

**Dialogic System Software and SDK for
Windows NT
DNA 2 Release Catalog**

Copyright © 1998 Dialogic Corporation
05-0891-003



PRINTED ON RECYCLED PAPER

Copyright Notice

Copyright 1998 Dialogic Corporation. All Rights Reserved.

All contents of this document are subject to change without notice and do not represent a commitment on the part of Dialogic Corporation. Every effort is made to ensure the accuracy of this information. However, due to ongoing product improvements and revisions, Dialogic Corporation cannot guarantee the accuracy of this material, nor can it accept responsibility for errors or omissions. No warranties of any nature are extended by the information contained in these copyrighted materials. Use or implementation of any one of the concepts, applications, or ideas described in this document or on Web pages maintained by Dialogic may infringe on one or more patents or other intellectual property rights owned by third parties. Dialogic does not condone or encourage such infringement. Dialogic makes no warranty with respect to such infringement, nor does Dialogic waive any of its own intellectual property rights which may cover systems implementing one or more of the ideas contained herein. Procurement of appropriate intellectual property rights and licenses is solely the responsibility of the system implementor. The software referred to in this document is provided under a Software License Agreement. Refer to the Software License Agreement for complete details governing the use of the software.

Trademarks

All names, products, and services mentioned herein are the trademarks or registered trademarks of their respective organizations and are the sole property of their respective owners. Dialogic and the Dialogic logo are registered trademarks of Dialogic Corporation. The following are also trademarks of Dialogic Corporation:

Antares, Antares 2000/50, Antares 3000/50, Antares 6000/50, Board Locator Technology (BLT), D/21D, D/21E, D/21H, D/41D, D/41EPCI, D/41ESC, D/41H, D/80SC, D/160SC, D/160SC-LS, D/240SC, D/240SC-T1, D/240SC-2T1, D/240PCI-T1, D/300SC-E1, D/300SC-2E1, D/300SC-2E1-75, D/300SC-2E1-120, D/300PCI-E1, D/320SC, D/480SC-2T1, D/600SC-2E1, D/600SC-2E1-75, D/600SC-2E1-120, DCB/320SC, DCB/640SC, DCB/960SC, DIALOG/4, DTI/240SC, DTI/241SC, DTI/300SC, DTI/301SC, DTI/480SC, DTI/481SC, DTI/600SC, DTI/601SC, Dialogic Audio Conferencing, Dialogic Configuration Manager (DCM), Dialogic Native Architecture (DNA), DualSpan, GlobalCall, Global Tone Detection (GTD), Global Tone Generation (GTG), Global Dial Pulse Detection (GDPD), ICAPI, LSI/81SC, LSI/161SC, Modular Station Interface (MSI), MSI/SC, MSI/80SC, MSI/80SC-R, MSI/160SC, MSI/160-R, MSI/240SC, MSI/240SC-R, PerfectPitch, PerfectCall, PerfectLevel, PerfectVoice, PerfectDigit, ProLine/2V, Positive Answering Machine Detection (PAMD), Positive Voice Detection (PVD), Signal Computing System Architecture (SCSA), SCbus, The Dialogic Edge, Transaction Record, Universal Dialogic Diagnostics (UDD), VFX/40, VFX/40ESC, VFX/40ESCplus, VFX/40SC, VFX/PCI and WorldView. For a comprehensive trademark listing please visit our website at <http://www.dialogic.com/legal.htm>

CompactPCI is a registered trademark of the PCI Industrial Computer Manufacturers Group, CT Bus is a trademark of the Enterprise Computer Telephony Forum, Intel and Pentium are registered trademarks of the Intel Corporation, Vari-A-Bill is a registered trademark of AT&T, Borland C++ is a registered trademark of Borland International Inc., Visual C++, Windows, Windows95, and Windows NT are registered trademarks of Microsoft Corporation.

Part Number: 05-0891-003
Publication Date: March, 1998
Dialogic Corporation
1515 Route 10
Parsippany NJ 07054

Technical Support

Phone: 973-993-1443

Email: CustEng@dialogic.com

Fax: 973-993-8387

Visit our Technical Support website at <http://support.dialogic.com>

For **Sales Offices** and other contact information, visit our website at <http://www.dialogic.com>

Table of Contents

Release Overview	7
What's Hot.....	7
What's New	8
New Hardware Support.....	8
New Feature Descriptions	9
New Hardware Descriptions	10
New and Revised Documents.....	12
Master Feature List	13
Operating Environment	13
Hardware Environment	13
Software Environment.....	13
System	14
Voice (Record and Playback Voice Files)	14
FAX	15
Signaling	15
Tone Detection/Generation.....	15
Call Analysis	15
Network	16
GlobalCall	17
ISDN	17
Switching.....	19
Conferencing.....	20
Universal Dialogic Diagnostics (UDD)	21
TAPI 2.0/WAV support	22
Supported Hardware.....	23
Documents	27
Fixes	28
Release Issues	29
Hardware Restrictions	29
Compatibility Issues with DNA1 Production Release	29
System Startup.....	29
Miscellaneous	30
Caller ID	30
MSI/SC.....	30
Global DPD	30
ISDN.....	31
TAPI/WAVE 2.0.....	31

Dialogic System Software and SDK for Windows NT DNA 2 Release Catalog

Compatibility Issues	32
Known Problems	34
Restrictions and Limitations	36
SMP	36
PCI	36
Performance Counters	36
ISDN does not support the following features:	36
GlobalCall	36
VOICE	38
FAX	38
Installation/Configuration/Diagnostics	39
Programming Guidelines	41
Voice	41
TAPI/WAVE 2.0	42
MSI/SC	44
DCB/SC	44
DualSpan boards	45
Documentation Details	45
Documentation Additions	45
dx_fileerrno()	46
dx_sendevt()	47
dx_setdevuio()	49
Documentation Corrections	52
Call Analysis	53
dx_playvox() Code Example	53
dx_recf() Mode Argument	54
DX_XPB Data Structure	54
Speed and Volume Control	54

Release Overview

This catalog addresses the highlights of the Dialogic System Software and SDK (Software Development Kit) for Windows NT -- DNA Version 2 production release. This Intel release supports SCbus (the standard bus for communication within an Signal Computing System Architecture (SCSA) mode) and stand-alone boards.

This release is fully backward compatible with releases since System Release 4.25SC for Windows NT. The baseline for this release is Dialogic System Software and SDK for Windows NT-- DNA version 1. Some products and features (PBXpert32, Tone Set Files (TSF), Syntellect/Automated Attendant, Cadenced Tone, SCX160, TextTalk and D/42) that were available in past Streams-based releases are not available with this release although they may be mentioned in the base documentation set.

- Please visit our website at http://support.dialogic.com/releases/winnt/dna2/Release_Catalog.htm for up-to-date information on this release.
- Also visit our **Technical Support** website at <http://support.dialogic.com>.
- For **Sales Offices** and other contact information, visit our website at <http://www.dialogic.com>.

What's Hot

- GlobalCall
- Integrated Services Digital Network (ISDN)

What's New

- Full support for the Antares platform and SDK
- Peripheral Component Interconnect (PCI) support
- Dialogic Configuration Manager (DCM) API - Customized Installation and Configuration

New Hardware Support

PCI Boards:

- D/240PCI-T1
- D/300PCI-E1
- D/41EPCI
- VFX/PCI

High Density T-1/E-1 ISDN Network Interface:

- DTI/481SC
- DTI/601SC

Antares:

- Antares 2000/50
- Antares 3000/50
- Antares 6000/50

New Feature Descriptions

GlobalCall:

Integrated GlobalCall development software provides a common call control interface for applications, regardless of the signaling protocol needed to connect to the local telephone network. It is ideal for high density E-1 solutions and ISDN services where supported hardware and signaling technology can vary widely from country to country because it provides a consistent, high-level interface to the user and handles individual protocol requirements transparently.

The GlobalCall API supports ICAPI (T-1 and E-1 CAS) and the ISDN call control library.

NOTE: Unlike the ISDN call control library, ICAPI may only be accessed through GlobalCall.

