REV B reflects several changes made to the software. (same)

WHERE USED: LIST BELOW ALL DOCUMENTS, PARTS AND ASSEMBLIES THAT ARE AFFECTED BY THIS CHANGE ORDER.

<table>
<thead>
<tr>
<th>DOC/PART/ASSY NUMBER</th>
<th>CUR REV</th>
<th>NEW REV</th>
<th>TITLE/DESCRIPTION</th>
<th>MATERIAL DISPOSITION CODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>VD9950914-00</td>
<td>A</td>
<td>B</td>
<td>OD/OT Driver Guide</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 5 5 5 5 5</td>
</tr>
</tbody>
</table>

Instructions:
Rework VD9950914-00 Rev A to Rev B

Change pages reproduced in house.

Please Note:
1) Replace GREEN SHEETS with PRESENT INDEX TABS.
2) Replace Goldend with Software Registration Card.
3) Replace Blue Sheet with Acknowledgement Card.
Thank you for your purchase of an Emulex optical subsystem. We have included a set of two software packages to assist you in making the best use of your optical subsystem under VMS. These software packages center around two device drivers: the OD (optical disk) and OT (optical tape) drivers. This cover letter is to help you decide which of the two software packages would be best suited to your environment and application needs.

Both of the software packages include a driver that remaps the blocks on the optical medium to hide the write-once nature of the optical drive. The OD driver makes the optical drive appear to the system as a magnetic (read/write) disk. Its main advantages are the ability to support optical and magnetic drives on the same controller, and the ability to randomly access files and records on the disk. The OT driver makes the optical drive appear to the system as a magnetic tape drive. Its main advantage is the ability to support multiple media in a backup save set: that is, to fill up one medium and have the save set continue on the next medium.

Both of the software packages can be used on the same system on two different controllers, or on the same system at different times (though this requires rebooting the system to change drivers). The media produced by either package cannot be read by the other package. We anticipate that most customers will use only one of the packages; to determine which software package most suits your needs, consider the following factors:

- Do you need to run optical and magnetic drives from the same Emulex controller? If so, you will need the OD driver.
- Do you need the ability to continue a backup save set from one optical medium to another? If so, you will need the OT driver.
- Do you need random and/or simultaneous multi-user access to your data? If so, you will need to use the OD driver and the DEC file system. The OT driver, being a tape emulation, only allows one user to mount the device at a time, and only allows sequential access to the files on tape.

The manuals provided will give a fuller description of the operation of the OD and OT software packages. Please read the information contained in the tabbed section, SOFTWARE WARRANTY AND INFORMATION. If you have any questions, please contact your local sales representative or Emulex Technical Support at 714-662-5600.
EMULEX OD/OT DRIVER
LICENSED SOFTWARE LIMITED WARRANTY

Media and Operation Limited Warranty. Emulex Corporation ("LICENSOR") warrants that, for a period of ninety (90) days from the date of shipment to the original LICENSEE ("LICENSEE"), the tape or disk media on which the Licensed Software ("SOFTWARE") is furnished will be, under normal use and service, free from defects in materials and workmanship. The warranty is limited to operation of the Software with only the particular hardware, and only those particular computer operating systems expressly supported by LICENSOR. LICENSOR does not warrant that the operation of the Software will be adequate for LICENSEE’s particular use, or will be uninterrupted or error free.

Liability and Remedy Limitations, Disclaimer. LICENSOR’s entire liability and LICENSEE’s exclusive remedy under this warranty, which is subject to LICENSEE providing LICENSOR with a written description of the defects, will be to use reasonable best efforts to remedy defects covered by this warranty within a reasonable period of time or, at LICENSOR’s option, to refund the amounts paid by LICENSEE and terminate any further responsibility. LICENSOR shall not be required to provide on-site installation or integration for any Software revisions made hereunder.

EXCEPT FOR THE ABOVE EXPRESS LIMITED WARRANTIES, LICENSOR MAKES AND LICENSEE RECEIVES NO WARRANTIES, EXPRESS, OR IMPLIED, STATUTORY, OR OTHERWISE COMMUNICATED TO LICENSEE, AND LICENSOR SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL LICENSOR BE LIABLE TO LICENSEE FOR ANY SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES, INCLUDING ANY LOST PROFITS OR LOST SAVINGS ARISING OUT OF THE USE OR INABILITY TO USE SUCH SOFTWARE, EVEN IF LICENSOR OR AN AUTHORIZED DEALER OR DISTRIBUTOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY OTHER PARTY.

SOME STATES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES SO THE ABOVE MAY NOT APPLY TO YOU.

Transfer Restrictions. The benefits of this warranty are not transferable to any third party without the prior written consent of LICENSOR.

Miscellaneous. This warranty supersedes any representations or prior agreements on this subject, and may be amended only in a writing specifically denoted as an amendment to this warranty, and signed by LICENSOR. This warranty is governed by the laws of the State of California.

Should you have any question concerning this warranty or if you desire to contact LICENSOR for any reason, please contact in writing: Emulex Corporation, 3545 Harbor Boulevard, Costa Mesa, CA 92626.

(8/88)
This is a license agreement between you, the ("LICENSEE"), and Emulex Corporation, the ("LICENSOR") concerning the Emulex OD/OT Driver Software ("Software"). BY LOADING OR USING THIS SOFTWARE YOU ARE AGREEING TO BECOME BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, DO NOT LOAD OR USE THE SOFTWARE, AND PROMPTLY RETURN ALL ITEMS (INCLUDING DISKS OR TAPES, WRITTEN MATERIALS) FOR A FULL REFUND.

Grant of License. Emulex, as LICENSOR, grants to you, the LICENSEE, a nonexclusive right to use and display this copy of the SOFTWARE on a single CPU at a single location. This License Grant applies only to the particular version of the Software specified on the License Grant, and does not allow loading or use of any other version or release of the Software.

Ownership of Software. LICENSEE owns the magnetic or other physical media on which the Software is originally or subsequently recorded or fixed, but LICENSOR retains title and ownership of the Software recorded on the original media copy(ies), as provided to LICENSEE, or subsequent copies of the Software, regardless of the form or media in or on which the original and other copies may exist. This LICENSE is not a sale of the original Software or any copy.

Copy Restrictions. The Software and the accompanying written materials are copyrighted. Unauthorized copying of the Software or the written materials is expressly forbidden. LICENSEE may be held legally responsible for any copyright infringement that is caused or encouraged by a failure to abide by the terms of this agreement.

Use Restrictions. LICENSEE may load the Software onto a computer system [including its disk or tape storage device(s)]. LICENSEE may create backup copies for its own use in the case of the loss of other copies of the Software. LICENSEE agrees to maintain all Emulex copyright and other markings on any copy made of the Software. LICENSEE agrees not to otherwise use, load, copy or modify the Software.

Transfer Restrictions. LICENSEE agrees that it will not transfer this Software to anyone without the prior written consent of LICENSOR. Any authorized transferee of the Software shall be bound by the terms and conditions of this agreement, and any applicable License Agreement. In no event may LICENSEE transfer, assign, rent, lease, sell or otherwise dispose of the Software on a permanent or temporary basis except as expressly provided in this agreement.

Update Policy. LICENSOR may create, from time to time, updated versions as new releases of the Software. At its option LICENSOR may provide updates to LICENSEES who have paid an update fee. A SEPARATE LICENSE GRANT FOR EACH INDIVIDUAL PRODUCT, SPECIFYING THE PARTICULAR RELEASE VERSION OF THE SOFTWARE, IS REQUIRED BEFORE THE USER MAY LOAD OR USE THAT VERSION OF THE SOFTWARE.
Warranty. The warranty for the Software is limited to that expressly stated in the "Emulex Software Limited Warranty" published for the particular Software release version, and subject to all stated exclusions in that warranty.

U.S. Government Restricted Rights. This Software and documentation is provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subdivision (b)(3)(ii) of the Rights in Technical Data and Computer Software clause at 252.227-7013 Contractor/manufacturer is Emulex Corporation, 3545 Harbor Boulevard, Costa Mesa, CA 92626.

Miscellaneous. This agreement supersedes any prior agreements or representations on this subject, and may be amended only in a writing signed by Emulex. This agreement is governed by the laws of the State of California. This agreement is effective until terminated, and will automatically (without notice from Emulex) terminate if User fails to comply with any provision, at which time User shall destroy all written materials and software copies.

Should you have any question concerning this agreement, or if you desire to contact Emulex for any reason, please contact in writing: Emulex Corporation, 3545 Harbor Boulevard, Costa Mesa, CA 92626.
ODDRIVER
SOFTWARE PRODUCT DESCRIPTION

Description. ODDRIVER is an optical disk device driver that runs under the VMS and MicroVMS operating systems. ODDRIVER allows the operating system to use the optical disk as if it were a non-removable magnetic disk device: that is, ODDRIVER responds to disk-specific I/O requests, reading the writing and optical disk as necessary, and returning status comparable to that of a magnetic disk.

Features:

- Supports the VMS Files-11 disk structure level 1 and level 2, including the initialization, re-initialization, mounting, and dismounting of volumes, and the creation, reading, writing, and deletion of disk files.

- Supports user utilities that issue virtual, logical, and physical I/O requests directly to the device.

- Includes a configuration utility, ODCONFIG, that allows the user to determine the memory allotment to ODDRIVER.

- Includes a recovery utility, ODRECOVER, to allow the user to recover from a system or subsystem error and continue to use the volume (not all data may be recoverable).

- Supports a historical mount capability to allow the user to read the volume as it existed at a specific date and time.

Hardware Required. ODDRIVER works with Emulex optical subsystems consisting of a UC04/MI or UC14/MI and any optical drive available from Emulex.

Software Required. ODDRIVER works with VMS and MicroVMS, version 4.4 and above.

Warranty and Exclusions. The standard Emulex software warranty is provided by Emulex with the purchase of a license for the product. The warranty is described in detail in the documentation accompanying the product.

Emulex does not warrant that the software licensed shall be error-free, that the software shall operate with any hardware and software other than as specified in this Software Product Description, or that the software shall operate with any user-written program without accommodations in the user’s code.

(8/88)
Description. OTDRIVER is an optical disk device driver that runs under the VMS and MicroVMS operating systems. OTDRIVER allows the operating system to use the optical disk as if it were a TS11 tape device: that is, OTDRIVER responds to tape-specific I/O requests, reading and writing the optical disk as necessary, and returning status comparable to that of a TS11 tape controller.

Features:

- Supports the VMS commands INITIALIZE, MOUNT, and DISMOUNT, and the VMS utilities BACKUP, COPY, and EXCHANGE.
- Supports user utilities that issue I/O requests directly to the tape device.
- Includes a configuration utility, OTCONFIG, that allows the user to determine the memory allotment to OTDRIVER.
- Includes a recovery utility, OTRECOVER, to allow the user to recover from a system or subsystem error and continue to use the volume (not all data may be recoverable).
- Supports a historical mount capability to allow the user to read the volume as it existed at a specific date and time.

Hardware Required. OTDRIVER works with Emulex optical subsystems consisting of a UC04/MO or UCl4/MO and any optical drive available from Emulex.

Software Required. OTDRIVER works with VMS and MicroVMS, version 4.4 and above.

Warranty and Exclusions. The standard Emulex software warranty is provided by Emulex with the purchase of a license for the product. The warranty is described in detail in the documentation accompanying the product.

Emulex does not warrant that the software licensed shall be error-free, that the software shall operate with any hardware and software other than as specified in this Software Product Description, or that the software shall operate with any user-written program without accommodations in the user's code.
The Optical Disk (OD) and Optical Tape (OT) Drivers optical software update service entitles the subscriber to receive any updates and new revisions to the software that may be released during a specific 12-month period.

Purchase of an Emulex LX400-series optical disk subsystem includes the update subscription for the first 12-month period, beginning with the date of purchase. Subsequent renewals of the subscription must be purchased prior to expiration; each renewal extends the service for an additional 12-month period, commencing on the anniversary of the original ship date.

An Acknowledgement Card is delivered as evidence of enrollment in the subscription program. It serves to indicate that the customer’s Purchase Order for the software update service (or for products which include the update service) was received and processed. This procedure also incorporates the User Registration and Renewal Reminder program, which is intended to remind users when the renewal date is approaching.

NOTE

Users who choose not to renew the subscription are not entitled to receive new revisions of the software, including revisions needed in order to remain compatible with new releases of DEC’s VMS operating system software. Non-subscribers who later determine that a new revision of OD and/or OT Driver(s) are necessary will purchase a new software license at the future list price.

Contact Emulex Sales or Order Administration for current pricing and ordering information for the OD and OT Driver optical software update service subscription.
## Engineering Change Order

**No.** 5151  **Date** 18 Oct 88

### Equipment Affected:

OD and OT Drivers

### Effectivity:

Initial Release

### Reason for Change:

Initial Release

### Document Number

<table>
<thead>
<tr>
<th>REV</th>
<th>Title/Description</th>
<th>Next Order</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>OD and OT Drivers User's Guide</strong></td>
<td>5 4 4 4 4 4</td>
</tr>
<tr>
<td></td>
<td><strong>OD/OT Acknowledgement Card</strong></td>
<td>5 4 4 4 4 4</td>
</tr>
</tbody>
</table>

**Instructions:**

Refer to the Assembly Sheet for placement of manuals, cards, tabs, and legal documentation.

Print and distribute VD9950914-00 Rev A and VD9952701-00 Rev A.

Reproduced in house.

