

**ODI**

**NetWare Server Driver**

**Development Toolkit**

**Driver Specification Version 3.1**

Part Number: 107-000007-001  
Document Version 1.00  
March 15, 1993

## Disclaimer

Novell, Inc. makes no representations or warranties with respect to the contents or use of this manual, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

© Copyright 1992 and 1993 by Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express prior written consent of the publisher.

Novell, Incorporated  
122 East 1700 South  
Provo, Utah 84606

## Trademarks

Novell has made every effort to supply trademark information about company names, products, and services mentioned in this document. Trademarks indicated below were derived from various sources.

Ethernet is a trademark of Digital Equipment Corporation, Intel Corporation, and Xerox Corporation.

IBM, Micro Channel, and Token-Ring are trademarks of International Business Machines Corporation.

Intel is a trademark of Intel Corporation.

Novell, NetWare, and RX-Net are registered trademarks of Novell, Inc.

Phar Lap is a trademark of Phar Lap Software, Inc.

---

# Table of Contents

## Chapter 1: Introduction

Open Data-Link Interface . . . . .	1-2
Link Support Layer . . . . .	1-2
Multiple Link Interface Drivers . . . . .	1-3
NetWare Loadable Modules . . . . .	1-3
Driver Modules . . . . .	1-4
Novell Provided Support Modules . . . . .	1-4
Developer Provided Module . . . . .	1-5
Loading Driver Modules . . . . .	1-6
Development Process . . . . .	1-7
Driver Related Files . . . . .	1-7

## Chapter 2: HSM Overview

HSM Components . . . . .	2-2
HSM Procedures . . . . .	2-2
HSM Data Structures and Variables . . . . .	2-4
HSM Design Considerations . . . . .	2-5
Hardware Issues . . . . .	2-5
Coding Issues . . . . .	2-6
Code and Data Space . . . . .	2-8
Multichannel Adapters . . . . .	2-10
Recommended Support . . . . .	2-10
Optional Support . . . . .	2-11

## Chapter 3: HSM Data Structures and Variables

Introduction . . . . .	3-2
Global Data Access . . . . .	3-2
Driver Parameter Block . . . . .	3-3
Driver Configuration Table . . . . .	3-9
Driver Frame Data Space . . . . .	3-9
Driver Adapter Data Space . . . . .	3-21
Driver Statistics Table . . . . .	3-21
Media Specific Counters . . . . .	3-27
Driver Firmware . . . . .	3-35
Driver Keywords . . . . .	3-37
Driver Keyword Enhancements . . . . .	3-39

---

## Chapter 4: MSM / TSM Data Structures and Variables

Introduction	4-3
MSM Equates	4-4
MSMVirtualBoardLink	4-4
MSMStatusFlags	4-5
MSMTxFreeCount	4-6
MSMMaxFrameHeaderSize	4-7
MSMPhysNodeAddress	4-8
Public Variables	4-9
LogicalToPhysical	4-9
PhysicalToLogical	4-9
Data Structures	4-10
Receive Control Blocks	4-11
Transmit Control Blocks	4-16
Event Control Blocks	4-22

## Chapter 5: HSM Procedures

Introduction	5-2
Initialization	5-3
DriverInit	5-3
Packet Reception	5-10
Reception Methods	5-10
DriverISR	5-15
DriverPoll	5-21
Packet Transmission	5-22
Transmission Methods	5-22
DriverSend	5-24
Multi-Operating System Support	5-25
DriverEnableInterrupt	5-26
DriverDisableInterrupt	5-26
Control Procedures	5-27
DriverReset	5-28
DriverShutdown	5-29
DriverMulticastChange	5-30
DriverPromiscuousChange	5-32
DriverStatisticsChange	5-34
DriverRxLookAheadChange	5-35
DriverManagement	5-36
Timeout Detection	5-37
DriverTxTimeout (RX-Net)	5-37
DriverAES / DriverCallBack	5-38
Removal	5-39
DriverRemove	5-39

---

## Chapter 6: TSM Procedures

Introduction	6-2
<TSM>GetNextSend	6-3
<TSM>GetRCB	6-4
<TSM>ProcessGetRCB	6-6
<TSM>FastProcessGetRCB	6-8
<TSM>RcvComplete	6-10
<TSM>FastRcvComplete	6-11
<TSM>RegisterHSM	6-12
<TSM>SendComplete	6-13
<TSM>FastSendComplete	6-14
<TSM>UpdateMulticast	6-15
RXNetTSMGetRCB	6-16
RXNetTSMRcvEvent	6-19
RXNetTSMFastRcvEvent	6-20