ISDN:

New Network Interface features for ISDN provide voice adaptation layer and enhancements on the following boards: D/240SC-T1, D/240SC-2T1, D/300SC-E1, D/300SC-2E1, D/480SC-2T1, D/600SC-2E1, DTI/240SC, DTI/241SC, DTI/300SC, DTI/301SC, DTI/480SC, DTI/481SC DTI/600SC and the DTI/601SC. New features include:

- Vari-A-Bill
- Advice of Charge (NTT switch only)
- Call-by-Call service selection
- Dialed Number Identification Service (DNIS)
- Subaddressing
- Non-Facility Associated Signaling (NFAS)
- Automatic Number Identification (ANI) on demand
- DialView suite

DCM API - Customized Installation and Configuration

The API for Dialogic Configuration Manager (DCM) and the Dialogic OEM Installation Tool (DOIT) enable users to create customized and unattended installation and configuration programs for their customers.

PCI support

Dialogic now offers PCI bus support on several of its products.

Antares Speech Platform and SDK support

Antares is a powerful Digital Signal Processor (DSP) expansion board based on the SCbus standard. Antares works in conjunction with the Dialogic series of industry standard SCbus products. This release contains all support necessary to install, configure, download and run the Antares boards. It does not include any technologies available through third party vendors (ASR, TTS, etc.) other than a demo play-record technology. Please note the following:

- The installation and configuration procedures for Antares differ from those for other Dialogic boards. Please consult the *Getting Started* booklet that accompanies the CD-ROM for details.
- **Current users of Antares should be aware of the procedures for converting existing configuration data to the new data format employed by DCM. If you do not carefully follow the procedure outlined in the *Getting Started* booklet, you may lose your existing configuration data.**
- The Antares software components include: host drivers, libraries and header files necessary to run an Antares application. These components must be present in order to operate the Antares technology (such as those provided by third-party technology developers).
- The technology developer's package may require the use of a special hardware device called a *dongle*. The dongle is a security device enabling use of the product. Please consult the documentation provided by the technology developer for additional information.
- Antares is not part of the Typical installation option. To install Antares, please choose either the Custom Installation option or the Complete Installation option. For additional information, please access the Installation online helpfile.

New Hardware Descriptions

- D/240PCI-T1 - SCbus-compatible 24-channel voice and T-1 network interface board in PCI form factor with play/record, tone, and call progress analysis
- D/300PCI-E1 - SCbus-compatible 30-channel voice and 75- or 120- Ohm E-1 network interface board in PCI form factor with play/record, tone, and call progress analysis
- D/41EPCI - SCbus-compatible 4-channel analog PCI voice processing board
- VFX/PCI - SCbus-compatible 4-channel analog PCI voice and fax processing board

High Density T-1 / E-1 ISDN Network Interface:

- DTI/481SC - provides the equivalent of two DTI/241SC boards in a single slot with dual T-1 digital network interface; tone and call progress support

- DTI/601SC - provides the equivalent of two DTI/301SC boards in a single slot with dual E-1 digital network interface; tone and call progress support

Antares Open DSP Platform:

Four-DSP open platform solutions supporting third-party speech technologies such as Automated Speech Recognition (ASR) and Text-to-Speech (TTS) are available. The ISA form factor boards have varying on-board memory configurations:

- Antares 2000/50 - 512K SRAM; 4Mb DRAM
- Antares 3000/50 - 512K SRAM; 8Mb DRAM
- Antares 6000/50 - 2Mb SRAM; 8Mb DRAM

Please note: The relative advantage in using one memory configuration over another is dependent upon the specific third party technology product being used.

Dialogic Performance Counters for the Windows NT Performance Monitor

This is an installable option enabling the user to view and monitor the performance of Dialogic boards using the Windows NT Performance Monitor. The application file **Perfctl.exe** turns the performance counters on and off. Please see the DCM online help file for additional information.

New and Revised Documents

- *MSI/SC Programming Reference for Windows NT*
- *Primary Rate Software Reference for Windows NT*
- *Dialogic Hardware Diagnostics User's Guide*
- *GlobalCall E-1/T-1 Technology User's Guide for UNIX and Windows NT*
- *GlobalCall ISDN Technology User's Guide for UNIX and Windows NT*
- *GlobalCall Country Dependent Parameters (CDP) Reference*
- *GlobalCall API Software Reference for UNIX and Windows NT*
- *Dialogic System Software and SDK for Windows NT -- DNA Version 2 Release Reference*
- *Dialogic Audio Conferencing Software Reference for Windows NT*
- *Getting Started* (jewelcase booklet accompanying the CD-ROM)

Master Feature List

Operating Environment

Hardware Environment

This software release has the same minimum system requirements as Windows NT Workstation and Windows NT Server (versions 3.51 or 4.0) for Intel-based processors, with the following additional requirements:

- an additional 75 MB of available hard disk space
- an additional 32 MB of RAM recommended for Windows NT Server (for a total of 64 MB RAM)

A faster processor and additional RAM can provide performance improvements.

Software Environment

- Windows NT 3.51 (Workstation or Server) with or without Service Pack(s)
- Windows NT v.4.0 (Workstation or Server) with or without Service Pack(s)
- Microsoft Visual C++ 2.x, 4.x
- Borland C++ 4.5, 5.0

System

- Open System Architecture
- Dialogic Configuration Management (DCM)
- SCbus support
- Serial number and Silicon Serial number retrieval for Dialogic boards
- Board Locator Technology (BLT)
- Universal Dialogic Diagnostics (UDD)
- File Manipulation Routines
- Dialogic Performance Counters
- Online documentation provided in Windows WinHelp and PDF format on the CD-ROM
- Extended demos and sample programs
- Compiler Support (Borland C++ and Microsoft Visual C++)
- TAPI/WAV 2.0 support

Voice (Record and Playback Voice Files)

- Industry standard Voice Compression/Decompression
- Bulk data buffer sizing
- Transaction Record
- Multiple data formats for VOX and WAV audio files
- Speed and Volume Control
- PerfectPitch Voice Playback Speed Control
- PerfectLevel Voice Playback Volume Control
- Automatic Gain Control (AGC)
- PerfectVoice Voice Coding

FAX

- Transmit and Receive TIFF
- Transmit and Receive Raw
- ASCII-to-FAX (transmit only)
- Fax Error Correction (ECM) switch
- Scan Line Error Correction
- MR and MMR Advanced Compression
- Subaddress fax routing

Signaling

Tone Detection/Generation

- Analog Display Services Interface (ADSI)
- Dual Tone Multi Frequency (DTMF)
- Multi Frequency (MF)
- Global Tone Detection (GTD)
- Global Tone Generation (GTG)
- Global Dial Pulse Detection (GDPD)
- Caller ID

Call Analysis

- Basic Call Analysis
- PerfectCall Call Analysis
- Positive Voice Detection (PVD)
- Positive Answering Machine Detection (PAMD)
- Cadence Detection
- Frequency Detection
- Loop Current Detection
- Dial Tone Detection
- Special Information Tones (SIT) Frequency Detection

Dialogic System Software and SDK for Windows NT DNA 2 Release Catalog

- Ringback Detection
- Busy Tone Detection
- FAX or Modem Tone Detection
- Silence Detection

Network

- GlobalCall - ICAP
- Analog
- E-1
- T-1
- TS16
- E & M
- Ground Start
- Loop Start Originate

GlobalCall

ICAPI (E-1 CAS and T-1 Robbed Bit)

Protocols must be ordered separately. Please contact your sales representative for further information.

ISDN (T-1/E-1)

Protocols are included on the CD-ROM.