### Disposition Note:

**Originator**
L. Bridges

**Design Services**

**Project Engineer**

**Manufacturing**

**Quality Control**

**Field Service**

**Test Engineering**

**Certification**
# TABLE OF CONTENTS

## SIX  INTRODUCTION

6.1 OVERVIEW ................................................................................. 6-1
6.2 DESCRIPTION OF THE OT DRIVER .............................................. 6-1
6.2.1 Advantages of the OT Driver .................................................. 6-4
6.2.2 Disadvantages of the OT Driver .............................................. 6-4
6.2.3 Software Components of the OT Driver ................................. 6-5
6.3 DISTRIBUTION MEDIA ............................................................... 6-5
6.4 COMPATIBILITY AND REQUIREMENTS ....................................... 6-6
6.5 RELATED DOCUMENTATION ....................................................... 6-6

## SEVEN  OT DRIVER INSTALLATION

7.1 OVERVIEW ................................................................................. 7-1
7.2 CONVENTIONS AND ABBREVIATIONS ........................................ 7-1
7.3 INSTALLATION PROCEDURE ...................................................... 7-1
7.4 CREATING THE CONNECTOT.COM FILE ................................. 7-2
7.5 USING SYSGEN TO CONFIGURE OT DEVICES ............................ 7-3

## EIGHT  THE OTCONFIG UTILITY

8.1 OVERVIEW ................................................................................ 8-1
8.2 OTCONFIG COMMANDS ............................................................. 8-1
8.2.1 EXIT Command ................................................................ 8-1
8.2.2 HELP Command ................................................................ 8-1
8.2.3 SET Command ................................................................ 8-2
8.2.4 SHOW Command ................................................................ 8-3
8.3 OTCONFIG MESSAGES .............................................................. 8-3
8.3.1 Possible OTCONFIG Messages ............................................. 8-3
8.3.2 Status of the Specified OT Device ........................................ 8-5

## NINE  THE OTRECOVER UTILITY

9.1 OVERVIEW ................................................................................ 9-1
9.2 OTRECOVER OPERATION .......................................................... 9-1
9.3 OTRECOVER MESSAGES .......................................................... 9-2
COMPATIBILITY

10.1 OVERVIEW .................................................. 10-1
10.2 OPERATIONAL DIFFERENCES FROM A TAPE SUBSYSTEM .......... 10-1
10.2.1 Command Execution Times .................................. 10-1
10.2.2 Insuring Data Integrity with Writecheck ...................... 10-1
10.2.3 Pre-Initializing Virgin Disks ................................ 10-2
10.2.4 No Standalone BACKUP .................................... 10-2
10.3 I/O FUNCTION CODES ......................................... 10-3
10.4 FUNCTION CODE MODIFIERS .................................... 10-6
10.5 RETURN STATUS ................................................ 10-7

SAMPLE INSTALLATION ............................................. C-1

SAMPLE CONNECTOT.COM FILE ..................................... D-1

TABLES

2-1 OT Driver Distribution Media .................................. 6-6
2-2 Optical Tapes Supported by OT Driver Software .................. 6-7

FIGURES

2-1 Normal VMS Software Components (without the OT Driver) .... 6-2
2-2 VMS Software Components with the OT Driver ................. 6-3
6.1 Overview

This manual is a guide to the installation and use of the VMS Optical Disk Driver (OT Driver) on Digital Equipment Corporation (DEC) VAX and MicroVAX computers. It consists of the following sections:

- Section Six (Introduction) presents an overview of the OT Driver.
- Section Seven (Installation) gives installation instructions for OT Driver.
- Section Eight (OTCONFIG) describes the OT Driver configuration utility and its commands.
- Section Nine (OTRECOVER) describes the OT Driver media recovery utility and its commands.
- Section Ten (Command Compatibility) describes the I/O functions, modifiers, and status compatibility issues between the OT Driver and the DEC DU/PU class and port drivers.
- Appendix C (Sample Installation) shows an example of the OT Driver installation procedure.
- Appendix D (Sample CONNECTOT.COM File) provides an example of the command procedure required to connect the OT Driver. The example comes from the sample installation in Appendix C.

6.2 Description of the OT Driver

Optical disks supported by the OT Driver provide high capacity-storage on removable, reasonably priced media, but with the restrictions of write-once technology. This means that the medium cannot support the disk file structure of any DEC operating system. This normally requires the application programmer to assume the responsibility for maintaining the integrity of blocks on the medium. The OT Driver relieves the application programmer of this responsibility.

The "optical tape" driver supports standard tape QIO functions, and remaps those functions to Mass Storage Control Protocol (MSCP) disk functions. This causes the logical (OT) device to appear as a tape drive, both to the VMS operating system and the system and application programs. It also allows the user to run BACKUP, COPY, etc. In addition, the user can run applications that use the Magnetic Tape Ancillary Control Process (MTACP) for file-structured use of the "tape," while the optical disk provides such advantages as faster positioning and higher capacity.
Normal VMS disk operations, i.e., without the OT Driver, utilize the software components illustrated in Figure 2-1.

![Diagram of Normal VMS Software Components (without the OT Driver)](VD994-1791)

Requests for tape commands, such as reading, writing, spacing records and files, and rewinding, are sent directly through the tape driver to the tape controller. Requests for file handling such as creating, modifying, and closing files are sent to the magtape ACP for servicing. The ACP keeps track of the ANSI structure of the tape, and translates high-level requests (such as "open a file") to the lower-level requests for reading, spacing, and writing data. The ACP then sends these lower-level requests to the tape driver, which sends them on to the tape controller.

With the addition of the OT Driver, requests to the OT tape device utilize the software components illustrated in Figure 2-2.
The OT Driver accepts all the commands supported by the DEC TS11 subsystem. Requests for ACP functions, such as IO$ ACCESS and IO$ CREATE are routed to the magtape ACP as with the TS11 driver. Requests for reading and writing utilize the same Function Decision Table (FDT) routines that the TS11 driver uses. This makes the function validation and preprocessing identical to that of the TS11 driver. Once a tape command has been broken down to its most basic representation, it is executed by a combination of tasks from two categories:

1. Performance of optical disk I/O operations
2. Reading and modifying the Tape Structure Table (The Tape Structure Table describes how the tape data has been stored on the disk.)

Advantages of the OT Driver

The OT Driver offers the following advantages:

1. Using the disk as a tape device allows the optical disk to be used as a viable BACKUP medium. Without intelligent use of the medium, the optical disk could only hold one BACKUP save set per optical medium. With OT Driver, multiple save sets can be written onto a single optical disk, and/or save sets can extend across multiple output media if required, as with a tape backup.

2. The OT Driver is much simpler and more software transparent than an optical disk that emulates a magnetic disk using a new disk ACP. Also, the tape structure tables that are stored on the disk take up much less disk overhead than the corresponding disk directories and file headers for a disk ACP.
3. The OT Driver is very efficient, doing direct I/Os to and from the buffers of the calling process.

4. The OT Driver supports all standard tape QIO functions, allowing it to support the VMS programs BACKUP, COPY, and EXCHANGE without modification. It also supports any user programs that wish to address the device as a tape drive.

5. The OT Driver maintains the "tape" structure, including the size of the records and placement of file marks, in memory. Thus REWIND, SKIP RECORD, and SKIP FILE commands are almost instantaneous, since they require no actual data transfer. This allows very fast access to the save sets on the disk without having to read through the whole disk (a true tape drive would have to read through the tape). This saves both time and CPU overhead.

6.2.2 Disadvantages of the OT Driver

The optical tape approach does not provide the random access functionality of a disk emulation. It does provide very high capacity removable storage, with the file and save set flexibility required by archival storage applications.

The "tape" blocks that are written to the disk are not compressed. They are zero-filled to the block size required by the drive (512- or 1024-bytes). For example, writing an 80-byte ANSI header record actually fills a 512- or 1024-byte block. BACKUP defaults to 8192-byte blocks and COPY to 2048-byte blocks, so these programs do not waste data space. This minor problem is easily bypassed through correct selection of block size. The "tape" blocks are analogous to real tape drives, where larger block sizes produce more efficient use of tape.

6.2.3 Software Components of the OT Driver

The OT Driver contains the following software components:

1. The OT Driver: This is the device driver. It allows you to run SYSGEN to create a Unit Control Block (UCB) for the OTnn device, and to issue $QIO I/O requests to the device. The OT Driver calls standard Function Decision Table (FDT) routines within VMS, then reads the tape structure table, issues disk I/Os, and modifies the tape structure table.

2. OTCONFIG: This program allows the system manager to modify the following parameters that affect the mapping of data on the disk.
a. Optical disk block size  
b. The amount of memory allocated to the tape structure table  
c. Setting a special date to which to revert, i.e., to mount the volume and have it appear as it did at that date and time—the "Historical Mount" feature

3. OTRECOVER: This program is used in the event of a VMS system failure or subsystem failure, to make the optical medium usable again. If such a failure occurs while writing to the OTnn device, data blocks will have been written to the disk that are not yet recorded on the disk's mapping table. The disk map will not have been updated to reflect the space used on the disk.

The presence of these "hanging" data blocks prevents use of the remainder of the disk. The OTRECOVER program cannot recover these "hanging" data blocks, but it can revector around them to make the rest of the medium usable again. NOTE: Other mapping techniques are also susceptible to loss of data in the event of system failure during a write operation.

4. KITINSTAL.COM: This is the command procedure that installs and builds the three optical tape programs listed above. It is compatible with the DEC VMSINSTAL command procedure.

6.3 Distribution Media

Table 2-1 lists the OT Driver distribution media for Versions 4.X and Version 5.0.

Table 2-1. OT Driver Distribution Media, Versions 4.X and 5.0

<table>
<thead>
<tr>
<th>Emulex P/N Version 4.X</th>
<th>Emulex P/N Version 5.0</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VD9960714-00</td>
<td>VD9960719-00</td>
<td>5.25-inch floppy diskette (MicroVAX only)</td>
</tr>
<tr>
<td>VD9962014-00</td>
<td>VD9962019-00</td>
<td>TK50 tape cartridge (MicroVAX only)</td>
</tr>
<tr>
<td>VD9960314-00</td>
<td>VD9960319-00</td>
<td>9-track 1600 BPI magtape</td>
</tr>
<tr>
<td>VD9960514-00</td>
<td>VD9960519-00</td>
<td>RX01 8-inch floppy diskette (1178x only)</td>
</tr>
<tr>
<td>VD9960414-00</td>
<td>VD9960419-00</td>
<td>TU58 cartridge (11750 only)</td>
</tr>
</tbody>
</table>
6.4 Compatibility and Requirements

The OT Driver is compatible with the MicroVAX II, VAX 11750, 11780, 11785, and certain CPUs in the VAX 8x00 family. If the host computer is a MicroVAX II, it must have a Q-Bus with an Emulex UC04/MO host adapter connected to an optical drive. If it is any other VAX, it must have a Unibus adapter with an Emulex UC14/MO host adapter plugged into the Unibus and connected to an optical drive. Table 2-2 lists the optical drives that the OT Driver software supports.

Table 2-2. Optical Drives Supported by OT Driver Software

<table>
<thead>
<tr>
<th>Mfg</th>
<th>Model</th>
<th>Sector Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimem</td>
<td>1000</td>
<td>1024 bytes</td>
</tr>
<tr>
<td>OSI</td>
<td>LD 1200</td>
<td>1024 bytes</td>
</tr>
</tbody>
</table>

OT Driver software is compatible with the DEC MicroVMS and VMS operating systems, versions 4.4 and above. VMS systems include the BACKUP program and the magtape ACP by default. The Program Development (SYSP) option of MicroVMS is necessary for full ANSI magtape support, as the magtape ACP is included in that option.

Media created under the OT Driver are not compatible with media created under any other optical software.

If you have an MSCP disk server enabled, ensure that the UC controller is set to a non-standard address. This prevents the server from attempting to utilize the optical device as a magnetic disk.

6.5 Related Documentation

The following related documents are available:

- UC04/MO Host Adapter Manual, P/N UC0451003-00
- UC14/M Host Adapter Manual, P/N UC1451003-00

They may be ordered directly from:

Emulex Corporation  
3545 Harbor Boulevard  
Costa Mesa, CA 92626  
(714) 662-5600  TWX 910-595-2521
Section 7

OT DRIVER INSTALLATION

7.1 Overview

This section explains how to install the OT Driver under VMS and MicroVMS. The term VMS refers to both VMS and MicroVMS.

7.2 Conventions and Abbreviations

Operator input appears in boldface type, in order to distinguish it from operating system messages and prompts.

The symbol <return> signifies the carriage return key.

"OTnn" and "OTcu" refer to the VMS device mnemonic for the drive on your system, typically "OTA0".

7.3 Installation Procedure

This subsection gives the steps, including commands, required for OT Driver installation.

1. Log into the system manager’s account.

NOTE

Before invoking VMSINSTAL under MicroVMS, you must install the System Programming Option (SYSP). To ensure that SYSP has been installed, check the system directory for the ACP of the Magnetic Tape (MTAAACP.EXE) and the system symbol table (SYS.STB) files. To make this check, use the following commands:

   $ DIR SYS$SYSTEM:SYS.STB < return >
   $ DIR SYS$SYSTEM:MTAAACP.EXE < return >

2. Mount the distribution medium and invoke the VMSINSTAL PROCEDURE:

   $ @SYS$UPDATE:VMSINSTAL * ddcu: < return >

   where "ddcu:" is the VMS device name for the device where you have loaded the distribution medium. If you do not specify the kit name (*) or distribution device (ddcu:), the VMSINSTAL command procedure will prompt you for them.
3. Supply answers requested by the Emulex installation procedure.

NOTE

OTDRIVER is not supported for operation on MicroVAX I systems. If you attempt to install the software on a MicroVAX I system, the following warning will print: The driver does not work if loaded on a MicroVAX I system. The installation will continue to completion, but will only allow you to install the software for transporting to another system.

If you are updating an existing OT Driver to a new version, the install procedure will delete the old files. Copy these files to another account if you wish to retain them.

