Set B-channel xx to transmit on stream ss timeslot  
tt, B - optional alerting tone

### **Layer 3 “D” Commands for NT Spans**

**DAxxp** ALERtIng message on B channel xx, progress  
ind. p

**DCxx(p)(#)** CONNect message on B channel xx, opt. progress  
p, opt. connected #

**DCxxA** CONNect ACK message on B channel xx

**DDxx** DISConnect message on B channel xx, normal  
clearing, tones off

**DDxxcc(rrrr)(#)** DISConnect message on B channel xx, cause cc,  
opt. call ref. rrrr, opt. connected #

**DFxxAid#** FACILITY, ctActive, id - invoke id, # - connected  
number

**DFxxCide#(A)** FACILITY, ctComplete, id - invoke id, e - end  
designator, # - redirection number, opt. A if  
alerting

**DFxxF1idro#** FACILITY, divertingLegInformation1, id -invoke  
id, r - reason, o - subscription option, # - diverting  
to #

**DFxxF3id** FACILITY, divertingLegInformation3, id - invoke  
id

**DFxxlid** FACILITY, ctIdentify invoke, id - invoke id

**DFxxlid,c-#** FACILITY, ctIdentify results, id - invoke id, c -  
call identity, # - redirecting party number

**DFxxLEidee** FACILITY, RLT return error ee, id - invoke id

**DFxxLJidpp** FACILITY, RLT reject, pp - problem, id - invoke  
id

**DFxxLR02** FACILITY B-channel xx, RLT Third Party accept

**DFxxN(n)** FACILITY on B channel xx, opt. notification ind.

<b>DFxxPid,c-#</b>	FACILITY B-channel xx, pathReplacePropose, id - invoke id, c - call identity, # - redirecting #
<b>DFxxRid,c-#</b>	FACILITY, ctInitiate, id - invoke id, c - call identity, # - redirecting party number
<b>DFxxSidm#/#</b>	FACILITY (SSCT) on B-channel xx, id - invoke id, m - mode, # - transfer to number, /# - transferred #
<b>DFxxTidIdcid</b>	FACILITY (ECT identity results), xx - B-channel, id - invoke id, dcid - D-channel identifier
<b>DFxxUid#</b>	FACILITY, ctUpdate, id - invoke id, # - redirection number
<b>DGxxArrrr</b>	RETRieve ACKnowledge on B chnl. xx, call ref. rrrr
<b>DGxxHrrrr</b>	RETRieve on B channel xx, call ref. rrrr
<b>DGxxRccrrrr</b>	RETRieve REJect cause cc, call reference rrrr
<b>DHxx</b>	HOLD message on B channel xx
<b>DHxxA</b>	HOLD ACKnowledge message on B channel xx
<b>DHxxRcc</b>	HOLD REJect message on B channel xx, cause cc
<b>DNxxn(#)</b>	NOTIFY, notification ind. opt. calling number #
<b>DPxx(p)</b>	CALL PROCEeding message on B channel xx, optional progress indicator p
<b>DPxxPccp</b>	PROGress message on B channel xx, cause cc, progress indicator p
<b>DQxx</b>	Query default Directory Number for B-channel xx
<b>DRxxcc(rrrr)(#)</b>	RELEase COMplete message on B channel xx, cause cc, opt. call ref. rrrr, opt. connected #
<b>DRxxRcc(rrrr)(#)</b>	RELEase on B channel xx, cause cc, opt. call ref. rrrr, optional connected number #
<b>DSxxbp#(/#)</b>	SETUP message, bearer capability b, progress indicator p, calling number(/called number)
<b>DTAt[text]</b>	Add display text to buffer with display tag t
<b>DTC</b>	Clear display text buffer
<b>DTDt[text]</b>	DMS-100 display, t - name type
<b>DTNt[text]</b>	Create new display text element in buffer with display tag t

<b>DTQA[<i>text</i>]</b>	Add display (ANSI T1.607) text to buffer, <i>text</i>
<b>DTQC</b>	Clear display buffer
<b>DTQL[<i>text</i>]</b>	Create new display (ANSI T1.607) in buffer, <i>text</i>
<b>DXxx</b>	Status Query for B-channel <i>xx</i>
<b>DZxx(A)</b>	RESTART interface <i>xx</i> , opt. A restart all interfaces
<b>DZxxC</b>	RESTART specified B-channel <i>xx</i>
<b>D@F2idcr#</b>	Put divertingLegInformation 2 in buffer, <i>id</i> - invoke <i>id</i> , <i>c</i> - diversion count, <i>r</i> - reason, <i>#</i> - diversion <i>#</i>
<b>D@Nido&lt;<i>name</i>&gt;</b>	Put <i>name</i> in buffer, <i>id</i> - invoke <i>id</i> , <i>o</i> - operation
<b>D@nido&lt;<i>name</i>&gt;</b>	Put <i>name</i> in buffer (Telcordia Generic), <i>id</i> - invoke <i>id</i> , <i>o</i> - operation
<b>D@I&lt;<i>CallID</i>&gt;</b>	Put RLT Call Id in buffer
<b>D#C[#]</b>	Put Calling Party <i>#</i> in buffer
<b>D#Rr[#]</b>	Put Redirecting <i>#</i> in buffer, redirect reason <i>r</i>

**Layer 3 “D” Commands for CI Spans**

<b>DAxxp(rrrr)</b>	ALERTing message on B channel <i>xx</i> , progress indicator <i>p</i> , opt. call reference <i>rrrr</i>
<b>DCxxp(rrrr)(#)</b>	CONNect message on B channel <i>xx</i> , progress <i>p</i> , opt. call ref. <i>rrrr</i> , opt. connected <i>#</i>
<b>DCxxA</b>	CONNect ACK message on B channel <i>xx</i>
<b>DDxx</b>	DISConnect message B channel <i>xx</i> , normal clearing
<b>DDxxcc(rrrr)(#)</b>	DISConnect message on B channel <i>xx</i> , cause <i>cc</i> , opt. call ref. <i>rrrr</i> , opt. connected <i>#</i>
<b>DFxxAid#</b>	FACILITY, ctactive, <i>id</i> - invoke <i>id</i> , <i>#</i> - connected number
<b>DFxxCide#(A)</b>	FACILITY, ctcomplete, <i>id</i> - invoke <i>id</i> , <i>e</i> - end designator, <i>#</i> - redirection number, opt. A if alerting
<b>DFxxDidr[#]</b>	FACILITY (CD) on B-channel <i>xx</i> , <i>id</i> - invoke <i>id</i> , <i>r</i> - deflection reason, <i>#</i> - number deflected to

<b>DFxxEidrrrr</b>	FACILITY (ECT) on B-channel xx, id - invoke id, rrrr - call reference of third party
<b>DFxxF1idro#</b>	FACILITY, divertingLegInformation1, id - invoke id, r - reason, o - subscription option, # - diverting to #
<b>DFxxF3id</b>	FACILITY, divertingLegInformation3, id - invoke id
<b>DFxxIid</b>	FACILITY, ctIdentify invoke, id - invoke id
<b>DFxxIid,c-#</b>	FACILITY, ctIdentify results, id - invoke id, c - call identity, # - redirecting party number
<b>DFxxL&lt;CallID&gt;</b>	FACILITY (RLT Third Party) on B-channel xx
<b>DFxxPidc-#</b>	FACILITY B-channel xx, pathReplacePropose, id - invoke id, c - call identity, # - redirecting #
<b>DFxxRid,c-#</b>	FACILITY, ctInitiate, id - invoke id, c - call identity, # - redirecting party number
<b>DFxxSidm#/#</b>	FACILITY (SSCT) on B-channel xx, id - invoke id, m - mode, # - transfer to number, /# - transferred #
<b>DFxxTidrrrr</b>	FACILITY (ECT) on B-channel xx, id - invoke id, rrrr - call reference of third party
<b>DFxxTidrrrr-dcid</b>	FACILITY (ECT) xx -B-channel, id - invoke id rrrr - call ref., dcid - D-channel identifier
<b>DFxxTidI</b>	FACILITY (ECT - identifier request), xx - B-channel, id - invoke id
<b>DFxxUid#</b>	FACILITY, ctdate, id - invoke id, # - redirection number
<b>DGxxArrrr</b>	RETRieve ACKnowledge on B channel xx, call ref. rrrr
<b>DGxxHrrrr</b>	RETRieve on B channel xx, call reference rrrr
<b>DGxxRccrrrr</b>	RETRieve REJECT cause cc, call reference rrrr
<b>DHxx</b>	HOLD message on B channel xx
<b>DHxxA</b>	HOLD ACKnowledge message on B channel xx
<b>DHxxRec</b>	HOLD REJECT message on B channel xx, cause ss
<b>DKxxk(k)</b>	INFORMATION message, keypad digit(s) k(k)

<b>DPxx(p)</b>	CALL PROCEEDing message on B channel xx, optional progress indicator p
<b>DPxxPccp</b>	PROGRESS message on B channel xx, cause cc, progress indicator p
<b>DQxx</b>	Query default Directory Number for B-channel xx
<b>DRxxcc(rrrr)(#)</b>	RELEASE COMPLETE message on B channel xx, cause cc, opt. call ref. rrrr, opt. connected #
<b>DRxxRcc(rrrr)(#)</b>	RELEASE message on B channel xx, opt. cause cc, call reference rrrr, opt. connected #
<b>DSxxbp(#)</b>	SETUP message, overlap sending, bearer cap. b, progress p, opt. calling party #
<b>DSxxbp(#)/#</b>	SETUP message, bearer capability b, progress p, opt. calling party number #, called party number #
<b>DTDt[<i>text</i>]</b>	DMS-100 display, t - name type
<b>DXxx</b>	Status Query for B-channel xx
<b>DZxx(A)</b>	RESTART interface xx, opt. A restart all interfaces
<b>DZxxC</b>	RESTART specified B-channel xx
<b>D@F2idcr#</b>	Put diverting LegInformation 2 in buffer, id - invoke id, c - diversion count, r - reason, # - diversion #
<b>D@Mido#(/#)(=m)</b>	Put MWI operation in buffer, id - invoke id, o - operation, user #, opt. msg. center #, opt. m number of messages
<b>D@Nido&lt;name&gt;</b>	Put name in buffer, id - invoke id, o - operation
<b>D@nido&lt;name&gt;</b>	Put name in buffer (Telcordia Generic), id - invoke id, o - operation
<b>D@R</b>	Put invoke RLT Operation Indication in buffer
<b>D#C[#]</b>	Put Calling Party # in buffer
<b>D#Rr[#]</b>	Put Redirecting # in buffer, redirect reason r

**Interrupt Control Commands**

<b>IN</b>	enable transmit interrupts and messages
<b>IF</b>	disable transmit interrupts and messages

### Conference Commands

<b>Kahhooooiiii(anto)</b>	Conference to handle hh, input timeslot iiiii, output on timeslot oooo, attenuation a, noise threshold n, tone mode t, output mode o
<b>KDoooo</b>	Disable conference output to timeslot oooo
<b>KEiiii(cc)</b>	Enable DTMF detection on conferenced timeslot iiiii, cc = clamping time, 0.2 sec. increments
<b>KEiiiiD</b>	Disable DTMF detection on conferenced timeslot iiiii
<b>KGhhan</b>	Set attenuation to a and noise threshold to n for conference handle hh
<b>KIhhiiii(ant)</b>	Add input timeslot iiiii to conference with handle hh attenuation a, noise threshold n, tone mode t
<b>KMhhoooo(an)</b>	Monitor conference with handle hh on timeslot oooo, attenuation a, noise threshold n
<b>KUhh</b>	Dissolve conference with handle hh
<b>KXhhiiii</b>	Remove timeslot iiiii as an input to conference hh

### Layer 3 Message Commands

<b>LCddsstt</b>	Layer 3 command for span dd, ss = SAPI, tt = TEI
-----------------	--

### MVIP Compatibility Commands

<b>MBhhD</b>	Disable alerting tone detector/generator hh
<b>MBhhE</b>	Enable alerting tone detector/generator hh
<b>MDhhD</b>	Disable DTMF detector hh (MVIP) (00-FF)
<b>MDhhE(cc)</b>	Enable DTMF detector hh (MVIP) (00-FF)
<b>MGhh(ds)</b>	Generate the dial string (ds) with generator hh (MVIP) (00-FF)
<b>MKhhccm(anto)</b>	Conference control, hh - conference handle, cc - CCA, m = D - disable, m = E - enable a - attenuation, n - noise threshold, t - tone mode, o - output mode
<b>MObssttD</b>	Set_output disable mode, bsstt - output terminus

<b>MObssttEbsstt</b>	Set_output enable mode, bsstt - output terminus, bsstt - input terminus
<b>MObssttPpp</b>	Set_output pattern mode, bsstt - output terminus, pp - pattern value
<b>MTD</b>	Disable output to the CT Bus (tristate)
<b>MTE</b>	Enable output to the CT Bus (tristate)

**Query Commands**

<b>QA</b>	Query alarms
<b>QEdd</b>	Query error counts for span dd
<b>QHHSstt</b>	Query T8110, sstt = H.100 stream and timeslot
<b>QHLSstt</b>	Query T8110, sstt = local bus stream and timeslot
<b>QObsstt</b>	Query Output for terminus bsstt
<b>QPdx(msg)</b>	Send message to DSP d, send only bits 0-3 of x
<b>QRxx</b>	Query call reference and state of B-channel xx
<b>QXxxx</b>	Query SCbus transmit timeslot for B-channel xxx

**Reset Commands**

<b>RA</b>	reset all (resets spans, DSP functions, H.100 bus)
<b>RBxx</b>	reset B-channel xx (Layer 3)
<b>RD</b>	reset DSP (resets DSP chip only)
<b>RPdd</b>	reset span dd

**Setup Commands**

<b>SAddaaabb</b>	Set Network Specific Facility info for span dd
<b>SBabcd</b>	Set bit rate for streams 0-3, 4-7, 8-11, and 12-15 0 - 2.048 MHz. 1 - 4.096 MHz. 2 - 8.192 MHz.
<b>SCmsabb(c)</b>	Set clock mode m submode s, arguments a, bb, & c
<b>SDxx(#)</b>	Set the default directory number for B channel xx
<b>SEa</b>	Set Encoding mode a, M = Mu-Law, A = A-Law
<b>SFddfzbs</b>	Set framing for span dd, f = framing, D - D4, E - ESF z = zero suppression, A - AMI, B - BZ8S

	b = build out, 0-7
	s = signaling, N - None, P - PRI, R - robbed bit
<b>SI</b> ttffffllffffllnnffnff	Set custom information (Call Progress) tone ii, ffff = freq. 1st & 2nd tone, ll = level, 1st & 2nd tone, nn = 1st & 2nd on duration, ff = 1st & 2nd off dur.
<b>SL</b> (ll...ll)	Set protocol layer for each span where l values are 2 - Layer 2 support 3 - Layer 3 support A - Avaya support D - DMS-100 Pre NI-1 support N - National ISDN-2 support S - Siemens CoreNet support T - Telcordia Generic Requirements support
<b>SMB</b>	Transfer configuration buffer to active memory
<b>SMC</b>	Clear the EEPROM configuration contents
<b>SML</b>	Load the EEPROM contents onto the board
<b>SMRa</b>	Read segment a from dual-ported memory
<b>SMS</b>	Save the configuration in EEPROM
<b>SMWa</b>	Write segment a to dual-ported memory
<b>SN</b> (ic..ic)	Set Non-facility associated signalling information
<b>SP</b> xxtd(pn)	Set line protocol for channel xx; t = line type, E - E&M, G - ground start, L - loop start, N - none; d = direction, O - FXO (CO), S - FSX (CPE); p = optional address protocol, I - immediate start, W - wink start; n = optional number address digits, n = M for MF signaling
<b>SS</b> abcdefgh	Set system options a-h to 'Y' or 'N' a - disable auto CONNECT ACK b - separate calling party # message c - NSAP or user specified subaddress format d - disable Layer 3 timers e-h reserved



<b>ST(xx...xx)</b>	Set span types for each span where x values are: C - Customer Interface (CI) N - Network Termination (NT) U - Undefined/unused * - No change to port type
<b>STab</b>	Set bus termination, a = H.100 bus, b = MVIP bus
<b>SXstt</b>	Set SCbus base timeslot to stream s, timeslot tt
<b>SX</b>	Clear SCbus base timeslot assignment

**TEI Management Commands**

<b>TDdttt</b>	Disconnect Link for TEI tt on span dd
<b>TEdttt</b>	Establish Link for TEI tt on span dd

**Version Requests**

<b>VA</b>	Checksum of alternate segment request
<b>VC</b>	Version request
<b>VD</b>	DSP version request

**Download Commands**

<b>@xxxx</b>	Download 1K block to address xxxx
<b>@Es</b>	Erase segment s
<b>GA</b>	Jump to Alternate Program
<b>GM</b>	Jump to Main Program
<b>@Ws</b>	Write from RAM to segment s

**Diagnostics**

<b>XBddEB</b>	Insert a single bit error on span dd
<b>XBddEIr</b>	Set BERT error rate to r on span dd
<b>XBddPpp..pplcc</b>	Set BERT pattern pp..pp, pattern length ll, alternate word count cc, on span dd
<b>XBddRC</b>	Request BERT receive error count on span dd
<b>XBddRD</b>	Disable BERT receive function on span dd
<b>XBddREei</b>	Select BERT receive mode e, invert bits i
<b>XBddRTttttt</b>	Enable BERT receive function on timeslots ttttt
<b>XBddS</b>	Resynchronize BERT receiver on span dd

<b>XBddTD</b>	Disable BERT transmit function on span dd
<b>XBddTEei</b>	Select BERT transmit mode e, invert bits i
<b>XBddTTttttt</b>	Enable BERT transmit function on timeslots ttttt
<b>XCddc</b>	Generate loopback code c on span dd (D4 NT only)
<b>XCddcc</b>	Generate bit-patterned data link message cc on span dd (ESF only)
<b>XCdd400000</b>	Clear per channel loopback mode 4
<b>XFnn</b>	Set diagnostic control flags to nn
<b>XLddm</b>	Set loopback mode m
<b>XLdd4aabbcc</b>	Set per channel loopback on span dd; aa, bb, & cc specify channel
<b>XOddrrnn</b>	Set DS2155 register rr to nn for span dd
<b>XSxxs(C)</b>	Set Service status for B-channel xx, s = status I - in service, M - maintenance, O - out of service; opt. C if for specified B channel, else whole span
<b>XSxxsD</b>	Set Service status for D-channel xx, s = status I - in service, M - maintenance, O - out of service;

#### **5.4.4 Responses from the H.100 T1/PRI ISDN Board**

##### **Acknowledgments**

<b>IA</b>	Acknowledge interrupts enabled
<b>RA</b>	Reset all acknowledged
<b>RPdd</b>	Reset port dd acknowledged
<b>SMx</b>	EEPROM operation x = 0 - failure, 1 success

##### **Alarm Events**

<b>ABdd</b>	Blue alarm event on dd
<b>AGdd</b>	Green alarm event on dd (no alarms active)
<b>ARdd</b>	Red alarm event on dd
<b>AYdd</b>	Yellow alarm on dd

**Layer 3 “D” Responses for NT Spans**

<b>DAxxprrrr</b>	ALERTing on B channel xx, progress indicator p, call reference rrrr
<b>DCxxprrrr(#)</b>	CONNect on B channel xx, prog. ind p, call ref. rrrr, opt. connected #
<b>DCxxArrrrr</b>	CONNect ACKnowledge on B channel xx, call ref. rrrr
<b>DDxxcc(rrrr)(#)</b>	DISConnect B channel xx, cause cc, opt. call ref. rrrr, opt. connected #
<b>DFxxcid(tt)</b>	FACILITY message on xx, component c, invoke id, opt. tag value tt
<b>DFxxAid#</b>	FACILITY, ctActive, id - invoke id, # - redirecting number
<b>DFxxCide#(A)</b>	FACILITY, ctComplete, id - invoke id, e - end designator, # - redirection number, opt. A if alerting
<b>DFxxDidr[#]</b>	FACILITY (CD) on B-channel xx, id - invoke id, r - deflection reason, # - number deflected to
<b>DFxxEidrrrr</b>	FACILITY (ECT) on B-channel xx, id - invoke id, rrrr - call reference of third party
<b>DFxxF1ro#</b>	FACILITY, divertingLegInformation1, r - reason, o - subscription option, # - diverting to number
<b>DFxxF2cr#(/nam)</b>	FACILITY divertingLegInformation2, c - diverting count, r - diversion reason, # - diverting number, /name - optional diverting name
<b>DFxxF3id</b>	FACILITY, divertingLegInformation3, id - invoke id
<b>DFxxIid</b>	FACILITY, ctIdentify invoke, id - invoke id
<b>DFxxIid,c-#</b>	FACILITY, ctIdentify results, id - invoke id, c - call identity, # - redirecting party number
<b>DFxxMido#(/#)(=m)</b>	FACILITY MWI operation, id - invoke id, o - operation, user #, opt. msg. center #, opt. m number of msgs.
<b>DFxxNo&lt;name&gt;</b>	FACILITY Name element on xx, operation o, name

<b>DFxxNR</b>	FACILITY RLT Operation Indication element
<b>DFxxNT&lt;CallID&gt;</b>	FACILITY RLT Third Party element
<b>DFxxPid,c-#</b>	FACILITY B-channel xx, pathReplacePropose, id - invoke id, c - call identity, # - redirecting #
<b>DFxxRid,c-#</b>	FACILITY B-channel xx, ctInitiate, id - invoke id, c - call identity, # - redirecting party #
<b>DFxxSidm#/#</b>	FACILITY (SSCT) on B-channel xx, id - invoke id, m - mode, # - transfer to number, /# - transferred #
<b>DFxxTidrrrr</b>	FACILITY (ECT) on B-channel xx, id - invoke id, rrrr - call reference of third party
<b>DFxxTidrrrr-dcid</b>	FACILITY (ECT), xx - B-channel, id - invoke id, rrr - call ref., dcid - identifier
<b>DFxxTidI</b>	FACILITY (ECT - identity request), xx - B- channel, id - invoke id
<b>DFxxUid#</b>	FACILITY, CTUpdate, id - invoke id, # - redirection number
<b>DGxxArrrr</b>	RETRieve ACKnowledge on B chnl. xx, call ref. rrrr
<b>DGxxHrrrr</b>	RETRieve on B channel xx, call ref. rrrr
<b>DGxxRccrrrr</b>	RETRieve REJect cause cc, call reference rrrr
<b>DHxxrrrr</b>	HOLD message on B channel xx, call reference rrrr
<b>DHxxArrrr</b>	HOLD ACKnowledge message on B channel xx, call reference rrrr
<b>DHxxRccrrrr</b>	HOLD REJect message on B channel xx, cause cc, call reference rrrr
<b>DKxxk(k)</b>	INFORmation message, keypad digit(s) k(k)
<b>DPxxprrrr</b>	CALL PROCEeding message on B channel xx, progress indicator p, call reference rrrr
<b>DPxxPccp(rrrr)</b>	PROGress message on B channel xx, cause cc, progress indicator p, opt. call ref. rrrr
<b>DQxx(DN)</b>	Default DN for B channel xx, response to DQ query