ISDN

Dialogic ISDN Package 2 level supports:

- DualSpan support
- Vari-A-Bill
- Advice of Charge (NTT switch only)
- Call-by-Call service selection
- Subaddressing
- Non-Facility Associated Signaling (NFAS)
- Automatic Number Identification (ANI) on demand
- DialView suite

Dialogic System Software and SDK for Windows NT DNA 2 Release Catalog

The following table shows ISDN-supported protocols (TE = Terminal Equipment, NT = Network Termination):

Protocol	Dialogic Name	Equipment Type
1TR6	1TR6	TE*
TR1449/TR41459	4ESS	TE
TR41449/TR41459	5ESS	TE
ETS1300 (VN4 in France)	CTR4 (Euro ISDN)	TE
BTNR-190-1985	DASS2	TE
ETS300-102 (VN4 in France)	NE1	NT*
TR41449/TR41459	NT1	NT
INS-Net 1500	NTT	TE
TS-01141 1990	TPHE	TE
VN3	VN	TE
VN3	VN NT	NT
TS-0141 1990	TPHNT	NT

Master Feature List

ISDN does not support the following features:

- Bulk Data Buffer Sizing
- Loop Current Detection
- GDPD
- Caller ID
- ADSI

ISDN does not support all digital network interface (dtlib.h) functions. The functions listed below are supported in the current ISDN firmware.

- ATDT_DNLDVER()
- ATDT_IDLEST()
- **dt_close()**
- **dt_getevtmsk()** (DTG_T1ERREVT & DTG_E1ERREVT only)
- **dt_open()**
- **dt_setalarm()**
- **dt_setevtmsk()** (DTG_T1ERREVT & DTG_E1ERREVT only)
- **dt_setidle()**
- **dt_setparm()**
- **dt_tstcom()**
- **dt_tstdat()**
- **dt_xmitarm()**
- **dt_xmittone()**

Switching

SCbus routing

Conferencing

- Flexible conferencing
- Hidden training
- Extended connections
- Monitoring
- Coach/Pupil monitoring
- Full duplex
- Individual volume control
- High port density
- Active Talker status
- On-board digit detection with tone clamping

Universal Dialogic Diagnostics (UDD)

Universal Dialogic Diagnostics (UDD) is a diagnostic utility for testing the functionality of Dialogic hardware. Refer to the *Dialogic Hardware Diagnostics User's Guide* for information about using the UDD utility.

UDD supports the following boards:

- D/21D
- D/41D
- D/21H
- D/41H
- D/41ESC
- D/80SC
- D/160SC
- D/160SC-LS
- D/240SC
- D/240SC-T1
- D/240SC-2T1
- D/300SC-2E1
- D/300SC-E1
- D/320SC
- D/480SC-2T1
- D/600SC-2E1
- DTI/240SC
- DTI/241SC
- DTI/300SC
- DTI/301SC
- DIALOG/4
- MSI/80SC
- MSI/80SC-R
- MSI/160SC
- MSI/160SC-R
- MSI/240SC

Dialogic System Software and SDK for Windows NT DNA 2 Release Catalog

- MSI/240SC-R
- ProLine/2V
- VFX/40SC
- VFX/40ESC
- VFX/40ESCplus
- VFX/PCI
- D/41EPCI
- D/240PCI-T1
- D/300PCI-E1

Restrictions

- Diagnostic tests are performed only on the D/41E baseboard.
- UDD does not differentiate between a DIALOG/4 board and a D/41D board when displaying the Dialogic configuration, running diagnostics and displaying results. The diagnostic tests and results are still valid.
- UDD identifies VFX-series fax boards by the corresponding D/41E-series baseboard; for example, VFX/40ESC is identified as D/41ESC. Diagnostic tests are performed on the D/41E-series baseboard only and not on the fax daughtercard.

TAPI 2.0/WAV support

TAPI/WAV requires that you have installed Windows NT 4.x or a later version of Windows NT that includes TAPI 2.0. You must activate the Dialogic Telephony Service Provider (TSP) and WAVE driver via the appropriate control panel applets.

Supported Hardware

SC in the board name implies SCbus connectivity.

Product	Description
D/21D	2-channel analog voice board with play/record, tone and call progress analysis
D/41D	4-channel analog voice board with play/record, tone and call progress analysis
DIALOG/4	4-channel, half-size form factor, analog voice board with analog loop start interfaces (PC-AT height)
D/21H	2-channel, half-size form factor, analog voice board with play/record, tone and call progress analysis (PC-AT height)
D/41H	4-channel, half-size form factor, analog voice board with play/record, tone and call progress analysis (PC-AT height)
ProLine/2V	2-channel analog voice board with play/record, tone and call progress analysis; an audio jack and a 2/3 size form factor (PC-XT height)
D/41ESC	4-channel board with call processing and analog loop-start interfaces
D/41EPCI	4-channel analog SCbus-compatible PCI voice processing board
VFX/40ESC	4-channel analog voice and 14,400 bps fax solution
VFX/40ESCplus	4-channel analog voice and 14,400 bps fax solution with an additional 128K DSP memory
VFX/40SC	4-channel analog voice and 9600 bps fax solution
VFX/PCI	4-channel analog SCbus-compatible; PCI voice and fax processing solution
D/80SC	8-channel voice board with play/record, tone, and call progress analysis
D/160SC	16-channel voice board with play/record, tone, and call progress analysis

Dialogic System Software and SDK for Windows NT DNA 2 Release Catalog

Product	Description
D/160SC-LS	16-channel voice and analog loop start network interface board with play/record, tone, and call progress analysis
D/160SC-LS-IDPD	16-channel voice and analog loop start network interface board with play/record, tone, call progress analysis, and Global DPD support
D/240PCI-T1	24-channel voice and T-1 network interface board in PCI form factor with play/record, tone, and call progress analysis
D/240SC	24-channel voice board with play/record, tone, and call progress analysis
D/240SC-2T1	24-channel voice board with play/record, tone, and call progress analysis and 2 network interfaces (48 ports)
D/240SC-IDPD	24-channel voice board with play/record, tone, call progress analysis and Global DPD support
D/240SC-T1	24-channel voice and T-1 network interface board with play/record, tone, and call progress analysis
D/240SC-T1-IDPD	24-channel voice and T-1 network interface board with play/record, tone, call progress analysis, and Global DPD support
D/300PCI-E1-75	30-channel voice and 75- Ohm E-1 network interface board in PCI form factor with play/record, tone, and call progress analysis
D/300PCI-E1-120	30-channel voice and 120- Ohm E-1 network interface board in PCI form factor with play/record, tone, and call progress analysis
D/300SC-2E1-75	30-channel voice board and dual E1 75- Ohm network interface board with play/record, tone and call progress analysis
D/300SC- 2E1-120	30-channel voice board and dual E1 120- Ohm network interface board with play/record, tone and call progress analysis
D/300SC-E1	30-channel voice board and 75- or 120- Ohm E-1 network interface board with play/record, tone, and call progress analysis
D/300SC-E1-75-IDPD	30-channel voice board and 75- Ohm E-1 network interface board with play/record, tone, and call progress analysis, and Global DPD support

Supported Hardware

Product	Description
D/300SC-E1-120-IDPD	30-channel voice board and 120- Ohm E-1 network interface board with play/record, tone, call progress analysis, and Global DPD support
D/320SC	32-channel voice board with play/record, tone, and call progress analysis
D/320SC-IDPD	32-channel voice board with play/record, tone, call progress analysis, and Global DPD support
D/480SC-2T1	48-channel voice board with play/record, tone, and call progress analysis; and 2 network interfaces with 48 ports
D/600SC-2E1-75	60-channel voice board with play/record, tone, and call progress analysis; 2 network interfaces with 60 ports (75- Ohm)
D/600SC-2E1-120	60-channel voice board with play/record, tone, and call progress analysis; 2 network interfaces with 60 ports (120- Ohm)
DTI/240SC	24-channel T-1 network interface board
DTI/241SC	24-channel T-1 network interface board with tone and call progress analysis
DTI/300SC	30-channel 75- or 120- Ohm E-1 network interface board
DTI/301SC	30-channel 75- or 120- Ohm E-1 network interface board with tone and call progress analysis
DTI/480SC	provides the equivalent of two DTI/240SC boards in a single slot with a dual T-1 digital network interface board
DTI/481SC	provides the equivalent of two DTI/241SC boards in a single slot with a dual T-1 digital network interface board with tone and call progress analysis
DTI/600SC	provides the equivalent of two DTI/300SC boards in a single slot with a dual E-1 digital network interface (75- or 120- Ohm)
DTI/601SC	provides the equivalent of two DTI/301SC boards in a single slot with a dual E-1 digital network interface board with tone and call progress analysis
LSI/81SC	8-channel analog loop start network interface board with tone and call progress analysis