## Chapter 7: MSM Procedures and Macros

Introduction	7-2
MSMAlertFatal	7-3
MSMAlertWarning	7-4
MSMAlloc	7-5
MSMAllocPages	7-6
MSMAllocateRCB	7-7
MSMDisableHardwareInterrupt (macro)	7-9
MSMDriverRemove	7-10
MSMDoEndOfInterrupt (macro)	7-11
MSMEnableHardwareInterrupt (macro)	7-12
MSMEnablePolling	7-13
MSMEndCriticalSection (macro)	7-14
MSMFree	7-16
MSMFreePages	7-17
MSMGetCriticalStatus (macro)	7-18
MSMGetCurrentTime (macro)	7-19
MSMGetHardwareBusType (macro)	7-20
MSMGetProcessorSpeedRating (macro)	7-21
MSMGetRealModeWorkspace (macro)	7-22
MSMInitAlloc	7-24
MSMInitFree	7-25
MSMParseCustomKeywords	7-26
MSMParseDriverParameters	7-30
MSMPrintString	7-34
MSMPrintStringFatal	7-35
MSMPrintStringWarning	7-36
MSMPSemaphore (macro)	7-37
MSMReadEISACfg	7-38
MSMReadPhysicalMemory	7-39
MSMRealModeInterrupt (macro)	7-40

---

MSMRegisterHardwareOptions	7-41
MSMRegisterMLID	7-43
MSMRescheduleLast (macro)	7-44
MSMReturnDriverResources	7-45
MSMReturnNotificationECB (macro)	7-46
MSMFastReturnNotificationECB (macro)	7-46
MSMReturnRCB (macro)	7-47
MSMScheduleAESCallBack	7-48
MSMScheduleIntTimeCallBack	7-49
MSMServiceEvents (macro)	7-50
MSMServiceEventsAndRet (macro)	7-51
MSMSetHardwareInterrupt	7-52
MSMStartCriticalSection (macro)	7-53
MSMVSemaphore (macro)	7-54
MSMWritePhysicalMemory	7-55

## Chapter 8: Hub Management Interface

Introduction	8-3
Hub Hardware	8-4
Hub Management Software	8-6
Hub Management Objects	8-8
10BaseT Object Packages	8-9
Token-Ring Object Packages	8-11
Object Identifiers	8-14
10BaseT Object Identifiers	8-15
Token-Ring Object Identifiers	8-21
Information Tables	8-23
Hub Information Table	8-24
Group Information Table	8-28
Hub Management MLIDs	8-30
Implementing HMI Support in MLIDs	8-30
Implementing HMI Only	8-31
Managing External Hubs	8-31
HMI Mode Selection	8-32
Command Processing	8-33
Hub Command ECB	8-33
Hub Command Sequence	8-36
Design Tips and Notes	8-36
Notification Processing	8-37
Notification Types	8-37
Notification Generation Sequence	8-38
Notification ECB	8-39
Hub Management Pseudocode	8-41
DriverInit	8-41
DriverISR or DriverPoll	8-41
DriverManagement	8-42
References	8-44

---

## Chapter 9: Source Routing Support

Introduction	9-2
Driver Source Routing Support	9-2
Calling the Source Routing Module from DriverSend	9-3
Calling the Source Routing Module from DriverISR	9-4
Load Board Number	9-6
Unload Board Number	9-7
Clear Source Routing Table	9-8
Change Unknown Destination Address Route	9-9
Change General Broadcast Route	9-10
Change Multicast Broadcast Route	9-11
Change Broadcast Response Type	9-12
Change Source Routing Update Table Timer	9-13
Remove Node from Source Route Table	9-14
Loading ROUTE.NLM	9-15

## Appendices

### Appendix A: Building the HSM

Development Process	A-2
Creating the Source Files	A-2
Assembling the Source Files	A-2
Linking the Object Files	A-2
Loading the Driver	A-6

### Appendix B: The NetWare Debugger

Introduction	B-2
Invoking the Debugger	B-3
Debug Commands	B-4
Help	B-4
"." Commands	B-4
Breakpoints	B-4
Memory	B-6
Register Manipulation	B-8
Input/Output	B-9
Miscellaneous	B-10
Debug Expressions	B-12
Grouping Operators	B-13
Conditional Evaluation	B-13
Symbolic Information	B-14