<b>DRxxcc(rrrr)(#)</b>	RELEase COMplete message B channel xx, cause cc, opt. call reference rrrr for calls not assigned a B-channel, opt. connected #
<b>DrxxRcc(rrrr)(#)</b>	RELEase message, cause cc, opt. call reference rrrr for calls not assigned B channel, opt. connected #
<b>DSxxbp(#)</b>	SETUP message, bearer capability b, progress p, opt. calling #, (overlap sending)
<b>DSxxbp(#)/#</b>	SETUP message, bearer capability b, progress p, optional calling #, called # (Enbloc sending)
<b>DTxxt[text]</b>	Display text with display tag t for B-channel xx
<b>DXxxrrrr</b>	STATUS QUERY message on B-channel xx
<b>DXxxccssrrrr</b>	STATUS message on B channel xx, cause cc, call state ss, call reference rrrr
<b>DZxxA</b>	RESTART all interfaces associated with B-channel
<b>DZxxC</b>	RESTART specified B-channel xx
<b>DZxxI</b>	RESTART interface starting with B-channel xx
<b>DZxxAA</b>	RESTART ACKnowledge, all interfaces
<b>DZxxAC</b>	RESTART ACKnowledge, specified B-channel xx
<b>DZxxAI</b>	RESTART ACKnowledge, interface starting with xx
<b>D#xxC[#]</b>	Calling party # on B-channel xx
<b>D#xxRr[#]</b>	Redirecting # on B-channel xx, r - redirecting reason

**Layer 3 “D” Responses for CI Spans**

<b>DAxxprrrr</b>	ALERTing on B channel xx, progress indicator p, call reference rrrr
<b>DCxxprrrr(#)</b>	CONNect on B channel xx, progress indicator p, call reference rrrr, opt. connected #
<b>DCxxArrrr</b>	CONNect ACKnowledge on B chnl. xx, call ref. rrrr
<b>DDxxcc(rrrr)</b>	DISConnect on B channel xx, cause cc, opt. call ref. rrrr

<b>DFxxcid(tt)</b>	FACILITY message on xx, component c, invoke id, opt. tag value tt
<b>DFxxAid#</b>	FACILITY, ctActive, id - invoke id, # - redirecting number
<b>DFxxCide#(A)</b>	FACILITY, ctComplete, id - invoke id, e - end designator, # - redirection number, opt. A if alerting
<b>DFxxF1ro#</b>	FACILITY, divertingLegInformation1, r - reason, o - subscription option, # - diverting to number
<b>DFxxF2cr#(/nam)</b>	FACILITY divertingLegInformation2, c - diverting count, r - diversion reason, # - diverting number, /name - optional diverting name
<b>DFxxF3id</b>	FACILITY, divertingLegInformation3
<b>DFxxIid</b>	FACILITY, ctIdentify invoke, id - invoke id
<b>DFxxIid,c-#</b>	FACILITY, ctIdentify results, id - invoke id, c - call identity, # - redirecting party number
<b>DFxxNo&lt;name&gt;</b>	FACILITY Name element on xx, operation o, name
<b>DFxxNEidee</b>	FACILITY RLT return error ee, id - invoke id
<b>DFxxNI&lt;Callid&gt;</b>	FACILITY RLT Operation Indication result Callid
<b>DFxxNJidppee</b>	FACILITY RLT Reject on B-channel xx, id - invoke id, pp - problem tag, ee - error
<b>DFxxNT</b>	FACILITY RLT Third Party accepted on B-channel xx
<b>DFxxPidc-#</b>	FACILITY B-channel xx, pathReplacePropose, id - invoke id, c - call identity, # - redirecting #
<b>DFxxRidc-#</b>	FACILITY B-channel xx, ctInitiate, id - invoke id, c - call identity, # - redirecting party #
<b>DFxxSidm#/#</b>	FACILITY (SSCT) on B-channel xx, id - invoke id, m - mode, # - transfer to number, /# - transfered #
<b>DFxxTidIdcid</b>	FACILITY (ECT -identifier results), xx - B-channel, id - invoke id, dcid - identifier
<b>DFxxUid#</b>	FACILITY, ctUpdate, id - invoke id, # - redirection number

<b>DGxxArrrr</b>	RETRieve ACKnowledge on B chnl. xx, call ref. rrrr
<b>DGxxHrrrr</b>	RETRieve on B channel xx, call ref. rrrr
<b>DGxxRccrrrr</b>	RETRieve REJect cause cc, call reference rrrr
<b>DHxxrrrr</b>	HOLD message on B channel xx, call reference rrrr
<b>DHxxArrrr</b>	HOLD ACKnowledge message, call reference rrrr
<b>DHxxRccrrrr</b>	HOLD REJect message, cause cc, call reference rrrr
<b>DNxxn(#)</b>	NOTIFY message on B-channel xx, notification indicator n, optional. calling #
<b>DPxxprrrr</b>	CALL PROCEeding message on B channel xx, progress indicator p, call reference rrrr
<b>DPxxPccp(rrrr)</b>	PROGress message on B channel xx, cause cc, progress indicator p, opt. call ref. rrrr
<b>DQxx(DN)</b>	Default DN for B channel xx, response to DQ query
<b>DRxxcc(rrrr)(#)</b>	RELEase COMplete message, cause cc, optional call reference rrrr for calls not assigned a B-channel
<b>DRxxR(rrrr)(#)</b>	RELEase message, opt. call reference rrrr for calls not assigned a B channel
<b>DSxxbp#(/#)</b>	SETUP message on B channel xx, bearer capability b, assigned B channel xx, progress indicator p, calling #, called # optional
<b>DSxxAprrrr</b>	SETUP ACKnowledge, prog. ind. p, call ref. rrrr
<b>DTxxl[text]</b>	Display (ANSI T1.607) , line l, text
<b>DTxxt[text]</b>	Display text with display tag t for B-channel xx
<b>DXxxrrrr</b>	STATUS QUERY message on B-channel xx
<b>DXxxccssrrrr</b>	STATUS message, cause cc, call state ss, call ref. rrrr
<b>DZxxA</b>	RESTART all interfaces associated with B-channel
<b>DZxxC</b>	RESTART specified B-channel xx
<b>DZxxI</b>	RESTART interface starting with B-channel xx

<b>DZxxAA</b>	RESTART ACKnowledge, all interfaces
<b>DZxxAC</b>	RESTART ACKnowledge, specified B-channel xx
<b>DZxxAI</b>	RESTART ACKnowledge, interface starting at xx
<b>D#xxC[#]</b>	Calling party # on B-channel xx
<b>D#xxRr[#]</b>	Redirecting # on B-channel xx, r - redirecting reason

### **Error Messages**

<b>ECxx</b>	Clock error bit xx
<b>EDddy</b>	Diagnostic error y on span dd
<b>EKhhee</b>	Conference error ee on conference handle hh
<b>ELddy</b>	Layer 1 or 2 error y on span dd
<b>EMddy</b>	MDL error y on span dd (Layer 2 protocol errors)
<b>ESddss</b>	Line Interface Unit event bits ss on span dd
<b>ETxx</b>	Address digit timeout on B-channel xx
<b>ETxxe</b>	Layer 3 timer timeout event e on B-channel xx
<b>EWxx</b>	Wink timeout on B-channel xx
<b>EXddss</b>	Elastic store event bits ss on span dd

### **Layer 3 Message Received Response**

<b>LCddsstt</b>	Layer 3 command on D channel dd, ss = SAPI, tt = TEI
-----------------	--

### **Query Responses**

<b>QAaaaaaaa</b>	Query Alarm response, a - alarm state G - no alarm Y - Yellow alarm B - Blue alarm R - Red alarm
<b>QEdd(data)</b>	Query Error Counts response, span dd
<b>QHsstt0c0ttt00dd</b>	Reply to T8110 H.100 bus query, c - control bits, ttt - tag, dd - data memory contents
<b>QHLsstt0c0ttt00dd</b>	Reply to T8110 local bus query, c - control bits, ttt - tag, dd - data memory contents



<b>QObssttm(bsstt)</b>	Query_output reply, bsstt - output terminus, m - mode (bsstt) input terminus
<b>QPd(text)</b>	DSP diagnostic responses from DSP d
<b>QRxxxxrrrrss</b>	Query B-channel reference response rrrr = call reference, ss = call state
<b>QXxxxstt</b>	SCbus transmit timeslot for B-channel xxx
<b>QXxxxZ</b>	No SCbus transmit timeslot set for B-channel xxx

### **B-Channel State Change Messages**

<b>SBxx</b>	Channel xx busy (response to CBxx message)
<b>SCxx</b>	Connect on B-channel xx acknowledged
<b>SDxxxdd...d</b>	Address digits ddd..d received on B-channel xx
<b>SExx</b>	Tone string on B-channel xx completed
<b>SEXhh</b>	Tone string from generator hh ended
<b>SFxx</b>	Off hook detected on channel xx (robbed bit)
<b>SHxx</b>	Hold on B-channel xx acknowledged
<b>SIxx</b>	Disconnect on B-channel xx acknowledged
<b>SLxx</b>	Listen on B-channel xx acknowledged
<b>SMxx</b>	Audio enable on channel xx acknowledged
<b>SNxx</b>	On-hook detected on channel xx (robbed bit)
<b>SRxx</b>	Ringing detected on channel xx (robbed bit)
<b>STxxd</b>	DTMF digit d detected on B-channel xx
<b>STiiiiid</b>	DTMF digit d detected on input timeslot iiii
<b>STXhhd</b>	DTMF digit d detected by detector hh (MVIP)
<b>SWxx</b>	Wink detected on channel xx
<b>SXxx</b>	Transmit on B-channel xx acknowledged

### **Diagnostic Responses**

<b>VAxxxx</b>	Checksum of the alternate segment
<b>VCxxxxyyyyPTN8</b>	Version response, 8 span board
<b>VCxxxxyyyyPTN4</b>	Version response 4 span board xxxx = checksum of main segment, yyyy = version number, PT = board type, N = North American standards
<b>VDxxxx</b>	DSP version xxxx

**U(msg)** An undefined or unparseable message response

**Maintenance Responses**

**XBddRCeeeeeeeeccccc** BERT receive error count on span dd,  
eeeeee - number of errors, cccccc - number of bits

**XLddc** Loopback code x detected on span dd (D4 CI only)

**XLddcc** Bit-pattern message cc detected on ESF span dd

**XSxxs(C)** Service status change for B-channel xx, s = service status, optional C if only for specified channel

**XSxxsD** Service status change for D-channel on span xx

**XSxxAs(C)** Service status change acknowledge for B-channel xx, s = service status, opt. C if for specified channel

**XSxxAsD** Service status change acknowledge for D-channel on span xx

**XXdd(data)** Performance Report Message on span dd

## **6.0 The H.100 Bus & Clock Modes**

The Infinity Series H.100 T1/Primary Rate ISDN Board provides a means of connecting T1/PRI B-channels to the digital H.100 computer telephony bus. Through this bus, the B-channels can be connected to other H.100 compatible boards. To accomplish this, the board has complete access to all streams and timeslots on the bus. It is capable of operating in a variety of clock modes compatible with H.100 operation. In addition, the board is capable of interoperating with legacy MVIP and SCbus boards.

### **6.1 The H.100 Bus**

The H.100 bus consists of 32 Pulse Code Modulation (PCM) streams operating at an 8.192 MHz. clock rate. Each stream contains 128 timeslots, for a total of 4096 timeslots. In addition to the PCM data signals, there are a number of bit, frame, and network reference signals that are used to synchronize the operation of multiple boards. For interoperation with the legacy SCbus, MVIP-90 bus and the H-MVIP bus there are some additional clock signals that are included on the bus.

For the purposes of commands, a particular H.100 timeslot is referred to by a four digit hexadecimal number. The first two digits are the stream number, while the last two digits are the timeslot within the stream. Streams range from 00h to 1Fh, and timeslots from 00-7Fh.

The physical H.100 bus is a 68 conductor ribbon cable that connects the various boards in the system. As in any such bus, termination is important for its proper operation. The board at each end of the H.100 cable must have the proper termination installed or enabled, while any board between the ends must not terminate the bus. For the H.100

T1/Primary Rate ISDN Board, termination is enabled using a command of the form **STab** where a controls the H.100 termination and b the MVIP bus termination. Termination is enabled if a is “E” and disabled if a is “D”.

### 6.1.1 Legacy Bus Compatibility

The H.100 specification provides for inter-operability with several common legacy PCM busses. These include the SCbus, the MVIP-90 bus and the H-MVIP bus. Because these busses run at different bit rates than the 8.192 MHz. of the H.100 bus, provisions exist in the specification to run the first 16 streams at either 4.096 MHz. or 2.048 MHz. For inter-operability with the SCbus, these streams typically should be run at 4.096 MHz. (2.048 MHz. and 8.192 MHz. are also possible choices) and with the MVIP-90 bus they should be run at 2.048 MHz. H-MVIP runs these streams at either 2.048 or 8.192 MHz. depending on whether MVIP-90 compatibility is desired.

On the H.100 T1/Primary Rate ISDN Board, the bit rate of the first 16 streams is set using the “SB” command. This command takes the form **SBabcd** where the parameters a, b, c, and d select the bit rate for streams 0-3, 4-7, 8-11, and 12-15 respectively. The choices for these parameters are:

- 0 - 2.048 MHz.
- 1 - 4.096 MHz
- 2 - 8.192 MHz.

Thus to operate with the SCbus at 4.096 MHz. the command would be **SB1111** and to operate with the MVIP-90 bus **SB0000**. The default selection for these streams is the H.100 rate of 8.192 MHz.

When operating in a compatibility mode, the timeslot in board commands range from 00 to the maximum number of timeslots allowed by the bit rate. At 2.048 MHz. timeslots within a stream are numbered

### *The H.100 T1/Primary Rate ISDN Board*

00-1Fh and at 4.096 the timeslots are 00-3Fh. MVIP bus streams are numbered 00-0Fh. This numbering corresponds to the DSo/DSi convention according to the following table:

<b>H.100 stream</b>	<b>MVIP-90 stream</b>	<b>MVIP-95 stream</b>	<b>H.100 stream</b>	<b>MVIP-90 stream</b>	<b>MVIP-95 stream</b>
00h	DSo0	HDS0	08h	DSo4	HDS8
01h	DSi0	HDS1	09h	DSi4	HDS9
02h	DSo1	HDS2	0Ah	DSo5	HDS10
03h	DSi1	HDS3	0Bh	DSi5	HDS11
04h	DSo2	HDS4	0Ch	DSo6	HDS12
05h	DSi2	HDS5	0Dh	DSi6	HDS13
06h	DSo3	HDS6	0Eh	DSo7	HDS14
07h	DSi3	HDS7	0Fh	DSi7	HDS15

## 6.2 Clock Modes

The H.100 bus specification defines a variety of clock signals. Two clock signals CT bus A and CT bus B are provided for redundancy. In addition, a signal called CT\_NETREF is defined which may be referenced to an external clock source such as a T1 or E1 span. This signal exists to aid in recovery if the primary clock source should fail. The specification also includes clock signals for compatibility with both the MVIP-90 and SCbus.

The clock mode must be set before any connections can be made with other boards. The clock mode is set using the Set Clock command "SCmsabbc", where m is the clock mode, s is the sub-mode, and a, bb, and c are additional arguments used to select clock sources and specify compatibility modes. The default clock mode on a power up is to

provide a local clock, but to neither source clock signals to the bus or derive the clock from the bus. The possible clock modes are:

- 0 no clocks to or from the bus
- 1 clocks slaved to the CT bus
- 2 the board is clock master CT bus clock A
- 3 the board is clock master CT bus clock B
- 4 the board is secondary master for CT bus clock A
- 5 the board is secondary master for CT bus clock B

Connections are possible only when all boards within a system are synchronized to the same clock. Only one board in a system can provide the H.100 bus clock. The other boards in the system must slave their internal clocks to the master. If the H.100 T1/Primary Rate ISDN Board is to use the H.100 bus clock, this clock must be provided by another board before switching can be accomplished.

### **6.2.1 Slave Mode**

In the Slave Mode, the H.100 T1/Primary Rate ISDN Board derives its clocks from one of the clock signals on the CT bus. The clock signal is selected with the submode argument in the **SC** command. The possible clock signals are:

- 0 - CT bus clock A
- 1 - CT bus clock B
- 2 - SCbus clock at 2 MHz.
- 3 - SCbus clock at 4 MHz.
- 4 - SCbus clock at 8 MHz.
- 5 - MVIP-90 clock signal
- 6 - CT bus clock A, auto-fallback mode
- 7 - CT bus clock B, auto-fallback mode

Argument *a* is used to set the CT\_NETREF mode, while argument *bb* is used to select the source of CT\_NETREF. The choices for argument *a*

### *The H.100 T1/Primary Rate ISDN Board*

are:

- 0 - No CT\_NETREF output
- 1 - CT\_NETREF output is enabled

The CT\_NETREF source is specified by argument *bb*, and can be any of the T1/Primary Rate ISDN interfaces that is operating as a TE (terminal endpoint).

### **6.2.2 Primary Master Mode**

In modes 2 or 3, the board supplies the CT master clocks A or B respectively. Other boards on the H.100 bus will synchronize to one of these clocks. The source of the clock is selected by the submode argument *s*. The choices are:

- 0 - freerun, the board's internal clock
- 1 - CT\_NETREF
- 2 - CT\_NETREF\_2 (not present on the H.100 bus)
- 3 - a local network, any CI T1/PRI interface
- 4 - a local network, any CI T1/PRI interface with auto-fallback to CT\_NETREF

For submode 1 and 2, argument *bb* will select the frequency of the CT\_NETREF signal. The choices is:

- 00 - 8 kHz. (frame rate)

Note that on earlier modes, other CT\_NETREF rates were supported, but that this is not possible with the current switching chip.

For submodes 3 and 4, argument *bb* will select one of the T1/PRI interfaces. This interface must be operating as a CI (customer interface). For submode 4, the optional argument *c* will specify the frequency of CT\_NETREF.

For all modes, argument a will select the legacy bus compatible clocks that the board will supply. The options are:

- 0 - no compatibility clocks
- 1 - SCbus clocks at 2 MHz.
- 2 - SCbus clocks at 4 MHz.
- 3 - SCbus clocks at 8 MHz.
- 4 - MVIP 90 clocks
- 5 - H-MVIP clocks

### **6.2.3 Secondary Master Modes**

When a board is operating as a secondary master, it uses the other clock signal as a source, i.e. if a board is the secondary master for CT clock B, it uses CT clock A as a source and provides CT clock B. If the primary clock fails, the secondary master then becomes the clock master. Typically, one board will be set as the master for clock A and another board as the secondary master for clock B, or vice versa. If the clock source specified by the submode is either of the CT\_NETREF signals or a local network, the board will automatically fall back on that source if the primary clock source should fail. If set to free-run, it will fall back to a PLL that was locked to the primary master clock.

In all secondary master modes, if the primary master fails, the board will automatically become the new primary master. If the original primary master is restored, the clock mode for the original secondary master must be reset. A board should only be set to be a secondary master after the primary master is established

When operating in secondary master mode, the arguments s, a, and bb are the same as when operating as a primary master.



### **6.2.4 Clock Fallback**

The H.100 Specification details a scheme for automatically recovering from a clock failure. One of the CT bus clocks, either A or B is designated the master clock. The other clock is the secondary master and is generated by a different board than the primary clock. While the primary clock is valid, the secondary clock is locked to it. If the primary clock should fail, the secondary clock takes over using a local oscillator, CT\_NETREF, or a local network as the source. Boards that are slaves should automatically fall back to the secondary clock. After a failure of the master clock, system software should designate new primary and secondary clocks. The new primary may be the previous secondary clock master. For Infinity Series H.100 boards, this will involve sending a set clock command with the new primary clock information.

When an Infinity Series board is set for automatic fallback, the board will automatically switch to the secondary clock if the primary clock fails. When this occurs, the board will send an “EC” message indicating the failure. When the application designates a new primary master, it should send a new clock mode command to the board even though auto-fallback may have occurred.

### **6.2.5 Clock Errors**

If the board detects a problem with the clocks, it will generate a clock error, which notifies the application that it should take appropriate action. Clock errors are reported in the Clock Error Bit message, **ECxx** where the **xx** is a hexadecimal value in which each bit identifies the specific error. A value of 1 indicates an error condition. The bits are as follows:

<u>bit</u>	<u>Error Description</u>
0	CT bus clock A
1	CT bus clock B
2	SCbus clocks

3	MVIP bus clocks
4	Master PLL error
5	Always set for compatibility
6	Failsafe bit

### 6.2.6 Clock Termination

For proper operation of the clocks, it is important that the clock lines be terminated on the board at each end of the H.100 bus. Failure to do so may cause noise in the audio signals. Clock termination is controlled by a command of the form **STab** where a controls the H.100 clock line termination and b controls the MVIP clock line termination. These parameters should be set to “E” to enable the termination and “D” to disable the termination. The default is for the terminations to be disabled. Note, that only the boards on the end of the H.100 cable should have the termination enabled.

## 6.3 Configuration Information

Information on the clock mode setting, stream rates, and other configuration settings is available in the dual-ported memory in an eight byte block beginning at an offset of 1F18h. The first four bytes are the clock mode, the submode, and the a and bb arguments from the set clock command **SC**. The next byte contains the stream rate information from the **SB** command with bits 0-1 containing the value for streams 0-3, bits 2-3 for streams 4-7, and so on. Bits 0 and 1 of the sixth byte indicate the state of the H.100 and MVIP termination, respectively, with a value of 1 being the enabled state. If bit 7 of the seventh byte is 1, bits 0-3 indicate the T1 interface that is providing the reference for clocks on the board. If bit 7 is 0, the clocks are not being derived from an interface. The eighth byte contains the clock error status bits. These are in the same order as in the **EC** clock error message (Sec. 6.2.5).

## 7.0 T1 Framing & Clocking

T1 spans provide a means of interconnecting telephony equipment at a bit rates of 1.544 Mbps. These are commonly used in North America (the European equivalent is an E1 span). The physical aspects of T1 interfaces (also known as a DS1 interface) are given in a set of ANSI specifications designated the T1.403 series. These include T1.403 which covers the electrical characteristics, T1.403-01 which covers Layer 1 of the T1 interface when used as a Primary Rate ISDN, and T1.403-02 which defines the use of robbed bit signaling. Additional standards cover other aspects such as the upper layers of PRI interfaces and are dealt with in subsequent sections.

This section will give a brief description of the first physical layer and how they relate to the inter-workings of the H.100 T1/Primary Rate ISDN board and the application. It is not meant as an exhaustive reference or definition. For this, the reader is referred to the relevant specifications.

### 7.1 DS1 Electrical Interface

The physical characteristics of the DS1 Interface are defined in *ANSI T1.403-1999 - Network and Customer Installation Interfaces - DS1 Electrical Interface*. When dealing with a Primary Rate ISDN interface this is referred to as Layer 1. The DS1 electrical interface deals with the functions of timing and synchronization. It also provides the following services, signaling, data transmission, error detection, and maintenance functions.