Dialogic System Software and SDK for Windows NT DNA 2 Release Catalog

Product	Description
LSI/161SC	16-channel analog loop start network interface board with tone and call progress analysis
DCB/320SC	conferencing board with 32 DSP-based conferencing resources
DCB/640SC	conferencing board with 64 DSP-based conferencing resources
DCB/960SC	conferencing board with 96 DSP-based conferencing resources
MSI/80SC	conferencing board with 8 modular station interfaces
MSI/80SC-R	conferencing board with 8 modular station interfaces and ringing capability
MSI/160SC	conferencing board with 16 modular station interfaces
MSI/160SC-R	conferencing board with 16 modular station interfaces and ringing capability
MSI/240SC	conferencing board with 24 modular station interfaces
MSI/240SC-R	conferencing board with 24 modular station interfaces and ringing capability
Antares 2000/50	512K SRAM; 4Mb DRAM
Antares 3000/50	512K SRAM; 8Mb DRAM
Antares 6000/50	2Mb SRAM; 8Mb DRAM

Documents

Hardcopy Documentation:	Part Number:
Converting a Windows NT App from PEB to SCbus	05-0576-002
System Software and SDK for Windows NT DNA 2 " <i>Getting Started</i> " CD-ROM Jewelcase booklet	05-1062-001
Dialogic Hardware Diagnostics User's Guide	05-1089-001
Dialogic System Software and SDK for Windows NT - DNA 2 Release Catalog	05-0891-003
Dialogic System Software and SDK for Windows NT - DNA Release Reference	05-0774-004
SCbus Routing Function Reference for Windows NT	05-0439-002
SCbus Routing Guide for Windows NT	05-0289-004
Voice Software Reference for Windows NT	05-0168-007
GlobalCall E-1/T-1 Technology User's Guide for UNIX and Windows NT	05-0615-002
GlobalCall ISDN Technology User's Guide for UNIX and Windows NT	05-0653-002
GlobalCall Country Dependent Parameters (CDP) Reference	05-0870-001
GlobalCall API Software Reference for UNIX and Windows NT	05-0387-003
Primary Rate SW Reference for Windows NT	05-0481-003
FAX SW Reference for Windows NT	05-0172-006
MSI/SC SW Reference for Windows NT	05-0505-002
Dialogic Audio Conferencing SW Ref for Windows NT	05-0512-001
Digital Network SW Reference for Windows NT	05-0438-001

Fixes

- Fixed a problem with DF_ASCIIDATA - previously, when units of PEL were submitted to the structure, some data was ignored. Uninitialized parameters caused the fax session to abort.
- Previously, the sample.exe demo tried to open dtib5 when it detected a digital board, if more than one network interface board was in the system.
- Made an improvement so that the length of the beep before recording (RM_TONE) is now the same length as in previous Streams-based releases.
- PTR# 4695 - An improvement was made where previously a "click" was heard at the end of a recording session when terminated with a DTMF using 8kHz PCM recording mode.
- PTR# 5201 - Added CT_DFD41D product family code previously missing from dxxxlib.h.
- PTR# 4352 - Added function declarations for **r2_createsig()** and **r2_playbsig()** previously missing from dxxxlib.h.
- PTR# 5098 - The function **ms_geningex()** previously missing from MSI library (msilibmt.dll) is now available.
- PTR# 5015 - Fixed a problem where previously the function ATDX_TRCOUNT() returned a 0 at all times.
- PTR# 4988 - Fixed a problem with the DCM where previously attempting to change the SCbusClockMasterSource for a Network Interface board would cause a Dialogic System Service to fail.

Release Issues

Hardware Restrictions

Setting the A000 and B000 Segments: Even though you are physically able to set the jumpers and switches on Dialogic hardware configurable boards (D/21D, D/41D, DIALOG/4, ProLine/2V, D/21H and D/41H) to use a base memory address in the A000 or B000 segment (as described in the hardware installation card), the Board Configuration software does not support these segments. If you attempt to use the A000 and B000 segments, you may encounter conflicts, including video distortions.

Compatibility Issues with DNA1 Production Release

There is an issue with retaining the current configuration when upgrading from DNA 1 to DNA 2. If the user removes the prior release (DNA 1), as suggested in the installation instructions, existing configuration data will be lost.

For those who want to save existing configuration, install DNA2 over the top of DNA1. This will keep configuration settings intact.

System Startup

When the application is first invoked after a Dialogic Service Startup, if the application enables signaling transition notification via the **dt_setevtmsk()** function, a DTG_SIGEVT is posted automatically. The state of the receiving signaling is typically DTMM_AOFF and DTMM_BOFF, if enabled, as if the transition had just occurred. Subsequent generation of these events is only on signaling state change.

Miscellaneous

- The **dx_reciottdata()** and **dx_playiottdata()** functions do not support recording or playing to/from memory (iott.io_type = IO_MEM) with FileFormat set to FILE_FORMAT_WAVE.
- The ProLine/2V, D/21D, D/41D, D/21H and D/41H voice boards do not support the A-law PCM digitizing method for playing and recording.

Caller ID

Caller ID is not supported on D/21D, D/41D, DIALOG/4, VFX/40SC or VFX/40ESC boards.

MSI/SC

- If the coach speaks before any conversation has taken place between the client and the pupil, the client will hear some background noise for a fraction of a second. Under most circumstances, this will not be a problem since the coach usually will not need to speak before some conversation has taken place between the client and the pupil.
- A glare condition occurs when two parties simultaneously seize the same line. Although very rare, a glare condition may result if **ms_genring()** is issued asynchronously to a station at the same time the station is being placed off hook. If glare occurs in your application, **ms_genring()** will return successfully. However, it is followed by the event MSEV_NORING. The data associated with the event is E_MSBADRNGSTA, indicating that the station was off-hook when the ring was attempted. This issue is being addressed.

Global DPD

- GDPD is supported on: D/41ESC, D/41ESCplus, ProLine/2V, D/21H, D/41H, VFX/40ESCplus, D/80SC, D/160SC, D/160SC-LS, D/240SC, D/240SC-T1, D/240SC-2T1, D/320SC, D/300SC-E1, D/480SC-2T1, LSI/81SC, LSI/161SC, DTI/241SC, DTI/301Sc and D/600SC-2E1 (boards will have an -IDPD suffix).
- GDPD will work only on DPD-enabled boards via an enablement diskette unless -IDPD is part of the board name. To indicate that a board is DPD-enabled, apply the sticker provided with the Global DPD enablement package

to your board. Additionally, it is recommended that you write down the serial number of the DPD-enabled board for your records.

ISDN

NFAS is only supported by the 4ESS, 5ESS and DMS protocols.

TAPI/WAVE 2.0

The Dialogic TSP does not support the following TAPI line functions:

- **lineCompleteCall()**
- **lineDevSpecific()**
- **lineDevSpecificFeature()**
- **linePickup()**
- **lineRedirect()**
- **lineRemoveFromConference()**
- **lineSendUserUserInfo()**
- **lineSetDevconfig()**
- **lineSetMediaControl()**
- **lineSetTerminal()**
- **lineUncompleteCall()**

The WAVE input driver does not support the **waveinMessage** input device function and message.

The WAVE output driver does not support the following output device functions and messages.

- **waveOutGetPitch**
- **waveOutGetPlaybackRate**
- **waveOutMessage**
- **waveOutSetPitch**
- **waveOutSetPlaybackRate**

Compatibility Issues

Visit our website at

http://support.dialogic.com/releases/winnt/dna2/Release_Catalog.htm for up-to-date information on compatibility issues that affect this release.