---

## Appendix C: Installation Information File

Introduction . . . . .	C-2
Reserved Keywords . . . . .	C-3
Information File Overview . . . . .	C-4
General Syntax . . . . .	C-7
Language Translation . . . . .	C-9
Driver Section . . . . .	C-11
Driver Parameters . . . . .	C-16
Local Predefined Parameters . . . . .	C-28
Global Predefined Parameters . . . . .	C-32
Dependency Expressions . . . . .	C-33
Examples . . . . .	C-36
Server Driver Template . . . . .	C-41

## Appendix D: Media Frame Types

Frame and Protocol ID Table . . . . .	D-3
Frame Types . . . . .	D-4
802.2 Type I and Type II . . . . .	D-4
Default Frame Types . . . . .	D-5
Frame Formats . . . . .	D-5
Ethernet 802.2 . . . . .	D-6
Ethernet Raw 802.3 . . . . .	D-6
Ethernet SNAP . . . . .	D-7
Ethernet II . . . . .	D-7
Token-Ring 802.2 . . . . .	D-8
Token-Ring SNAP . . . . .	D-9
FDDI 802.2 . . . . .	D-10
FDDI SNAP . . . . .	D-11
PCN2 802.2 . . . . .	D-12
PCN2 SNAP . . . . .	D-13
RX-Net . . . . .	D-14

## Appendix E: LSL Support Procedures

Introduction . . . . .	E-2
LSLAddProtocolID . . . . .	E-3
LSLDeRegisterMLID . . . . .	E-5
LSLFastRcvEvent . . . . .	E-6
LSLFastSendComplete . . . . .	E-7
LSLGetMaximumPacketSize . . . . .	E-8
LSLGetSizedRcvECBRTag . . . . .	E-9
LSLHoldRcvEvent . . . . .	E-10
LSLRegisterMLIDRTag . . . . .	E-11
LSLReturnRcvECB . . . . .	E-13
LSLSendComplete . . . . .	E-14
LSLServiceEvents . . . . .	E-15
LSLUnBindThenDeRegisterMLID . . . . .	E-16



---

## Appendix F: OS Support Procedures

Introduction	F-2
AddPollingProcedureRTag	F-3
Alloc	F-4
AllocateMappedPages	F-5
AllocateResourceTag	F-6
AllocBufferBelow16Meg	F-7
CancelInterruptTimeCallBack	F-8
CancelNoSleepAESProcessEvent	F-9
CancelSleepAESProcessEvent	F-10
ClearHardwareInterrupt	F-11
CPSemaphore	F-12
CRescheduleLast	F-13
CVSemaphore	F-14
DeAllocateMappedPages	F-15
DeRegisterHardwareOptions	F-16
DisableHardwareInterrupt	F-17
DoEndOfInterrupt	F-18
DoRealModeInterrupt	F-19
EnableHardwareInterrupt	F-21
Free	F-22
FreeBufferBelow16Meg	F-23
GetCurrentTime	F-24
GetHardwareBusType	F-25
GetProcessorSpeedRating	F-26
GetRealModeWorkSpace	F-27
GetServerPhysicalOffset	F-29
OutputToScreen	F-30
ParseDriverParameters	F-31
QueueSystemAlert	F-35
ReadEISAConfig	F-37
ReadRoutine	F-38
RegisterForEventNotification	F-40
RegisterHardwareOptions	F-42
RemovePollingProcedure	F-43
ScheduleInterruptTimeCallBack	F-44
ScheduleNoSleepAESProcessEvent	F-46
ScheduleSleepAESProcessEvent	F-47
SetHardwareInterrupt	F-48
UnRegisterEventNotification	F-50

## Appendix G: Canonical and Non-Canonical Address Formats

Introduction	G-2
Address Format	G-2
Node Addresses	G-3
Specifying the Node Address Format	G-3
Media Headers	G-5

---

Raw Sends . . . . .	G-5
Functional Addresses . . . . .	G-5
Other Address Format Issues . . . . .	G-5

## Appendix H: Standard Message Definitions

Introduction . . . . .	H-2
HSM Messages . . . . .	H-3
General Driver Messages . . . . .	H-3
Ethernet Driver Messages . . . . .	H-4
Token-Ring Driver Messages . . . . .	H-5
PCN2 Driver Messages . . . . .	H-7
TSM Messages . . . . .	H-8
MSM Messages . . . . .	H-8
LSL Messages . . . . .	H-8
LANSUP Messages . . . . .	H-9
ODINSUP Messages . . . . .	H-9
EXOS Messages . . . . .	H-9