7.4 Creating the CONNECTOT.COM File

The installation command procedure has the ability to build a command procedure in the system manager’s account in order to do the following:

a. Load the OT Driver automatically
b. Connect the units under SYSGEN
c. Use the OTCONFIG utility to set up the OT Driver parameters (option)

The full file specification for this file is SYS$MANAGER:CONNECTOT.COM.

After the installation command procedure has linked the driver and utility images, the command procedure will ask the following:

* Do you wish to create the CONNECTOT file [YES]?

Responding “N” or “NO” to this question will skip the creation of the CONNECTOT.COM file, including skipping the next prompts. Take the default response of “YES” to create the CONNECTOT command procedure, unless you have already been through the Emulex product installation and have created this command procedure.

If you wish to create the CONNECTOT file, the command procedure will prompt you for information regarding the OTAn: devices. Only the first optical controller will be configured automatically by the command procedure. If there are multiple controllers with optical drives that will be controlled by the the OT Driver software, you must edit the file SYS$MANAGER:CONNECTOT.COM, and insert additional lines for the additional devices.

The command procedure will prompt you for the CSR address and interrupt vector of the first controller on the Q-Bus or Unibus as follows:

* Enter octal CSR bus address for the OTAn: units [772150]?
* Enter octal vector address for the OTAn: units [154]?
The default answers for these questions are the default CSR and vector for the first MSCP controller on the bus. Valid responses are either that CSR and vector, or CSR addresses in the range (octal) 760334-760370 and vector addresses in the range (octal) 300-340. The command procedure will next prompt as follows:

* How many OTAn: units do you wish to configure (1-8) [1]?

Each unit that you specify will have a corresponding SYSGEN CONNECT statement added to the CONNECTOT command procedure. For each unit that you specify, you will be asked the following:

* Does unit OTAn: have a 1024-byte minimum write size [YES]?

Optical disks have either a 512-byte or 1024-byte minimum write size. If the optical disk that corresponds to the unit number has a 1024-byte minimum write size, answer "YES." Each unit to which you answer "YES" will have a corresponding OTCONFIG SET OTAn/BLOCK_SIZE=1024 statement added to the CONNECTOT command procedure. See subsection 1.4 for block size information on Emulex-supplied optical drives.

Appendix C of this document contains a sample run of the installation command procedure. Appendix D contains a sample CONNECTOT.COM file.

### Using SYSGEN to Configure OT Devices

OTDRIVER controls devices that are normally controlled by the DEC DU/PU MSCP disk class and port driver. Two steps are required to allow this. First, the SYSGEN program must not be allowed to configure the devices as DU devices; second, the SYSGEN program must be used to configure the OT devices.

These steps are accomplished by changes to the system configuration command file SYS$MANAGER:SYCONFIG.COM. This command file is used to make sitespecific changes to the device configuration. Edit this command file and insert the following lines:

```sh
$ STARTUPS$AUTOCONFIGURE ALL = = 0
$ RUN SYS$SYSTEM:SYSGEN
AUTOCONFIGURE ALL/EXCLUDE=(PUx,DUx)
EXIT
$ @SYS$MANAGER:CONNECTOT
```

Line 1 sets a global symbol to keep the system startup command procedure from doing a SYSGEN AUTOCONFIGURE ALL to configure all devices. Line 2 starts the SYSGEN program. Line 3 specifies auto-configure for all devices except those excluded by the EXCLUDE switch. For example, if the devices to be controlled by the OT Driver would normally show up as DUC7: and DUC8: (on the third MSCP controller on the bus), substitute the letter "C" for the "x" in the third line above.
Line 5 of the changes invokes the CONNECTOT.COM command file that was built during the installation. For each device to be connected, CONNECTOT.COM will execute a SYSGEN command of the following form:

```
SYSGEN > CONNECT  
OTAn/ADAPTER = UB0/CSR = %0772150/VECTOR = %0154
```

The first unit to be connected will also cause the controller to go through MSCP initialization. This raises the Interrupt Priority Level (IPL) to 31 (maximum) for a period of 3-15 seconds, suspending all other activity on the system for that period of time. You may wish to use the "SET TIME" command after executing CONNECTOT.COM, to make sure the system time is set correctly.
8.1 Overview

The OTCONFIG utility is used to set software parameters that affect the mapping of data onto the optical disk and to display the current settings of those parameters. The device name "OTA0:" is used in the examples in this section.

8.2 OTCONFIG Commands

OTCONFIG is installed in the SYS$SYSTEM account. It is an interactive utility that accepts the commands EXIT, HELP, SET, and SHOW, and has the following prompt:

```
OTCONFIG >
```

Any of these commands can be abbreviated to letter(s) that make(s) them unique. The SET and SHOW commands require one parameter: the OTnn device. Where parameters are required, they must be entered on the same line as the command. OTCONFIG will not prompt for unspecified parameters, and it will print an error message if parameters are left unspecified.

Device names may be entered with or without the terminating colon (:) character.

8.2.1 The EXIT Command

The EXIT command terminates the OTCONFIG program, and returns control to the command interpreter. Control-Z (Z) also terminates the program.

8.2.2 The HELP Command

The HELP command prints a brief (one-screen) help message summarizing the format of the commands. It takes no parameters, and sublevels of help are not available.

8.2.3 The SET Command

The SET command modifies the software parameters that affect the mapping of data onto the optical device. It takes one parameter, the "OTnn" device name, and any combination of the following qualifiers:
OTCONFIG Commands

OTCONFIG > SET OTA0/BLOCK SIZE = 512
OTCONFIG > SET OTA0/BLOCK SIZE = 1024

This qualifier sets the minimum block size that the optical disk can write. All optical drives can read single 512-byte sectors and portions thereof. Some optical drives must write 1024-byte sectors, as they cannot individually address 512-byte sectors for a write. The default setting for this parameter is 512 bytes. If your optical drive has a 512-byte block size, you do not need to use this qualifier.

OTCONFIG > SET OTA0/BUFFER = 65536

This qualifier sets the amount of non-paged memory to allocate for the tape structure table when the device is mounted. The amount of data that can be stored on the device is affected by both the physical size of the device and the size of the tape structure table. When either the tape structure table or the physical device approaches capacity, end-of-tape status is returned to the user. A good rule of thumb is to allocate 200 bytes, plus 100 bytes for every file (ANSI file or BACKUP save set) that will be stored on the device at any one time.

OTCONFIG > SET OTA0/REVERT = 01-JAN-1988:10:00
OTCONFIG > SET OTA0/NOREVERT

Data on an optical disk is vectored by the OT Driver, but it is never actually erased. Specifying the /REVERT= date qualifier will cause the device to revert to the last tape structure table that was written before the specified date and time.

The /REVERT date can be modified only while the device is dismounted. It will affect all mounts on the device until the device is dismounted and the /NOREVERT qualifier is used to turn off this feature. When a device is mounted, and a /REVERT date has been specified, the device will be software write-locked for the duration of the mount.

The SHOW Command

The SHOW command shows which parameters are set for the OTnn device. It has the following form:

OTCONFIG > SHOW OTA0

The SHOW command displays the OT Driver flags, the optical disk block size, and the buffer size settings. If the device is mounted and the Tape Structure Table is in memory, the SHOW command will display optical disk block usage information and buffer usage information.
8.3 OTCONFIG Messages

8.3.1 Possible OTCONFIG Messages

The following messages are possible from OTCONFIG:

%OTCONFIG-E-DEVMOUNTED, device is mounted

A SET command cannot be executed with the /BLOCK_SIZE or /REVERT qualifier because the device is currently mounted; changing these parameters would corrupt data on the device.

%OTCONFIG-E-ERRGETDVI, error getting information on device

An error was encountered using either the $GETDVI (get device information) system service or the kernel mode routine that reads device information from the UCB. This message will be followed by a VMS error message explaining the error.

%OTCONFIG-E-ERRSETINF, error setting information on device

An error was encountered in the kernel mode routine that sets device information in the UCB. Use the SHOW command to ascertain the current settings of the device. This message will be followed by a VMS error message explaining the error.

%OTCONFIG-E-ILLDEVNAM, illegal device name, should be OTcu:

A SET or SHOW command was executed, and the device name specified was not of the form "OTcu:". Re-enter with a valid device name.

%OTCONFIG-E-INVBLKSIZ, invalid block size, must be 512 or 1024

A SET command with the /BLOCK_SIZE qualifier was executed, with a block size specified that was not 512 or 1024 bytes. Re-enter the command with a block size value of either 512 or 1024.

%OTCONFIG-E-INVBUFSIZ, invalid buffer size specified

A SET command with the /BUFFER qualifier was executed, with an invalid buffer size specified. The minimum size of the buffer is 4,096 bytes, and the maximum is 1,048,576 bytes. Re-enter the command with a buffer size within these bounds.
%OTCONFIG-E-NOCHANGES, no changes have been made to the device

An error was detected in the format of the SET command before any part of the command was executed, thus none of the command was executed. Correct the qualifier or parameter, and re-enter the command.

%OTCONFIG-E-NOQUAL, no qualifiers specified on SET command

The SET command was entered without any qualifiers. Re-enter the command with at least one qualifier.

%OTCONFIG-E-NOTSUPUV1, not supported on the MicroVAX I

OTDRIVER, and thus OTCONFIG, are not supported on the MicroVAX I processor. Using SYSGEN to connect the OT Driver on the MicroVAX I will result in the device being offline. Attempting to run the OTCONFIG program on the MicroVAX I will result in this error message.

%OTCONFIG-E-UNKNOWN, device state is unknown and may be corrupted

A catastrophic error was detected setting device information, and the data in the UCB for the device may be corrupted. Use the SHOW command to display the parts of the UCB that it shows. Reboot the system, and reload the OT Driver if the device continues to return errors.

### 8.3.2 Status of the Specified OT Device

The SHOW command will return the status of the specified OT device. The physical device name will be shown first, optionally followed by one or more flag bits. These flag bits are as follows:

- Device is in process of dismount.
- Device currently has maps loaded.
- Device is in process of mount.
- Device will revert to mount status from a specified date/time.
  - Revert date and time = dd-mmm-yyyy hh:mm:ss.hh
- Device has been written on since last MOUNT command.

If no flag bits are currently set, no message is printed. The optical disk block size is then printed, followed by the buffer size currently being used, and the amount of buffer requested for the next MOUNT command.
Optical disk block size = n bytes.
Buffer (non-paged pool) currently allocated = w bytes.
Buffer (non-paged pool) in use = x bytes.
Buffer is (non-paged pool) remaining = y bytes.
Buffer (non-paged pool) remaining for next mount = z bytes

The usage of the optical disk is summarized in the following messages:

Total blocks on device = x.
Blocks used on device = y.
Blocks free on device = z.

The block figures are all in 512-byte blocks, even if the /BLOCK_SIZE qualifier has been set to 1024 bytes.

The buffer information is printed to allow the system manager to more accurately estimate the amount of buffer required for the files on a given volume. If the amount of memory allocated is too large, non-page pool is wasted. If the amount of memory is too small, the buffer is filled and the OTDRIVER returns EOT (end-of-tape) status to the application. This allows the application to continue writing to another volume, though there may be remaining blocks on the first volume that cannot be accessed without re-initializing the volume.
9.1 Overview

The OTRECOVER utility is used to analyze and/or repair the "tape" structure on a disk following a system crash or fatal optical disk subsystem failure. The OT Driver keeps the Tape Structure Table (TST) in memory until a dismount or available command is issued to the OTnn tape unit.

At that time, the TST is written immediately after the last data block on the disk. A revector block is written immediately preceding the data blocks just written on the disk; space for the revector block is left by the OT Driver when the device is mounted. The revector block is used to point over that group of data blocks to the next TST.

In the event of a system crash or subsystem failure, data blocks to which no TSTs are pointing may have been written on the optical disk. This is because the data blocks are written as the commands are issued, but the TST is not written until the dismount.

The OTRECOVER utility will find the last valid TST, and duplicate it right after the "hanging" data blocks on the device. This returns the "tape" to the structure that it had immediately before the last mount, allowing the user to access the files already on the tape, and make use of the unused portion of the disk. The OTRECOVER utility cannot recover the blocks that are hanging.

9.2 OTRECOVER Operation

OTRECOVER requires almost no operator intervention. To use OTRECOVER, first ensure that the OTnn device is not mounted. Invoke OTRECOVER with the following command:

$ RUN SYSSYSTEM:OTRECOVER

OTRECOVER prints a header describing the program, then prompts as follows:

Do you wish to continue <No> ? YES<return>

Answer "Y" or "YES" to continue with the process. OTRECOVER prompts you as follows for the name of the device to analyze:

Name of tape (OTnn) device to check ? OTA0<return>
OTRECOVER does not prompt for any more input. OTRECOVER performs the following steps to analyze and/or repair the TST:

1. Read logical block 0, and check the information stored there to confirm that the initialization of the pack meets the following criteria:
   a. It has already been performed.
   b. It used the correct (OT Driver) software.
   c. It was performed with a compatible version of the OT Driver.

2. Starting at logical block 2, read in a revector block, and use it to point to a TST.

3. If the revector block is readable, and it checks out as a revector block, read in and verify the TST.

4. If the revector block is readable, but does not check out as a revector block, the disk is unrepairable, as there is no way to reuse the revector block.

5. If the revector block is unreadable, the last TST read is the correct one. Allocate one block following the last TST for the new revector block, and make sure that block is empty.

6. Starting just beyond the new revector block, start reading data blocks, until a read error is encountered. This read error signifies the end of the written data blocks on the disk.

7. If you reach the end of the usable part of the disk before you get the read error, the disk is not repairable, but the existing files on the "tape" should be readable.

8. If the read error occurs immediately after the last revector block, no new data blocks were written on the "tape." The disk does not need to be repaired.

9. If any hanging data blocks are found, read until the first unwritten block is found. Write the valid TST into the first free block, and write the new revector block into the pre-allocated block. The disk is now repaired.