Also consult:

- The Dialogic Products and Services Guide
 - The latest WorldView CD-ROM
 - Your sales representative
 - Dialogic Technical Support at 973-993-1443
-
- Due to improvements incorporated into the MSI/SC-R during the installation process of Windows NT Dialogic Native Architecture Version 2 software, a semi-permanent change to the MSI/SC-R hardware (EEPROM) renders it incompatible with earlier Streams-based system releases. These modifications are retained even when the board is powered off. Once the board is modified in this manner, it will no longer operate with the previously released software. All MSI/SC products with the ringing option installed are affected by this upgrade situation (MSI/80SC-R, MSI/160SC-R and MSI/240SC-R). If this situation occurs, please contact Dialogic Customer Engineering at (973) 993-1443.
 - If threads are created and killed during normal application processing, either of two compatibility problems may occur: (1) If you issue a multi tasking function such as **dx_playiottdata()**, **dx_reciottdata()** or **dx_getdig()** asynchronously with a thread, and that thread subsequently exits or dies before the function completes, the function never returns a completion event even if another thread is waiting for it. (2) If you open devices within a thread and that thread subsequently exits or dies, the device might no longer receive unsolicited events even if other threads are waiting for those events and the device handle is still valid.

Release Issues

- Application Compatibility: Applications previously developed on Windows 95 for Dialogic supported boards in a “stand-alone” configuration (using the on-board analog network interface) will work with this software release. Also, binary compatibility exists for an application previously developed on Windows NT and Windows 95 if either of the following are true:
 - The application uses only a common set of features and functions that are supported on both systems, or
 - The application is designed to work on either system with the presence of unsupported features and functions handled through the Compatibility Library, that allows run-time linking of source code.
- All demos and makefiles provided in this release have been compiled and linked with Microsoft Visual C++ 4.x.
- On-Hook Audio Path: The D/41E series, D/21H, D/41H, ProLine/2V and VFX40ESCplus boards provide an on-hook audio path (this hardware feature is provided for Caller ID support). If you port an application from the D/41D voice board to one of these boards, your application may detect silence on/off events while the channel is still on-hook. You can adapt your application for this situation by using the **dx_setevtmask()** function to disable silence on/off events while the channel is on-hook.
- Several of the guides provided with this release contain information to support SCbus and PEB hardware products. Although PEB products are referenced in these manuals, PEB configurations are not supported in this release.
- **FAX:** The Fax parameter FC_STOPCH is no longer supported. Attempting to set or retrieve the value of this parameter with the function **fx_setparm()** will result in an error. **fx_setparm()** will return -1 with no extended error information available.

Known Problems

Visit our website at

http://support.dialogic.com/releases/winnt/dna2/Release_Catalog.htm for the most current information regarding unresolved PTRs (Problem Tracking Reports) associated with this release.

- **Antares Macro Redefinitions**

A conflict with the `equat` exists because it is redefined in the `antares.h` file. While this does not affect functionality, it will cause a warning message when compiling a file that has a dependency on the `Antares.h` file. For example:

antares.h

```
\dlgcdev\dialogic\inc\antares.h (518): warning C4005: 'MAXERRORLENGTH':  
macro redefinition. This macro was previously defined in  
\msdev\include\mmsystem.h →(83) #define MAXERRORLENGTH 256.
```

- The subaddress fax routing functionality is not working properly. Specifically, the receiver end cannot receive subaddress information since the function `fx_getparm()` fails to get the `FC_REMOTESUBADDR` parameter.
- **PTR# 5041: `dt_getevt()`** does not pull network T-1 alarm events. Use `sr_waitevt()` instead.
- **FAX:** When a `fx_send()` is issued after a blind dial and the end user never picks up, the function fails as expected; however the error code is set to `ERROR_NONE`.
- **System Start Up:** In previous releases of Dialogic software, when a single board failed to download, all other boards would be downloaded and the system would successfully start. Under this release, a single download failure will cause the start up process to abort.
- **System Start Up:** The time slot assignment program does not put information into the event view on pass or fail. This program is run after the boards are downloaded and started. If the startup service fails, and the boards have both been successfully downloaded and started, the timeslot assignment program is at fault. Failure of this program usually indicates an IRQ conflict.
- **DCM** does not adjust the `BLTId` parameter when a board's rotary switch setting is changed. In this case, the `BLTId` parameter must be updated by the user from the System property sheet in DCM.
- **DCM** does not automatically detect BLT boards if the system contains a hardware configurable board set to the base memory address `D8000`. To resolve this problem, the user should:

Release Issues

1. Click the **Add Device** option from the **Action** pull-down menu on DCM's main window.
2. Select the appropriate BLT board model from the Add Hardware wizard.
3. When the DCM Property Window appears, select the System property sheet.
4. Set the BLTAddress parameter to a value other than D8000.
For instructions on selecting the appropriate address, consult *Avoiding Memory Conflicts*, that is available through the Search or Index tab of DCM's online help.
5. Exit and re-start DCM; the auto-detect feature will then detect all other BLT boards.
 - **UDD:** The loop start front end test in UDD takes up to 20 minutes to complete on a D/160SC-LS board.
 - **Voice:** When called on analog front end boards, the **dx_play()**, **dx_record()**, **dx_getdtmf()** and **dx_playtone()** functions will fail to perform the first time they are called on a channel. This applies to all analog front end boards, even if the front end is not in use (i.e., board is in resource mode). Issuing a **dx_sethook()** before these function calls will eliminate the problem. Note that the function will return successfully and subsequent calls to these functions will work properly.
 - **Voice Library:** **dx_bldXX()**, **dx_addtone()** and **dx_initcallp()** are global functions and have the potential "bad tone template." Workaround: Semaphore protect these calls from within threads.
 - There is the possibility of poor conference quality (cracking and popping sound) when using the MSI/SC board with the following boards. If you experience this, contact the Dialogic RMA Department for instructions on returning your board. Affected boards are: D/160SC-LS (pn 99-0395-006), D/160SC-LS-IDPD (pn 99-0930-001), LSI/81SC (pn 99-0796-001), LSI/161SC (pn 99-0797-002).

Restrictions and Limitations

Visit our website at http://support.dialogic.com/releases/winnt/dna2/Release_Catalog.htm for the most current information regarding restrictions and limitations associated with this release.

SMP

Do not use the IRQ5 setting on a Dialogic board in an SMP environment. A Windows NT kernel crash (blue screen) could occur.

PCI

Installing and configuring Dialogic PCI boards in a system which already contains Dialogic ISA boards may present some issues for device names. Please consult the DCM online help for complete details.

Performance Counters

Please note that overall system performance is impacted when using the Performance Counters feature.

ISDN does not support the following features:

- Bulk data buffer sizing
- Loop current detection
- GDPD
- Caller ID
- ADSI

GlobalCall

Please note: Analog technology is not supported in this release.

Alarm Handling: The GlobalCall API does not currently enable the application to keep track of all lower level alarm conditions which occur on the line nor does the GlobalCall API currently support any function that can be used to send an alarm

to the line. The GCEV_BLOCKED event will indicate when an alarm condition exists on the line and will indicate what condition caused the first alarm event. However, the application will not receive additional events that report subsequent alarm conditions.

This limitation primarily affects applications using a drop and insert configuration; when certain alarm conditions are detected the application may need to generate an alarm. The application cannot use any current GlobalCall function to send an alarm to the line.

When a network board is in a terminate configuration, the sending of an alarm in response to a loss of sync condition is handled automatically.

For additional details about the GCEV_BLOCKED and GCEV_UNBLOCKED events, see the GlobalCall API Software Reference for UNIX and Windows NT.