DS1 interfaces are digital circuits that run at either a 1.544 Mbps. bit rate. A major problem and the reason for many of the design choices in these circuits is insuring the proper interpretation of data after it has

*The H.100 T1/Primary Rate ISDN Board*

been transmitted an extended distance over a connection consisting of physical wiring. This wiring imposes a variety of limitations such as loss, capacitance, reflections from impedance mismatches, and so on. The complexities of the DS1 interface arises from the methods used to overcome these problems.

The major issue is one of clocking. The two ends of a DS1 interface do not share a common clock. Instead, all clocking and framing information is contained in the DS1 signal. The network end, which provides the clocking for the interface, is typically tied into a hierarchy of clocks going to a master source provided by the National Bureau of Standards. The customer end derives its clock from clues that are present in the signal sent over the DS1 interface from the network.

The second of a DS1 bit stream is divided into 8,000 segments called frames. Each frame consists of 192 bits of data and one framing bit. These 192 bits are divided into 24 8-bit channels. These frames allow the DS1 interface to carry multiple channels of voice sampled at the 8000 8 bit sample rate for a rate of 64 kbps for each channel. Of course, the channels can also be used to carry data instead of voice. In addition to the 192 bits of data per frame, an additional framing bit is added for reliability and maintenance purposes.

Historically, the DS1 format has evolved over time as attempts were made to improve performance and reliability. This has resulted in several formats being in use at the same time. The two major formats are D4 and ESF. The difference between the formats is largely how frames are grouped together into what are called superframes. Superframes are used to define the location in the bit streams of bits used for signaling or maintenance. The D4 format has a superframe that consists of 12 frames. The ESF format which stands for Extended SuperFrame has a superframe that consists of 24 frames.

In addition to framing format, there are several other consideration. One of these is insuring that there is no DC offset to the signal. To do this,

a scheme called Alternate Mark Inversion or AMI, is used. In this scheme, every other mark or one signal is indicated by a negative rather than a positive voltage. Another concern is zero suppression. If too many zeros are transmitted in a row, the signal can degrade. To prevent this, the bit encoding scheme is modified to prevent this from occurring. The scheme commonly used for this purpose is B8ZS or Bipolar with 8-Zero Substitution. This document will not go into the details of these schemes, but for an interface to work it is important that both ends use the same framing and zero suppression formats.

It is also necessary to match impedances and levels to minimize reflections and insure the proper interpretation of signals. This is called “line build out”. The characteristics of the interface should be tailored to the length and type of wire used to connect the two ends.

## 7.2 Configuring the Interfaces

Several steps are needed to configure the DS1 interfaces on the H.100 T1/PRI Board. The first of these is to specify whether the interface is to act as the Network or NI end or as the Customer or CI end. This is important because the CI end derives its clock from the interface while the NI end generates the clocks for the interface. Specifying which end the interface is to be is done with the “**ST**” command. This command consists of “**ST**” followed by a character for each of the interfaces on the board. This character can be a “**C**” for the customer end, an “**N**” for the network end, a “**U**” for an unused or undefined interface, or an “**\***” if no change is to be made. As an example, if the first two interfaces are to be set as NIs, the next four as CIs and the remaining are not being used, the command would be **STNNCCCCUU**. Note that for a board with only four interfaces, only four characters would need to be sent.

The next step is to specify the framing format for each interface. This is done with a command of the form **SFddfzbs** where dd is the interface or span number, f is a character defining the framing format, z is a

character defining the zero suppression scheme, b specifies the line build out, and s specifies the signaling format. The choices for framing format are “D” for D4 and “E” for ESF. The choices for zero suppression are “A” for AMI and “B” for B8ZS. The signaling format can be “N” for no signaling, “R” for robbed bit signaling, or “P” for Primary Rate ISDN. The buildout choices are specified with a number between 0 and 7 according to the following table:

0	DSX-1 (0 to 133 feet)/0dB CSU
1	DSX-1 (133 to 266 feet)
2	DSX-1 (266 to 399 feet)
3	DSX-1 (399 to 533 feet)
4	DSX-1 (533 to 655 feet)
5	-7.5dB CSU
6	-15dB CSU
7	-22.5dB CSU

Another configuration issue is setting the bus clock mode. If one or more interfaces are acting as CIs and are connected to the network, then one of the interfaces should serve as the source for the bus clock, and the board should act as the master. If there is more than one board with T1 interfaces connected to the network, then one board should be the master and the other boards should slave to the CT bus. If there is more than one CI interface, then it may be desirable to make choices as to a primary and secondary clock source to allow a graceful fallback in case of a problem with the primary interface. (See Sec. 6.0 for details)

## **7.3 Alarms**

Alarms are used to indicate that there is a problem with the interface. There are three different alarm conditions which are designated by color, Yellow, Blue, and Red, with Red being the most severe. Green indicates that no alarm condition has been detected. The meaning of the alarms is as follows:

*The H.100 T1/Primary Rate ISDN Board*

Yellow Alarm	Remote Alarm Indication (RAI), a specific pattern is detected that depends on the framing mode
Blue Alarm	Alarm Indication Signal (AIS), an unframed all-ones signal is detected
Red Alarm	Loss Of Signal (LOS)

The Yellow and Blue alarms can be generated by the interface. The Yellow alarm or RAI should be sent when the incoming signal is effectively lost, such as when the interface detects a Red alarm. The Blue alarm should be sent when the originating signal has been lost, or when an action is taken that will cause a service disruption, for example when a line loopback has been implemented.

The alarm state of an interface can be determined in several ways. Each interface has a status LED associated with it, (See Sec. 3.3). This LED can be green, yellow, blue, red, or dark. Green indicates that there is no alarm condition detected. The other colors indicate the most serious alarm condition detected, i.e. red is more important than blue is more serious than yellow. Dark is reserved for undefined spans. The status of these LEDs is also present in dual-ported memory at an offset of 1D00h. There are 8 bytes, one for each span. Bits 0-3 represent the green, yellow, blue and red alarm states respectively. Bit 7 is set to a 1 on NT interfaces and to a 0 for CI and undefined interfaces.

The alarm states may also be determined by sending the query message "QA". The response message will have the form **QAaaaaaaa** where each "a" represents the alarm state of a span starting with span 0. The alarm states are indicated by a 'G', 'Y', 'B' or 'R' for green, yellow, blue, or red alarm condition.

Finally, a message is sent every time an alarm changes state. This message takes the form **ARdd**, **ABdd**, or **AYdd** for red, blue, yellow alarms respectively, where dd is the span number. If all alarms are cleared, an **AGdd** message is sent.

Yellow and Blue alarms can be sent under application control. This is done with a message of the form **AYdds** or **ABdds** for the yellow and blue alarm where dd is the span number and s is the alarm state, “S” to set the alarm, and “C” to clear the alarm.

## 7.4 Line Interface Status Events

When certain physical conditions are detected by the Line Interface Unit, an event message will be generated. These events reflect open or short circuit conditions in the wiring and are not dependent on the framing or clocking of the interface. When such an event is detected, a message of the form **ESddss** is sent where dd is the span number and ss is a value indicating which condition or conditions exist. The bit values in ss are:

<u>bit</u>	<u>description</u>
1	an open circuit has been detected between the transmit tip and ring outputs
2	the transmit current limiter has been activated
3	the receive carrier signal is lost
4	the jitter attenuator limit has been reached

The other bits are reserved for future use. Note, that this message is sent whenever any of the event bits change.

## 7.5 Elastic Store Buffer Events

The synchronization of clocks on the two ends of the span may not be perfect. This can be due to the design of the equipment, noise, or jitter in the clocks. To compensate for this, each span on the H.100 T1/PRI Board is equipped with an elastic store buffer in each direction. Under some circumstances, the elastic store buffer may be unable to handle the clock differences. When this occurs, an elastic store event will be



generated. This causes a message of the form **EXddss**, where dd is the span number, and ss is a value indicating which event or events has occurred. Each bit in ss indicates a different event:

<u>bit</u>	<u>description</u>
0	RSLIP - receive elastic store slip event
1	RESEM - receive elastic store buffer empties, frame repeated
2	RESF - receive elastic store buffer full, frame deleted
3	TSLIP - transmit elastic store slip event
4	TESEM - transmit elastic store buffer empties, frame repeated
5	TESF - transmit elastic store buffer full, frame deleted

## 7.6 Maintenance Functions

Several maintenance functions are available to test whether a span is operating correctly. These functions conform to those which are described in *ANSI T1.403-1999*, and include provisions for performing loopbacks and collecting performance data. In general, these functions are initiated by the NT (central office) end of the span, or involve data that is sent from the CI (customer) end to the NT.

### 7.6.1 D4 Loopback Codes

Loopback codes are repeating patterns of bits that are sent from the NT to the CI to activate or deactivate loopbacks. D4 framing uses a 3 or 5 bit pattern that persists for at least 5 seconds to send the loopback request codes. The detection of 5 seconds of the pattern is considered to be the request for loopback.

Loopback codes may be sent from an NT interface using a command of the form **XCddc** where dd is the span number and c is the code mode. Valid values for c are, “0” for no code, “1” to activate a loopback, and

“2” to deactivate the loopback. Timing of the loopback codes is the responsibility of the application. For D4, the code must be sent for at least 5 seconds.

When a CI interface receives a loopback code for 5 seconds it will respond with a message of the form **XLddc** where dd is the span number and c is the code. Valid codes are “1” to activate the line loopback and “0” to deactivate the line loopback. A “0” indicates the code has stopped. The application is responsible for activating and deactivating the loopback with the appropriate “XL” command.

### **7.6.2 ESF Bit-Pattern Messages**

The ESF format uses a 16 bit pattern on the ESF facilities data link that repeats at least 10 times to transmit messages using the Facilities Data Link. These messages consist of 6 bits plus 10 framing bits, a 0 before and after the 6 bit code and eight consecutive 1s. The meanings of the 6 bit codes are defined in *ANSI T1.403-1999*, Table 4. Bit-patterned messages may be sent from either the NT or the CI, but certain messages are restricted to one or the other end. Three message values, 00h, 15h, and 1Fh are considered priority messages. These messages shall be transmitted for as long as a condition exists, but not for less than 1 second.

Most of the messages are concerned with activating or deactivating different loopbacks, however, some messages are reserved for other purposes which are beyond the scope of this document. The messages that are concerned with loopbacks and alarms are given in the table below:

#### Priority Messages

00h	RAI
15h	Loopback retention and acknowledge
1Fh	RAI-CI

**Command and Response Messages**

07h	Line loopback activate
1Ch	Line loopback deactivate
0Ah	Payload loopback activate
19h	Payload loopback deactivate
09h	Reserved for network use (loopback activate)
12h	Universal loopback (deactivate)
17h	ISDN line loopback (NT2)
10h	CI/CSU line loopback (NT1)
0Eh	For network use (indication of NT1 power off)

Bit patterned messages are sent with a command of the form **XCddcc** where dd is the span and cc is the message code. The bit pattern can be cleared with a message of the form **XCddFF**. It is the applications responsibility to time the message. A received bit pattern message is indicated by a message of the form **XLddcc** where dd is the span and cc is the message. An RAI message causes a Yellow alarm and no “XL” message will be sent. With one exception, messages will be reported when the pattern has cleared. That exception is the loopback retention and acknowledge message 15h. The message will be reported when it is detected. A message code of 55h will be reported when this message clears. The application is responsible for responding to bit patterned loopback messages by activating or deactivating loopbacks with the “XL” command.

**7.6.3 Loopback Control**

Two loopback types are defined in *ANSI T1.403-1999*, a line loopback and a payload loopback. A line loopback involves transmitting received framing and channel data, while a payload loopback does not loopback the framing and CRC6 bits are not looped back. Payload loopbacks are only supported with ESF framing. A third loopback form, the framer loopback is also supported by the H.100 T1/PRI board.

A loopback may be activated on a span by sending a command of the form **XLddm** where dd is the span number and m is the loopback mode. The following table gives the valid values for m:

0	clear loopbacks
1	line loopback (CI) or local loopback (NT)
2	framer loopback
3	payload loopback
4	per channel payload loopback

With a line loopback, data received from the line is looped back to the span. Data from the span is also transmitted to the internal telephony bus. With a local loopback, data from the internal telephony bus will be looped back to the telephony bus as well as being transmitted to the span. When a framing loopback is activated, data from the internal telephony bus is looped back to the telephony bus and a Blue alarm (AIS) is sent to the span. Payload loopbacks involve looping back data (excluding framing bits) from the span back to the span. The data is also transmitted to the internal telephony bus. With a per channel payload loopback, only data from selected channels is looped back to the span.

The payload loopback loops all 24 channels. Specific channels may be looped back using a command of the form **XLdd4aabbcc**. The hexadecimal arguments aa, bb, and cc define a bit map indicating which channels are to be looped back. Channels 0-7 are defined by aa, 8-15 by bb, and 16-23 by cc. The least significant bit in these arguments represents the lowest numbered channel. A set bit indicates the channel is to be looped back. To clear this loopback, a message of the form **XCdd4000000** must be sent. The command **XCdd0** will not clear this loopback.

#### **7.6.4 Performance Report Messaging**

*ANSI T1.403-1999* defines a Facilities Data Link or FDL for ESF framing. This data link may be used to send unacknowledged frames of

data using the LAPD format and the framing bits. The primary use of the data link is to send performance data from the CI to the NT to indicate whether the interface is operating correctly.

The Performance Report Message should be sent by an CI with ESF framing once a second and contains performance information about each of the last four seconds. The information sent includes the number of CRC errors, and whether a slip, line code violation, framing bit error, or framing event has occurred in that second. These messages are enabled by sending a command of the form **XF02**.

For interfaces set as an NT, the PRM messages are not normally passed up to the application to avoid excessive messaging. However, they can be turned on by sending a message of the form **XF01**. They can be turned off by sending a message of the form **XF00**. The messages to the application take the form **XXddd1d1d2d2d3d3d4d4** where dd is the span number, and d1d1 is the data for the first second, d2d2 is the data for the second second, and so on. The data for each second is sent as a 4 digit hexadecimal number where the bits are:

G3 LV G4 U1 U2 G5 SL G6	m.s.b of data
FE SE LB G1 R G2 Nm NI	l.s.b. of data

where:

G1 = 1	CRC error event count = 1
G2 = 1	CRC error event count >1 & < 6
G3 = 1	CRC error event count > 5 & < 11
G4 = 1	CRC error event count > 10 & < 101
G5 = 1	CRC error event count > 100 & < 320
G6 = 1	CRC error event count > 320
SE = 1	Severely errored framing event > 0 (FE = 0)
FE = 1	Framing synchronization bit error > 0 (SE = 0)
LV = 1	Line code violation > 0
SL = 1	Slip event > 0

LB = 1	Payload loopback activated
U1,U2,R	Sent as 0s, may be modified
Nm,NI	modulo 4 sequence number

A typical message when a line code violation occurred in the second before the most recent second reported would appear as:

**XX000003400200010000**

### 7.6.5 BERT Testing

Bit Error Rate Testing or BERT is used to test and stress communications links to determine if there are problems with the transmissions facilities. It consists of sending a defined pattern and then detecting and counting any errors over an interval. A fixed rate of errors may be introduced at the transmit side to insure that the error detection at the receive end is functioning. The BERT pattern may be sent from one end to the other, or the far end may be put into a loopback so that the pattern is returned to the transmitting end.

Four commands are used to control the BERT transmit function. The first of these is a command of the form:

**XBddTEei**

which is used to set the pattern mode where dd is the span, e is the patten mode, and i selection the inversion mode for transmitted and received data. The pattern modes range from 0-7 and are:

- 0 Pseudorandom 2E7-1
- 1 Pseudorandom 2E11-1
- 2 Pseudorandom 2E15-1
- 3 Pseudorandom Pattern  $QR_{ss}$  - a  $2^{20} - 1$  pattern with 14 consecutive zero restriction
- 4 Repetitive pattern

*The H.100 T1/Primary Rate ISDN Board*

- 5 Alternating word pattern
- 6 Modified 55 octet (Daly) pattern as defined in ATIS T1-TRN25
- 7 Pseudorandom 2E9-1

Bits 0 and 1 of the inversion mode *i* are used to control the inversion of the transmit and received data streams respectively, with a value of 0 being normal and 1 being inverted.

If mode 4 (repetitive pattern) is selected, the pattern is set by a command of the form:

**XBddPppppppppllcc**

where *dd* is the span number, the *pp..pp* argument defines a repetitive pattern that may range from 17 to 32 bits, *ll* is the length of the pattern, and *cc* is the alternating word count which is used in the alternating word mode to select the number of times a pattern defined by the first 16 bits is repeated before changing to an alternate pattern defined by the second 16 bits.

The command to select which timeslots will be used for transmitting the BERT data is defined by a command of the form:

**XBddTTttttt**

where *dd* is the span number and the *ttttt* argument is a bit map of the timeslots with the most significant byte defining timeslots 0-7 and so on. To enable the BERT transmit function, the pattern would first be defined, then the pattern mode, and finally the timeslots would be selected. To disable the BERT transmit function, a command of the form:

**XBddTD**

should be issued where dd is the span number.

To enable the BERT receive function, the mode is set using a command of the form:

**XBddREei**

which is used to set the pattern mode where dd is the span, e is the patten mode, and i selection the inversion mode for transmitted and received data. The modes are the same as for the “XBddTE” command defined above. The timeslots receiving data are then selected using a command of the form:

**XBddRTttttt**

where dd is the span number and ttttt is a bit map of the timeslots as defined above. The BERT receive function is disabled using a command of the form:

**XBddRD**

where dd is the span number.

The BERT error counters can be read using a command of the form:

**XBddRC**

where dd is the span number. The count will be returned in a message of the form:

**XBddRCeeeeeeccccccc**

where dd is the span number, eeeee is a 24 bit error count in hexadecimal, and ccccccc is a 32 bit received bit count in hexadecimal. The received bit count can be used to determine the ratio of errors to bits



received.

Two commands exist to introduce errors into the bit stream. To introduce a single bit error, a command of the form:

**XBddEB**

is used, where dd is the span number. To introduce errors at a fixed rate, a command of the form:

**XBddEIr**

is used where dd is the span number and r is the rate as follows:

0	no errors are inserted
1	10E-1
2	10E-2
3	10E-3
4	10E-4
5	10E-5
6	10E-6
7	10E-7

It should be noted that error rates greater than 10E-3 may cause problems with synchronization.

It may be necessary under some circumstances to resynchronize the BERT receiver. This is done with a command of the form **XBddS** where dd is the span number of the receiver.

It is beyond the scope of this document to discuss the details of BERT testing.

## 7.7 Robbed Bit Signaling

Robbed bit signaling is used on non-ISDN T1 spans for call control. It allows each channel on the span to emulate a variety of analog line types. As its name implies, it robs bits from each channel for call control information, the least significant bit from every eighth frame. Use of robbed bit signaling does result in a loss of bandwidth, that is only 56 kbps are available instead of 64 kbps. However, as it is the least significant bit, it does not cause a noticeable degradation for voice circuits. For a D4 frame, two bits are available, called A and B. For ESF, two additional bits are available, C and D. However, in most implementations, the C and D bits repeat the A and B bits.

These bits can be used to indicate hook status, ringing, and ground signaling. By using them, a channel can emulate the operation of an E&M, ground start, or loop start circuit. For ground start and loop start circuits, the two ends are not symmetric, that is one end is the central office and the other is the station. Thus which end of the circuit a channel is emulating must be specified. When using robbed-bit signaling, the line type must therefore be set for each channel. This is done using a command of the form **SPxxtd(pd)** where xx is the channel number, t is the line type, d is the direction, and p and d are additional parameters for address signaling protocols that may be supported (see Sec. 10.4.5). Line types that are supported are “E” for E&M, “G” for ground start, “L” for loop start and “N” for none. When none is selected, robbed bit signaling is not implemented. The direction specifies the end and can be either “O” for the central office end, or “S” for the station end. The p parameter allows for the support of immediate start “I”, or wink start “W” protocols, and the d parameter is the number of digits expected.

## **7.8 Primary Rate ISDN**

Primary Rate ISDN involves the use of a separate control channel for call control rather than using robbed bits. This increases the bandwidth to the full 64 kbps for each channel. It also allows complex messages to be transmitted, thus supporting more features and services. This is done by sacrificing one of the channels on the span for a call control data or “D” channel.

With Primary Rate ISDN, 23 channels are available for what are referred to as bearer or B channels. These channels can carry a variety of formats including speech, 3.1 kHz. audio, and unrestricted data. The 24th channel is used for the “D” channel. Call control messages are sent down this channel using the LAPD protocol. This is described in Sections 8.0 and 9.0.

Primary Rate ISDN always uses the ESF framing format and B8ZS zero suppression.

## **7.9 Non-Facilities Associated Signaling (NFAS)**

When connecting multiple Primary Rate ISDN spans to the public network, it may be advantageous to use a single D-channel to control multiple spans. This is called Non-Facilities Associated Signaling or NFAS, as call control is handled by a D-channel not associated with the span carrying the call.

To enable NFAS on a set of spans one span is selected as the primary span. This span will have a D-channel and the framing is set in the same manner as a normal PRI span. The framing for the other spans must be set for no signaling, i.e. “SFddEBbN” where dd is the span, and b is the buildout code. This will free up the twenty-fourth timeslot on the span for call handling. If the NFAS arrangement is to include a backup D-channel, the span with the backup must also have its’ framing set for

PRI signaling. However, the D-channel will not be used until backup procedures are initiated.

If Layer 3 call control messaging is to be handled by using the “LC” message procedure, nothing more needs to be done to set up the spans. However, the Channel Identity information element will need to include the appropriate interface identifier information.

If call control is being handled through “D” messages, i.e. the Protocol Level support is set other than “2”, then there is an additional step to setting up the spans. This is done through a command of the form **SNic..ic** where each pair of digits ic is used to control one span. There will need to be one pair for each of the spans on the board. The ‘i’ digit is used to indicate which interface identifier is associated with the span. The ‘c’ digit is used to indicate the span carrying the D-channel for the set of NFAS spans. Bit 3 of this digit must be set to indicate that the span is part of an NFAS arrangement. If a span is not part of an NFAS arrangement, it should be coded as ‘00’. As an example the following command sets up three spans as an NFAS arrangement with the first span carrying the D-channel. The other spans are not part of NFAS arrangements.

**SN0818280000000000**

If D-channel backup procedures are invoked, this command will need to be issued reflecting the new D-channel.

## 8.0 ISDN Layer 2 Protocol

Primary Rate ISDN conforms to a model defined in a set of standards (I.431, Q.921, Q.931 etc.) that is composed of a number of layers dealing with the physical interface, peer to peer signaling and call control. The Infinity Series H.100 T1/Primary Rate ISDN Board provides most of the control over the first two layers and optionally provides an interface to deal with the details of the third or call control layer. This section will give a brief description of the first two layers and how they relate to the inter-workings of the board and the application. It is not meant as an exhaustive reference or definition. For this, the reader is referred to the relevant specifications or Bellcore documents.