9.3 OTRECOVER Messages

The following messages are possible from OTRECOVER (in the error messages, the abbreviation TST stands for "tape structure table," which is the overall table of tape information).
%OTRECOVER-E-BADVERSION, driver version mismatch

The volume to be analyzed was initialized under a version of the OT Driver that is higher (later) than the version of OTRECOVER. OTRECOVER cannot continue; install the latest version of the the OT Driver software and retry the operation.

%OTRECOVER-E-DEVMOUNTED, device is mounted

OTRECOVER cannot allocate and analyze the device because the device is already mounted.

%OTRECOVER-E-ERRALLOC, error allocating device

An error was encountered trying to allocate the OTnn device. This message is followed by the VMS message explaining the error, which is typically caused by another user having allocated or mounted the device.

%OTRECOVER-E-ERRASSIGN, error assigning device

OTRECOVER encountered an error assigning a channel to the OTnn device. This message is followed by a VMS message explaining the error.

%OTRECOVER-E-ERRREADINF, error reading information block

OTRECOVER encountered an error reading the device information block at Logical Block Number (LBN) of the volume. Either the volume has never been used, and the information block is blank (and thus unreadable), or the information block is corrupt, and the volume is not repairable.

%OTRECOVER-E-ERRREADTST, error reading tape structure table

OTRECOVER encountered an error while reading the TST for the device, at a block where it is known to have been written. The pointers to that block cannot be changed, and the volume is not repairable.

%OTRECOVER-F-ERRGETDVI, error getting device information on device

OTRECOVER encountered an error trying to issue a $GETDVI system service to get information on the OTnn device. This message is followed by a VMS message explaining the error.
%OTRECOVER-E-ERRWRTREV, error writing revector block

OTRECOVER encountered an error trying to repair the device. Specifically, it received an error writing the revector block to point at the new TST that was written. The volume is probably not repairable.

%OTRECOVER-E-ERRWRTTST, error writing tape structure table

OTRECOVER encountered an error trying to repair the device. Specifically, it received an error writing the new TST in an unused part of the device. If the error can be corrected (such as a write-protected drive), retry the operation. If it cannot, the volume is not repairable.

%OTRECOVER-F-ILLDEVNAM, illegal device name, must be "OTcu:"  

The specified device name was not in the form "OTcu" or "OTcu:". Restart the program and enter the correct device name.

%OTRECOVER-E-INSFRESPC, insufficient free space to repair volume

The specified volume has already been written past the EOT marker block; thus there is insufficient space to write the new TST. Repairing the device would only bring it to the point of the last successful dismount, which could be achieved using the /REVERT qualifier of OTCONFIG.

%OTRECOVER-E-INVINFBLK, invalid INF block structure

OTRECOVER read block zero of the device, expecting it to contain information about the driver and driver version that initialized the volume. The block was readable, but did not appear to be a valid information block. The volume was not initialized with the OT Driver, and OTRECOVER should not be used on the volume.

%OTRECOVER-E-INVREVBBLK, invalid REV block structure

OTRECOVER read a block, expecting it to be a revector block, but its contents do not match the structure of a revector block. The volume is corrupt and not repairable.

%OTRECOVER-E-INVTSTBLK, invalid TST block structure

OTRECOVER read a block, expecting it to be a TST block, but the contents of the block do not match the format of the TST. The volume is corrupt and not repairable.
%OTRECOVER-F-MAPLOADED, device currently has maps loaded

OTRECOVER cannot allocate and analyze the device because the device currently has map structures loaded in memory.

%OTRECOVER-S-NOFIXREQD, no tape structure repair required

OTRECOVER found no “hanging” data blocks after the last valid TST; thus no tape structure repair is required for this medium.

%OTRECOVER-E-NOTAPEDEV, device is not a tape device

The specified device is not understood by VMS to be a tape device. This error usually means that an incorrect device name is specified.

%OTRECOVER-F-NOTFIXABLE, this medium cannot be repaired

The previous error message was of a type that indicates this disk medium cannot be repaired by OTRECOVER.

%OTRECOVER-E-NOTOTDVOL, this is not an OT Driver volume

OTRECOVER successfully read the information (INF) block, and the block checked out as a valid INF block, but the driver field of the INF block indicated that the volume is not an OT Driver volume. It is an Emulex OD Driver volume, and should be examined with the ODRECOVER utility.

%OTRECOVER-E-REMOTE, device OTnn is not on the local node

OTRECOVER cannot allocate and analyze the device because the device is not on the local node.

%OTRECOVER-S-REPAIRED, tape structure table has been repaired

OTRECOVER has successfully written a new TST and a new revector block. The tape structure has been repaired, and the device can now be INITed or MOUNTed.

In addition to the status message above, OTRECOVER prints the date/time and size of each TST as it is read. This information can be used to determine the date/time combinations when the device was dismounted, for use with the OTCONFIG SET/REVERT command. OTRECOVER also prints the device usage statistics for the volume in the same format that the OTCONFIG SHOW command does.
Section 10
COMPATIBILITY

10.1 Overview

The OTDRIVER most closely emulates the DEC tape subsystem. It returns the drive type code of "Foreign tape 7" to keep the error formatting program from interpreting the error log as if it came from a DEC driver. The Function Decision Table (FDT) routines within the driver were patterned after those of the TS11 driver to ensure compatibility. Where device status for certain functions, e.g., read reverse into Beginning of Tape (BOT), was undocumented or ambiguously documented, every effort has been made to ensure that the OT Driver returns the same status that the TSDRIVER would in a comparable situation.

This section covers compatibility issues, which fall into the following four areas:

a. Operational differences
b. I/O function codes
c. Function code modifiers
d. Return status
e. Error Log Formatting

10.2 Operational Differences from a Tape Subsystem

The OT subsystem has some operational differences from a tape subsystem. This subsection summarizes the major ones.

10.2.1 Command Execution Times

INIT, MOUNT, and DISMOUNT may take up to seven minutes for OT devices. The OT Driver loads map information on a MOUNT command, and writes map information to the disk on a DISMOUNT command. The INIT command uses IO$ PACKACK and IO$ AVAILABLE as an implied mount and dismount. INIT and MOUNT may also include delays caused by executing the IO$ SENSEMODE command.

On the other hand, positioning commands, such as SET MAGTAPE/Skip and SET MAGTAPE/REWIND, takes almost no time at all, since they do not have to physically move tape.

10.2.2 Insuring Data Integrity With Writecheck

You should always mount OT devices with writechecking enabled. Use the following command:
$ MOUNT/DATACHECK = WRITE

This is because the optical disk, not the operating system, assumes responsibility for revectoring bad blocks. The optical disk can detect and correct errors only if a datacheck operation is performed on all write operations.

NOTE

Failure to mount the OT device /DATACHECK = WRITE results in unreliable data being written to the device, and can result in corrupt and lost data.

10.2.3 Pre-Initializing Virgin Disks

Issuing a read function to a virgin (never written) disk results in a fatal drive error, just as issuing a read function to a virgin tape will. To prevent this data error, initialize virgin disks with the following command and qualifiers:

$ INITIALIZE/OVERRIDE = (ACCESS, EXPIRATION, OWNER)

This is consistent with recommendations in the VMS documentation for initializing virgin tapes. If you are going to be writing a multi-volume BACKUP or COPY set to the OTnn: device, Emulex recommends that you pre-initialize all the volumes to be used, with the command shown above. This will make the volume check and initialize more efficient. The following DCL command and qualifiers allow the automatic initialization of continuation volumes in an ANSI tape volume set:

$ MOUNT/INITIALIZE = CONTINUATION/AUTOMATIC

These qualifiers can be used, but Emulex recommends the pre-initialization of all volumes to be used.

10.2.4 No Standalone BACKUP

Standalone SYSGEN configures the OTnn devices as OTnn devices; thus the OT Driver is not supported for use with standalone BACKUP.

10.3 I/O Function Codes

The OTDRIVER supports the function codes listed below. Some of them are not documented in the DEC VMS I/O User’s Guide, but were included in the OT Driver because they are in TSDRIVER. Variances between the OT Driver and TSDRIVER that might be perceptible to the user are documented here.
IO$ READVBLK
IO$ READLBLK
IO$ READFBLK

Read block functions: rejects odd byte counts with status SS$ IVBUFSIZE, except for a byte count of 65535, which is rounded down to 65534. If the IO$M DATACHECK function modifier is set, that qualifier is passed on in the optical disk command.

IO$ WRITEVBLK
IO$ WRITELBLK
IO$ WRITEFBLK

Write block functions: rejects odd byte counts with status SS$ IVBUFSIZE. All I/O operations are done as direct I/O from the user's buffer. Writing a record when not at the end of the volume invalidates any records or file marks that would be after the data block just written. If the IO$M_DATACHECK function modifier is set (it always should be), that qualifier is passed on in the optical disk command.

IO$ ACCESS
IO$ CREATE
IO$ DEACCESS
IO$ MODIFY

ACP functions: these functions are sent to the magtape ACP for interpretation. They are not handled directly by the OT Driver, but may be handled indirectly if the magtape ACP converts them into read, write, and space requests.

IO$ AVAILABLE

Available function: writes the TST to the disk if it has been modified since the last pack acknowledge. Issues an AVAILABLE command to the optical drive.

IO$ CLEAN

No operation.

IO$ DRVCLR

No operation.

IO$ DSE

Data security erase: returns invalid function code error; not supported by the TS11, and not logical for optical drives.
**I/O Function Codes**

**IO$ _NOP**

No operation.

**IO$ PACKACK**
**IO$ _READPRESET**

Pack acknowledge function: issues a pack acknowledge to the device. If the TST has not already been loaded from disk, the pointers to the TSTs are followed until the last (valid) TST is located. That TST is loaded into memory.

**IO$ _READHEAD**
**IO$ _WRITEHEAD**

These are not normal tape commands, and are not supported by the DEC TSDRIVER. They are used for unmapped reading and writing of specific disk blocks. They are used by the OTRECOVER utility only, and are not supported for use by any other program.

**IO$ _REREADN**
**IO$ _REREADP**

Reread next block and reread previous block: issues normal I/O without changing the logical tape position.

**IO$ _REWIND**
**IO$ _RECAL**

Rewind function: this function is very fast, since it requires only a status change in the OT Driver, and no real I/O operation.

**IO$ _REWINDOFF**
**IO$ _UNLOAD**
**IO$ _ACPCONTROL!IO$ _DMOUNT**

Rewind/dismount function: this function causes the current TST to be written from memory to the optical disk, and a new revector block to be rewritten. The new TST is written only if it has been modified since the last IO$ PACKACK command. This command ends by issuing an UNLOAD function to spin down the optical disk.

**IO$ _SENSEMODE**
**IO$ _SENSECHAR**

Sense device characteristics: copies information from the device characteristics in the OTnn unit control block to the user's buffer. If the device is already online, the function is executed immediately. If the device is not, a delay of approximately 30 seconds is introduced into these commands. This is to ensure
that valid status has been transferred from the disk to the controller by the controller polling routine. This shows up most predominantly when mounting new volumes in a multi-volume BACKUP set.

**IOS$ SETMODE**

**IOS$ SETCHAR**

Set device characteristics: copies information from the user's buffer to the device characteristics in the OTnn UCB. These characteristics, such as tape format, parity, and density, can be set in the OTnn device, but the OT Driver never checks them, so they have no effect.

**IOS$ SKIPFILE**

**IOS$ SPACEFILE**

Skip file marks forward or reverse: this function does no actual I/O. Based on its search through the TST, it returns status.

**IOS$ SKIPRECORD**

**IOS$ SPACERECORD**

Skip records forward or reverse: this function does no actual I/O; it returns status based on its search through the TST.

**IOS$ WRITECHECK**

No operation.

**IOS$ WRITEOF**

**IOS$ WRITEMARK**

Write file mark: this function does no actual I/O; it merely updates the TST. Writing a file mark when not at the end of the volume invalidates any records or file marks that would be after the file mark just written.

**IOS$ WRITERET**

Write retry: simulates a space record reverse, erase tape, and write data; same as TS11.

**IOS$ WRTTMMKR**

Write take mark retry: simulates a space record reverse, and writes two tape marks.
The function codes listed above can take function code modifiers that affect the I/O operation. Consult the VMS I/O User’s Guide for a complete description of these function modifiers. Differences between the OT Driver and the TSDRIVER are summarized below.

**IO$M_ACCESS**

No difference.

**IO$M_CREATE**

No difference.

**IO$M_DATACHECK**

This qualifier is sent along to the optical disk device from the OT Driver. Thus a volume that is mounted /DATACHECK, or I/O operations that are issued with the IO$M_DATACHECK qualifier explicitly stated, result in a datachecked operation to the disk. The TSDRIVER does not support datachecks.

**IO$M_DMount**

No difference.

**IO$M_ERASE**

No difference. This qualifier is not supported by the TSDRIVER or the OT Driver.

**IO$M_INHEXTGAP**

This qualifier has no effect. Inter-record gaps are meaningless in the context of tape data mapped onto a disk.

**IO$M_INHRETRY**

This qualifier has no effect. Retries are handled entirely within the OT Driver.
**IO$M_NOWAIT**

This qualifier has no effect. It is used only on REWIND and UNLOAD commands. The rewind time is insignificant in the OT Driver, so there is no need to process the IO$M_NOWAIT qualifier.

**IO$M_REVERSE**

No difference.

---

### 10.5 Return Status

The first longword of the I/O status block contains a status code in the lower word, and a byte count or space count for most functions in the upper word. The status returned in this longword will be identical to that of the TSDRIVER.

The second longword of the I/O status block is filled with the device-dependent characteristics upon return from an I/O. These characteristics are summarized in the DEC VMS I/O User's Guide. The MT$M_PARITY, MT$M_DENSITY, and MT$M_FORMAT bits in this longword are settable by the user but are not used by the OT Driver.