E-1 applications may receive the 0x2050 (DTEV_MTFNCNPT) unexpected event. This event has been observed when running E-1 applications using the Chile protocol. This event will not affect the application and can be ignored. If your application receives this event, ignore it. If you are operating in callback mode, you must use a handler to remove the event from the SRL event queue by returning a 0. The following example code can be used to handle an unexpected event (when operating in the SRL polled mode (non-callback) and assuming that the SRL has been set to polled mode):

```
/* the following ignores unexpected events */
METAEVENT          metaevent;
.
.
.
for (;;) {
    sr_waitevt (-1L);          /* wait forever */
    /* Get the next event */
    if (gc_GetMetaEvent (&metaevent) != GC_SUCCESS) {
        /* process error */
    }
    if (metaevent.flags & GCME_GC_EVENT) {
        /* process GlobalCall event */
        .
        .
        .
    } else {
        /* process non-GlobalCall event */
        switch (metaevent.evtttype) {
            /* process all 'expected' events */
            .
            .
            .
        default:
            /* ignore all 'unexpected' events */
```

```
        break;
    }
}
}
}
When operating in the callback mode (both signal and non-
signal) the following handler can be used to process all
events:
int process_event_handler ( )
{
    if (gc_GetMetaEvent (&metaevent) !=GC_SUCCESS) {
        /* process error */
    }
    if (metaevent.flags & GCME_GC_EVENT) {
        /* process GlobalCall event */
        .
        .
    } else {
        /* process non-GlobalCall event */
        switch (evttype) {
            /* process all 'expected' events */
            .
            .
            default:
                /* ignore all 'unexpected' events */
                return 0;
                break;
        }
    }
}
```

VOICE

In the **dx_setuio()** function UIOBLK must remain in scope for the duration of the **dx_play()** / **dx_record()** function for the User I/O being specified. Also, the user must define the read, write and seek function or the **dx_setuio()** function will fail.

FAX

- When converting an ASCII file, the tab stops cannot be greater than 32.
- The upper limit on margins for a converted ASCII document is 512 pixels.
- The minimum page length allowed to an ASCII file is 52 in 0.1 inch units and 133 in millimeter units.

- FAX header format changes (DF_HDRDDATEFMT and DF_HDRTIMEFMT) must be specified prior to using DF_HDRDATETIME to place the date/time in the header.

Installation/Configuration/Diagnostics

General Considerations

- Users who have already installed version 1 of the Dialogic Native Architecture should not uninstall version 1. Rather, install version 2 over version 1. Uninstallation of version 1 will result in a total loss of the existing Dialogic Configuration Manager (DCM) configuration data.
- DCM provides a graphical user interface (GUI) for modifying configuration data. Users who wish to modify configuration data through their own applications should employ the DCM API, which is documented in the *Installation and Configuration Customization Guide for the Dialogic Native Architecture*. Under no circumstances should the registry keys be modified directly.
- Universal Dialogic Diagnostics (UDD) does not differentiate between a DIALOG/4 board and a D/41D board when displaying the Dialogic configuration, running diagnostics, and displaying results. The diagnostic tests and the results are still valid.
- UDD identifies VFX/40E-series fax boards by the corresponding D/41E-series baseboard; for example, VFX/40ESC is identified as D/41ESC. Diagnostic tests are performed on the D/41E-series baseboard only and not on the fax daughterboard.
- The Uninstall Utility will only remove the directories and files that were created with the installation of the Dialogic software. All directories and files that existed prior to the installation are not removed.

Error Messages When Starting the Dialogic Service with Digital Network Interface Boards

When you start the Dialogic Service, you may encounter downloading errors. The download error message for Digital Network Interface (DNI) boards requires explanation. An example of the typical download error message is:

D/480SC-2T1 (ID 11) download failed

The ID number corresponds to one of the DNI board's control processors (CPs). The CP indicated by the ID number is where the download failure occurred. To correlate the ID number with the board containing the faulty CP, consult the following table. The table indicates BLT-ID (rotary switch setting) of the board with the corresponding CP:

Board Rotary Switch Setting	ID Numbers			
	Network Span CP #1	Network Span CP #2	Resource CP	Resource CP
0	0	10	20	30
1	1	11	21	31
2	2	12	22	32
3	3	13	23	33
4	4	14	24	34
5	5	15	25	35
6	6	16	26	36
7	7	17	27	37
8	8	18	28	38
9	9	19	29	39
A	A	1A	2A	3A
B	B	1B	2B	3B
C	C	1C	2C	3C
D	D	1D	2D	3D
E	E	1E	2E	3E
F	F	1F	2F	3F

NOTE: If a D/41ESC board's SWI is set to ON, its ID will be 11.

Programming Guidelines

- **All technologies:** Passing an invalid handle (channel or device) to any function will result in an application exception. This will cause the application to terminate.
- Define **_CROSS_COMPATLIB_** preprocessor directive using the cross compatibility library for compiling under Borland C to avoid compilation without warning messages.
- When an application is first invoked after a Dialogic Service Startup, if the application enables signaling transition notification via the **dt_setevtmsk()** function, a DTG_SIG EVT is posted automatically. The state of the receiving signaling is typically DTMM_AOFF and DTMM_BOFF, if enabled, as if the transition had just occurred. Subsequent generation of these events is only on signaling state change.
- The **#include <windows.h>** statement must precede the **#include <srllib.h>** statement in all of your application code.
- If one function specifies a digit as a termination condition and the following function specifies a digit as its initiation condition (through the **dx_setparm()** DXCH_DTINITSET channel parameter), the initiating digit must be different from the terminating digit. In the case where the application design requires that the terminating and initiating digits be the same, do not use **dx_setparm()** DXCH_DTINITSET to start the second function. Instead, use the **dx_getdig()** function to retrieve the initiating digit and then initiate the function (thus simulating the DXCH_DTINITSET initiation condition).

Voice

- The **dx_reciottdata()** and **dx_playiottdata()** functions do not support recording or playing to/from memory (iott.io_type = IO_MEM) with FileFormat set to FILE_FORMAT_WAVE.
- Speed Control for the D/160SC-LS-IDPD, D/300SC-E1-75-IDPD and D/320SC-IDPD boards cannot be used while the Global DPD feature is enabled. You can adjust the speed before or after placing or receiving a call that uses the Global DPD feature. If any speed control adjustments are received by the board firmware while the Global DPD is enabled, the firmware will return a 0x0B error code that has no equate in the dxxxlib.n header file.
- When using the **dx_addtone()**, **dx_blddt()**, **dx_initcallp()**, **dx_blddt()**, **dx_blddtcad()**, **dx_bldst()**, **dx_bldstcad()**, and **dx_initcallp()** functions in a multithreaded environment, you must serialize these calls. These functions are global and not channel or device specific. Therefore, when one of these

functions is called in a given thread and the same function is being called in a second thread using different values of the tone template, a thread might be pre-empted and jump to a different thread, corrupting the data. Workaround: Semaphore protect **dx_addtone()**, **dx_blddt()**, **dx_blddtcad()**, **dx_bldst()**, **dx_bldstcad()**, and **dx_initcallp()** to avoid any corruption of the global data that results from calling any of these functions.

- When a voice resource is not listening to a network device, it may report spurious silence transitions and ring events if the events are enabled. Workaround:
 1. Disable the ring and silence detection on unrouted/unlistened channels using the **dx_setevtmsk()** function.
 2. When you need to change the resource currently connected to your network device, do a half duplex disconnect of the current resource and a full duplex connect on the new resource using the appropriate listen/unlisten functions or the convenience functions **nr_scroute()** and **nr_scunroute()**.
- If Caller ID is enabled, on-hook signal detection (DTMF, MF and Global Tone Detection) will not function.
- Detecting DTMF digits when a caller on one voice board is routed in a conversation on an analog line with a caller on another voice board (analog inbound/outbound configuration). If either caller sends a DTMF digit, both voice channels will detect the DTMF digit. This occurs because the network functionality of the analog front end on the voice board cannot be separated from the voice functionality in an analog connection between two callers. This makes it impossible to determine which caller sent the DTMF digit.