### 8.1 Layer 1

The physical characteristics of the Primary Rate Interface are defined by I.431 and are referred to as Layer 1. Layer 1 deals with the functions of timing and synchronization. It also provides the following services, activation/deactivation, D-channel access for signaling, transmission of the B and D channels, and Layer 1 status indication to Layer 2.

In North America, the framing used for Primary Rate ISDN is always ESF with B8ZS zero suppression. Details of framing may be found in Section 7. Twenty-four timeslots are supported. The first 23 of these are designated as B-channels and are used for passing voice and data information. The twenty-fourth timeslot is the D-channel and is used for call control. Schemes exist for grouping multiple Primary Rate spans with a single D-channel on one of the spans used for control, but this is not currently supported on the H.100 T1/Primary Rate ISDN Board.

The transmission of voice, data, and call control information depends on

*The H.100 T1/Primary Rate ISDN Board*

the proper synchronization of the physical interface. For details of this layer see Sec. 7.0.

## **8.2 Layer 2**

Layer 2 deals with the exchange of messages between peer devices and is referred to as the Data Link Layer. It is defined by the Q.921 specification. The protocol for the establishment of data links and the exchange of messages is called LAPD. On a Primary Rate Interface, LAPD messages or frames are exchanged over the 64 kHz. D-channel in the last timeslot of the span.

Primary Rate ISDN normally employs only one data link on the D-channel at the same time. The data link is identified by a two part address called the Data Link Connection Identifier or DLCI. The two parts are the Service Access Point Identifier or SAPI, and the Terminal Endpoint Identifier or TEI. SAPI values can have a range of 0-63 and TEI values can range from 0-127. However, for the purposes of the Primary Rate ISDN interfaces only a single address is normally used, SAPI = 0, TEI = 0. Because only a single fixed DLCI is used, no TEI management procedures are employed.

LAPD messages or frames can take three formats. Information or I frames are used to perform information transfers between Layer 3 entities. Supervisory or S frames are used to perform data link supervisory control functions such as the acknowledgment of I frames, requesting the retransmission of I frames, or request the suspension or resumption of the transmission of I frames. The third format is Unnumbered or U frames which provide additional data link control functions and unnumbered information transfers for unacknowledged information transfers.

## Supervisory Frames

- RR** Receive Ready, used to acknowledge I frames  
**RNR** Receiver Not Ready, used to suspend transmission  
**REJ** Reject, reject an I frame due to an error

## Unnumbered Frames

- SABME** Set Asynchronous Balanced Mode Extended Command  
**DISC** Disconnect Command  
**UA** Unnumbered Acknowledgment Response  
**DM** Disconnect Mode Response  
**UI** Unnumbered Information Command  
**FRMR** Frame Reject Response

To exchange Information frames, a data link must first be established. This is done by a terminal requesting a TEI using a UI frame. The network can grant the TEI also using a UI frame. There are a number of possible states a data link can have once a TEI has been assigned. To exchange I frames, the link must be in the multi-frame established state. This is achieved by one end or the other sending an SABME frame and receiving a UA response. Once the data link is in the multi-frame state it can remain there indefinitely unless an error is detected or a request is made to disconnect from the multi-frame state.

Each I frame contains two state variables  $N(S)$  and  $N(R)$  which are sequence numbers for the I frame sent and the next I frame that is expected to be received. The supervisory frame that acknowledges the I frame has corresponding variables. These are used to insure that no I frames are lost. They can also be used to request the retransmission of an I frame should an error in transmission occur. Each frame also includes Frame Check Sequence or FCS field to insure the integrity of the I frame contents.

The H.100 T1/Primary Rate ISDN Board manages all the details of the Layer 2 protocol. Acknowledgments and supervisory frames are sent as

*The H.100 T1/Primary Rate ISDN Board*

needed and recovery procedures initiated without requiring the intervention of the host application. The necessary timers are maintained on the board to handle retransmissions as specified by Q.921. Under normal circumstances, an application need only deal with the information field using “LC” commands and responses (see section 8.4).

It is not necessary for the application to deal directly with the Layer 2 data link states. However, the state information for data link is available in the dual-ported memory. For details see section 8.3. If a Layer 2 error condition is detected that can not be handled with the normal recovery procedures, an “EM” error message will be sent by the board to the application allowing the application to take further action such as initiating a TEI management procedure or restarting the port. See section 11.2 for details of these messages.

### **8.3 Layer 2 States**

Each of the spans has eight bytes reserved for Layer 2 state information beginning at an offset of 1C00h. The first byte for a span represent the Layer 2 states for the data link, the other bytes are reserved. These states are:

- 1 TEI unassigned
- 4 TEI assigned, a TEI has been assigned, but multi-frame operation has not been established
- 5 Awaiting multi-frame operation, an SABME frame has been sent and awaiting a UA frame acknowledgment
- 6 Awaiting release from multi-frame state, a DISC frame has been sent and awaiting a DM frame
- 7 Multi-frame operation, exchange of I frames is possible
- 8 Timer recovery, a timer has expired and recovery procedures are in progress



## 8.4 TEI Management

As Primary Rate ISDN uses a single, fixed data link, there are no management procedures to request or remove TEI assignments. The PRI interface is also a point-to-point interface with only a single terminal device connected to the network. Therefore there is no need for procedures to verify TEI assignments.

Initially, the data link is in the TEI assigned state. An attempt to send a message will place the data link in the multi-frame established state. However, the link can be disconnected or reestablished by using commands of the form **TDxxtt** and **TExxtt** where xx is the span number and tt is the TEI (00). A data link that is disconnected is placed in the TEI assigned state. If an attempt is made to send a Layer 3 message on a data link in the TEI assigned state, an attempt to reestablish the multi-frame established state will automatically be made by the board.

## 8.5 Sending and Receiving Layer 3 Messages

When the protocol level for a port is set at Layer 2 using the “SL” command, Layer 3 messages are sent and received using the auxiliary mailboxes. To send a message, the application must first check to make sure that the transmit flag is 0. The contents of the information field are placed in the auxiliary transmit mailbox at an offset of 1402h bytes. The length of the information field is placed at an offset of 1400h bytes. This is a sixteen bit value with the low order byte first. An “LC” message of the form **LCxxsstt** is then placed in the main transmit mailbox where xx is the port number, ss is the SAPI, and tt is the TEI of the data link to be used. The transmit flag is then set to 01h.

When a Layer 3 message is received, an “LC” message of the form **LCxxsstt** will be placed in the receive mailbox. This indicates that a Layer 3 message for the DLCI (ss,tt) on port xx is available in the auxiliary receive mailbox. The length of the message will be at an offset

of 1600h and the contents of the information field will begin at 1602h.

There is no restriction on the contents of the Layer 3 message other than a maximum length of 260 octets. Messages may only be sent for a DLCI with a SAPI of 0 and a TEI of 0 on a Primary Rate span.

As an example to send a Q.931 CONNect message for a call reference of 1, the following bytes would be placed in the auxiliary mailbox:

<u>offset</u>	<u>bytes</u>	<u>description</u>
1400h	04h	length
1401h	00h	high order byte of length
1402h	08h	Q.931 message
1403h	01h	call reference length of 1
1404h	81h	call reference of 1
1405h	07h	CONNect message type

To send this message with a DLCI of (0,0) on span 3, the command “LC030000” would then be placed in the main transmit mailbox and the transmit flag would then be set to 01h. Receiving a message is the mirror image of this process.

## 8.6 Transmit Queue Size

Each span has a separate queue for information frames containing Layer 3 messages. Under heavy traffic conditions, these queues, which are 29 messages deep, may become full. When this happens, additional messages for that span are discarded until room becomes available on the queue. In order to prevent this, the number of messages on the queue is available in dual-ported memory beginning at an offset of 1C80h, with consecutive bytes reserved for each span. It is recommended that if heavy traffic conditions are anticipated, that the application check this count and not attempt to generate additional Layer 3 messages for that span if the count exceeds a threshold value. The

appropriate threshold value will depend on the application to allow for latencies in the system, but 25 is probably a reasonable number. Under normal traffic conditions, this count can be ignored.

this page intentionally left blank

## 9.0 Using “D” Messages for Layer 3

### 9.1 Q.931 Messages

Call control for ISDN interfaces is handled at the Network Layer, also known as Layer 3. The specification that defines the message set used for call control is Q.931. Specifics for the U.S. implementation of the ISDN interface can be found in the ANSI document *Integrated Services Digital Network (ISDN) - Layer 3 Signaling Specification for Switched Circuit Bearer Service for Digital Subscriber Signaling System Number 1 (DSS1) (ANSI T1.607-1998)* and related ANSI documents for supplemental services.

Most switches currently deployed in North America use a common subset of Q.931 messages and procedures to simplify the interworking of equipment. This subset is called National ISDN, and has been implemented in a set of phases referred to as NI-1, NI-2, and NI-3. The details of these phases are described in various Telcordia (formerly Bellcore) documents. Some switches support additional features called Supplementary Services which are detailed in Q.932 and other specifications. Some of these are part of National ISDN 2, while others are part of standards such as QSIG or are proprietary.

The message set defined in Q.931 consists of a number of command messages to establish and break down a call. These include **SETUP** to initiate a call, **ALERTing**, **CALL PROCeeding**, and **PROGRess**, to handle the various stages involved in accepting or rejecting a call, **CONNect** to complete the connection, and **DISConnect**, **RELEASE**, and **RELEASE COMplete** to break down a call. Additional messages are provided as acknowledgments, **SETUP ACKnowledge** and

**CONNect ACKnowledge.**

Features beyond the basics of making a call are supported by additional messages. A call may be placed on hold and retrieved using the messages **HOLD** and **RETrieve**. These messages have corresponding acknowledgment messages, **HOLD ACKnowledge** and **RETRIEVE ACKnowledge**, as well as rejection messages, **HOLD REJect** and **RETreive REJect**. **INFORmation** messages can be used for a variety of purposes such as sending text for display, activating features or indicators, or exchanging configuration information. Additional messages are defined for maintenance or management purposes including **NOTIFY**, **STATus**, **STATus ENQuiry**, **FACILITY** and **REGister**.

In addition to the basic codes for the various message types, each Q.931 message consists of a number of information elements. These information elements are used to contain items such as the called and calling numbers, whether a call is speech or data, the B channel involved, and other information about the progress of a call. For some messages such as the **SETUP** message a large number of information elements are possible, many of which are optional or only used for certain types of calls. As an example a typical **SETUP** message from a central office switch contains eight information elements and ninety octets of information.

Each active call has a call reference that is part of every Q.931 message dealing with that call. This allows for the juggling of multiple calls on a B-channel. The call reference is granted by the originating party and remains in effect until the call is finally released, no matter what happens in the interim. However, a call reference is only valid at that particular interface in messages between the network termination and the terminal equipment.

Q.931 also describes a state model for the progress of a call. The states are defined by which type of message has been received or sent. State

information may be exchanged through STATus messages as a check on the operation of the terminal and to insure that the terminal and network are in agreement on the condition of a particular call. For each state of a call, only certain message types are appropriate.

## **9.2 “D” Command & Response Messages**

The Q.931 message set can be complicated, with many options and information elements to deal with. On the H.100 T1/Primary Rate ISDN Board, an application can directly access the information field of a Q.931 message by using the Layer 2 support level. In some cases this may be desirable when dealing with equipment that does not support National ISDN 1 or has non-standard features or messages. However, the H.100 T1/PRI Board does provide a simpler interface through the Layer 3 support level for those instances when only simple call control is needed for equipment that conforms to NI-1, NI-2, and NI-3..

The Layer 3 support level is achieved using “D” channel commands and responses. These messages consist of an initial “D”, a second character indicating the Q.931 message type, for example an “A” for an ALERTing message or a “C” for a CONNect message, a two digit hexadecimal number indicating the B channel, and for some messages additional arguments. The 23 B channels for span 0 are numbered 00-16, the B channels for span 1 are numbered 20-36, and so on. Each of the additional arguments are typically encoded as a single character or as a pair of hexadecimal digits in the case of the Cause element or Call references. The intent is to keep the messages short and simple with only the necessary information and allow the on board firmware to fill in the details of the actual Q.931 messages sent. The “D” messages are similar in their organization to other XDS messages such as the “C” and “S” messages used to control the B-channel connections on the H.100 T1/PRI Board and ports on other XDS line boards. This simplifies the overall task of the application.

“D” channel messages and the corresponding Q.931 messages are given

*The H.100 T1/Primary Rate ISDN Board*

in the following list:

<b>DAxx</b>	ALERTing
<b>DCxx</b>	CONNect
<b>DCxxA</b>	CONNect ACKnowledge
<b>DDxx</b>	DISConnect
<b>DFxx</b>	FACILITY
<b>DGxx</b>	RETrieve
<b>DGxxA</b>	RETrieve ACKnowledge
<b>DGxxR</b>	RETrieve REJect
<b>DHxx</b>	HOLD
<b>DHxxA</b>	HOLD ACKnowledge
<b>DHxxR</b>	HOLD REJect
<b>DIxx</b>	INFOrmation
<b>DKxx</b>	Keypad element INFOrmation message
<b>DNxx</b>	NOTIFY
<b>DPxx</b>	CALL PROCEEDing
<b>DPxxP</b>	PROGress
<b>DRxx</b>	RELease COMplete
<b>DRxxR</b>	RELease
<b>DSxx</b>	SETUP
<b>DXxx</b>	STATus
<b>DYxx</b>	REGister

Because of the fact that the required information elements in specific messages are not identical for the network and customer or user side, the argument details are different of NT and CI ports. A complete list of messages and arguments can be found in section 5.4.

To further simplify the application interface, the board will automatically send an acknowledgment message. A CONNect ACK is sent where needed in response to a CONNect message. A RELease is sent in response to a DISConnect, and a RELease COMplete is sent in response to a RELease message. A SETUP ACK or a CALL PROCEEDing message is sent in response to a SETUP message from a



terminal.

The on board firmware normally hides call references from the application. Calls are instead associated with B channels. If incoming calls do not have a B channel assignment, the board will make that assignment. However, when the association of a call and a B channel is not maintained, as in the case when a call is placed on hold, the call reference is passed on to the application so that it can be used to track the calls progress or be used to retrieve the call. Where used, the call reference consists of a four digit hexadecimal number. The most significant bit of this number is cleared when used by the originating side of the call and set when used by the terminating side of the call.

The “D” message set provides the application with a simple interface that gives the application complete control of the call but frees it and the programmer from being forced to deal with the details of coding Q.931 messages.

## **9.3 Information Elements**

Information elements are used to encode information that may be required to handle a call. In “D” messages on the H.100 T1/PRI Board, these elements are usually represented by single argument characters. Only those elements that are needed are included in the messages. Elements that are unambiguous or constant are encoded by the board in the Q.931 messages but are not part of the “D” messages. Some elements that are rarely used are not part of the “D” message scheme. Details on which arguments are used in which specific message can be found in Section 5.4.

### **9.3.1 Bearer Capability**

Bearer capability is used to describe the type of call being placed. This argument is present in both SETUP commands and responses. The five options for bearer capability are:

*The H.100 T1/Primary Rate ISDN Board*

- A** 3.1 kHz audio, 64 kbps, circuit mode, Mu-Law
- D** Unrestricted digital information, 64 kbps, circuit mode
- E** ECMA (QSIG) unrestricted digital data
- R** Rate adaption from 56 kbps, 64 kbps, circuit mode
- S** Speech, 64 kbps, circuit mode, Mu-Law

### **9.3.2 Call Reference**

The call reference element is used to keep track of a specific call when handling multiple calls on an interface. The call reference for Primary Rate ISDN is a number between 0001-7FFFh. When used by the terminating side of a call, the most significant bit of the reference is set, i.e. the range is from 8001-FFFFh. The call reference is assigned by the originating side of the call. The H.100 T1/PRI Board will automatically assign call references for calls that originate from the board. The call references are assigned sequentially. Note that the call reference is not present in all messages, but is only employed when needed to keep calls straight.

### **9.3.3 Cause**

The Cause element is present in a number of different messages. Its purpose is to give a reason why a particular message such as a DISConnect was issued. Some causes such as Normal Clearing are part of a normal call while other causes may indicate an error condition. The cause element is quite lengthy, and therefore the argument is represented by a two digit hexadecimal number in “D” messages. The first digit of the cause serves to divide the causes into different classes, i.e. a ‘0’ indicates a cause related to dialing. The list of causes are:

- 01** Unallocated Number
- 02** No route to specified transit network
- 03** No route to destination
- 06** Channel unacceptable

*The H.100 T1/Primary Rate ISDN Board*

- 07** Call awarded and being delivered in an established channel
- 10** Normal clearing
- 11** User busy
- 12** No user responding
- 13** User alerting, no answer
- 15** Call rejected
- 16** Number changed
- 1A** Non-selected user clearing
- 1B** Destination out of order
- 1C** Invalid number format (incomplete address)
- 1D** Facility rejected
- 1E** Response to STATus ENQuiry
- 1F** Normal, unspecified
- 22** Circuit/channel congestion
- 29** Temporary failure
- 2A** Switching equipment congestion
- 2B** Access information discarded
- 2C** Requested channel not available
- 2F** Resource unavailable, unspecified
- 32** Requested facility not subscribed
- 39** Bearer Capability not authorized
- 3A** Bearer Capability not presently available
- 3F** Service or Option not available, unspecified
- 41** Bearer capability not implemented
- 45** Requested facility not implemented
- 4F** Service or option not implemented, unspecified
- 51** Invalid call reference value
- 58** Incompatible destination
- 60** Mandatory information element is missing
- 61** Message type non-existent or not implemented
- 63** Information element non-existent or not implemented
- 64** Invalid information element contents
- 65** Message not compatible with call state
- 66** Recovery of timer expiry
- 6F** Protocol error, unspecified

7F Interworking, unspecified

### **9.3.4 Address Numbers & Subaddresses**

The called or calling party numbers in “D” messages are represented by a string of digits. Additional information that is encoded into the information element such as the type of number or numbering plan are added by the firmware based on the context and length. If the calling party number is omitted for SETUP messages from a CI port or the called party number is omitted for SETUP messages from an NT port the default directory number may be used if it has been programmed.

The default number type and numbering plan is always “Unknown.” Under some circumstances, it may be necessary to change this. This can be done by appending a character to the beginning of the digit string. Appending an “I” to the string will set the number type to International and the numbering plan to ISDN. Appending an “N” will set the number type to National and the numbering plan to ISDN. Appending an “n” will set the number type to National and the numbering plan to Unknown which is required by some switches. For called party numbers, appending an “S” will set the number type to Subscriber and the numbering plan to ISDN.

The default presentation indicator for calling party numbers is “presentation allowed.” Appending a “P” to the beginning string will set the presentation indicator to “presentation restricted.” The “P” can be appended either to the calling party number in the “DS” command or to the default directory number. If a “P” is not followed by any digits, the calling party number element will be present with “presentation restricted” but no digits. The “P” may not be used in conjunction with a character to set the numbering plan as described in the previous paragraph.

A subaddress number may also be added to the address string. This is done by attaching a hyphen as a separator and the subaddress digits to

the address string. For example, in the string “5551212-1234”, 5551212 is the address number and 1234 is the subaddress. Subaddresses are used to indicate extensions when calling to or from a private exchange such as a PBX.

To support supplementary services, address strings may represent the connected number and subaddress, or a redirecting number and subaddress. The connected address, which may appear in message types such as the CONNect, is always indicated by a “#” character as a separator, i.e. “#5551212”. This may be used to indicate that the connected party is not the party originally called, as when a call has been forwarded or transferred.

The redirecting number is an address from which a call has been deflected or forwarded, and usually appears in a SETUP message. Because of message length limitation, this number is indicated in a separate message of the form **D#xxRr[#]** where r is the reason for the redirection. This message will appear following the “DS” message for the SETUP message. Valid redirect reasons are, “A” for always, “B” for busy, “D” for deflected, “N” for no answer, and “O” for out of order. More than one redirecting number may be present. To include a redirecting number in a SETUP message from an NT port, a command message of the form **D#Rr[#]** must be sent prior to the “DS” command for the SETUP message. The redirecting number and subaddress will be added to the SETUP message when it is sent.

### **9.3.5 Progress Indicator**

Progress indicators are used to indicate something about the nature of the interworking of a call. This can be information about the fact that the call may not be an end to end ISDN call, or it may indicate the presence of inband information such as call progress tones or announcements. This element is present only in commands to NT ports or responses on CI ports as it is never found in messages sent by a user. The codes for a progress indicator are:

- C** Call is not end to end ISDN, call progress information may be available inband
- D** Destination address is non-ISDN
- O** Origination address is non-ISDN
- I** Inband information or appropriate pattern now available
- W** Delay in response at destination interface
- N** no progress indicator

### **9.3.6 Notification Indicator**

The Notification Indicator information element is used in the NOTIFY and some other types of messages to impart information relating to a call. In “D” messages, it is indicated by a single character:

- A** Call Transfer - Alerting call
- B** User Bridged onto call
- D** Privacy Disabled
- E** Privacy Enabled
- F** Call is Forwarded
- H** Call on Hold
- h** Remote Hold
- I** Idle resource or monitored user
- M** Monitoring discontinued
- N** No notification indicator or unknown
- R** Call Retrieved from Hold
- r** Remote Hold Release
- T** Call Transfer - Active call
- U** Update of Service Profile
- W** Call Waiting

### **9.3.7 Network Specific Facilities**

Some carriers offer special routing facilities that may provide lower tariffs when placing outgoing calls into the network. To access these

facilities, the SETUP message must include a Network Specific Facilities information element. This information element consists of a Carrier Identification Code and a facilities code specified by the network. The carrier code consists of three ASCII digits assigned as part of the North American Numbering Plan. As an example, AT&T is identified by “288” which spells out “ATT” on the keys of a touch tone telephone. The facilities code is a single octet. For instance AT&T’s MEGACOMM service is indicated by the octet 0E3h.

If the Network Specific Facilities information element is present, the network will use that facility when routing outbound calls. If the element is not present, the default facility will be used.

A Network Specific Facilities element may be specified by a command of the form **SAddabchh** where dd is the span, abc is the Carrier Identification Code, and hh is a pair of hexadecimal digits representing the facilities code. This command will affect all outgoing calls made on the span. As an example “SA00288E3” would set the network specific facilities to AT&T MEGACOMM for span 00. A command of the form **SAdd** will specify that no network specific facilities element is included in SETUP messages.

## **9.4 Default Directory Numbers**

Each B-channel has a default directory number associated with it. These numbers are set using the “SD” command (see section 4.3). This number may be up to fifteen digits. It also may include a seven digit subaddress. On CI ports, it is used for the calling number element in the SETUP message for calls originating from the port. If no number is defined for the default, the calling number element will be absent from the SETUP message. For NT ports, the directory number is used for the default called number element in SETUP messages for calls originating from the board and terminating on the terminal. If the application wishes to use a number other than the default, this can be encoded in the

“DS” message after the calling number with a “/” as a separator.