The following return status codes may have special significance if returned from the OT Driver.

**SS$_BEGOFTAPE**

Returned if the last space reverse function encountered BOT.

**SS$_DATALATE**

Returned instead of SS$ PARITY if the IO$ READHEAD function is issued for a blank (unwritten) sector of the optical disk.

**SS$_DEVOFFLINE**

This status is returned when the SYSGEN and/or the driver unit initialization routine cannot properly configure a unit. The device will show up as "Offline" in the $SHOW DEVICE display. This error can be caused by any of the following situations:
a. The controller not responding at the selected CSR address  
b. An MSCP initialization sequence error  
c. An attempt to connect the device on a MicroVAX I processor  
d. The OT Driver not being able to recognize the controller type

SS$_DRVERR  
Returned if an IO$_WRITERET function is issued at BOT.

SS$_ENDOFFILE  
Returned if the last read or space record command encountered a file mark.  
Returned if a read reverse command is issued at BOT.

SS$_ENDOFTAPE  
Returned if the last write function crossed the EOT marker block, which is 1000 blocks before the maximum logical block number on the volume. This block is used because some utilities, including BACKUP, write data past the EOT marker.

SS$_ENDOFVOLUME  
Returned if the last space file forward function encountered a double tape mark signifying the logical end of volume.

SS$_FORMAT  
Set if errors are incurred or internal corruption is detected when trying to read in the TST from the disk.

SS$_ILLIOFUNC  
Same as TS11. Returned for an invalid function code.

SS$_INSFMEM  
Returned if there is insufficient non-paged dynamic memory to allocate space for the TST.
SSS_IVBUFLEN

Returned for odd byte count read or write requests. Read requests for 65535 bytes are rounded down to 65534 bytes, and do not return this error.

SSS_NORMAL

Same as TS11. Normal (success) return status.

SSS_TAPEPOSLOST

Returned if the last space or read command moved the tape past the logical end of volume. Returned if the TST in the OT Driver memory is determined to be corrupt. No read or write commands are processed if the tape is determined to be lost. Issue a rewind or unload to the tape device to clear the lost status.

SSS_VOLINV

Returned for a virtual I/O function to a device that is not volume valid. Also returned as the status from MOUNT on fatal drive map errors after eight retries. The OT Driver returns SS$_FORMAT errors to MOUNT, but MOUNT signals the error as SS$_VOLINV.

10.6 Error Log Formatting

The OT Driver logs all device errors into the error log except for reads of blank optical sectors. The ANALYZE/ERROR LOG command formats these error entries as "Unknown device" errors, meaning they were logged from a driver other than the standard DEC driver. At the end of the normal information describing the I/O request, the user, and the system, a series of longwords describing the status is printed. The most important of these longwords are longword 5, which contains the function and status, and longword 10, which contains the physical block number on the optical device where the error occurred.

Longword 5 should be broken into two parts: the low order byte (rightmost two hex digits) is the function that was executing when the error occurred. It should be one of the following hex values:

- 88 = Available command.
- 89 = Online command.
- 92 = Erase command.
- A0 = Writecheck command.
- A1 = READ command.
- A2 = WRITE command

Compatibility 10-9
The high order word of longword 5 (leftmost four digits) indicate the type of error that occurred. It should be one of the following hex values:

- xxx1 = Anything ending in 1 indicates a driver error and should be reported to Emulex Technical Support
- 00xA = Controller error
- 0023 = Unit offline
- 002B = Seek incomplete error
- 0043 = Unit offline
- 0048 = Header compare error
- 004B = Controller error
- 0068 = Read data sync error
- 0069 = Non-existent memory error
- 006B = Drive positioning error
- 008B = Loss of read/write ready
- 00AB = Loss of drive clock
- 00E8 = Uncorrectable ECC error
- 00EB = Drive detected error
- 01x8 = ECC error
- 010B = Controller error
- 1006 = Unit is software write protected
- 2006 = Unit is hardware write protected
C.1 Sample Installation

The following is an example of the OT Driver installation.

Username: SYSTEM
Password:

Welcome to VMS V4.6

$ @SYS$UPDATE:VMSINSTAL

VAX/VMS Software Product Installation Procedure V4.6

It is 21-JUL-1987 at 12:00.
Enter a question mark (?) at any time for help.

* Are you satisfied with the backup of your system disk [YES]?
* Where will the distribution volumes be mounted: MSA0:

Enter the product to be processed from the first distribution volume set.
* Products: *
Please mount the first volume of the set on MSA0:
* Are you ready? YES
%MOUNT-I-MOUNTED, EMULEX mounted on _MSA0:

The following products will be processed:

EMULEX V1.0

Beginning installation of EMULEX V1.0 at 12:03

%MVMINSTAL-I-RESTORE, Restoring product saveset A...

Emulex Software Installation for VAX/VMS V4.4 and above
COPYRIGHT (C) 1988 EMULEX CORPORATION
ALL RIGHTS RESERVED.

This volume contains Emulex software and installation procedures to support Emulex OT (optical tape) pseudo-driver on the VAX/VMS operating system. This software is for use in conjunction with the Emulex UC0x and UC1x optical disk subsystems on the MicroVAX II and on Unibus VAX processors.

%EMULEX-I-LINKING, now linking the OT Driver image
%EMULEX-I-IGNOREMSG, ignore "no user transfer address" warning
%LINK-W-USRTFR, image VMIS$ROOT:[SYSUPD.EMULEX010]OTDRIVER.EXE;1 has no user transfer address
%EMULEX-I-LINKING, now linking OTCONFIG and OTRECOVER images
%EMULEX-I-MOVING, now moving images to system account
This command procedure can create a command procedure to CONNECT the OTAn: device units under SYSGEN and automatically run the OTCONFIG program to set the block size of the device from 512-bytes to 1024-bytes if desired. If you wish to connect a second controller for use with the OT Driver, edit the command file after it is built. Consult the Emulex OT Driver User’s Guide for more information if required.

NOTE

The UC controller NOVRAM will report eight logical units, though you need only connect the actual number of physical optical drives present.

* Do you wish to create the CONNECTOT file [YES] ? YES
* Enter octal CSR bus address for the OTAn: units [772150] ? 760334
* Enter octal vector address for the OTAn: units [154] ? 300
* How many OTAn: units do you wish to configure (1-8) [1] ? 2
* Should unit OTA0: be configured for mapping by default [NO] ? YES
* Does unit OTA0: have a 1024-byte minimum write size [NO] ? YES
* Does unit OTA1: have a 1024-byte minimum write size [NO] ? NO

%EMULEX-I-MOVING, now moving CONNECTOT.COM to the system manager account

The command procedure to configure the OTnn: devices has been copied to SYSSMANAGER:CONNECTOT.COM. If you wish to configure more devices, or to modify the OTCONFIG parameters for any device, edit that command file.

To prevent the OTnn: devices from being automatically configured by VMS as DUnn: devices, edit the file SYSSMANAGER:SYCONF.COM, and insert the following lines:

```
$ STARTUP$AUTOCONFIGURE ALL = = 0
$ RUN SYSSSYSTEM:SYSGEN
   AUTOCONFIGURE ALL.EXCLUDE = (PUx,DUx)
EXIT
$ @SYSSMANAGER:CONNECTOT
```

The first line prevents STARTUP.COM from trying to configure all devices on the system automatically. The next three lines automatically configure all devices except the MSCP disk devices that would correspond to the OTnn: devices (substitute the DUnn: controller designator for the “x” in the AUTOCONFIGURE line above). The fifth line causes the command procedure that you just built, CONNECTOT.COM, to be executed to configure the OTnn: devices, and set up the parameters with OTCONFIG.

%VMSINSTALL-I-MOVEFILES, Files will now be moved to their target directories...

Installation of EMULEX V1.0 completed at 12:10

Enter the products to be processed from the next distribution volume set.

* Products: EXIT < return >

VMSINSTALL procedure done at 12:11
D.1 Sample CONNECTOT.COM File

The following is the CONNECTOT.COM command procedure that was generated by the sample installation in Appendix C.

$! Emulex OT device connection command procedure:
$!
$! To connect more OTnn devices into your system, add
$! CONNECT commands between the RUN and EXIT commands:
$!
$ RUN SYS$SYSTEM:SYSGEN
CONNECT OTA0/ADAPTER = UB0/CSR = %o760334/VEC = %o300
CONNECT OTA1/ADAPTER = UB0/CSR = %o760334/VEC = %o300
EXIT
$!
$! Each OTnn device that has been connected should be
$! configured here using the OTCONFIG utility.
$!
$ RUN SYS$SYSTEM:OTCONFIG
SET OTA0/BLOCK_SIZE = 1024
EXIT
$!
$! End of CONNECTOT command procedure.
Reader’s Comments

Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publication.

Manual Part Number __________________________ Rev. __________________________

What is your general reaction to this manual? In your judgment is it complete, accurate, well organized, well written, etc.? Is it easy to use?

____________________________________________________________________________________

____________________________________________________________________________________

____________________________________________________________________________________

What features are most useful?

____________________________________________________________________________________

____________________________________________________________________________________

____________________________________________________________________________________

What faults or errors have you found in the manual?

____________________________________________________________________________________

____________________________________________________________________________________

____________________________________________________________________________________

Does this manual satisfy the need you think it was intended to satisfy? Does it satisfy your needs? Why?

____________________________________________________________________________________

____________________________________________________________________________________

____________________________________________________________________________________

☐ Please send me the current copy of the Controller Handbook, which contains the information on the remainder of EMULEX's controller products.

Name ____________________________________________ Street __________________________

Title ____________________________________________ City __________________________

Company ________________________________________ State/Country ______________________

Department _____________________________________ Zip __________________________

Additional copies of this document are available from:

Emulex Corporation
3545 Harbor Boulevard
P.O. Box 6725
Costa Mesa, CA 92626
Attention: Customer Services
Copyright (c) 1988 Emulex Corporation

The information in this manual is for information purposes and is subject to change without notice.

Emulex Corporation assumes no responsibility for any errors which may appear in the manual.

Printed in U.S.A.
# TABLE OF CONTENTS

## ONE INTRODUCTION

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>OVERVIEW</td>
<td>1-1</td>
</tr>
<tr>
<td>1.2</td>
<td>DESCRIPTION OF THE OD DRIVER</td>
<td>1-1</td>
</tr>
<tr>
<td>1.2.1</td>
<td>Advantages of the OD Driver</td>
<td>1-3</td>
</tr>
<tr>
<td>1.2.2</td>
<td>Disadvantages of the OD Driver</td>
<td>1-4</td>
</tr>
<tr>
<td>1.2.3</td>
<td>Software Components of the OD Driver</td>
<td>1-4</td>
</tr>
<tr>
<td>1.3</td>
<td>DISTRIBUTION MEDIA</td>
<td>1-5</td>
</tr>
<tr>
<td>1.4</td>
<td>COMPATIBILITY AND REQUIREMENTS</td>
<td>1-5</td>
</tr>
<tr>
<td>1.5</td>
<td>RELATED DOCUMENTATION</td>
<td>1-6</td>
</tr>
</tbody>
</table>

## TWO OD DRIVER INSTALLATION

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>OVERVIEW</td>
<td>2-1</td>
</tr>
<tr>
<td>2.2</td>
<td>CONVENTIONS AND ABBREVIATIONS</td>
<td>2-1</td>
</tr>
<tr>
<td>2.3</td>
<td>INSTALLATION PROCEDURE</td>
<td>2-1</td>
</tr>
<tr>
<td>2.4</td>
<td>CREATING THE CONNECTOD.COM FILE</td>
<td>2-2</td>
</tr>
<tr>
<td>2.5</td>
<td>USING SYSGEN TO CONFIGURE OD DEVICES</td>
<td>2-3</td>
</tr>
</tbody>
</table>

## THREE THE ODCONFIG UTILITY

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>OVERVIEW</td>
<td>3-1</td>
</tr>
<tr>
<td>3.2</td>
<td>ODCONFIG COMMANDS</td>
<td>3-1</td>
</tr>
<tr>
<td>3.2.1</td>
<td>The EXIT Command</td>
<td>3-1</td>
</tr>
<tr>
<td>3.2.2</td>
<td>The HELP Command</td>
<td>3-1</td>
</tr>
<tr>
<td>3.2.3</td>
<td>The SET Command</td>
<td>3-2</td>
</tr>
<tr>
<td>3.2.4</td>
<td>The SHOW Command</td>
<td>3-3</td>
</tr>
<tr>
<td>3.3</td>
<td>ODCONFIG MESSAGES</td>
<td>3-3</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Possible ODCONFIG Messages</td>
<td>3-3</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Status of the Specified OD Device</td>
<td>3-5</td>
</tr>
</tbody>
</table>

## FOUR THE ODRECOVER UTILITY

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>OVERVIEW</td>
<td>4-1</td>
</tr>
<tr>
<td>4.2</td>
<td>ODRECOVER OPERATION</td>
<td>4-1</td>
</tr>
<tr>
<td>4.3</td>
<td>ODRECOVER MESSAGES</td>
<td>4-2</td>
</tr>
</tbody>
</table>

Table of Contents xi
FIVE  COMPATIBILITY

5.1  OVERVIEW ......................................................... 5-1
5.2  OPERATIONAL DIFFERENCES FROM A MAGNETIC DISK SUBSYSTEM ... 5-1
5.2.1  Command Execution Times ........................................... 5-1
5.2.2  Insuring Data Integrity with Writecheck ............................... 5-1
5.2.2.1  Initialize Command .................................................. 5-2
5.2.3  No Standalone Backup .................................................. 5-2
5.3  I/O FUNCTION CODES .................................................. 5-2
5.4  FUNCTION CODE MODIFIERS .......................................... 5-5
5.5  RETURN STATUS ......................................................... 5-6

A  SAMPLE INSTALLATION ................................................ A-1

B  SAMPLE CONNECTOD.COM FILE .......................................... B-1

TABLES

1-1  OD DRIVER DISTRIBUTION MEDIA ....................................... 1-5
1-2  OPTICAL DRIVES SUPPORTED BY OD DRIVER SOFTWARE ............... 1-5

FIGURES

1-1  NORMAL VMS SOFTWARE COMPONENTS (WITHOUT OD DRIVER) ....... 1-2
1-2  VMS SOFTWARE COMPONENTS WITH THE OD DRIVER .................. 1-3
1.1 Overview

This manual is a guide to the installation and use of the VMS Optical Disk Driver (OD Driver) on Digital Equipment Corporation (DEC) VAX and MicroVAX computers. It consists of the following sections:

- **Section One (Introduction)** presents an overview of the OD Driver.
- **Section Two (Installation)** gives installation instructions for OD Driver.
- **Section Three (ODCONFIG)** describes the OD Driver configuration utility and its commands.
- **Section Four (ODRECOVER)** describes the OD Driver media recovery utility and its commands.
- **Section Five (Command Compatibility)** describes the I/O functions, modifiers, and status compatibility issues between OD Driver and the DEC DU/PU class and port drivers.
- **Appendix A (Sample Installation)** shows an example of the OD Driver installation procedure.
- **Appendix B (Sample CONNECTOD.COM File)** provides an example of the command procedure required to connect the OD Driver. The example comes from the sample installation in Appendix A.