TAPI/WAVE 2.0

- This Dialogic software release of TAPI and WAVE requires that you have installed Windows NT 4.x or a later version of Windows NT, which includes TAPI 2.0.
- You must activate the Dialogic Telephony Service Provider (TSP) and WAVE driver according to the instructions "Installing and Removing the Dialogic TSP and WAVE Driver for TAPI and WAVE."
- **lineClose()** Will not automatically drop any calls in progress on the line; the application should perform a **lineDrop()** to disconnect any active calls before performing **lineClose()**.
- **lineGatherDigits()** A) 31 digits maximum. B) If both the inter-digit and first digit time-out parameters are specified, only the higher of the two specified values will be used to set both parameters.

Release Issues

- **LineGenerateTone()** Allows up to two LINEGENERATETONE entries, but if different dwCadenceOn and dwCadenceOff values are specified for the two entries, the average of the values will be used to generate the tone.
- **LineGetNewCalls()** The LINECALLSELECT_ADDRESS parameter is supported.
- **lineInitialize()** If you use this function, TAPI will see your application as being compliant with TAPI Version 1.4 (lineNegotiateAPIVersion() will always return Version 1.4), even if you try to negotiate Version 2.0. If you want to negotiate TAPI Version 2.0, you must use lineInitializeEx().
- **lineInitializeEx()** If you want to negotiate TAPI Version 2.0 or later, you must use this function rather than lineInitialize().
- **lineMonitorMedia()** Not relevant; has no effect on Dialogic boards.
- **lineMonitorTones()** This function is valid only if the call is in a "connected" state.
- **lineNegotiateExtVersion()** No device-specific extensions are currently implemented in this version of the TSP.
- **lineOpen()** The LINEMAPPER parameter is supported. This function supports the same media types as supported by Dialogic hardware.
- Calling **lineMonitorTones()** with lpToneList=NULL does not always cancel monitoring of the tone list currently in effect. In this case, you will continue to receive the LINE_MONITORTONE message until you drop the call. When the call is dropped, the tone lists effective for this call are canceled.
- The Dialogic TSP does not support phone-related functions and messages, except for stubs that specify that the function is not available. The reason for this is that the underlying hardware does not model a phone.
- The Dialogic TSP cannot enable or disable PerfectCall Call Progress Analysis for a specific channel. The Dialogic TSP is only capable of enabling or disabling PerfectCall Call Progress Analysis for all channels on all Dialogic boards in a system.
- When developing TAPI applications with conferencing capabilities using Dialogic boards, treat each channel of a Dialogic board as if it were a standard analog telephone. Telephone switches cannot differentiate between Dialogic board channels and analog telephones.
- The Dialogic TSP does not support the following TAPI line functions:
 - **lineCompleteCall()**
 - **lineDevSpecific()**
 - **lineDevSpecificFeature()**
 - **linePickup()**
 - **lineRedirect()**
 - **lineRemoveFromConference()**

- **lineSendUserUserInfo()**
- **lineSetDevConfig()**
- **lineSetMediaControl()**
- **lineSetMediaMode()**
- **lineSetTerminal()**
- **lineUncompleteCall()**
- The WAVE input driver does not support the waveInMessage input device function and message.
- The WAVE output driver does not support the following output device functions and messages:
 - **waveOutGetPitch**
 - **waveOutGetPlaybackRate**
 - **waveOutMessage**
 - **waveOutSetPitch**
 - **waveOutSetPlaybackRate**

MSI/SC

- If the coach speaks before any conversation has taken place between the client and the pupil, the client will hear some background noise for a fraction of a second. Under most circumstances, this will not be a problem since the coach usually will not need to speak before some conversation has taken place between the client and the pupil.
- A glare condition may occur when two parties simultaneously seize the same line. Although very rare, a glare condition may result if **ms_genring()** is issued asynchronously to a station at the same time the station is being placed off hook. If glare occurs in your application **ms_genring()** will return successfully. However, it is followed by the event MSEV_NORING. The data associated with the event is E_MSADRNGSTA, indicating that the station was off hook when the ring was attempted.
- In order to use the SCbus routing convenience functions **nr_scroute()** and **nr_unscroute()** with the MSI/SC device, the preprocessor directive DTISC must be defined when compiling the sctools.c file.

DCB/SC

- If a DSP handle is to be used for active talker retrieval, the DSP handle must be opened before conference resource changes are made on that DSP. This must be done to ensure that the resource table associated with the DSP handle contains all conference information.
- Echo cancellation is not supported. Using analog front end products with the DCB/SC for creating conferences will result in poor audio quality.

DualSpan boards

- Genload reports that only one span on a DualSpan board was downloaded even though both spans were downloaded. If one span fails to load, an error message will be displayed.

Documentation Details

This release is fully backward compatible with releases since System Release 4.25SC for Windows NT. Although the baseline for this release is Dialogic System Software and SDK 97.01 for Windows NT, some products and features (D/42 and TextTalk) that were available in 97.01 are not available with this release. In addition, other products, features and functions that were supported subsequent to the 97.01 release may not be supported in this release although they may be mentioned in the documentation set (Syntellect/Automated Attendant, Cadence Tone Detection/Generation and Tone Set File).

Documentation Additions

The following functions were previously undocumented:

dx_fileerrno()

Note: available since 4.25SC. Voice DLL(libdxxmt.dll)

dx_fileerrno() - Syntax

Name:	int dx_fileerrno (void)
Inputs:	none
Returns:	errno
Includes:	dxxlib.h srllib.h
Mode:	Synchronous

dx_fileerrno() - Description

The **dx_fileerrno()** function obtains the system error value reported by the Voice DLL. The system error value in the **errno** parameter has meaning only within the context of the run-time library. As an example, if one calls **dx_fileopen()** and the function fails, the correct **errno** parameter value can only be obtained from the context of the C run-time library that the Dialogic DLL loads at run time, which can differ from the version of the MSVC++ library that linked with the application at compile time.

This function can be used for any system error reported by the Voice DLL. For example, if a **dx_open()** failed, you would use **dx_fileerrno()** to obtain the system error value.

dx_fileerrno() - Example

```
rc=dx_fileopen(fileName, O_RDONLY);  
if (rc == -1) {  
    printf('Error opening %s, errno: %d\n', fileName,  
dx_fileerrno( ) );  
}
```

dx_sendevt()

dx_sendevt() - Syntax

Name:	int dx_sendevt (dev, evttype, evtdatap, evtlen, flags)	
Inputs:	int dev	valid Dialogic device handle
	long evttype	Type of event to be sent
	void *evtdatap	Pointer to data block associated with evttype
	short evtlen	Length of the data block in bytes
	unsigned short flags	SENDSelf/SENDOTHERS/ SENDALL
Returns:	0 if successful	
	-1 error return code	
Includes:	dxxxlib.h	
	srllib.h	
Mode:	Synchronous	

dx_sendevt() - Description

The **dx_sendevt()** function allows Inter-Process Event Communication. The event type parameter **evttype** and its associated data are sent to one or all processes that have the **dev** parameter device opened. The block pointed to by **evtdatap** cannot be greater than 256 bytes and hence **evtlen** cannot contain a number smaller than 0 and bigger than 256. The **evtdatap** parameter can be NULL and the **evtlen** parameter 0 if there is no data associated with an event type. The **flags** parameter determines which processes are going to receive this event. The following values are valid for the **flags** parameter:

EVFL_SENDSelf	/* Only the process calling dx_sendevt() will receive the event */
EVFL_SENDOTHERS	/* All processes that have the device opened except the process calling dx_sendevt() will receive the event */
EVFL_SENDALL	/* All processes that have the device opened will receive the event */

The events generated by this function can be retrieved using **sr_waitevt()**, by registering an event handler via **sr_enbhdr()** or by calling **dx_getevt()** to catch the event if the **evttype** is set to TDX_CST.

The application can define the **evtttype** and **evtdata** to be any values as long as **evtttype** is greater than 0x1FFFFFFF and less than 0x7FFFFFFF0. The only exception to this rule is the use of this function to stop **dx_wtring()** and **dx_getevt()** by sending TDX_CST events. To unblock a process waiting in **dx_wtring()** or **dx_getevt()**, send an event of type TDX_CST to that process. The **evtlen** will be the size of DX_CST structure and **evtdata** will point to a DX_CST structure with **cst.cst_event** set to DE_STOPWTRING or DE_STOPGETEVT as the case may be.

dx_sendevt() - Cautions

This function will fail if an invalid device handle is specified.