It is possible to use the default directory number, but attach a subaddress other than the default subaddress. This is done by omitting the calling or called number, but including the subaddress digits after a hyphen in the place normally reserved for the number in the “DS” message. For example, the message “DS00SN-1234/5551234” from a CI port would use the default number for the calling party number but would use 1234 as the calling party subaddress.

## 9.5 Display Text

Display Text is sometimes included in a message to impart additional information such as the calling party name, the status of the call, or features. Text can be sent to a terminal in an INFOrmation message or it can be included as an information element in another message. Text is normally only sent in messages from the NT (CO) side.

Most displays have one or two lines of twenty characters and text is usually sent in segments of this length. Each segment includes a tag, which indicates what the nature of the text is, i.e. called party number, cause, or generic text. This structure has been assumed for the purposes of the “D” message set.

When sending text, a buffer is used to build the Display Text element. Each segment of the text is sent in a “DT” command. When a “D” command is issued for a message such as SETUP, the text in the buffer will be sent as part of the message, and the buffer will be cleared. The “DT” command can take several forms. To begin filling the buffer the form **DTNt[*text*]** is used. This will clear the buffer of existing text and create a new element with one segment of text with a tag of t. Additional text segments may be added with the form **DTAt[*text*]**. Multiple segments may be added as long as the total length of the display text element is less than 128 octets. Note, that as some octets are



used for the segment tags and lengths that this represent fewer than 128 characters. Each segment typically has no more than 20 characters to match displays, but this is not a hard limit. Valid tag codes are:

<b>@</b>	Text	<b>O</b>	Called party name
<b>B</b>	Continuation	<b>P</b>	Original call name
<b>C</b>	Called address	<b>Q</b>	Redirecting name
<b>D</b>	Cause	<b>R</b>	Connected name
<b>E</b>	Progress Indicator	<b>S</b>	Originating restrictions
<b>F</b>	Notification Indicator	<b>T</b>	Date & time of day
<b>G</b>	Prompt	<b>U</b>	Call appearance ID
<b>H</b>	Accumulated digits	<b>V</b>	Feature address
<b>I</b>	Status	<b>W</b>	Redirection name
<b>K</b>	Inband	<b>X</b>	Redirection number
<b>L</b>	Calling Address	<b>Y</b>	Redirecting number
<b>M</b>	Reason	<b>Z</b>	Original called number
<b>N</b>	Calling party name	<b>[</b>	Connected number

Text that is sent to a port set as a CI will be passed to the application in a response message of the form **DTxxt[*text*]** where xx is the B-channel assigned the call is assigned to, t is the text tag, and the text is the text for that tag. If more than one text segment is present in the display text information element, multiple “DT” messages will be sent. These messages will follow immediately after the “D” response message for the message that contained the display text element.

A second text format is defined in ANSI T1.607. This is the “Display” element as distinguished from the “Display Text” element. This format is defined in Q.931 and the QSIG standards. In this format, text consists of unformatted characters up to a maximum of 20 or 80 (depending on the implementation). There are no tags indicating the type of text.

In this format, text is buffered as for Display Text. However, the command to put text in the buffer takes the form **DTQL[*text*]**. To add text to text already in the buffer, the command takes the form

**DTQA[*text*]**. These commands are sent before the command for the Layer 3 message that is to include the text. The response message for text contained in a message takes the form **DTxxl[*text*]** where “1” is the line number. Line “1” will contain the first 27 characters, line “2” the second 27, and so on till the end of the text.

Pre NI-1 Nortel DMS-100 and DMS-250 switches used a proprietary form of the Display information element which could be sent from either the network or customer side to pass name information. To send information a command of the form **DTDt[*text*]** is used to place the text in the buffer, where t is a tag indicating the type of text. The allowed types are “C” for Calling Party Name, “O” for Originating Party Name, and “X” for Connected Party Name. When a Display element in this format is received, a response of the form **DTxxt[*text*]** will be sent where xx is the B-channel, and t is the name type as indicated above.

## **9.6 Call Handling Procedures**

This section will describe the basic procedures used to establish and clear a call. The proper messages and their order will be given.

### **9.6.1 Call Establishment**

A call is initiated with a SETUP message. The SETUP message contains a number of pieces of information. The first item is the Bearer Capability. This describes the type of call, i.e. Speech, Data, etc. The second element is the Channel Identity which specifies the B-channel. These items are mandatory. Calls from a central office may also include a progress indicator. This is used to indicate that a call is not ISDN from end to end. The SETUP message may also contain the calling party number, though this is optional. Primary Rate ISDN SETUP messages will always have the called party number. Finally, SETUP messages from a CO may include a display text element. Other information elements are listed as optional in the specifications, but are

not typically used in basic call handling. These elements are not supported in the H.100 T1/PRI “D” message set.

When originating a call on a CI span, the SETUP command takes the form **DSxxbp(#)/#** where xx is the B-channel, b is the bearer capability, p the progress indicator, # is the optional calling party number, and /# is the called party number. If the calling party number is not specified in the “DS” message the default number associated with the B-channel will be used. This number is set using the “SD” command. If no default number has been specified and the calling party number is not included in the “DS” message, this element will be omitted.

The “D” command used on an NT span is similar. It takes the form **DSxxbp#(/#)** where xx is the B-channel, b is the bearer capability, p is the progress indicator, # is the calling party number and /# is the optional called party number. If the progress indicator is not needed because of interworking conditions, it should be coded as “N” or no progress indicator to be included. If the called party number is not specified in the “DS” message, the default number will be used. If there is no default number for the B-channel, one should be specified in the “DS” command.

When receiving a call, the response message is also a “DS” message. For a CI span it will take the form **DSxxbp#/#**, and for an NT span it will take the form **DSxxb(#)/#** where the parameters are as defined above.

Several different messages can be sent in response to a SETUP message. These are CALL PROCeeding, PROGress, ALERting, or CONNect. The order of these messages is fixed, though not all may be sent for each call. A CALL PROCeeding message is sent by the terminating end to indicate that enough information has been received to process the call. A PROGress message is sent to indicate that an interworking situation is present, for example, the destination is not an ISDN device. The ALERting message is sent when the call is presented to the user, and

the CONNect message is sent when the user answers the call. Because each message reflects the call state, a CALL PROCEEDing or PROGRess message should never be sent once an ALERTing message has been sent, and a CALL PROCEEDing, PROGRess, or ALERTing message should not be sent once a CONNect message has been sent.

If a call is to be rejected, a RELease COMplete message should be sent if no other message, for example, an ALERTing message has been sent previously. If one of the messages above has been sent, then a DISConnect message should be used. The reason for rejecting the call should be specified using the cause parameter.

The CALL PROCEEDing message takes the form **DPxx(p)** where xx is the B-channel and p is the optional progress indicator. The PROGRess message takes the form **DPxxPccp** and is sent only from the network side. The ALERTing message takes the form **DAxx(p)** where xx is the B-channel and p is a progress indicator. The ALERTing response message will also include the call reference.

The CONNect message takes the form **DCxx(p)** where xx is the B-channel, and p is the optional progress indicator. CONNect response messages will take the form **DCxxprrrr** where xx is the B-channel, p is the progress indicator, and rrrr is the call reference. The network will respond to CONNect messages by sending a CONNect ACKnowledge message which takes the form **DCxxArrrr**. This message is optional for the terminal side.

### 9.6.2 Call Clearing Procedures

Call clearing procedures fall into two cases. In one the far end initiates the clearing, and in the other, the near end is the initiator. In either case, the initiator sends a DISConnect message. The other end of the connection then responds with a RELease message. The initiator then sends a RELease COMplete. At this point, the call is no longer in existence, the call reference is no longer valid, and the B-channel is no

longer in use. The “D” command to send a DISConnect has the form:

**DDxxcc**

where xx is the B-channel and cc is the cause value.

The response to this should be a RELEase message which will cause the XDS Layer 3 Protocol to send a “DR” message to the application. It will also automatically send a RELEase COMplete message. An example of disconnecting is:

xmt: **DD1210** DISConnect, cause normal clearing  
rcv: **DR12R** RELEase message

The XDS Layer 3 Protocol software will automatically respond to a DISConnect message by sending the RELEase response. The other end should then respond with a RELEase COMplete. An example of this message sequence is:

rcv: **DD1210** DISConnect, normal clearing  
rcv: **DR1200** RELEase COMplete, no cause specified

Note that there is no need for the application to send a RELEASE message.

Calls should be disconnected using the DISConnect message except in a few circumstances. When originating a call, and no ALERTing or CALL PROCEEDing message has been received in response, or when a B-channel is not assigned to the call, the call should be released using a RELEase COMplete message. The following is an example where no response has been received:

xmt: **DS06SN5551234** SETUP message, called #5551234  
no response  
xmt: **DR0610** RELEase COMplete, normal clearing

It is possible for calls that are not assigned to a B-channel to be cleared. This is done by specifying the call reference. The DISConnect command takes the form **DDxxccrrrr** where rrrr is the call reference. The RELease command takes the form **DRxxRrrrr**, and the RELease COMplete command takes the form **DRxxccrrrr**. The responses take the same form, however, the B-channel will be coded as timeslot 23 on the span to indicate that the call is not associated with a B-channel, that is B-channel 17h, 37h, etc. will be used.

## 9.7 NT Call Handling Examples

This section and the one that follows will give some simple examples of the “D” messages that would be sent and received for a typical speech call. Only the “D” messages are shown. The “C” messages used to establish the audio paths will be explained in section 10.0.

### 9.7.1 Placing an Outbound Call

In this example, an outgoing call is initiated, answered, and then disconnected by the far end hanging up. The seventh B-channel (number 06) of the first span is used.

<u>commands</u>	<u>responses</u>	<u>description</u>
<b>DS06SN7654321</b>		SETUP message
	<b>DA060003</b>	ALERTing message
	<b>DC06N0003</b>	CONNect message
	<b>DD0610</b>	DISConnect message
	<b>DR0600</b>	RELease COMplete

In the “DS” command, the 06 indicates the B channel, in this case the seventh channel for the first span, the “S” indicates a bearer capability of speech, the “N” indicates that there is no progress indicator, and the calling number is 7654321. The far end sends an ALERTing message indicating that it has accepted the call and is alerting the user. The

CONNect message indicates the user has answered the call. The DISConnect occurs when the user hangs up. The cause code indicates normal clear. This causes the board to send a RELease message. The far end finally responds with a RELease COMplete message to end the call. The cause code of 00 indicates there was no cause code.

### 9.7.2 Receiving an Inbound Call

In this example, a call is placed by the far end. Enbloc sending is always used for PRI.

<u>commands</u>	<u>responses</u>	<u>description</u>
	<b>DS06SN5551000/5551212</b>	SETUP message
<b>DP06</b>		CALL PROCeeding
<b>DA06I</b>		ALERTing message
<b>DC06</b>		CONNect message
<b>DD0610</b>		DISConnect message
	<b>DR06R00</b>	RELease message

The initial SETUP message indicates that this is a speech call. The digits following the “N” are the calling party number. B-channel 06 was selected. A CALL PROCeeding message was sent in response indicating that the SETUP message had been received and there was enough information to process the call. When the user was notified of the call, an ALERTing message is sent with the progress indicator for inband tones. Ringback would be played to the B-channel to notify the caller. A CONNect follows when the call is answered. A “DD” command is issued with a cause of normal clearing to end the call. The far end responds with a RELease message. This causes the board to send a RELease COMplete message to end the call.

## 9.8 CI Call Handling Examples

Call handling from the perspective of the CI side is similar to that for an NT. However, there are some differences in which messages are sent and which information elements are present.

### 9.8.1 Originating a Call

In this example, the board is originating a call. As the board is a piece of automatic equipment, it is assumed that the “enbloc” sending method is always used for originating calls.

<u>commands</u>	<u>responses</u>	<u>description</u>
<b>DS10S/5551000</b>		SETUP message
	<b>DP10N</b>	CALL PROCeeding message
	<b>DA10I0001</b>	ALERTing message
	<b>DC10N0001</b>	CONNect message
	<b>DD10I0</b>	DISConnect message
	<b>DR10000</b>	RELEase COMplete message

The SETUP command indicates that the call is a speech call to 5551000. There is no progress indicator. The network responds with a CALL PROCeeding message with no progress indicator. This means that the network has accepted the call and is passing it on. When the far end has accepted the call and is alerting the user, an ALERTing message is sent with a progress indicator of inband tones. The CONNect is sent after the far end answers the call. When the far end disconnects the network sends a DISConnect to the board with a cause of normal clearing and a signal of tones off. This causes the board to send a RELEase. The network then responds with a RELEase COMplete message to end the call. This has a cause value of 00 indicating no cause element was present.



### 9.8.2 A Terminating Call

SETUP messages from the network always use “enbloc” sending to send the called number. It is up to the terminal to accept the call if the called number is part of its programming.

<u>commands</u>	<u>responses</u>	<u>description</u>
	<b>DS10SN5551000/5552000</b>	SETUP message
<b>DA10</b>		ALERTing message
<b>DC10</b>		CONNect message
	<b>DC10A8001</b>	CONNect ACKnowledge
<b>DD1010</b>		DISConnect
	<b>DR10R</b>	RELEase

In this example, the SETUP message from the network indicates a speech call with no progress indicator and no signal. The called number is 5551000 and the called number is 5552000. The application sends a “DA” command to cause an ALERTing message to be sent. There are no arguments for the alerting message for a CI port. A CONNect follows which causes the network to respond with a CONNect ACKnowledge. A DISConnect message with normal clearing is sent to hang up on the call. The network responds with a RELEase which causes the board to send a RELEase COMPLETE to end the call.

## 9.9 Call Processing Errors

Calls do not always progress smoothly. Sometimes there is an error in dialing, the called number is busy, or no route may be available. It is also possible for a terminal to request a facility that is not supported or subscribed to. Switching errors can occur or some other problem may arise creating an error situation.

In these cases the network will send a PROGRESS message. This message will indicate the nature of the problem and for speech or audio

calls may indicate the presence of an inband tone or message. To send a PROGRESS message to a terminal, the application would send a command of the form **DPxxPccp**, where xx is the B-channel of the call, cc is the cause of the PROGRESS message, and p is the progress indicator.

As an example, the message **DP06P11I**, would indicate that the terminal trying to place a call on the 06 B-channel has dialed a busy number (cause 11). The “I” indicates that there is an inband tone present. It is the application’s responsibility to provide the inband signal using a “CP062” command (see section 9.0).

For ports set as a TE, a response of the form **DPxxPccp** will indicate that a problem has occurred. The cause will be indicated by the cc, and the progress indicator and signal will indicate if any error tones or messages are available.

## 9.10 Layer 3 Timers

Several timers are specified at Layer 3 to deal with cases where the expected response is not received. These timers prevent calls from just hanging indefinitely. If the timer times out, a specific action is taken to either elicit a response or clear the call. The following table lists the supported timers.

No action relating to the board is required on the part of the application when a timeout occurs, as the firmware will send the appropriate message. Of course the application must take the appropriate internal actions such as removing a call from internal tables or clearing the near end connection. Layer 3 timers may be disabled by setting system option 4 to ‘Y’.

When a timer times out, the board will send a message of the form **ETxxe** where xx is the B-channel, and e is a value specifying the timer. The values of e are:

- 0 T303 has expired the first time
- 1 T303 has expired the second time
- 2 T310 has expired
- 3 T305 has expired
- 4 T308 has expired the first time
- 5 T308 has expired the second time
- 6 T316 has expired
- 7 T322 has expired

Timer	timeout value	cause for initiation	cause for termination	action on expiry
1st T303	4 sec.	SETUP is transmitted	CALL PROC, ALERT., CONNect or RELease COMplete	retransmit SETUP
2nd T303	4 sec.	SETUP retransmitted	as above	send RELease
T305	30 sec.	DISConnect sent	DISConnect, RELease, RELease COMplete	send RELease
1st T308	4 sec.	RELease sent	RELease, RELease COMplete	retransmit RELease
2nd T308	4 sec.	RELease retransmitted	as above	send RESTART
T310	10 sec.	CALL PROC received	ALERTing, CONNect or DISConnect	send DISConnect
T316	120 sec.	RESTART	RESTART ACK	retransmit RESTART
T322	4 sec.	STATUS ENQUIRY	STATUS, DISC. RELease RELease COM	send DISConnect

## **9.11 Supplementary Services**

Supplementary Services are additional features that go beyond basic call handling. They may include the provision of additional information about a call such as an identifying name associated with one of the parties to a call, or they involve routing procedures such as transferring a call or call deflection.

Supplementary Services have been defined by a number of entities. Proprietary services have been specified by the various switch vendors, and are unique to a specific vendor or switch type. Those services that fall under the various phases of National ISDN have been defined by Telcordia (formerly Bellcore) in documents generated by that body. In Europe, an effort known as QSIG has defined a number of services to promote interoperability between Private Branch Exchanges (PBXs). These are detailed in documents from the European Computer Manufacturer’s Association (ECMA), the European Telecommunications Standards Institute (ETSI) and the International Standards Organization (ISO). These three bodies have tried to harmonize their documents with the ISO documents taking precedence. While QSIG was initially a European effort, many switch vendors have incorporated QSIG features into switches available in North America.

Supplementary Services are usually enabled through a subscription parameter if connecting to the PSTN or through an administration procedure if connecting to a PBX. To determine which services are available and if they are enabled, it will be necessary to consult with the PSTN, the administrator, or the switch vendor.

### **9.11.1 Facility Messages**

Many Supplementary Services use FACILITY messages to initiate or acknowledge an action. These messages include a Facility information element which is used to contain the information necessary for the

service. Each Facility element includes a component. Components come in four types, Invoke, Return Result, Return Error, and Reject. The Invoke component is used to initiate a service. It includes an invoke identity tag which is used to identify a specific instance of an invocation. This tag is used in subsequent response messages to indicate that the response is for that specific invocation. An invoke tag is a unique value from 01h to FFh (in some cases 0h to 0FFFFh) and must not be used by another invocation until the requested service has been completed. A Return Result component is used to acknowledge an invoke request, and it may or may not include additional information. A Return Error component is used to indicate that an error has occurred in the service invocation and will include an error code indicating the reason. A Reject component is used to indicate that an invoked service has been rejected, and will contain a problem code indicating the cause of the rejection. This will usually be because the facility information element can not be understood.

Each service has its own format and argument structure, which will be described in subsequent sections. However, the H.100 T1/PRI Board provides a generic form for FACILITY response messages where a more detailed format is not required. The generic format for the FACILITY message is:

**DFxxcid(tt)**

where xx is the B-channel of the call for which the service is being invoked, c is the component type, id is the invoke id tag, and tt is an optional value such as an operation type or error value. The component types are:

- I**     invoke
- R**     return result
- E**     return error
- J**     reject

As an example of the message sent for a return result component:

rcv: **DF00R47**            return result, invoke id 47

Note that in this case no value was returned as part of the return result. If an error condition had occurred the response would have been:

rcv: **DF00E4703**        return error, invoke id 47, error 3 “not available”

If a reject component was received the response would be:

rcv: **DF00J4701**        reject, invoke id 47, unrecognized operation

It should be noted that due to the complexity of some of the components, the Facility elements for all services are not supported at this time. Those that are supported are detailed in the following sections.

### **9.11.2 Name Identification Service**

The Calling or Connected Party Name may be part of a message if the Name Identification Presentation supplementary service is supported. This feature is described in *ANSI T1.641-1998* and *Telcordia GR-1367-Core Issue 3, March 2000*. The QSIG specifications for this feature are contained in *ECMA-163* and *ECMA-164*, while the ISO documents are *ISO/IEC 13864* and *ISO/IEC 13868*.

The Name Identification information is carried in a Facility information element as part of a SETUP or CONNECT message. If present, it appears as a message of the form **DFxxNo[name]** where xx is the B-channel assigned to the call, o is a number representing the operation, and name is the string of characters. The operation values are 0 for calling name, 1 for called name, 2 for connected name, and 3 for busy name. This string is normally limited in length to 15 or 20 characters. The “DF” message will follow the “DS” or “DC” message with which

is associated.

To include a calling or connected name in a SETUP or CONNECT message, the name is sent in a command of the form **D@Nido[name]** where id is a hexadecimal invoke identity in the range 00-FFh, o is the operation code as defined above, and the name is the character string to be sent. This command must be sent before the “DS” or “DC message with which it is to be associated.

The Telcordia Generic Requirements specifies a slightly different form for the Facility information element than that specified by QSIG. This form can be sent using a command of the form **D@nido[name]** where all the fields are as described in the paragraph above.

### **9.11.3 Release Link Transfer**

Release Link Transfer or RLT is a proprietary feature of some Nortel switches. It is available on the DMS-100 NI-1 switch and some models of the DMS-250. It is not available on DMS-100 NI-2 switches (Explicit Call Transfer may be used instead). This feature is described for the DMS-100 in the Nortel document *NIS-A211-1 ISDN Primary Rate User-Network Interface Specification*. When connecting to the DMS-100 switch, the switch level must be set to “D” with the “SL” command.

The RLT procedure consists of two steps. First, when placing a call to the number the call is to be transferred to, a facility information element with an operation of RLT Operation Indication is included in the SETUP message. If the switch decides the call can be transferred, it will return a Call ID in a facility element in either an ALERting or PROGRESS message. Which message will be used depends on whether there is an interworking situation, but it will always be the first message received. This Call ID is not the same as the call reference. When the call is to be transferred, which must be after an ALERting or CONNect message has been received for this call, a FACILITY message is sent using the call

reference of the first call and a facility information element including the Call ID of the second call with an operation value of RLT Third Party. If accepted, the two parties will then be connected and the two calls will be cleared using the normal disconnect procedure.

To include the RLT Operation Indication in a SETUP message, the command **D@R** must be issued immediately before the “DS” message for the SETUP. The Call ID will be returned in a message of the form **DFxxNI<CallID>** where the Call ID is represented as six or eight hexadecimal digits. To complete the transfer a message of the form **DFxxL<CallID>** should be sent on the B-channel of the call to be transferred with the Call ID being the same digits as received. If the RLT is accepted, a FACILITY message of the form **DFxxNT** will be received. If there is an error, a message of the form **DFxxNEidee** will be received where xx is the B-channel, id is the invoke id which will be 01 if the error comes at the first step and 02 during the second, and ee is the error. Valid error codes are:

- 10 RLT Bridge Fail
- 11 RLT Call ID Not Found
- 12 RLT Not Allowed
- 13 RLT Switch Equipment Congestion

If the RLT is rejected, a message of the form **DFxxNJidppee** where xx is the B-channel, id is the invoke id, pp is the problem code, and ee is the error code. Normally only an invoke id of 00, a problem code of 82 and an error code of 02 is valid.