1.2 Description of the OD Driver

Optical disks supported by the OD Driver provide high-capacity storage on removable, reasonably priced media, with the restrictions of write-once technology. This means that the medium cannot support the disk file structure of any DEC operating system. This normally requires the application programmer to assume the responsibility for maintaining the integrity of blocks on the medium. The OT Driver relieves the application programmer of this responsibility.

The OD Driver makes the optical (write once) disk appear to the VMS operating system as a magnetic (read/write) disk device. It supports standard disk QIO functions, remapping them to Mass Storage Control Protocol (MSCP) disk functions. It reectors write requests to unwritten physical disk blocks, and reectors read requests to the physical block where the requested logical block was last written. It keeps track of mapping information on the same volume on which the data is stored.
Normal VMS disk operations, i.e., without the OD Driver, utilize the software components illustrated in Figure 1-1.

![Diagram showing software components](image)

Figure 1-1. Normal VMS Software Components (without the OD Driver)

Requests for disk commands, such as reading and writing, are sent directly through the disk driver to the disk controller. Requests for file handling, such as creating, modifying, and closing files, are sent to the disk Ancillary Control Process (ACP) for servicing. The ACP keeps track of the Files-11 structure of the disk, and translates high-level requests, such as "open a file," to the lower-level requests for reading and writing data. The ACP then sends these lower-level requests back to the disk driver, which sends them on to the disk controller.

With the addition of the OD Driver, requests to the OD disk device utilize the software components illustrated in Figure 1-2.
The OD Driver accepts all the commands supported by the DEC DU/PU driver. Requests for ACP functions, such as IO$ ACCESS and IO$ CREATE, are routed to the disk ACP, as with the DEC DU/PU driver. Requests for reading and writing utilize the same function decision table (FDT) routines that the DEC driver uses. This makes the function validation and preprocessing identical to that with the DEC driver. Once a command has been broken down to its most basic representation, it is executed by tasks from two categories:

1. Performance of disk I/O operations
2. Reading and modifying the disk maps

1.2.1 Advantages of the OD Driver

The OD Driver offers the following advantages:

1. Making the disk seem "re-writeable" allows the optical disk to be used as a viable online storage medium. Without intelligent use of the medium, the optical disk could not support the file structure of the various DEC operating systems. With the OD Driver, disks can be re-initialized, re-written, and files can be deleted, etc., as with a magnetic disk. The OD Driver uses the same Files-11 disk structure used by the DU/PU class and port drivers.

2. The OD Driver is much simpler and more software-transparent than an optical disk that emulates a magnetic disk using a new disk ACP. Also, the disk maps that are stored on the disk take up much less disk overhead than the corresponding disk directories and file headers for a disk ACP.
Description of the OD Driver

3. The OD Driver is very efficient, doing direct I/Os to and from the buffers of the calling process.

4. The OD Driver supports all standard disk QIO functions, allowing it to support the VMS commands INITIALIZE, DIRECTORY, COPY, RMS file commands, etc.

5. The OD Driver does not require any temporary storage of data on magnetic media.

6. The ODDriver supports unmapped magnetic disk drive operation concurrently with mapped WORM optical drives attached to the same EMULEX host adapter.

1.2.2 Disadvantages of the OD Driver

Partial blocks written to the disk are zero-filled to the block size required by the drive (512 or 1024 bytes). For optical disks that support 512-byte sectors, this is not a problem. For optical disks that have a 1024-byte sector, this can mean a waste of half the usable space. This can be easily avoided, however, by INITializing the device with an even cluster size. Emulex recommends that optical devices be initialized with a cluster size of 4 or greater.

1.2.3 Software Components of the OD Driver

The OD Driver contains the following software components:

1. The OD Driver: This is the device driver. It allows you to run SYSGEN to create a Unit Control Block (UCB) for the ODnn device, and to issue $QIO I/O requests to the device. The OD Driver calls standard Function Decision Table (FDT) routines within VMS, then reads the disk maps, issues disk I/Os, and modifies the maps accordingly.

2. ODCONFIG: This program allows the system manager to modify the following parameters that affect the mapping of data on the disk:
   a. Whether to map disk blocks or not
   b. Optical disk block size
   c. The amount of memory allocated to the disk maps
   d. Setting a special date to which to revert, i.e., to mount the volume and have it appear as it did at that date and time—the "Historical Mount" feature

3. ODRECOVER: This program is used in the event of a VMS system failure or subsystem failure, to make the optical medium usable again. If such a failure occurs while writing to the ODnn device, data blocks will have been written to the disk that are not yet recorded on the disk’s mapping table. The disk map may not have been updated to reflect the space used on the disk.
The presence of these "hanging" data blocks will prevent use of the remainder of the disk. The ODRECOVER program cannot recover these "hanging" data blocks, but it can revector around them to make the rest of the medium usable again.

NOTE:

Other mapping techniques, including the ACP approach, are also susceptible to loss of data in the event of system failure during a write operation.

4. KITINSTALL.COM: this is the command procedure that installs and builds the three optical disk programs listed above. It is compatible with the DEC VMSINSTALL command procedure.

1.3 Distribution Media

Table 1-1 lists the OD Driver distribution media for versions 4.X and 5.0

Table 1-1. OD Driver Distribution Media, versions 4.X and 5.0

<table>
<thead>
<tr>
<th>Emulex P/N Version 4.X</th>
<th>Emulex P/N Version 5.0</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VD9960714-00</td>
<td>VD9960719-00</td>
<td>5.25-inch floppy diskette (MicroVAX only)</td>
</tr>
<tr>
<td>VD9962014-00</td>
<td>VD9962019-00</td>
<td>TK50 tape cartridge (MicroVAX only)</td>
</tr>
<tr>
<td>VD9960314-00</td>
<td>VD9960319-00</td>
<td>9-track 1600 BPI magtape</td>
</tr>
<tr>
<td>VD9960514-00</td>
<td>VD9960519-00</td>
<td>RX01 8-inch floppy diskette (1178x only)</td>
</tr>
<tr>
<td>VD9960414-00</td>
<td>VD9960419-00</td>
<td>TU58 cartridge (11750 only)</td>
</tr>
</tbody>
</table>

1.4 Compatibility and Requirements

The OD Driver is compatible with the MicroVAX II, VAX 11750, 11780, 11785, and certain CPUs in the VAX 8x00 family of computers. If the host computer is a MicroVAX II, it must have a Q-Bus with an Emulex UC04/MO host adapter connected to an optical drive. If it is any other VAX, it must have a Unibus adapter with an Emulex UC14/MO host adapter plugged into the Unibus and connected to an optical drive.

The optical drives listed in Table 1-2 are supported by the OD Driver software.
Table 1-2. Optical Drives Supported by OD Driver Software

<table>
<thead>
<tr>
<th>Mfg</th>
<th>Model</th>
<th>Sector Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimem OSI</td>
<td>1000</td>
<td>1024 bytes</td>
</tr>
<tr>
<td></td>
<td>LD 1200</td>
<td>1024 bytes</td>
</tr>
</tbody>
</table>

OD Driver software is compatible with the DEC MicroVMS and VMS operating systems, versions 4.4 and above.

Media created under the OD Driver are not compatible with media created under any other optical software.

If you have an MSCP disk server enabled, ensure that the UC controller is set to a non-standard address. This prevents the server from attempting to utilize the optical device as a magnetic disk.

1.5 Related Documentation

The following related documents are available:

- UC04/MO Host Adapter Manual P/N: UC0451003-00
- UC14/MO Host Adapter Manual P/N: UC1451003-00

They may be ordered directly from:

Emulex Corporation
3545 Harbor Boulevard
Costa Mesa, CA 92626
(714) 662-5600  TWX 910-595-2521
2.1 Overview

This section explains how to install the OD Driver under VMS and MicroVMS. The term VMS refers to both VMS and MicroVMS.

2.2 Conventions and Abbreviations

Operator input appears in **boldface** type, in order to distinguish it from operating system messages and prompts.

The symbol `<return>` signifies the carriage return key.

"ODnn" and "ODcu" refer to the VMS device mnemonic for the drive on your system, typically "ODA0".

2.3 Installation Procedure

This subsection gives the steps, including commands, required for OD Driver installation.

1. Log into the system manager's account.

   **NOTE**
   
   Before invoking VMSINSTAL under MicroVMS, you must install the System Programming Option (SYSP). To ensure that SYSP has been installed, check the system directory for the system symbol table (SYS.STB) file. Use the following command to make this check:

   `$ DIR SYS$SYSTEM:SYS.STB< return >`

2. Mount the distribution medium, and invoke the VMSINSTAL PROCEDURE:

   `$@SYS$UPDATE:VMSINSTAL * ddcu:< return >`

   where "ddcu:" is the VMS device name for the device where you have loaded the distribution medium. If you do not specify the kit name (*) or distribution device (ddcu:), the VMSINSTAL command procedure will prompt you for them.

3. Answer questions requested by the Emulex installation procedure.
NOTE

OD DRIVER is not supported for operation on MicroVAX I systems. If you attempt to install the software on a MicroVAX I system, the following warning will print: **The driver does not work if loaded on a MicroVAX I system.** The installation will continue to completion, but will only allow you to install the software for transporting to another system.

If you are updating an existing OD Driver to a new version, the install procedure will delete the old files. Copy these files to another account if you wish to retain them.

### 2.4 Creating the CONNECTOD.COM File

The installation command procedure has the ability to build a command procedure in the system manager's account in order to do the following:

1. Load the OD Driver automatically
2. Connect the units under SYSGEN
3. Use the ODCONFIG utility to set up the OD Driver parameters (option)

The full file specification for this file is SYS$MANAGER:CONNECTOD.COM.

After the installation command procedure has linked the driver and utility images, the command procedure asks the following:

* Do you wish to create the CONNECTOD file [YES]?

Responding "N" or "NO" to this question will skip the creation of the CONNECTOD.COM file, including skipping the next prompts. Take the default response of "YES" to create the CONNECTOD command procedure, unless you have already been through the Emulex product installation and have created this command procedure.

If you wish to create the CONNECTOD file, the command procedure will prompt you for information regarding the ODAn: devices. Only the first optical controller will be configured automatically by the command procedure. If there are multiple controllers with optical drives that will be controlled by the OD Driver software, you must edit the file SYS$MANAGER:CONNECTOD.COM, and insert additional lines for the additional devices.

The command procedure will prompt you for the CSR address and interrupt vector on the Q-Bus or Unibus of the first controller, as follows:

* Enter octal CSR bus address for the ODAn: units [772150] ?
* Enter octal vector address for the ODAn: units [154] ?
The default answers for these questions are the default CSR and vector for the first MSCP controller on the bus. Valid responses are either that CSR and vector, or CSR addresses in the range 760334-760370 (octal), and vector addresses in the range 300-340 (octal). The command procedure will next prompt as follows:

* How many ODAn: units do you wish to configure (1-8) [1]?

Each unit that you specify will have a corresponding SYSGEN CONNECT statement added to the CONNECTOD command procedure. For each unit that you specify, you will be asked the following:

* Should unit ODAn: be configured for mapping by default [NO]?

Answer YES to this question if the specified unit is an optical disk that you will normally be mapping with the OD Driver. This will cause an ODCONFIG SET ODAn/MAPPED command to be added to the CONNECTOD.COM file for execution when the system is started. For each unit, you will also be asked the following:

* Does unit ODAn: have a 1024-byte minimum write size [YES]?

Optical disks have either a 512-byte or 1024-byte minimum write size. If the optical disk that corresponds to that unit number has a 1024-byte minimum write size, answer "YES." Each unit to which you answer "YES" will have a corresponding ODCONFIG SET ODAn/BLOCK_SIZE=1024 statement added to the CONNECTOD command procedure. See subsection 1.4 for sector size information on Emulex-supplied optical drives.

Appendix A of this document contains a sample run of the installation command procedure. Appendix B contains a sample CONNECTOD.COM file.

2.5 Using SYSGEN to Configure OD Devices

The OD Driver controls devices that are normally controlled by the DEC DU/PU MSCP disk class and port driver. Two steps are required to allow this. First, the SYSGEN program must not be allowed to configure the devices as DU devices; second, the SYSGEN program must be used to configure the OD devices.