No event will be generated if event type value is greater than 0x7FFFFFFF0.

dx_sendevt() - Example

```
#include "srllib.h"
#include "dxxxlib.h"
int          dev;          /* Dialogic device handle */
DX_CST      cst;          /* TDX_CST event data block */

/* Open board 1 channel 1 device */
if ((dev = dx_open("dxxxB1C1", 0)) == -1) {
    printf("Cannot open channel dxxxB1C1.  errno = %d",
    errno);
    exit(1);
}

/* Set up DX_CST structure */
cst.cst_event = DE_STOPGETEVT;
cst.cst_data  = 0;

/* Send the event to all other processes that have dxxxB1C1
open */
if (dx_sendevt(dev, TDX_CST, &cst, sizeof(DX_CST),
EVFL_SENDOOTHERS) == -1) {
    printf("Error message = %s", ATDV_ERRMSGP(dev));
    exit(1);
}
```

dx_sendevt() - Errors

If the function returns -1 to indicate an error, use the SRL Standard Attribute function **ATDV_LASTERR()** to obtain the error code or you can use **ATDV_ERRMSGP()** to obtain a descriptive error message. The error codes returned by **ATDV_LASTERR()** are:

Equate	Returned When
EDX_BADPARAM	Invalid parameter.
EDX_SYSTEM	System error. Check the global variable errno for more information.

dx_sendevt() - See Also

- dx_getevt()
- sr_waitevt()

dx_setdevuio()

dx_setdevuio() - Syntax

Name:	int dx_setdevuio(chdev, setuiop, getuiopp)	
Inputs:	int chdev	Dialogic channel device handle
	DX_UIO *setuiop	Pointer to user I/O routines structure
	DX_UIO **getuiopp	Pointer to return pointer for user I/O routines structure
Returns:	0 if successful; -1 if failure	
Includes:	srllib.h	
	dxxlib.h	
Mode:	synchronous	

dx_setdevuio() - Description

The **dx_setdevuio()** function will allow an application to install and retrieve user-defined I/O functions on a per Dialogic channel device basis. The user I/O functions installed using this function will be used in all subsequent I/O operations performed on the channel even if the application installs global user I/O functions for all devices using the **dx_setuio()** function. The user I/O functions are installed by installing a pointer to a **DX_UIO** structure which contains addresses of the user-defined I/O functions.

The first argument to the function is **chdev**, the descriptor of a Dialogic channel device. This specifies the channel for which the user-defined I/O functions will be installed. The second argument, **setuiop**, is a pointer to an application-defined global DX_UIO structure which contains the addresses of the user-defined I/O functions. This pointer to the DX_UIO structure will be stored in the Dialogic Voice DLL for the specified chdev channel device. The application must not overwrite

the DX_UIO structure until **dx_setdevuio()** has been called again for this device with the pointer to another DX_UIO structure. The third argument, **getuiopp**, is the address of a pointer to a DX_UIO structure. Any previously installed I/O functions for the chdev device are returned to the application as a pointer to DX_UIO structure in getuiopp. If this is the first time **dx_setdevuio()** is called for a device, then getuiopp will be filled with the pointer to the global DX_UIO structure which may contain addresses of the user-defined I/O functions that apply to all devices.

Either of **setuiop** or **getuiopp** may be NULL, but not both at the same time. If getuiopp is NULL, the **dx_setdevuio()** function will only install the user I/O functions specified via the DX_UIO pointer in setuiop but will not return the address of the previously installed DX_UIO structure. If setuiop is NULL, then the previously installed DX_UIO structure pointer will be returned in getuiopp but no new functions will be installed.

dx_setdevuio() - Cautions

The DX_UIO structure pointed to by setuiop must not be altered until the next call to **dx_setdevuio()** with new values for user-defined I/O functions.

For proper operation, it is the application's responsibility to properly define the three DX_UIO user routines: **u_read**, **u_write** and **u_seek**. Please see the *Voice Software Reference for Windows NT* for more information on the DX_UIO structure.

dx_setdevuio() - Error Codes

If the function returns -1 to indicate an error, use the SRL Standard Attribute function ATDV_LASTERR() to obtain the error code or you can use ATDV_ERRMSGP() to obtain a descriptive error message. The error codes returned by ATDV_LASTERR() are:

Equate	Returned When
EDX_BADPARAM	Invalid parameter.
EDX_BADDEV	Invalid Device Descriptor.

dx_setdevuio() - Example

```
#include "windows.h"
#include "srllib.h"
#include "dxxxlib.h"

int chdev;          /* channel descriptor */
DX_UIO devio;      /* User defined I/O
functions */
DX_UIO *getiopp;   /* Retrieve I/O functions */

int appread(fd, ptr, cnt)
```

Documentation Details

```
    int          fd;
    char         *ptr;
    unsigned     cnt;
}
    printf("appread: Read request\n");
    return(read(fd, ptr, cnt));
}

int appwrite(fd, ptr, cnt)
    int          fd;
    char         *ptr;
    unsigned     cnt;
{
    printf("appwrite: Write request\n");
    return(write(fd, ptr, cnt));
}

int appseek(fd, offset, whence)
    int          fd;
    long         offset;
    int          whence;
{
    printf("appseek: Seek request\n");
    return(lseek(fd, offset, whence));
}

main(argc, argv)
    int          argc;
    char        *argv[];
{
    /* Open channel */
    if ((chdev = dx_open("dxxxBlC1",0)) == -1) {
        printf("Cannot open channel\n");
        printf("errno = %d\n",errno);
        exit(1);
    }

    . /* Other initialization */
    .
    /* Initialize the device specific UIO structure */
    devio.u_read  = appread;
    devio.u_write = appwrite;
    devio.u_seek  = appseek;
}
```

```
/* Install the applications I/O routines */
if (dx_setdevuio(chdev, &devio, &getiop) == -1) {
    printf("error registering the UIO routines = %d\n",
ATDV_LASTERR(chdev) );
    }}
```

dx_setdevuio() - See Also

dx_setuio()

Documentation Corrections

Call Analysis

The following corrections apply to the defines for the Call Analysis Parameter structure (DX_CAP) field **ca_intflg** in the *Voice Software Reference for Windows NT*:

- DX_OPTEN should be documented as obsolete (it applied only to non-DSP boards such as the D/41B and D/21B voice boards; use DX_OPTNOCON for DSP boards).
- DX_PVDOPTEN should be documented as obsolete (it applied only to non-DSP boards such as the D/41B and D/21B voice boards; use DX_PVDOPTNOCON for DSP boards).
- DX_PAMDENABLE description should say "Enable PAMD and PVD."
- DX_PAMDOPTEN description should say "Enable PAMD, PVD and DX_OPTNOCON."
- DX_OPTDIS description should say "Disable SIT Detection, PAMD and PVD."

dx_playvox() Code Example

The final "If" statement of the Code Example in the Voice Software Reference API, should appear as follows:

```
if (dx_playvox(chdev, "HELLO.VOX", &tpt, NULL, EV_SYNC) == -1) {
```

dx_recf() Mode Argument

The Voice Software Reference for Windows NT states **dx_recf()** is a Synchronous/Asynchronous function, however the function can only run in Synchronous mode (refer to PTR# 5171).

DX_XPB Data Structure

The Voice Software Reference for Windows NT defines one of the DX_XPB data structure fields as nBitsPerSample; however the correct DX_XPB field is defined in dxxxlib.h is wBitsPerSample (refer to PTR# 5236).

Speed and Volume Control

The following correction applies to the Speed Modification Table in the *Voice Software Reference for Windows NT*.

The table entries for the increase positions should be changed as follows:

Table Entry	Default Value (%)	Absolute Position
increase[5]	-128 (0x80)	6
increase[6]	-128 (0x80)	7
increase[7]	-128 (0x80)	8
increase[8]	-128 (0x80)	9
increase[9]	-128 (0x80)	10