The following example shows the messaging from the CI side:

<u>commands</u>	<u>responses</u>	<u>description</u>
	<b>DS00SN/5551212</b>	SETUP message of first call
<b>DC00</b>		CONNECT for first call
	<b>DC00A8001</b>	CONNect ACK
<b>D@R</b>		RLT Operation Indication



<b>DS01SN/5551234</b>		SETUP message for second call
	<b>DP01I</b>	CALL PROCEEDing for 2nd call
	<b>DA01I</b>	ALERTing message for 2nd call
	<b>DF01NI123456</b>	Call ID of second call
<b>DF00L123456</b>		FACILITY RLT Third Party
	<b>DF00NT</b>	FACILITY RLT accepted result
	<b>DD0010</b>	DISCONNect for 1st call
	<b>DD0110</b>	DISCONNect for 2nd call
	<b>DR0010</b>	RELease COMplete for 1st call
	<b>DR0110</b>	RELease COMplete for 2nd call

The H.100 T1/PRI board may also emulate the NT side of an RLT. When this is done, a message of the form **DFxxNR** will indicate the RLT Operation Indication invoke component. The Call ID is sent in a command of the form **D@I<CallID>** which must be sent immediately before the ALERTing or PROGRESS message for the second call. A received message of the form **DFxxNT<CallID>** will indicate an RLT Third Party invoke component. The return result component is sent with a command of the form **DFxxLR02**. Note that it is the responsibility of the application to manage the Call ID values and to clear the calls after making the transfer.

### **9.11.4 Enhanced Explicit Call Transfer**

Enhanced Explicit Call Transfer (ECT), also called Two B-channel Transfer, is a mechanism to join the parties associated with the far end of two calls together while the near end parties drops out and the B-channels are released. One of the calls must be in the connect state, while the other call may be in either the connect state if an incoming call, or in the connect or alerting states if an outgoing call. This feature is included in HI-2 and is described in *ANSI T1.643-1998* and *Telcordia GR-2865-Core Issue 3, March 2000*.

In a typical application, a party A places an incoming call to the interface. A call is then placed to party B using a different B-channel.

This call can be in either the alerting state (ringing) or the connect state (answered). The transfer is then initiated from the interface. If the transfer is accepted, party A and party B are connected together and both calls are cleared and the B-channels released.

To invoke Explicit Call Transfer (ECT), the initiating interface sends a FACILITY message for one B-channel which contains the call reference of the call on the other B-channel as part of a facility element. If one of the calls in the alerting state, the Facility message must be sent for the B-channel of the connected call. The “D” message to do this takes the form **DFxxTidrrrr** where xx is the B-channel of one of the calls, id is an invoke identity, and rrrr is the call reference of the other call. The invoke id is a number in the range 00-7F which is used to identify the request for transfer. It is only valid until the transfer has either been granted or rejected.

In the following example, an incoming call is connected on the 02 B-channel. A second call is then made on the 06 B-channel. When that call is answered, a FACILITY message is sent to request the ECT. The request is granted and both calls are released. This example is shown from the CI side.

<u>commands</u>	<u>responses</u>	<u>description</u>
	<b>DS02SN/5551234</b>	Incoming SETUP message
<b>DA02</b>		ALERTING for 1st call
<b>DC02</b>		CONNect for 1st call
	<b>DC02A8001</b>	
<b>DS06SN/5551212</b>		SETUP message for 2nd call
	<b>DA06I0003</b>	ALERTing message
	<b>DC06N0003</b>	CONNECT message
<b>DF02T130003</b>		FACILITY invoking ECT, id=13
	<b>DD0210</b>	DISConnect for 1st call
	<b>DR0200</b>	RELease COMplete for 1st call
	<b>DD0610</b>	DISConnect for 2nd call

**DR0600**

RELease COMplete for 2nd call

The FACILITY message used to invoke the call transfer has an invoke id of 13 and uses the call reference of the second call, 0003. There may be a received FACILITY message with a Return Results component. If the transfer request is not accepted, there should be a Facility message with either an Error or Reject component.

If the two calls involved in the transfer use different D-channels, a different form of the command is used. This form is **DFxxTidrrrr-dcid** where dcid is the D-channel identity. The D-channel identity may be discovered by sending a message of the form **DFxxTidI** where xx is the B-channel served by the D-channel in question and id is the invoke id. The identity will be returned in a message of the form **DFxxTidIdcid** where xx is the B-channel, id is the invoke id used in the request, and dcid is the D-channel identity. Note, as the D-channel identity is a static value, there is no need to request it for every call.

An earlier form of this feature, Explicit Call Transfer is still used on some legacy switches. For this service, the command should take the form **DFxxEidrrrr**.

### **9.11.5 Call Deflection**

Call Deflection is a Supplementary Service that allows an incoming call to be diverted to another destination. This is used, for example, when the called party is busy or unavailable. Call Deflection is described in *ANSI T1.642-1995, Telcordia GR-1310-CORE, or SR-4494 2000 NIPRI CPE Generic Guidelines Issue 1, December 1999.*

To deflect a call, the command is of the form **DFxxDidr[#]** where xx is the B-channel of the call, id is the invoke id, r is the reason, and # is the directory number of the destination the call is being deflected to. If the call deflection request is accepted, the call will be cleared in the normal manner with a DISConnect message. Valid reasons for deflection are:

<b>B</b>	user busy
<b>D</b>	call deflection
<b>N</b>	no response
<b>U</b>	unconditional
<b>X</b>	unknown

As an example, the following shows a call being deflected because the user is busy:

<u>command</u>	<u>response</u>	<u>decription</u>
	<b>DS04SN/5551212</b>	SETUP message
<b>DA04</b>		ALERTing message
<b>DF04D07B5551234</b>		FACILITY message
	<b>DF04R800113</b>	FACILITY, Return Results
	<b>DD0410</b>	DISConnect message
	<b>DR0400</b>	RELEase COMplete message

The FACILITY message has an invoke id of 07, a deflection reason of “user busy” and a destination number of 555-1234. A FACILITY message with a Return Results component is received indicating that the call deflection request has been accepted.

### 9.11.6 QSIG Call Diversion and Call Forwarding

QSIG Call Diversion and Call Forwarding (QSIG-CF) are services that allow a user to send calls for that user to another number. The calls may be forwarded unconditionally (CFU), if the user is busy (CFB), if there is no answer at the user (CFNR - Call Forwarding No Reply), or on a call by call basis (CD - Call Deflection). The ECMA documents describing these services are *ECMA-173* and *ECMA-174* and the ISO documents are *ISO/IEC 13872* and *ISO/IEC 13873*.

With the Call Forwarding service, a call is placed from the originating party to the diverting user. This is called Leg 1. A call can then be placed from the PBX serving the diverting user to the forwarding

destination which is called forward switching, or it can be placed directly from the PBX serving the originating party which is called rerouting. In either case, this is called Leg 2. The connection from the destination to the originating party is called Leg 3. The messages involved in call forwarding inform the calling and destination users that diversion has occurred.

From the viewpoint of the H.100 T1/PRI Board, there are two cases, either a call is diverted to the board, or a call originating on the board is diverted. It is assumed that the board is acting as a CI connected to a PBX.

In the case where a call is diverted to the board, there will be a SETUP message containing a Facility information element with diverting Leg 2 information. This will be indicated by a message of the form **DFxxF2cr#(/name)** immediately following the “DS” SETUP response message, where xx is the B-channel, c is the diverting count, i.e. the number of times the call has been diverted, r is the diversion reason, # is the number the call was diverted from, and /name is an optional diverting party name. Diversion reasons can be “B” for busy, “D” for deflected, “N” for no answer, or “U” for unconditional, that is all calls are being forwarded. The application should respond with a diverting Leg 3 information command message of the form **DFxxF3id** where xx is the B-channel of the SETUP message, and id is an invoke id.

In the case where a call originating from the board is diverted there will be two response messages. The first will be a diverting Leg 1 information FACILITY message of the form **DFxxF1ro#** where xx is the B-channel of the call, r is the diversion reason, o is the subscription option and # is the number the call was diverted to. Subscription option values are “0” for no notification, “1” notification without diverted to number, and “2” for notification with diverted to number. The second response message will take the form **DFxxF3** indicating a diverting Leg 3 information FACILITY message on B-channel xx.

### 9.11.7 QSIG Call Transfer by Join

QSIG Call Transfer by Join (QSIG-CT) is a service that allows a PBX to notify another PBX that it has joined two separate calls into a single call. The notified PBX can then inform the user that a join has taken place by updating a display or through other means. The ECMA documents describing this service are *ECMA-177* and *ECMA-178* and the ISO documents are *ISO/IEC 13865* and *ISO/IEC 13869*.

In call transfer by join a call is placed from party A to party B with the two parties in different switches. A call is then placed from party B to party C which is in a switch other than the one serving party B. When the transfer takes place, A and C are connected together and B drops out of both calls. Note that both paths from A and C to the switch serving B remain.

To notify the switches serving A and C that a transfer has taken place and they are now connected together, a FACILITY message is sent to each switch with a Facility information element with the ctComplete operation. This element contains the new party number that the other party is now connected to, that is, the message to the switch serving A contains C's number and vice versa. The switch can then inform the parties of the number using a display or other means.

To send a ctComplete FACILITY message a command is issued of the form **DFxxCide#**. where xx is the B-channel of the call, id is an invoke id, e is the end designator and # is the number. The end designator can be either 'P' for Primary which would be A in the above example, or 'S' for Secondary which would be C. The default coding for the number is as an publicPartyNumber, nationalNumber. In some cases where the number is an internal extension this may need to be encoded as a privatePartyNumber, pISNSpecificNumber. This can be done by preceding the number with a 'P', for example, "P1234".

If party C has not yet answered, an 'A' should be added to the end of the

message to the primary end to inform the switch that the call is in the alerting state. In this case, a ctActive FACILITY message should be sent to the primary end when party C goes to the connect state. This is done with a message of the form **DFxxAid#** where xx is the B-channel of the primary end call, id is the invoke id, and # is the party number of C.

Some switches may send a ctUpdate Facility message after receiving a ctComplete. The transferring switch should forward this message on to the other leg of the transfer with identical information. The ctUpdate message takes the form **DFxxUid#** where id is the invoke id and # is the redirection number. This message takes the same form for both the command and response.

### **9.11.8 QSIG Path Replacement**

QSIG Path Replacement is a service which allows a connection involving multiple PBXs to be replaced with a simpler path. In particular, a connection created by a Call Transfer by Join which joins two calls to another PBX can be replaced by an internal connection in the PBX which contains the two end points. This situation is sometimes called a “trombone” connection for obvious reasons. The PBX with the end points will recognize that a trombone connection has been created using a stimulus such as receiving two ctComplete FACILITY messages with complementary redirecting numbers. It will then initiate the path replacement by sending FACILITY messages with a pathReplacePropose operation.

This facility message will take the form **DFxxPid,c-#** where xx is the B-channel, id is the invoke id, c is the call identity, and # is the redirection number. Note that the call identity can be more than one digit long. The PBX that joined the two calls should respond by echoing the pathReplacePropose message on the other call to signal acceptance. The initiating PBX will then create an internal connection and clear the calls. The default type for the redirection number is unknown. A letter

preceding the number indicates a different type. The letter “N” indicates a public National number, while an “L” or a “P” indicate a private local or private PISN Specific number type.

The following is an example of a Call Transfer by Join followed by a Path Replacement as seen from the perspective of the joining PBX:

<u>commands</u>	<u>responses</u>	<u>description</u>
	<b>DS00SN1234/1000</b>	SETUP for primary call
<b>DA00</b>		ALERTing message
<b>DC00</b>		CONNect message
<b>DS02SN/1212</b>		SETUP for secondary call
	<b>DA02I0001</b>	ALERTing message
<b>DF00C12P1212A</b>		ctComplete, primary call
<b>DF02C13S1234</b>		ctComplete, secondary call
	<b>DC02N0001</b>	CONNect message
<b>DF00A141212</b>		ctActive message
	<b>DF00P15,47-1234</b>	pRPropose message
<b>DF02P15,47-1234</b>		pRPropose message
	<b>DR0010</b>	RELEase COMplete
	<b>DR0210</b>	RELEase COMplete

### 9.11.9 QSIG Call Transfer by Rerouting

QSIG Call Transfer by Rerouting is another method of accomplishing a transfer on PBXs that support the service. The ECMA documents describing this service are *ECMA-177* and *ECMA-178* and the ISO documents are *ISO/IEC 13865* and *ISO/IEC 13869*.

The scenario for Call Transfer by Rerouting is similar to that for Call Transfer by Join and Path Replacement, except that the Rerouting is initiated by the local switch. This is done by the local switch requesting a call identity for the call placed to party C. This call identity is then used to initiate the rerouting.



The call identity is obtained by sending a FACILITY message using the call reference of the call to party C and an invoke component with a ctIdentify operation. This is done with a command of the form **DFxxIid** where xx is the B-channel of the call to party C and id is the invoke identity. The remote switch will send a FACILITY message with a result component using the same invoke id. The result component will contain the call identity as well as the redirecting party number. This will be in a response message of the form **DFxxIid,c-#** where xx is the B-channel, id is the invoke identity, c is the call identity and # is the redirecting party number. The call identity is typically several digits long. The rerouting is then initiated using a command of the form **DFxxRid,c-#** where xx is the B-channel, id is the invoke identity, and c and # are the call identity and redirecting party number obtained using the ctIdentify command.

The following is an example of a Call Transfer by Rerouting from the perspective of the local PBX:

<u>commands</u>	<u>responses</u>	<u>description</u>
	<b>DS00SN1234/1000</b>	SETUP for primary call
<b>DA00</b>		ALERTing message
<b>DC00</b>		CONNect message
<b>DS02SN/1212</b>		SETUP for secondary call
	<b>DA02I0001</b>	ALERTing message
	<b>DC02N0001</b>	CONNect message
<b>DF00I14</b>		ctIdentify message
	<b>DF00I14,47-1234</b>	ctIdentify result message
<b>DF02R15,47-1234</b>		ctInitiate message
	<b>DR0010</b>	RELEase COMplete
	<b>DR0210</b>	RELEase COMplete

### 9.11.10 QSIG Single Step Call Transfer

QSIG Single Step Call Transfer (QSIG SSCT) is a service that allows a user to transfer a call to another user without first placing a second call

to the transferred user. It is a service intended for PBXs rather than for PSTN connections. The ECMA documents describing this feature are *ECMA-299* and *ECMA-300* and the ISO documents are *ISO/IEC 19459* and *ISO/IEC 19460*.

As its name implies, the QSIG SSCT service involves a single step. It is initiated by sending a FACILITY message with the call reference of a connected call which includes an invoke component with an *ssctInitiate* operation with the number of the transferred-to party. If accepted, the switch places a call to this party. Two modes are allowed. In the first the call is transferred upon the switch receiving an ALERTing message, while in the second mode the transfer does not take place until a CONNect message is received.

To initiate a transfer, a command of the form **DFxxSidm##** is issued, where xx is the B-channel of the call to be transferred, id is the invoke id, a two digit hexadecimal number, m is the mode, either ‘A’ if the transfer is made upon receiving an ALERTing message or ‘C’ if it is made upon receiving a CONNect message, the first ‘#’ is the transferred-to parties number, and the second ‘#’ is the number of the transferred call that is to be presented to the transferred-to party.

If the transfer is successful, then a DISConnect message will be received by the transferring user to clear the call. If the transfer is not accepted, a message of the form **DFxxEidee** will be received where id is the invoke id in the command initiating the transfer and ee is an error code.

### **9.11.11 QSIG Message Waiting Indication**

The QSIG Message Waiting Indication allows a device such as a voice mail server to control an indication for a user on a PBX so that the user may know if there are any messages waiting for that user. The form of the indicator may be a display, a distinctive dial tone, or other means. The ECMA documents describing this service are *ECMA-241* and *ECMA-242* and the ISO documents are *ISO/IEC 15505* and *ISO/IEC*

15506.

Controlling Message Waiting Indicators involves placing a call to the PBX serving the user. The SETUP message will include a Facility information element to activate or deactivate the indicator. The Bearer Capability should be set to ECMA unrestricted digital data and no B-channel should be assigned. No audio path is necessary for this call.

Immediately before sending the “DS” command for the SETUP message, a command message of the form **D@Mido#(#)(=m)** should be sent to construct the Facility information element, where id is the invoke id, o is the operation, either ‘A’ to activate or ‘D’ to deactivate the message waiting indicator, and # is the user number. Optionally this message may contain a number for the message center which will take the form /#, and/or ‘=m’ where m is the number of messages waiting. The SETUP message should then be sent. The command to do this takes the form **DSxxEN#/#** where xx is the B-channel, the first ‘#’ is the calling party number and the second ‘#’ is the called party number which corresponds to the hunt group of the user. Note that to select a channel id of no B-channel, the B-channel should be set to a channel one more than the highest channel on that span, i.e. 17 for span 0, 37 for span 1 etc.

The called switch should respond with a CALL PROCeeding and CONNect message as normal. The CONNect message should include a Facility information element with a Return Result component to acknowledge the operation. The call should then be cleared in the normal way.

As an example of activating call waiting for station 1234 which belongs to hunt group 1000:

<u>command</u>	<u>response</u>	<u>description</u>
<b>D@M03A1234</b>		build Facility i.e.
<b>DS17EN/1000</b>		SETUP message

	<b>DP17N0004</b>	CALL PROCeeding message
	<b>DC17N0004</b>	CONNeCT message
	<b>DF17R03</b>	FACILITY, Return Results
<b>DR17R</b>		RELease message
	<b>DR1700</b>	RELease COMplete

### 9.11.12 HOLD & RETrieve

The Call Hold service provides a means for interrupting a call and then later restoring it. It is defined in *Q.932* as well as in *ANSI T1.616-1992 Call Hold Supplementary Service*. This service is not part of National ISDN-2 and is not defined in a Telcordia document for a PRI interface.

An active ISDN call can be placed on hold to free up a B-channel for use by another call. The held call remains active and retains its call reference, but the connection is broken. The connection can be restored by retrieving the call and assigning a new B-channel for the circuit. This does not have to be the same B-channel as was originally used.

An interface connected to an NT port can initiate a hold by sending a HOLD message. This results in the “D” response message **DHxxrrrr**, where xx is the B-channel and rrrr is the call reference. The call reference is important as it is the only way a call can be retrieved once it is placed on hold. Once a call has been placed on hold, it is the responsibility of the application to retain the call reference until the call is retrieved or released. To confirm the hold, the application would send a message of the form **DHxxA** to acknowledge the hold. If it did not want to accept the hold, it would send a message of the form **DHxxRcc**, where cc is the reason the hold was rejected. Once the HOLD ACK message has been sent, the B-channel is available for a new call.

To retrieve the held call, the terminal would send a RETrieve message. This would cause a “D” response message of the form **DGxxrrrr**, where xx is the B-channel and rrrr is the call reference. If the CI did not choose a B-channel, the board would assign one if available. To allow

the retrieval, the application would send the command **DGxxArrrr**. To reject the retrieval, the application would send the command **DGxxRccrrrr** where cc is the cause for rejecting the retrieval.

When the span on the H.100 T1/PRI Board is a CI, it can request that a call be placed on hold in a similar manner. The application would send a command of the form **DHxx** where xx is the B-channel of the call to be held. If the network responds with a HOLD ACK, the board would send a response message of the form **DHxxArrrr** where rrrr is the call reference. A rejection would result in a response message of the form **DHxxRccrrrr**, where cc is the cause for rejection.

The call would be retrieved by the application sending a command of the form **DGxxrrrr**, where xx is an available B-channel and rrrr is the call reference of the held call. This message should not be sent if a B-channel is not available. If the call is retrieved the network will send a RETrieve ACK message which will result in a response message of the form **DGxxArrrr**. If the network sends a RETrieve REJect message, the response message would be **DGxxRccrrrr**, where cc is the cause for the rejection.

In the examples above, it is the terminal that initiates the hold. It is also possible for the network side to place a call on hold to free the B-channel. However, only the terminal side can retrieve a call.

It is possible to release a held call by sending a RELease COMplete. This might be done if the far end disconnects. In this case, the “DR” command or response message would include not only the cause, but the call reference. It would be of the form **DRxxccrrrr**, where rrrr is the call reference.

## 9.12 Layer 3 Maintenance Messages

Several Layer 3 messages are provided for maintenance purposes. These consist of the RESTART message which is used to reset a B-channel or interface and the SERVICE message which is used to mark a B-channel as available or unavailable.

### 9.12.1 RESTART & RESTART ACKNOWLEDGE

The Restart Procedure is used to reset a B-channel or span to insure that both sides of the span are in agreement as to the state of B-channel or span by placing the B-channel or channels in the null state. This is done if there has been some indication that there has been a problem which can not be resolved by the normal clearing procedures. It is sometimes done upon recovery of a persistent Layer 1 error.

The side receiving a RESTART message should clear the indicated B-channels and send a RESTART ACKnowledge message. Upon receiving a RESTART ACKnowledge message the initiating side should clear the indicated B-channels. The Restart procedure is symmetrical and can be sent by either the NT or CI side.

A RESTART message can be send by issuing a command of the form **DZxx** to reset a span, **DZxxC** to reset the indicated B-channel, or **DZxxA** to reset all the spans associated with the D-channel in an NFAS arrangement.

The reception of a RESTART message is indicated by a response message of the form **DZxxA** if all spans associated with the D-channel are to be reset, **DZxxC** if only the indicated B-channel is to be reset, or **DZxxI** if all B-channels on the span starting with the indicated B-channel are to be reset.

The reception of a RESTART ACKnowledge message is indicated by

response messages of the form **DZxxAA**, **DZxxAC**, or **DZxxAI**, for reset all spans, reset a specific B-channel, or reset B-channels on a span starting with the indicated span, respectively.

It is not necessary for the application to take any action to clear the B-channels on the board. Of course, the application must take any internal actions that are necessary to reflect the fact that the B-channels have been reset.

### **9.12.2 SERVICE & SERVICE ACKNOWLEDGE**

The SERVICE message is used to indicate B-channel availability or to indicate a D-channel switchover in NFAS arrangements where there is a backup D-channel.

To initiate a change in the status of a channel or channels, a SERVICE message is sent. The other side of the interface replies with a SERVICE ACKnowledge message. A channel can be marked as being in service, out of service or in a maintenance state. Details of the procedures associated with these messages can be found in Annexes F and U of *ANSI T1.607*.

The command to change the status of a B-channel takes the form **XSxxsC** for an individual B-channel, or **XSxxs** for an entire span where xx is the B-channel and s is the state. For a D-channel the command takes the form **XSxxsD** where xx is the first B-channel on the span and s is the state. Allowed values for the state are “I” for in service, “O” for out of service, and “M” for maintenance.

The response message indicating a SERVICE message has been received takes the form **XSxxsC** for a single B-channel, **XSxxs** for all B-channels on a span, and **XSxxsD** for a D-channel. The SERVICE ACKnowledge message will automatically be sent by the board. The response message indicating a SERVICE ACKnowledge message has been received takes the form **XSxxAsC** for a single B-channel, **XSxxAs**

for all B-channels on a span, and **XSxxAsD** for a D-channel.

It is the responsibility of the application to mark the availability of B-channels and to avoid initiating calls on channels that are out of service. It is also the responsibility of the application to ensure that the correct D-channel is used on an NFAS arrangement by sending the appropriate values in the “SN” command.