These steps are accomplished by changes to the system configuration command file SYS$MANAGER:SYCONFIG.COM. This command file is used to make site-specific changes to the device configuration. Edit this command file, and insert the following lines:

```
$ STARTUP$AUTOCONFIGURE ALL = = 0
$ RUN SYS$SYSTEM:SYSGEN
AUTOCONFIGURE ALL/EXCLUDE=(PUx,DUx)
EXIT
$ @SYS$MANAGER:CONNECTOD
```
Line 1 sets a global symbol to keep the system startup command procedure from doing a SYSGEN AUTOCONFIGURE ALL to configure in all devices. Line 2 starts the SYSGEN program. Line 3 specifies autoconfigure for all devices except those excluded by the EXCLUDE switch. For example, if the devices to be controlled by the OD Driver would normally show up as DUC7: and DUC8: (the third MSCP controller on the bus), substitute the letter “C” for the “x” in the third line above.

Line 5 invokes the CONNECTOD.COM command file that was built during the installation. For each device to be connected, CONNECTOD.COM will execute a SYSGEN command with the following form:

```
SYSGEN > CONNECT
ODAn/ADAPTER = UB0/CSR = %0772150/VECTOR = %0154
```

The first unit to be connected will also cause the controller to go through MSCP initialization. This raises the Interrupt Priority Level (IPL) to 31 (maximum) for a period of 3-15 seconds, suspending all other activity on the system for that period. You may wish to use the "$SET TIME" command after executing CONNECTOD.COM, to make sure that the system time is set correctly.
3.1 Overview

The ODCONFIG utility is used to set software parameters that affect the mapping of data onto the optical disk, and to display the current settings of those parameters. The device name "ODA0:" will be used in the examples in this section.

3.2 ODCONFIG Commands

ODCONFIG is installed in the SYS$SYSTEM: account. It is an interactive utility that accepts the commands EXIT, HELP, SET, and SHOW, and has the following prompt:

```
ODCONFIG>
```

Any of these commands can be abbreviated to one or more letters that make them unique. The SET and SHOW commands require one parameter: the ODnn device. Where parameters are required, they must be entered on the same line as the command. ODCONFIG will not prompt for unspecified parameters, and it will print an error message if parameters are left unspecified.

Device names may be entered with or without the terminating colon (:) character.

3.2.1 The EXIT Command

The EXIT command terminates the ODCONFIG program, and returns control to the command interpreter. Control-Z (Z) also terminates the program.

3.2.2 The HELP Command

The HELP command prints a brief (one-screen) help message summarizing the format of the commands. It takes no parameters, and sublevels of help are not available.
3.2.3 The SET Command

The SET command modifies the software parameters affecting the mapping of data onto the optical device. It takes one parameter, the "ODnn" device name, and any combination of qualifiers described below.

**ODCONFIG > SET ODA0/MAPPED**
**ODCONFIG > SET ODA0/NOMAPPED**

This qualifier turns mapping on and off for the device. If mapping is on, the OD Driver assumes responsibility for remapping write requests to unwritten parts of the disk, and remapping read requests to the area of the last write. If mapping is off, the user application must assume responsibility for the mapping of the data. For normal use, optical disks should be set /MAPPED, and magnetic disks on the same controller should be set /NOMAPPED. The exception to this is when the ODRECOVER utility must be used on the device. The optical disk must be set /NOMAPPED for the running of ODRECOVER, and should be set back /MAPPED after ODRECOVER has repaired the volume.

**ODCONFIG > SET ODA0/BLOCK SIZE = 512**
**ODCONFIG > SET ODA0/BLOCK SIZE = 1024**

This qualifier sets the minimum block size that the optical disk can write. All optical drives can read single 512-byte sectors and portions thereof. Some optical drives must write 1024-byte sectors, as they cannot individually address 512-byte sectors for a write. The default setting for this parameter is 512 bytes. If your optical drive has a 512-byte block size, you need not use this qualifier.

**ODCONFIG > SET ODA0/BUFFERS = 24**

This qualifier sets the number of 1024-byte buffers to allocate from non-paged memory for the disk maps when the device is mounted. Each buffer maps 330 contiguous logical disk blocks. A rule of thumb is to allocate one buffer for each discontinuous data stream (file or indexed file area) that will be read or written on the device at any given time.

Allocating too many buffers will waste non-paged memory, and it increases the damage that can be done to the volume by a system crash. Allocating too few buffers will result in wasted disk space as updated buffers are purged from memory back to the disk, only to be loaded back into memory when they are needed again.

The SHOW command will display disk utilization (see below). As a general rule, the number of blocks used for maps should be under 1% of those used for data.
ODCONFIG > SET ODA0/REVERT=01-JAN-1988:10:00
ODCONFIG > SET ODA0/NOREVERT

Data on an optical disk is vectorized by the OD Driver, but it is never actually erased. Specifying the /REVERT=date qualifier will cause the device to use only maps that were written before the specified date and time. In order to get all the files on the disk that you are seeking, set the /REVERT time slightly after the dismount time of that disk.

The /REVERT date can be modified only while the device is dismounted. This date will affect all mounts on the device until the device is dismounted. The /NOREVERT qualifier is used to turn off this feature. When a device is mounted and a /REVERT date has been specified, the device will be software write-locked for the duration of the mount.

3.2.4

The SHOW Command

The SHOW command shows which parameters are set for the ODnn device. It has the following form:

    ODCONFIG > SHOW ODA0

The SHOW command displays the OD Driver flags, optical disk block size, and buffer settings. It displays optical disk block usage information if the OD Driver currently has maps loaded in memory (if the device is mounted).

3.3

3.3.1

ODCONFIG Messages

Possible ODCONFIG Messages

The following messages are possible from ODCONFIG:

%ODCONFIG-E-DEVMOUNTED, device is mounted

A SET command cannot be executed with the /BLOCK SIZE or /REVERT qualifier because the device is currently mounted, and changing these parameters would corrupt data on the device.

%ODCONFIG-E-ERRGETINF, error getting information on device

An error was encountered using either the $GETDVI (get device information) system service, or the kernel mode routine that reads device information from the UCB. This message will be followed by a VMS error message explaining the error.
%ODCONFIG-E-ERRSETINF, error setting information on device

An error was encountered in the kernel mode routine that sets device information in the UCB. Use the SHOW command to ascertain the current settings of the device. This message will be followed by a VMS error message explaining the error.

%ODCONFIG-E-ILLDEVNAM, illegal device name, should be ODcu:

A SET or SHOW command was executed, and the device name specified did not have the form "ODcu:". Re-enter with a valid device name.

%ODCONFIG-E-INVBBLKSIZ, invalid block size, must be 512 or 1024

A SET command with the /BLOCK SIZE qualifier was executed, with a block size specified that was not 512 or 1024 bytes. Re-enter the command with a block size parameter of either 512 or 1024.

%ODCONFIG-E-INVBUCNCT, invalid buffer count specified

A SET command with the /BUFFERS qualifier was executed, with an invalid number of buffers specified. The minimum number of buffers is eight, and the maximum is 127. Re-enter the command with a buffer count within these bounds.

%ODCONFIG-E-NOCANGES, no changes have been made to the device

An error was detected in the format of the SET command before any part of the command was executed, thus none of the command was executed. Correct the qualifier or parameter, and re-enter the command.

%ODCONFIG-E-NOCUAL, no qualifiers specified on SET command

The SET command was entered without any qualifiers. Re-enter the command with at least one qualifier.

%ODCONFIG-E-NOTSUPUV1, not supported on the MicroVAX I

The OD Driver, and thus ODCONFIG, are not supported on the MicroVAX I processor. Using SYSGEN to connect the OD Driver on the MicroVAX I results in the device being offline. Attempting to run the ODCONFIG program on the MicroVAX I results in this error message.
%ODCONFIG-E-UNKNOWN, device state is unknown and may be corrupted

A catastrophic error was detected while setting device information, and the data in the UCB for the device may be corrupted. Reboot the system, and reload the OD Driver if the device continues to return errors.

3.3.2 Status of the Specified OD Device

The SHOW command will return the status of the specified OD device. The physical device name will be shown first, optionally followed by one or more flag bits. These flag bits are as follows:

- Device can be enabled for mapping.
- Device currently has mapping turned OFF.
- Device currently has mapping turned ON.
- Device is in process of dismount.
- Device currently has maps loaded.
- Device is in process of mount.
- Device will revert to mount status from a specified date/time.
  Revert date and time = dd-mmm-yyyy hh:mm:ss.hh

The optical disk block size is then printed, followed by the number of buffers currently being used, and the number of buffers requested for the next MOUNT command:

- Optical disk block size = n bytes.
- Entry buffers currently allocated = x.
- Entry buffers requested for next mount = y.
- Each entry buffer is 1024 bytes of non-paged pool.

The usage of the optical disk is summarized in the next messages:

- Total blocks on device = w.
- Blocks used for data = x.
- Blocks used for maps = y.
- Blocks not yet written = z.

The block figures are all in 512-byte blocks, even if the /BLOCK_SIZE qualifier has been set to 1024 bytes.
4.1 Overview

The ODRECOVER utility is used to analyze and/or repair the mapping structure on a disk following a system crash or fatal optical disk subsystem failure. The OD Driver keeps mapping information in memory until a dismount or available command is issued to the ODnn unit (maps will also be written to disk as the mapping buffers in memory are filled and purged). At dismount, the updated maps are written to disk. See subsection 5.3 for information on purging maps from memory to disk.

In the event of a system crash or subsystem failure, data blocks to which no disk maps are pointing may have been written on the optical disk. This is because the data blocks are written as the commands are issued, but some maps will not be written until the dismount. The ODRECOVER utility will find the last map block, then duplicate it, but with an updated free block pointer pointing around any hanging data blocks.

Note that ODRECOVER can repair the map structure so the remainder of the optical volume is usable, but it cannot repair the directory structure, or the RMS structure of any files that were open at the time of the error. Use the "$ANALYZE/DISK_STRUCTURE" and "$ANALYZE/RMS_FILE" commands to verify the integrity of the file system after you use ODRECOVER.

4.2 ODRECOVER Operation

ODRECOVER requires almost no operator intervention. To use it, first ensure that the ODnn device is not mounted. Use ODCONFIG to set the device /NOMAPPED. Invoke ODRECOVER with the following command:

$ RUN SYS$SYSTEM:ODRECOVER

ODRECOVER will print out a header describing the program. It will then prompt as follows:

Do you wish to continue <No> ? YES<return>
Answer "Y" or "YES" to continue with the process. ODRECOVER prompts as follows for the name of the device to analyze:

Name of disk (ODnn) device to check? ODA0<return>

ODRECOVER does not prompt for any more input. It will perform the following steps to analyze and/or repair the mapping structures:

1. Read logical block 0, and check the information stored there to confirm that the initialization of the pack meets the following criteria:
   - It has already been performed.
   - It used the correct (OD Driver) software.
   - It was performed with a compatible version of the OD Driver.

2. Starting at the end of the volume, read the map blocks in reverse order until the program encounters a read. Update the pointer to the first free block from these map blocks.

3. Starting from the suspected first free block, read data blocks until the program encounters a read error. This read error signifies the end of the written data blocks on the disk.

4. If you reach the end of the usable part of the disk before you get the read error, the disk is not repairable, but the existing files on it should be readable.

5. If the read error occurs immediately on the first data block, no new data blocks were written on the volume, and the disk does not need repair.

6. If "hanging" data blocks are found, the last map block is read, the pointer to the first free block is updated, and the map block is written as a new map block. The device is now repaired.

4.3 ODRECOVER Messages

ODRECOVER will print out the device usage statistics for the volume in the same format that the ODCONFIG SHOW command does. The following messages are possible from ODRECOVER.
%ODRECOVER-E-BADVERSION, driver version mismatch

The volume to be analyzed was initialized under a version of the OD Driver that is higher (later) than the version of ODRECOVER. ODRECOVER cannot continue; install the latest version of the OD Driver software and retry the operation.

%ODRECOVER-E-DEVMOUNTED, device is mounted

ODRECOVER cannot allocate and analyze the device because the device is already mounted.

%ODRECOVER-E-ERRALLOC, error allocating device

An error was encountered trying to allocate the ODnn device. This message is followed by the VMS message explaining the error, which is typically caused by another user having allocated or mounted the device.

%ODRECOVER-E-ERRASSIGN, error assigning device

ODRECOVER encountered an error assigning a channel to the ODnn device. This message is followed by a VMS message explaining the error.

%ODRECOVER-E-ERRREADINF, error reading information block

An error was encountered reading the device information block at Logical Block Number (LBN) 0 of the volume. Either the volume has never been used, and the information block is blank (and thus unreadable), or the information block is corrupt, and the volume is not repairable.

%ODRECOVER-F-ERRGETDVI, error getting device information

ODRECOVER encountered an error trying to issue a $GETDVI system service to get information on the ODnn device. This message is followed by a VMS message explaining the error.

%ODRECOVER-F-ERRGETFLG, error getting OD flag information

ODRECOVER encountered an error trying to change mode to kernel in order to read the special OD flag information from the UCB. This message is followed by a VMS system error message. The error is typically caused by not having CMRKNL privilege.
%ODRECOVER-F-FINALREAD, error reading EBLK
ODRECOVER encountered an error re-reading the last mapping block (EBLK) for the final fix.

%ODRECOVER-F-FINALWRITE, error writing updated EBLK
ODRECOVER encountered an error writing the last mapping block (EBLK) for the final fix.

%ODRECOVER-F-ILLDEVNAM, illegal device name, must be "ODcu:"
The specified device name was not in the form "ODcu:" or "ODcu:". Restart the program, and enter the correct device name.

%ODRECOVER-E-INSFRESFC, insufficient free space to repair volume
There is insufficient free space on the volume to make a repair. The volume cannot be repaired, but the ODCONFIG SET/REVERT command can be used to recover any usable data on the volume.