## Appendix I: Sample ODI Server Driver Template

Equates . . . . .	I-3
Structures . . . . .	I-3
Multicast Table Structure . . . . .	I-3
Adapter Data Space Structure . . . . .	I-4
Statistics Table . . . . .	I-4
Driver Parameter Block . . . . .	I-5
Adapter Data Space Template . . . . .	I-6
Frame Data Space Template (Configuration Table) . . . . .	I-6
DriverInit . . . . .	I-8
DriverSend . . . . .	I-10
DriverCallBack . . . . .	I-12
DriverISR . . . . .	I-14
DriverEnableInterrupt . . . . .	I-18
DriverDisableInterrupt . . . . .	I-18
DriverMulticastChange . . . . .	I-19
DriverPromiscuousChange . . . . .	I-20
DriverReset . . . . .	I-21
DriverShutdown . . . . .	I-22
DriverRemove . . . . .	I-23

## Appendix J: v3.11 HSM Conversion

Introduction . . . . .	J-2
Conversion Process . . . . .	J-2

<b>Index</b> . . . . .	<b>Index-1</b>
------------------------	----------------

---

## Preface

This document provides the information necessary to develop the Hardware Specific Module (HSM) portion of a NetWare server driver. Drivers written using the information in this document conform to the Open Data-Link Interface (ODI) specification.

Because the ODI specification provides many powerful features, drivers are now more complex than they were in previous Novell specifications (NetWare v2.x, etc.). Novell has provided a LAN driver development toolkit containing the Media Support Module (MSM) and the Topology Specific Modules (TSMs). These modules allow the developer-written portion of the LAN driver, the HSM, to be as simple as possible and still retain all the features required by the ODI specification.

This document does not describe the full ODI specification, but explains the development of a driver using the Novell-provided development modules.

This document is organized as follows:

**Chapter 1** describes the NetWare environment. It consists of a brief look at the ODI architecture and introduction to ODI LAN driver structure.

**Chapter 2** provides an overview of the server driver HSM and its required functions.

**Chapter 3** describes the data structures and variables that the HSM developer must define.

**Chapter 4** describes the data structures and variables provided by the MSM and TSM for HSM development.

**Chapter 5** describes the procedures that the HSM developer must provide.

**Chapter 6** describes the TSM procedures available for HSM development.

**Chapter 7** describes the MSM procedures available for HSM development.

**Chapter 8** describes how to implement Hub Management support in the driver.

**Chapter 9** is an explanation of the source routing support of ROUTE.NLM.

**Appendix A** describes the process of assembling, linking, and loading an ODI NetWare server LAN driver.

**Appendix B** contains information on using the NetWare integrated debugger.

---

**Appendix C** describes the syntax required for the driver installation information file used by the Install utility to simplify driver setup and configuration.

**Appendix D** contains a table of defined frame types and protocol identifiers. It also provides information and diagrams for the different frame formats.

**Appendix E** describes the Link Support Layer (LSL) procedures available to the developer.

**Appendix F** describes the NetWare operating system procedures available to the developer.

**Appendix G** describes the use of canonical and non-canonical address formats in the driver.

**Appendix H** contains lists of standardized message numbers and their associated text strings. Message string standardization in drivers simplifies maintenance and provides for foreign-language enabling.

**Appendix I** contains the listing of a sample template for a NetWare ODI server driver HSM.

**Appendix J** contains brief instructions on the major points for converting a v3.11 HSM to a v4.0 HSM.

---

## Prerequisites

The developer should be experienced with assembly language programming for the Intel family of microprocessors and have a sound understanding of re-entrant coding, event-driven systems, and interrupt-driven device drivers. The developer should also be familiar with the 80386 microprocessor.

The NetWare operating system runs in both real and protected modes. LAN driver developers need to understand protected mode issues.

## Manual Conventions

All numbers in this document are decimal unless otherwise specified. Hexadecimal numbers are identified by a trailing 'h' (i.e. FFh). Where bit fields within a byte are specified, bit 0 is assumed to be the low-order bit.

The following data types are defined:

<b>byte</b>	1 byte unsigned integer
<b>char</b>	1 byte ASCII character
<b>offset</b>	32-bit non-segmented address

Note that numeric fields composed of more than 1 byte can be in one of two formats: high-low or low-high. High-low numbers contain the most significant byte in the first byte of the field, the next most significant byte in the second byte, and so on, with the least significant byte appearing last. Low-high numbers are stored in exactly the opposite order. The Intel microprocessors store numbers in low-high order.