It should be noted that the protocol discriminator used for SERVICE messages is not 08h indicating a Q.931 message. A protocol discriminator of 43h is specified by *ANSI T1.607* and the Telcordia NI-2 specifications. However, some pre NI-2 switches use a protocol discriminator of either 01h (some early AT&T switches) or 03h (Nortel pre NI-2). When connecting to these switches the switch level in the “SL” command should be set to “A” or “D” respectively. For NI-2 switches it should be set to “N”.



## **10.0 Controlling the B-Channels**

This section describes the process used for controlling the B-channels on the Infinity Series H.100 T1/Primary Rate ISDN Board. The steps necessary for configuring a board will be described. Basic functions such as making a connection, playing call progress and DTMF tones, and detecting DTMF tones will be described. In addition, examples of making incoming and outgoing calls combining several functions are included.

For interfaces operating as Primary Rate ISDN, control of the B-channels is completely independent of call control procedures involving Layer 3 messages. Neither “LC” or “D” commands affect the switching of the B-channels, and the commands described in this section do not result in any Layer 3 Q.931 messages being sent. For interfaces operating as T1 spans using robbed bit signaling, some aspects of the switching may be affected by changes in hook-status.

### **10.1 Overview of the Command Structure**

The command set described in this section are called the “C” Commands. Each of these commands instructs the board to take all the actions necessary to perform a function such as making a connection, playing a tone, or detecting DTMF digits. Thus, the command to detect DTMF digits will play dial tone to the B-channel, connect it to a DTMF detector, and activate that detector.

Many of the commands use a common form for the various arguments. The B-channel number is usually the first argument. The B-channel numbers begin at 00 for span 0, 20h for span 1, and so on. For T1 spans, there are 24 channels per span, numbered 00-17h, 20-37h etc. For PRI

spans, the last channel is used as a D-channel, and so only 23 channels are available. Channel assignments are given in the following table:

Span #	T1 Span Channels	PRI Span Channels
0	00-17h	00-16h
1	20-37h	20-36h
2	40-57h	40-56h
3	60-77h	60-76h
4	80-97h	80-96h
5	A0-B7h	A0-B6h
6	C0-D7h	C0-D6h
7	E0-F7h	E0-F6h

The arguments used to define the source or destination timeslots consist of a four digit hexadecimal number. The first two digits are used to indicate the stream. There are 32 streams defined on the H.100 bus and these range from 00 to 1F. The last two digits are used to indicate the timeslot. H.100 streams have 128 timeslots ranging from 00-7Fh. As an example, the argument **712** would refer to stream 7, timeslot 12h, or 18 decimal.

## **10.2 Legacy Computer Telephony Busses**

The H.100 bus specifications make provisions for connecting to legacy computer telephony busses such as the SCbus and the MVIP bus. When interoperating with the SCbus or MVIP bus, the first 16 streams are used for connecting to the legacy bus, and these streams may operate at less than the 8.192 MHz. bus rate. This means that there will be fewer than

*The H.100 T1/Primary Rate ISDN Board*

128 timeslots per stream.

### **10.2.1 SCbus Compatibility**

The SCbus is a 16 stream bus. Each stream on the bus normally operates at 4.096 MHz. and has 64 timeslots per stream for a total of 1024 timeslots. However, the SCbus may optionally run at either 2.048 MHz. with 32 timeslots per stream for a total of 512 timeslots or 8.192 MHz. with 128 timeslots per stream and a total of 2048 timeslots on the 16 streams. When inter-operating H.100 T1/Primary Rate ISDN board with the SCbus at either 2.048 or 4.096 MHz. the bit rate on the lower 16 H.100 streams must be set appropriately using the “SB” command.

The SCbus uses a 26 conductor ribbon cable. Because of this, an adapter must be used between the P2 connector on one of the H.100 boards and the SCbus.

### **10.2.2 SCbus Timeslot Assignment**

Typically, drivers and libraries conforming to the SCbus specification use a scheme called “timeslot assignment” to insure that no two devices are transmitting on the same timeslot. Not only will having two transmitters on the same timeslot degrade audio signals, but they also may damage some SCbus boards. To prevent this, each device or “port” is assigned a unique timeslot to transmit on during boot-up. For purposes of timeslot assignment, each B-channel can be considered a port.

Transmit timeslots are reserved on the H.100 T1/PRI Board using the “SX” command. The same command is used to reserve timeslots on XDS SCbus boards. The command takes the form **SXstt** where s is the stream and tt is the timeslot on that stream of the timeslot reserved for the first port on the board. As an example, if the first timeslot reserved for the board is 100 in decimal, then the message **SX124** would be sent (timeslot 100 corresponds to stream 1, timeslot 24h).

*The H.100 T1/Primary Rate ISDN Board*

When timeslots have been assigned, the actual output timeslot value is no longer used in commands that control outputs to the CT bus. Instead, the B-channel or port number is added to the base timeslot to determine the output timeslot used. For example, if the base timeslot is 100 decimal, B channel 21h would transmit on timeslot 133 decimal or 0205 in the sstt notation.

Normally, the timeslot assignment process is carried out as part of the initialization and loading of the driver. A configuration file is used to specify the number of timeslots to be reserved for the board. The function `xds_xmt_timeslot` is used for finding the transmit timeslot of a port. To aid this process, information on the number of reserved timeslots and the base timeslot is presented in the dual-ported memory. This information is available at the following locations:

1F10h	total number of timeslots for the board
1F12h	1st timeslot assigned to the board

The command **QXstt** can also be used to inquire as to which timeslot is reserved for a port stt. The reply takes the form **QXsttabb** where a is the stream and bb is the timeslot on the stream that is reserved for stt. If no timeslots have been reserved on the board, the response will take the form **QXsttZ**.

### 10.2.3 MVIP Compatibility

The MVIP-90 bus has 16 streams with 32 timeslots each. The streams run with a bit rate of 2.048 MHZ. When the H.100 T1/PRI Board is inter-operating with the MVIP bus, the bus rate on the lower 16 streams must be set using the “SB” command.

The MVIP-90 bus uses a 40 pin ribbon cable. To connect the H.100 T1/PRI Board to the MVIP-90 bus an adapter must be used. Only one adapter is allowed per system. The MVIP rules for termination must also be followed (see Section 4.2).

*The H.100 T1/Primary Rate ISDN Board*

Timeslots on the MVIP bus are normally paired, that is timeslot x on DSoy is paired with timeslot x on DSiy. One timeslot of a pair must be defined as an input and the other as an output. With most MVIP boards, an attempt to use both timeslots of a pair as inputs or as outputs will result in a conflict. The table in Section 6.1.1 gives the association between H.100 stream numbers and the DSi and DSo streams.

MVIP-90 applications normally assign timeslots dynamically. Connections that are not enabled are tri-stated.

The H-MVIP bus has 24 streams. The H-MVIP specification has several modes. One mode is compatible with the MVIP-90 specification, that is the lower 16 streams run at a 2.048 MHz. rate while streams 16-23 run at 8.192 MHz. This mode can be treated as the MVIP-90 case. In another mode, all 24 streams run at the same 8.192 MHz. rate as the H.100 bus. In this case, only the physical cabling between the H-MVIP bus and the H.100 bus and clock issues need to be addressed.

### **10.2.4 MVIP Compatibility Commands**

Several commands exist for compatibility with the MVIP-95 driver specification. This specification uses the concept of a “terminus” to define an input or output timeslot. The terminus argument consists of three parts, a bus, a stream within the bus, and a timeslot on that stream. In MVIP compatibility messages, a terminus is represented by a five character string. The first character indicates the bus. Valid bus selections are “H” for the H.100 CT bus, and “L” for the local bus. The first 8 local streams carry the audio for the B-channels. The next 8 local streams carry the audio for the voice resource channels. The remaining streams are used for the DSP resources. On boards with conferencing streams 10-13 are used by the DTMF generators and detectors and streams 14-17 are used for conferencing, while on boards without conferencing all eight streams 10-17 are used by the DTMF resources. Stream 19 carries the call progress tones.

<b>Local Stream</b>	<b>Input</b>	<b>Output</b>
00h	channels 00-17h	channels 00-17h
01h	channels 20-37h	channels 20-37h
02h	channels 40-57h	channels 40-57h
03h	channels 60-77h	channels 60-77h
04h	channels 80-97h	channels 80-97h
05h	channels A0-B7h	channels A0-B7h
06h	channels C0-D7h	channels C0-D7h
07h	channels E0-F7h	channels E0-F7h
08h	voice channels 00-1Fh	voice channels 00-1Fh
09h	voice channels 20-3Fh	voice channels 20-3Fh
0Ah	voice channels 40-5Fh	voice channels 40-5Fh
0Bh	voice channels 60-7Fh	voice channels 60-7Fh
0Ch	voice channels 80-9Fh	voice channels 80-9Fh
0Dh	voice channels A0-BFh	voice channels A0-BFh
0Eh	voice channels C0-DFh	voice channels C0-DFh
0Fh	voice channels E0-FFh	voice channels E0-FFh
10h	DTMF detectors 00-1Fh	DTMF generators 00-1Fh
11h	DTMF detectors 20-3Fh	DTMF generators 20-3Fh
12h	DTMF detectors 40-5Fh	DTMF generators 40-5Fh
13h	DTMF detectors 60-7Fh	DTMF generators 60-7Fh
14h	DTMF detectors 80-9Fh	DTMF generators 80-9Fh
15h	DTMF detectors A0-BFh	DTMF generators A0-BFh
16h	DTMF detectors C0-DFh	DTMF generators C0-DFh
17h	DTMF detectors E0-FFh	DTMF generators E0-FFh
18h	unused	call progress generators 00-1Fh

*The H.100 T1/Primary Rate ISDN Board*

Note that the arrangement of local streams is different than earlier versions of the board. Also, there are 128 more DTMF generators and more call progress generators available.

For compatibility with applications written for the previous generation board, JW1-4 may be installed. When this is done, the local stream assignments will match the older board. If this is done, only 128 DTMF generators may be accessed and conferencing resources may not be accessed using the “MO” and “MK” commands. Streams 10-17 that were routed to the mezzanine board are now routed to the voice resources on boards that are so equipped.

In the MVIP compatibility mode, connections are controlled using the Set Output command **MO**. This command takes the form **MObssttm**, where “bsstt” is the output terminus being controlled, and m is the mode. Valid modes are “D” for disable, “E” for enable, and “P” for pattern output. In the enabled mode, the input terminus follows the mode character, and in the pattern mode, a two digit hexadecimal number representing the value of the byte to be output follows the mode. As an example, the message “MOH0123EL0000” would enable a connection from the first channel of the first span to the H.100 timeslot 23h, stream 1. Connections can be made between the H.100 bus and the local bus, or between timeslots on the local bus within the same block.

In the MVIP compatibility mode, additional commands are needed to control the DSP resources. The DTMF detectors are controlled with a command of the form **MDhhm** where hh is the detector number or handle and m is the mode, either “D” to disable or “E” to enable the detector. When a detector is enabled, detected digits are reported in a message of the form **STXhhd** where hh is the detector number and d is the digit. To generate DTMF digits the command takes the form **MGhh(string)** where hh is the generator and “string” is the string of DTMF digits. This string may include pauses. Completion of the string is indicated by a message of the form **SEXhh** where hh is the generator number. To access the DSP resources, a “MO” Set Output command

must be issued to connect the resource to the desired B-channel or H.100 timeslot as well as issuing the resource control command. It is the responsibility of the application to manage the DSP resource in the MVIP compatibility mode.

As an example of detecting digits in the MVIP compatibility mode:

<u>commands</u>	<u>responses</u>	<u>description</u>
MOL0000EL1800		connect B-channel 00 to dial tone
MOL1001EL0000		connect DTMF detector 1 to B-channel 00
MD01E		enable DTMF detector 1
	STX011	digit 1 detected
MOL0000D		disable dial-tone
	STX012	digit 2 detected
	STX013	digit 3 detected
MOL1001D		disable input to detector 1
MD01D		disable DTMF detector 1

A query command **QObsstt** is also available to query the state of the output terminus "bsstt". This command corresponds to the Query\_Output command in the MVIP-95 specification. The response takes the form **QObssttm(bsstt)** where "bsstt" is the output terminus, "m" is the mode, and if the mode is enable, the second "bsstt" is the input terminus.



## 10.3 Configuring the Board

Several steps are necessary when configuring the board. The most important item is to select the clock mode. Only one board in the system can generate the Master Clock. If available, a T1 or Primary Rate ISDN span connected to the public switched network should serve as the master. This span may be on the H.100 T1/PRI Board or another board in the system. If no such span exists in the system, and one or more Basic Rate ISDN port configured as a terminal equipment interface that is connected to the public switched telephone network is available on another board, then one of these ports should be the source of the master clock. If neither of these conditions exist, then the H.100 T1/PRI board or another board in the system may generate the clock internally. (See Sec. 6.2 for clock mode details)

The clocks must be configured before any switching can take place. It should be noted that boards on the SCbus may fail to operate unless a master clock is provided, therefore, when inter-operating with SCbus boards, it may be necessary to configure the clock on the H.100 T1/PRI Board before trying to download software or issue commands to other boards on the SCbus. It is also important that the board providing the master clock be configured **last** after all other boards have had their clock modes set. This is required for the proper operation of the SC2000 chip used to interface to the SCbus on most SCSA boards.

The framing mode of each interface that is to be used must also be programmed. This is necessary to insure the correct clocking as well as the correct signaling format. (See Sec. 7.2 for details)

Several other steps may be necessary to configure individual ports. A port can be configured as unused or undefined, a network termination, or as terminal equipment. The “ST” command is used to configure the ports. It consists of characters, one for each port, representing the port type. An “N” defines a port as a “network termination” or NT (this is the CO side), a “C” as “customer interface” or CI (this is the CPE side),

*The H.100 T1/Primary Rate ISDN Board*

and a “U” as undefined or unused. As an example, if the first four ports are to operate as CIs, the next two as NTs, and the rest as unused ports, then the “ST” message to be sent would be:

**STCCCCNNUU**

If the span is to operate as a PRI interface, it will also be necessary to set the protocol level of the span. This is done with an “SL” message. Like the “ST” message, the “SL” message has four or eight characters that act as arguments for each of the ports. For the “SL” message the choices are “2” if only “LC” messages are to be used for call control, “3”, “A”, “D”, “N”, or “S” if “D” messages are to be used. The “A” is used for AT&T Custom switches, “D” for DMS-100 NI-1 switches, “N” for National ISDN-2 switches, “S” for Siemens CoreNet switches, and “T” for switches conforming to Telcordia Generic Requirements. The default is “2”.

If robbed-bit signaling is to be used on T1 spans, it is also necessary to program the channel type. This is done with the “SP” Set Protocol command. This takes the form **SPxxtd(pd)** where xx is the channel number, t is the type, d specifies the direction, and p and d are optional parameters used to specify an address protocol. Channel types are “E” for E&M, “G” for ground start, “L” for loop start, and “N” for none. The direction parameter is used to specify whether the channel is the “O” or FXO end or the “S” or FXS end. If the port is an E&M port, an optional address protocol may be specified for the collection of digits for incoming calls. The p parameter specifies whether the “I” or immediate start or “W” or wink start protocol is to be used and the d parameter is the number of digits that are expected. Details of the operation of the address protocol may be found in section 10.4.6.

The span type, framing, and protocol level information can be stored in the EEPROM to be loaded automatically on power up or a restart. See Section 4.3 for details.

## 10.4 Using the “C” Commands

The B-channels on the H.100 T1/Primary Rate ISDN Board can be set to provide full-duplex connections as well as half-duplex connections in either direction. They can also be used to send DTMF and call progress tones as well as provide DTMF detection.

Changes in the B-channel state are reported in state change messages. These consist of the letter “S” followed by a letter indicating the change, the B-channel number, and an optional information character. An example is the message **SI00** which indicates that B-channel 00 has returned to the idle state.

### 10.4.1 Making a Connection

A two-way connection can be made with a port using the Connect Command “**CC**”. This command takes the form **CCxxssttaabb**, where xx is the B-channel number, sstt is the destination stream and timeslot and aabb is the source stream and timeslot to be connected to. As an example, the message **CC0001020304** would connect B-channel 00 with timeslot 2, stream 1 being the transmit timeslot and timeslot 4, stream 3 the receive timeslot. Note that if a transmit timeslot had been reserved for the port using the “**SX**” command or if this had been done by a timeslot assignment program associated with a driver, the transmit timeslot portion of this command will be ignored and the reserved timeslot will be used. In addition to two-way connections, one-way connections are also possible in either direction. The Audit command “**CA**” is used to create a connection from a timeslot to a port. This command takes the form **CAxxsstt**, where xx is the B-channel number, ss is the stream, and tt the timeslot of the timeslot being audited. As an example, **CA000201** would direct timeslot 1 of stream 2 to B-channel 00.

The Transmit command “**CX**” is used to setup a B-channel to transmit

towards the H.100 bus. This command takes the form **CXxxsstt** where *xx* is the B-channel number, *ss* is the stream and *tt* is the destination timeslot. As an example of using a B-channel as an input, if B-channel 01 is to transmit on timeslot 5 of stream 6 the command **CX010605** would be sent.

A connection will be broken by a Disconnect command. The Disconnect command “**CDxx**” will disable the B-channel output and any H.100 bus connection. An idle state change message “**SIxx**” will signal that the B-channel has returned to the idle state.

A Hold or other B-channel command, such as playing a call progress tone, will also break a connection. The Hold command “**CHxx**” will disable audio to and from the B-channel. If the B-channel was connected to the H.100 bus, a silence pattern will be output to the H.100 bus. This pattern can be disabled by a “**CDxx**” command. It will also be cleared if a connect or disconnect command is issued for the B-channel.

### 10.4.2 Call Progress Tones

The on-board DSP can be used to play any of the standard Call Progress tones to a B-channel. The tones supported are the North American precise tones dial tone, busy tone, reorder tone, and audible ringback as well as the ETSI tones. Silence and a calibration tone of 1004 Hz. can also be played. The Call Progress command “**CP**” is used to play tones. This message is of the form **CPxxt**, where *xx* is the B-channel and *t* is the code of the tone. For example, the command **CP002** will play busy tone to B-channel 00. The tone will stop playing if another command is issued for the B-channel such as a connect.

### 10.4.3 Sending DTMF Tones

A string of DTMF tones can be sent using the “CT” command. Any of the 16 DTMF tones can be sent. Pauses may be embedded that are one or ten digit times (.2 or 2 sec.) Single tones of 697 and 941 Hz. are also possible. The tone string length can be up to 27 characters long. The command consists of CT followed by the B-channel number and the string of tones. As an example, the command **CT039P7654321** would send the digit “9”, pause two seconds, and then send the digits “7654321”.

On receiving the command, the board will respond with an “SL” state change message to indicate that the B-channel is listening to a DTMF generator. When the tone string has finished, the board will send an “SE” message to indicate the string is ended and place the B-channel in the hold state.

### 10.4.4 Sending MF Tones

A string of MF tones may be sent in a manner similar to DTMF tones using a command of the form **CKxxdd...d** where xx is the B-channel and dd...d is the digit string. In addition to the digits 0-9, MF also includes five special tones that are used to indicate the beginning and end of a sequence of digits. These tones and the character used to represent them in the dial string are:

KP	K	the start tone pair
STP	P	an ending tone
ST	S	a different ending tone
ST2P	*	a special code
ST3P	#	a special code

A typical address sequence consists of a KP, the digits and either an ST or STP. For example to send the address 608-555-1234, the command

would be **CK00K6085551234S**.

### **10.4.5 Detecting DTMF Tones**

The on-board DSP can be used to detect DTMF tones on a B-channel. The Listen for DTMF command takes the form **CLxxo**, where xx is the B-channel number and o is an option. If no option is given, dial tone will automatically be played to the B-channel until the first tone is detected. If the option is “Q”, then no dial tone will be played. If the option is “M”, then detection will take place without interrupting existing connections. The “F” option will disable detection that is in progress. As an example, the message **CL00** will enable DTMF detection on B-channel 00 and play dial tone.

As each tone is detected, the board will send a Tone detected state change message of the form **STxxd** where xx is the B-channel number and d is the tone. As an example, the message **ST005** indicates that the digit 5 has been detected on B-channel 00. All sixteen DTMF tones can be detected. Detection will remain active until a command is issued for the B-channel.

### **10.4.6 Robbed Bit Signaling**

On T1 spans, the channels can be set to emulate various analog line types using robbed bit signaling, which uses the least significant bit of every eighth frame to convey information. The line types that can be emulated include E&M, ground start, and loop start. The information that can be sent includes hook status, ground signaling and ringing. The channel emulation type is set using the “**SP**” command.

When a “C” command is issued for a channel set for robbed bit signaling, the robbed bit pattern appropriate for the line state is placed on the channel. For example, if a “**CH**” command is issued for an FXS loop start channel, the off-hook pattern is placed on the channel. If a

“CD” command is issued, the on-hook pattern is placed on the channel.

Changes in the robbed bit pattern sent from the other end are reported with the appropriate state change messages. An off-hook will be reported with an “SF” message and an on-hook will be reported with an “SN” message. If appropriate, the pattern sent will change, i.e. if a change to an on-hook pattern is detected by a channel that is off-hook, that channel will return an on-hook pattern and change state to the idle pattern state.

In addition to hook status, ringing can be emulated on loop and ground start channels. The FXO side can generate ringing towards an on-hook channel and the FXS side can detect ringing. Ringing is generated using a command of the form **CRxx** where xx is the channel number. This will initiate ringing. The board will handle the ring cadence, and stop ringing if the far end goes off-hook. Ringing can be stopped by sending a message of the form **CDxx**. The ring cadence is the standard 2 seconds on, four seconds off. If ringing is detected, it will be reported with a message of the form **SRxx** where xx is the channel number. One message will be generated for each ring cycle. If ringing should stop before the call is answered, a message of the form **SIxx** will be sent.

In the following example, ringing is detected, answered and then the call is disconnected. This example is seen from the perspective of a loop start FXS channel.

<u>command</u>	<u>response</u>	<u>description.</u>
	<b>SR00</b>	1st ring detected
	<b>SR00</b>	2nd ring detected
<b>CH00</b>		call answered by placing it on hold
<b>CC0002000100</b>		connect to H.100 bus
	<b>SC00</b>	connect acknowledge
<b>CD00</b>		call disconnected
	<b>SI00</b>	channel returns to the idle state

The same call from the perspective of an FXO channel would be:

<u>command</u>	<u>response</u>	<u>description</u>
<b>CR00</b>		start ringing
	<b>SF00</b>	off-hook detected
<b>CC0001000200</b>		connect to H.100 bus
	<b>SC00</b>	connect acknowledge
	<b>SN00</b>	on-hook detected
<b>CD00</b>		disconnect call
	<b>SI00</b>	channel returns to idle

### 10.4.7 Address Signaling Protocols

T1 channels are sometimes used to emulate E & M interfaces. In these applications, a single channel may be used to direct calls to more than one destination or for more than one phone number. In these applications which are sometimes called Direct Inward Dialing or DID, one or more digits of information is sent by the originating interface to indicate the ultimate destination. These digits are referred to as “address” digits and are sent as DTMF or MF signals on the channel. There are several standard protocols that define the timing of the digits, i.e. the time from the off-hook signal until the digits can begin and the time between digits, and the form of the acknowledgement or ready signal sent by the destination interface.