%ODRECOVER-E-INVINFBLK, invalid information (INF) block structure
ODRECOVER read block zero of the device, expecting it to contain information about the driver and driver version that initialized the volume. The block was readable, but was not a valid information block. The volume was not initialized with the OD Driver and ODRECOVER should not be used on the volume.

%ODRECOVER-F-MAPLOADED, device currently has maps loaded
ODRECOVER cannot allocate and analyze the device because the device currently has map structures loaded in memory.

%ODRECOVER-E-NODISKDEV, device is not a disk device
The specified device is not understood by VMS to be a disk device. This error usually occurs as the result of specifying an incorrect device name.
%ODRECOVER-S-NOFIXREQD, no disk structure repair required

ODRECOVER found no "hanging" data blocks after the last known written block. Thus no disk structure repair is required for this medium.

%ODRECOVER-F-NOTFIXABLE, this medium cannot be repaired

The previous error message was of a type that indicates that this disk medium cannot be repaired by ODRECOVER.

%ODRECOVER-E-NOTODDVOl, this is not an OD driver volume

ODRECOVER successfully read the information (INF) block, and it checked out as a valid INF block, but its driver field indicated that the volume is not an OD Driver volume. The volume is an Emulex OTDRIVER volume and should be examined with the OTRECOVER utility.

%ODRECOVER-E-REMOTE, device ODnn is not on the local node

ODRECOVER cannot allocate and analyze the device because the device is not on the local node.

%ODRECOVER-S-REPAIRED, disk structure table has been repaired

ODRECOVER has successfully written a new mapping block to reflect the actual physical blocks used on the volume. The device has been repaired, and should now be set /MAPPED with ODCONFIG.
Section 5
COMPATIBILITY

5.1 Overview

The OD Driver most closely emulates a DEC MSCP disk subsystem; it returns the drive type code of "Foreign disk 1" to keep the error formatting log from interpreting the error log as if it came from a DEC driver. The Function Decision Table (FDT) routines within the driver are patterned after those of the DU/PU driver to ensure compatibility. Where device status for certain functions was undocumented or ambiguously documented, every effort has been made to ensure that the OD Driver returns the same status that a magnetic disk would in a comparable situation.

This section covers compatibility issues, which fall into the following four areas:

a. Operational differences
b. I/O function codes
c. Function code modifiers
d. Return status
e. Error Log Formatting

5.2 Operational Differences from a Magnetic Disk Subsystem

The OD subsystem has some operational differences from a magnetic disk subsystem. This subsection summarizes the major ones.

5.2.1 Command Execution Times

INIT, MOUNT, and DISMOUNT can take up to seven minutes for OD devices. The OD Driver loads map information on a MOUNT command, and writes map information to the disk on a DISMOUNT command. The INIT command uses IO$PACKACK and IO$AVAILABLE as an implied mount and dismount.

5.2.2 Insuring Data Integrity With Writecheck

You should always initialize and mount OD devices with writechecking enabled. Use the following commands:

$ INITIALIZE/DATA CHECK=WRITE
$ MOUNT/DATA CHECK=WRITE

This is required because the optical disk, not the operating system, assumes the responsibility for revectoring bad blocks. The optical disk can detect and correct errors only if a datacheck operation is performed on all write operations.
NOTE
Failure to mount the OD device /DATACHECK = WRITE results in unreliable data being written to the device, which may corrupt and/or lose data.

5.2.2.1 Initialize Command
The Initialize command includes two important qualifiers: Cluster size and Nohighwater. Both should appear in the following command string:

$ Initialize/Data_Check = Write/Cluster = 4/Nohighwater

Cluster size is the value used by the operating system to determine the minimum number of storage blocks used per transfer. The number (4, if using 1024 byte minimum write) should be some multiple of two that is four or larger. Always use an even cluster number on a device with a 1024 byte minimum write; odd clusters waste a large amount of disk space.

The OD Driver correctly handles erase commands by zeroing out pointers so that erasing does not waste the data area of the disk. Nohighwater prevents the OD Driver from pre-zeroing files. Highwater marking serves no purpose on a write-once disk.

5.2.3 No Standalone BACKUP
Standalone SYSGEN configures the ODnn devices as DUnn devices; thus the OD Driver is not supported for use with standalone BACKUP.

5.3 I/O Function Codes
The OD Driver supports the function codes listed below (variations from magnetic disk operation that may be perceptable to the user have been documented).

All Data Transfers
Data transfers are not reordered as in standard MSCP implementations, because the OD Driver may need to execute multiple commands for segmented reads.
IO$ READVBLK
IO$ REA DLBLK
IO$ READPBLK

Read block functions: maximum byte count is 65535. If the IO$M_DATACHECK function modifier is set, it is passed along in the optical disk command. If a requested logical block is not mapped to a physical block, the user buffer will be filled with zeroes for that portion of the data transfer.

IO$ WRITEVBLK
IO$ WRITELBLK

Write block functions: maximum byte count is 65535. If the IO$M_DATACHECK function modifier is set, that qualifier is passed on in the optical disk command.

IO$ ACCESS
IO$ ACPCONTROL
IO$ CREATE
IO$ DEACCESS
IO$ DELETE
IO$ MODIFY
IO$ MOUNT

ACP functions: these functions are sent to the ACP for interpretation. They are not handled directly by the OD Driver, but may be handled indirectly if the ACP converts them into read and write requests.

IO$ AVAILABLE

Available function: writes any modified buffers to the disk. Issues an AVAILABLE command to the optical drive.

IO$ DRVCLR

This is not a normally supported command for MSCP disks. It is a special function for the OD Driver that causes all modified buffers in memory to write to the disk. User programs can periodically issue this command to minimize potential data block loss in the event of a system crash, but more disk space will be used for mapping information.

IO$ DSE

Data security erase: zeroes the pointers for the logical block numbers to be "erased." This command does not physically erase the data, which can be recovered with the ODCONFIG SET/REVERT command.
IO$_NOP

No operation.

IO$_PACKACK

Pack acknowledge function: issues a pack acknowledge to the device. If the
device does not already have maps loaded, memory is allocated for them, and a
mapping table is built from the map blocks stored on disk.

IO$_READHEAD
IO$_WRITEHEAD

These are not normal disk commands, and they are not supported by the DEC
DUDRIVER. They are used for unmapped reading and writing of specific disk
blocks. Only the ODRECOVER utility uses them, and they are not supported for
use by any other program.

IO$_READRCT

No difference from normal MSCP disk.

IO$_SENSEMODE
IO$_SENSECHAR

Sense device characteristics: copies information from the device characteristics in
the ODnn unit control block to the user’s buffer.

IO$_SETMODE
IO$_SETCHAR

Set device characteristics: copies information from the user’s buffer to the device
characteristics in the ODnn unit control block.

IO$_UNLOAD

Writes any updated maps to disk the same as the IO$ AVAILABLE command.
Issues an UNLOAD command to spin down the disk.

IO$_WRITECHECK

Writechecks the user buffer against the data on disk. For any portion of the
transfer where the logical block is not mapped to any physical block, the user
buffer is checked for all zeroes.
5.4 Function Code Modifiers

The function codes listed above can take function code modifiers that affect the I/O operation. Consult the VMS I/O User's Guide for a complete description of these function modifiers, which are listed below.

IO$M_ACCESS
No difference.

IO$M_CREATE
No difference.

IO$M_DATACHECK
This qualifier is sent to the optical disk device from the OD Driver. Thus a volume that is mounted /DATACHECK, or I/O operations that are issued with the IO$M_DATACHECK qualifier explicitly stated, will result in a datachecked operation to the disk.

IO$M_DMount
No difference.

IO$M_ERASE
The blocks in question are not physically erased on the volume. Instead, the pointers to those blocks are cleared.
5.5 Return Status

The first longword of the I/O status block contains a status code in the lower word, and a byte count for transfer functions in the upper word. The following return status codes may have special significance if returned from the OD Driver.

**SS$\_DATALATE**

Returned instead of SS$ PARITY if the IO$ READHEAD function is issued for a blank (unwritten) sector of the optical disk.

**SS$\_DEVOFFLINE**

This status is returned when the SYSGEN and/or the driver unit initialization routine cannot properly configure a unit. The device reads “Offline” in the $SHOW DEVICE display. This error can be caused by any of the following situations:

1. The controller not responding at the selected CSR address
2. An MSCP initialization sequence error
3. An attempt to connect the device on a MicroVAX I processor
4. The OD Driver not being able to recognize the controller type

**SS$\_DRVERR**

Can be returned for an actual drive error, or can indicate that an internal inconsistency in the map structures has been detected after the maps have been loaded from disk.

**SS$\_FORMAT**

Set if errors are incurred or internal corruption is detected when trying to read the mapping information from the disk.

**SS$\_ILLOFUNC**

No difference. Returned for an invalid function code.

**SS$\_INSFMEM**

Returned if there is insufficient non-paged dynamic memory to allocate space for the disk maps.

**SS$\_IVBUFLEN**

Returned for odd byte count read or write requests.
A.1 Sample Installation

The following is an example of the OD Driver installation.

Username: SYSTEM
Password:

Welcome to VMS V4.6

$ @SYS$UPDATE:VMSINSTAL

VAX/VMS Software Product Installation Procedure V4.6

It is 21-JUL-1987 at 12:00.
Enter a question mark (?) at any time for help.

* Are you satisfied with the backup of your system disk [YES]?
* Where will the distribution volumes be mounted: MSA0:

Enter the product to be processed from the first distribution volume set.
* Products: *
Please mount the first volume of the set on MSA0:.
* Are you ready? YES
%MOUNT-I-MOUNTED, EMULEX mounted on _MSA0:

The following products will be processed:

EMULEX V1.0

Beginning installation of EMULEX V1.0 at 12:03

%VMSINSTAL-I-RESTORE, Restoring product saveset A...

Emulex Software Installation for VAX/VMS V4.4 and above
COPYRIGHT (C) 1988 EMULEX CORPORATION
ALL RIGHTS RESERVED.

This volume contains Emulex software and installation procedures to support Emulex OD (optical disk) pseudo-driver on the VAX/VMS operating system. This software is for use in conjunction with the Emulex UC0x and UC1x optical disk subsystems on the MicroVAX II and on Unibus VAX processors.

%EMULEX-I-LINKING, now linking the OD Driver image
%EMULEX-I-IGNOREMSG, ignore "no user transfer address" warning
%LINK-W-USRTFR, image VMS$ROOT:[SYSUPD.EMULEX010]ODDRIVER.EXE;1 has no user transfer address
%EMULEX-I-LINKING, now linking ODCONFIG and ODRECOVER images
%EMULEX-I-MOVING, now moving images to system account
This command procedure can create a command procedure to CONNECT the ODAn: device units under SYSGEN, and automatically run the ODCONFIG program to set the block size of the device from 512-bytes to 1024-bytes if desired. If you wish to connect a second controller for use with the OD Driver, edit the command file after it is built. Consult the Emulex OD Driver User's Guide for more information if required.

NOTE

The UC controller NOVRAM will report eight logical units, though you need only connect the actual number of physical optical drives present.

* Do you wish to create the CONNECTOD file [YES] ? YES
* Enter octal CSR bus address for the ODAn: units [772150] ? 760334
* Enter octal vector address for the ODAn: units [154] ? 300
* How many ODAn: units do you wish to configure (1-8) [1] ? 2
* Should unit ODA0: be configured for mapping by default [NO] ? YES
* Does unit ODA0: have a 1024-byte minimum write size [NO] ? YES
* Should unit ODA1: be configured for mapping by default [NO] ? NO
* Does unit ODA1: have a 1024-byte minimum write size [NO] ? NO

%EMULEX-I-MOVING, now moving CONNECTOD.COM to the system manager account

The command procedure to configure the ODnn: devices has been copied to SYSS$MANAGER:CONNECTOD.COM. If you wish to configure more devices, or to modify the ODCONFIG parameters for any device, edit that command file.

To prevent the ODnn: devices from being automatically configured by VMS as DUnn: devices, edit the file SYSS$MANAGER:SYSCONFIG.COM, and insert the following lines:

```
$ STARTUP$AUTOCONFIGURE_ALL = = 0
$ RUN SYSSYSTEM:SYSGEN
   AUTOCONFIGURE ALL/EXCLUDE = (PUx,DUx)
EXIT
$ @SYSS$MANAGER:CONNECTOD
```

The first line prevents STARTUP.COM from trying to configure all devices on the system automatically. The next three lines automatically configure all devices except the MSCP disk devices that would correspond to the ODnn: devices (substitute the DUnn: controller designator for the "x" in the AUTOCONFIGURE line above). The fifth line causes the command procedure that you just built, CONNECTOD.COM, to be executed to configure the ODnn: devices, and set up the parameters with ODCONFIG.

%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...

Installation of EMULEX V1.0 completed at 12:10

Enter the products to be processed from the next distribution volume set.
* Products: EXIT < return >

VMSINSTAL procedure done at 12:11
B.1 Sample CONNECTOD.COM File

The following is the CONNECTOD.COM command procedure that was generated by the sample installation in Appendix A.

$! Emulex OD device connection command procedure:
$!
$! To connect more ODnn devices into your system, add
$! CONNECT commands between the RUN and EXIT commands:
$!
$ RUN SYS$SYSTEM:SYSGEN
CONNECT ODA0/ADAPTER = UB0/CSR = %o760334/VEC = %o300
CONNECT ODA1/ADAPTER = UB0/CSR = %o760334/VEC = %o300
EXIT
$!
$! Each ODnn device that has been connected should be
$! configured here using the ODCONFIG utility.
$!
$ RUN SYS$SYSTEM:ODCONFIG
SET ODA0/MAPPED
SET ODA0/BLOCK_SIZE = 1024
EXIT
$!
$! End of CONNECTOD command procedure.