The acknowledgement, if required, takes the form of a momentary off-hook signal or “wink”. If no acknowledgment is required before the digits are sent, it is referred to as “immediate start”. If a “wink” is required, it is referred to as “wink start.” The “wink” is typically 200 msec. long.

To set the protocol for a channel a command of the form “**SPxxt<sub>dpn</sub>**” is used where xx is the channel number, t is set to E for the E&M port type, d may be either “O” for FXO or “S” for FXS, p indicates the protocol for incoming calls, n is the number of digits, The protocol is

*The H.100 T1/Primary Rate ISDN Board*



indicated by an “I” for immediate start or a “W” for wink start. The number of digits indicated by n is typically 3 or 4, but can be any number from 1 to 15. For MF signaling, this parameter should be set to “M.” As an example, to set channel 0 for 4 DTMF digits, wink start, the command would be **SP00ESW4**.

When a protocol is set for a channel the on board processor takes care of all of the timing and acknowledgements. When the digits are received, they will be indicated in a message of the form **SDxxdd...d** where xx is the port number and dd...d are the collected digits. The message will not be sent until all of the digits are received. If an off-hook is detected, but the address digits do not follow within 10 seconds, an error of the form **ETxx** will be reported.

When making an outgoing call, the “CT” command can be used to send the address digits (sec. 9.4.3). The board will wait until the acknowledgement is received before sending the digits. If a wink is expected, and not received within 10 seconds, an error message of the form **EWxx** will be reported.

#### **10.4.8 Recording Alert Tone**

Many jurisdictions require that a periodic tone be played when a telephone conversation is being recorded to alert the parties of this fact. The H.100 T1/PRI Board provides a mechanism which will add such a tone to the audio stream from a B-channel towards the H.100 bus so that a party on the far end of the connection will hear the tone. The tone consists of a .5 sec 1400 Hz. tone repeated every 15 seconds.

To add a tone to a connection a “B” is appended to either the “CC” or “CX” messages. For example, the command **CC0301000200B** would introduce a alerting tone on the audio stream between B-channel 03 and H.100 timeslot 100. The alerting tone will be disabled by any subsequent command or by a change of the switching state.

If connections are made using the “MO” command, the alerting tone may be inserted using a command of the form **MBxxE** where xx is a detector/generator pair number. A command of the form **MBxxD** will disable the tone. To use this feature, a connection must be made from the B-channel to a DTMF detector and from the corresponding DTMF generator to the H.100 bus timeslot.

If an alerting tone other than the default is desired, the tone may be customized using the “SI” command for tone generator 20h (see Appendix C for details). For example, the command **SI2002581000003F05190000** would create a 600 Hz. tone of .5 sec. duration and with 2.5 sec. between tones. Note that for generator 20h the values for the on and off duration is in increments of 100 msec. not 50 msec. as it is for all other generators.

### 10.4.9 An Example of an Inbound Call

The following is an example of the steps involved in an inbound call to a CI PRI span. Both the “C” commands and state change messages will be shown. The “D” messages will also be shown to show the relative timing. The dialed digits will be detected, ringback played, and finally a connection made and broken when the phone hangs up.

<u>commands</u>	<u>responses</u>	<u>description</u>
	DS06SN5551212	SETUP message
CP063		play audible ringback
DA06IR		ALERTing message
	SL06	tone played to B-channel
CC0601020304		connect the B-channel
DC06		CONNect message
	SC00	connection confirmation
	DD0610	DISConnect message
	DR060F	RELease COMplete
CD06		disconnect the B-channel
	SI06	B-channel returned to idle state

### 10.4.10 An Example of an Outbound Call

This example shows the steps involved in an outbound call on a CI span. A SETUP message will be sent, then the port will be connected, and finally disconnected when a far end disconnect is detected.

<u>commands</u>	<u>responses</u>	<u>description</u>
DS06S7654321		SETUP message
	DA06IR0001	ALERTing message, inband tone
CX060102		transmit audio to H.100 bus
	SX06	transmit confirmation
	DC0601	CONNect message
CC0601020304		connect the B-channel
	SC06	connection confirmation
DD0610F		DISConnect message
	DR06R	RELease message
CD06		disconnect the port
	SI06	B-channel in idle state

### 10.4.11 An Example of Detecting DTMF Digits

Once a connection has been established, there may be a need to detect DTMF digits. An example is when the call is not an end to end ISDN call, and there is a need for the far end to signal the near end. In this example, DTMF detection is turned on, a string of digits is detected and then the connection is reestablished. Note that no “D” commands are issued because the call state has not changed.

<u>commands</u>	<u>responses</u>	<u>description</u>
CL06		listen for DTMF with dial tone
	SX06	DTMF detection enabled
	ST061	the digit “1” is detected, tone removed
	ST062	the digit “2” is detected
	ST063	the digit “3” is detected

CC0601020304

SC06

reestablishment of connection

connection confirmation

## 10.5 Voice Resources

The H.100 T1/PRI Board is equipped with 128 or 256 channels of voice record and 128 or 256 channels of voice playback capability. The number is dependent on the population level of the board. The operation of these channels is beyond the scope of this document. For further information consult the appropriate driver and library manuals for the operating system you are using.

## 10.6 Analog Audio Ports

The H.100 T1/PRI Board may be equipped with two analog audio ports. These ports can be used to connect an external analog audio source to a timeslot on the H.100 bus. The command to do this takes the form **AE0xsstt** where x is the audio port (either 0 or 1) and sstt is the H.100 stream and timeslot. The connection may be disabled with a command of the form **AD0x** where x is the audio port. The level of audio may be adjusted in a range of +3 to -18 dB in 3 dB steps using a command of the form **AG0xg** where x is the audio port and g is the gain parameter, with 0 representing a setting of +3 dB and 7 a setting of -18 dB. Bit 7 of the Configuration Flag Byte (at an offset of 1F08h in DPRAM) will be set to 1 on boards with analog audio ports.

## 10.7 Conferencing

Some configurations of the 4 span H.100 T1/PRI Board may be equipped with 128 ports of conferencing capability. This option is not available on boards configured for 8 T1 spans. Boards with conferencing will have bit 0 of the Configuration Flag Byte (offset 1F08h) in DPRAM set to 1.

*The H.100 T1/Primary Rate ISDN Board*

The conferencing feature is similar to that available on the H.100 MC3 and Conferencing Boards. The 128 ports may be configured as up to 63 n-way conferences with the only limitation being that the total of all parties in all conferences may not exceed 128.

Each conference is identified with a handle. The value of the handle ranges from 1-63 (01-3Fh). The handle is used to identify which conference a command applies to. Handles are assigned by the host application. Each conference input/output pair is defined by a Conference Control Address, or CCA. These are numbered 0-127 (00-7Fh). With the exception of the “MK” command, the CCAs are normally hidden from the application.

The commands to control a conference begin with “K”. There are commands to add or delete a party from a conference, to add an input only to a conference, to add an output from a conference, to dissolve a conference and to enable or disable DTMF detection and clamping. There is also an “MK” command which is used in conjunction with the “MO” command to control conferencing when the application uses the MVIP switching model.

The input and output levels of a conference party can be controlled with the attenuation parameter “a” according to the following table. The amount of attenuation required to maintain the desired level of audio quality depends on a number of factors such as the number of parties in a conference and the type and quality of the originating inputs.

A noise threshold can also be set for each conference input. When enabled, signals below a threshold will be suppressed. If the noise threshold parameter “n” is set to 0 the function is disabled. Values of 1 to 3 will raise the threshold to progressively higher levels. It should be noted that a high threshold level may introduce distortion and so should be used with discretion.

value of a	input attenuation	output attenuation
0	0 dB	0 dB
1	0 dB	3 dB
2	3 dB	0 dB
3	3 dB	3 dB
4	6 dB	0 dB
5	6 dB	3 dB
6	9 dB	0 dB
7	9 dB	3 dB

A periodic tone may be played to a conference. This may be useful to indicate that a conference is being recorded. This is controlled by the tone mode parameter “t”. If this is set to 0, no tone is played. If it is set to 1, a beep is played every 15 seconds. Values of 2 and 3 can be used to play customized tones. These customized tones can be programmed using the “SI” command for tones 11h and 12h.

Normally, a conference is configured so that the output of a conference includes the sum of the inputs of all parties of a conference except for the input that corresponds to the particular output. That is if a conference has three parties, A, B, and C, A will hear the B+C, B will hear A+C and C will hear A+B. This helps to prevent echo and feedback. However, under some circumstances it may be desirable to include the sum of all inputs. This is done by setting the output mode parameter o to 1.

### **10.7.1 Controlling a Conference**

A party is added to a conference with a command of the form **KAhhooooiiaanto** where hh is the conference handle, oooo is the output timeslot, and iiii is the input timeslot. The parameters a, n, t, and o are optional and may be used to control the attenuation, noise threshold, tone mode and output mode.

An input can be added to a conference with a command of the form **KIhhiiiiant** where hh is the conference handle and iiii is the input timeslot. The optional parameters a, n, and t may be used to control the attenuation, noise threshold and tone mode. A party may be an input to more than one conference.

An output from a conference with no corresponding input can be created using the monitor command. This command takes the form **KMhhooooan** where hh is the conference handle, oooo is the output timeslot and a and n are optional parameters to control the attenuation and noise threshold.

A conference can be dissolved using a command of the form **KUhh** where hh is the handle. This will remove all parties from a conference including input only and monitor output parties. A single party may be removed from a conference with a command of the form **KDoooo** where oooo is the output timeslot. To remove an input only party from a conference, a command of the form **KXhhiiii** is used where hh is the handle and iiii is the input timeslot. Note that as the handle is specified, this removes the party from only one conference and leaves other input connections unchanged.

It is possible to change the attenuation and noise threshold parameters for all parties of a conference. This is done with a command of the form **KGhhan** where hh is the handle and a and n are the attenuation and noise threshold parameters. Note that this command does not affect the tone and output modes of any party.

### 10.7.2 DTMF Detection and Clamping

As DTMF tones are often used to control conferences, DTMF detection is included as part of the conferencing. In addition, to minimize the control tones to other parties, these tones may be clamped, that is silenced once detected for the duration of the tone. This is done with a command of the form **KEiiii(cc)** where *iiii* is the input timeslot and *cc* is an optional clamping duration. This value is given in steps of .02 seconds. A command of the form **KEiiiiD** will disable the tone detection and clamping. Detected digits are reported in a message of the form **STiiiiid** where *iiii* is the input timeslot and *d* is the detected digit.

### 10.7.3 Conferencing Using the MK Command

An alternative method of establishing conferences is available using the “MK” command and the MVIP compatible switching model. In this method, it is the responsibility of the application to control the allocation of CCAs. Switching to and from the CCAs is accomplished using the “MO” command. The CCA’s are found on local streams 14-17h in place of DTMF detectors and generators 80-FFh. Control of the CCAs is done through a command of the form **MKhhccmanto** where *hh* is the handle, *cc* is the CCA, and *m* is the mode which can be either “E” to enable the CCA or “D” to disable the CCA. The optional *a*, *n*, *t*, and *o* parameters can be used to set the attenuation, noise threshold, tone mode, and output mode. The “MK” and “K” commands should not be used at the same time as there may be a conflict.

If the “MK” command is used to establish a conference, the DTMF detection and clamping can be controlled using a command of the form **MDhhmcc** where *hh* is detector number, *m* is the mode, either *D* to disable or *E* to enable detection, and *cc* is the optional clamping time parameter. Note that detectors 80-FFh correspond to CCA’s 00-7Fh. A detected digit will be reported with a message of the form **STXhhdd** where *hh* is the detector number and *d* is the detected digit.



# 11.0 Diagnostics & Error Messages

## 11.1 Diagnostic Commands

Several diagnostic commands are available:

- VA** Used to request the checksum of the firmware in the alternate segment of the board. This is returned in a message of the form `VAxxxx` where `xxxx` is the checksum of the firmware in the alternate segment of ROM.
- VC** Used to request the version of the firmware on the board. A message of the form `VCxxxxyyyyPTN8` is returned, where `xxxx` is the checksum of the firmware stored in the main segment of ROM, `yyyy` is a four-digit version number, `PTN8` indicates the board type and version (PCI T1/PRI North American firmware). For four span boards the board type is `PTN4`. This message takes the same form with all Infinity Series boards, and can be used to determine the configuration of the system.
- VD** Used to request the version of the DSP software. This is returned in a message of the form `VDxxxx`, where `xxxx` is the version number. All DSP's on the board use the same software version.
- QA** Queries the alarm status of the T1/PRI interfaces. The response takes the form `QAaaaaaaaa` where each `a` represents the alarm condition of one of the spans starting with span 0. The values of `a` are "G", "Y", "B", or "R" for

green (no alarm), yellow alarm, blue alarm, and red alarm respectively. An inactive span will be indicated by an “N”.

**QHbrrrr** Queries the T8110 switching chip for bus *b*, stream and timeslot *sstt*. The bus value *b* can be either “H” for the H.100 bus or “L” for the local bus. The results are returned in a message of the form *QHbsstt0c0ttt00dd* where *c* contains the control bits, *ttt* is the 12 bit tag value and *dd* is the contents of the data memory. This command refers to the details of the internal switching circuitry, and is ordinarily of limited use to the application.

**QRxx** Queries the call reference value and call state of the call currently assigned to B-channel *xx*. The results are returned as *QRxxrrrrss* where *xx* is the B-channel, *rrrr* is the call reference value, and *ss* is the call state. This command is only valid for Primary Rate ISDN interfaces.

**QXxxx** Queries the transmit timeslot reserved for B-channel *xxx*. This command is only valid if the board is operating in SCbus mode. The reply takes the form *QXxxxstt* where *stt* is the stream and timeslot for the B-channel. If no timeslot is assigned, the reply will be *QSxxxZ*. This command is common to all Infinity Series boards operating in the SCbus mode and all XDS SCSA boards. While the B-channel number has three digits to allow for boards with up to 1024 channels, only the values 000 through 33F are valid for the H.100 T1/Primary Rate ISDN Board.

## 11.2 Error & Event Messages

The board will detect a number of error conditions and respond with appropriate error messages. These messages are:

*The H.100 T1/Primary Rate ISDN Board*

- ABdd** A Blue Alarm event for span dd. An all-ones Alarm Indication Signal (AIS) has been detected.
- AGdd** A Green Alarm event for span dd. All alarm conditions have been cleared.
- ARdd** A Red Alarm event for span dd. A Loss of Signal (LOS) condition has been detected.
- AYdd** A Yellow Alarm event for span dd. A Remote Alarm Indication (RAI) has been detected.
- ECxx** A clock error bit event xx has occurred. The value xx is a hexadecimal number where the bits are as below (a value of 1 is an error). The Master PLL error bit will always be 1 unless the clock is derived from a span or CT\_NETREF.

<u>bit</u>	<u>description</u>
0	CT bus clock A
1	CT bus clock B
2	SCbus clocks
3	MVIP bus clocks
4	Master PLL error
5	Always set for compatibility
6	Failsafe bit

- EKhhxx** An error has occurred while attempting to make a conference using handle hh. If xx equals 01, a handle outside the range of 01-3Fh was used. If xx equals 02, all conference facilities available for that handle are in use. If xx equals 03, the timeslot is not an input to the conference with a handle hh.
- ELdde** An error has been detected in Layer 2 for span dd. The specific error is indicated by the value of e:

- 1: Timer T1 has expired indicating a persistent Layer 1 error condition on span dd.
- 2: Timer T2 has expired indicating a persistent Layer1 error condition recovery on span dd.
- 5: A Layer 2 N(R) state variable recovery has taken place

**EMdde** A Layer 2 protocol error has occurred on span dd. The error is indicated by the value of e. This corresponds to the Management Data Layer Indication in parentheses as defined by Q.921:

- 1: MDL error (A) unsolicited supervisory response (F=1)
- 2: MDL error (B) or (E) unsolicited DM response
- 3: MDL error (C) or (D) unsolicited UA response
- 4: MDL error (F) peer initiated reestablishment (SABME)
- 5: MDL error (G) or (H) unsuccessful retransmission of SABME or DISC
- 6: MDL error (I) unsuccessful retransmission of a status enquiry
- 7: MDL error (J) N(R) state variable error
- 8: MDL error (K) receipt of FRMR response

**ESddss** A Line Interface Unit status event has occurred on span dd with the event given by ss where the bit values are:

- 1: Transmit open circuit detected
- 2: Transmit current limit exceeded
- 3: Receive carrier loss detected
- 4: Jitter attenuator limit trip event

**ETxx** The expected number of DTMF address digits were not received on B-channel xx within 10 seconds on an incoming call using an address signaling protocol.

**ETxxe** A Layer 3 timer event has occurred on B-channel xx causing actions as defined in Q.931. The specific timer event is indicated by the value of e:

- 0: T303 has expired for the first time
- 1: T303 has expired for the second time
- 2: T310 has expired
- 3: T305 has expired
- 4: T308 has expired for the first time
- 5: T308 has expired for the second time
- 6: T316 has expired
- 7: T322 has expired

**EWxx** An expected wink signal was not received within 800 msec. for an outgoing call on B-channel xx.

**EXdds** An Elastic Store Buffer event has occurred on span dd with the event given by ss where the bit values are:

- 0: Receive elastic store slip event
- 1: Receive elastic store empty event
- 2: Receive elastic store full event
- 3: Transmit elastic store slip event
- 4: Transmit elastic store empty event
- 5: Transmit elastic store full event

**SM0** An EEPROM operation has failed. This indicates that either a read or write to the EEPROM was unsuccessful.

**SM1** An EEPROM operation successfully completed.

**U[*cmd*]** If the board does not recognize a command message, or if it does not have the appropriate number of arguments, the same message will be returned by the board preceded by a U to indicate an undefined message.

## 11.3 Diagnostic Tests

Several diagnostic tests can be run on the T1/PRI spans to check the operation of the span or the attached wiring. The commands to initiate these tests are:

**XCddc** This command sends the loopback code *c* out on span *dd* (D4 NT spans only). The code bit values are:

- 0: no code
- 1: request line loopback activation
- 2: request line loopback deactivation

**XCddcc** This command sends the bit-pattern message *cc* on span *dd*. This is valid only for ESF spans. (See Section 7.6.2 and *T1 403-1999* Table 4 for code values.)

**XFnn** Sets diagnostic control flags to *nn*. When set to 1, the diagnostic function is enabled. The functions for each bit are:

- 0: enables PRM message reporting
- 1: enables PRM message transmitting
- 2-7: reserved

**XLddm** This test puts span *dd* in the loopback mode specified by *m*:

- 0: no loopback
- 1: line loopback (CI) or remote loopback (NT)
- 2: framer loopback
- 3: payload loopback (all channels)

**XLdd4aabbcc** This test is used to generate a payload loopback for specific channels on span dd. The hexadecimal arguments aa, bb, and cc are used to specify which channels (0-7, 8-15, 16-23 respectively) are to be looped back. The lowest channel in each argument is the least significant bit.

The following test messages may be received during tests:

**XLddc** Loopback code c received on D4 span dd, code values are:

- 0: end of code
- 1: activate line loopback
- 2: deactivate line loopback

**XLddcc** Bit patterned ESF data link message cc detected on span dd. (See Section 7.6.2 and *T1 403-1999* Table 4 for code values.)

**XXdd(data)** A Performance Report Message has been received on span dd. The data will consist of four 4 digit hexadecimal numbers representing the values of error bits in the PRM message. (See Section 7.6.4 and *T1 403-1999* Fig. 6 for bit value meanings.)

this page intentionally left blank



## **Appendix A: Environmental Specifications**

The Infinity Series H.100 T1/Primary Rate ISDN Board meets the following environmental specifications:

TEMPERATURE EXTREMES:

Operating: 0°C (+32°F) to +50°C (+122°F).

Storage: -40°C (-40°F) to +70°C (+158°F).

AMBIENT HUMIDITY:

All boards will withstand ambient relative humidity from 0% to 95% non-condensing in both operating and storage conditions.

MECHANICAL:

All Infinity Series H.100 boards conform to the PCI-SIG mechanical specifications for full-length PCI cards.

MTBF:

50,000 hours.

ELECTRICAL REQUIREMENTS:

+5 volts  $\pm 5\%$  @ 1.5 amps maximum.

-5 volts, +3 volts, and  $\pm 12$  volts are not required.

this page intentionally left blank

## Appendix B: Custom Tones

The DSP firmware provides a wide variety of dialed and information tones that should cover the majority of situations. However, for those cases when the standard set of tones is not sufficient, it is possible to modify these tones to provide custom capabilities.

The standard set of informational or call progress tones consist of the 12 tones listed in the table below. These tones can be played to a B-channel using the **CP** command. Any one of these twelve tones plus four additional tones may be modified by using the **SI** command to create a custom call progress tone. However, if this is done to one of the standard tones, the original tone will no longer be available. Note that the effects of this command are removed if an “RA” or “RD” command is issued to the board.

**Standard Call Progress Tones**

tone	description	tone	description
00h	dial tone	06h	ETSI dial tone
01h	reorder	07h	congestion tone
02h	busy tone	08h	ETSI busy tone
03h	ringback	09h	ETSI ringback
04h	digital milliwatt	0Ah	U.K. ringback
05h	silence	0Bh	Japanese ringback

The command to create a custom call progress tone takes the form

**SI***ttffffllffffl***nnffnnff**

where *tt* is the number of the tone, *ffff* are the first and second frequencies used to make the tone, *ll* is the level of the first and second frequencies in -dBm, and *nn* and *ff* are the on and off times of the tone in 50 msec. increments. To create complex cadences, a second on and off time may be specified. The frequency range is from 0000-0CFFh or 0 to 3327 Hz. The levels *ll* of the two frequency components is given in -dBm and has a range of 00-3Eh or 0 dBm to -62 dBm. A value of 3Fh will disable the frequency allowing a single frequency tone to be created. The on and off times are specified in 50 msec. increments and has a range of 00-FEh or 50-12700 msec. A duration of value of 0FFh will cause a continuous tone.

As an example, the command **SI04028A0A01A40A05050000** will change the digital milliwatt (tone 04) to a tone composed of the frequencies 650 and 420 Hz. each at -10 dBm with a cadence of 250 msec. on and 250 msec. off.

It is also possible to send a custom tone on a one time basis to a specific B-channel. The command to do this takes the form:

**CV***xxffffllffffl***nnffrr**

where *xx* is the port number, *ffff* and *ll* are the frequencies and levels of the first and second tone as in the commands above, *nn* and *ff* are the on and off durations of the tone pair, and *rr* is the number of times the tones repeat. the on and off times are specified in 50 msec. steps and have a range of 00-50h or 0 msec. to 4.0 sec. The number of repetitions ranges from 01-FFh. If a value of 00h is used, the digit will be played one time. The frequencies and levels have the same ranges as for the SI command.

As an example, the command **CV04028A0A01A40A050507** would

send a tone consisting of 650 and 420 Hz. at -10 dBm, on and off durations of 250 msec. and repeating a total of seven times to B-channel 4.

this page intentionally left blank