Overland Data, Inc.

Installation, Configuration & DataTools

User Manual

Overland Data®

Installation, Configuration, & DataTools™ Manual

Overland Data, Inc.

8975 Balboa Avenue

San Diego, CA 92123-1599

P/N: 104078 Revision E
11/93

Copyright

All rights reserved. This manual or any part(s) thereof, may not be reproduced physically or electronically or distributed in any manner without prior written permission from Overland Data, Inc.

Disclaimer

This manual has been reviewed for accuracy and completeness. The descriptions and instructions it contains were accurate at the time this manual was produced. The tape drive configuration is subject to change without notice and may differ from the description herein. Overland Data assumes no liability for damages of any kind incurred from errors, omissions or discrepancies between the tape drive's actual operation and operation as described in the manual.

©Overland Data, Inc.

All Rights Reserved

Sales & Tech Support

Voice: (619) 571-5555 FAX: (619) 571-0982 BBS: (619) 571-3651

Quick Guide

Software

Installation, Configuration & DataTools Manual

DEPOT4	Window oriented, mouse-driven shell program for most applications
DEPOT2	"Batch" file application
FDUMP	To view tapes
LABEL	IBM or ANSI labeled tapes
TAPEUTL	Read fixed-length labeled tapes
TAPEUTL	Read variable-length labeled tapes

TapeView Manual (Optional)

TAPEVIEW	Select fields and perform format conversion

Local Hard Disk Backup

Installation, Configuration & Standard Software Manual

FLASHBAK	Hard disk-to-tape backup

Novell Network Backup (Optional)

Novabak Novaware Manual

NOVABACK	Powerful, friendly, window-oriented network backup and restore

Novell Network Backup (Optional)

Programming Tools Reference Manual

TAPE_MTO.SYS	Write tape programs in various languages using a device driver
ALLTAPE	Write tape programs using libraries in Microsoft C

Hardware

Controller Card

Installation, Configuration & Standard Software Manual

Use the above manual for all Overland Data supplied controllers and your parallel port.

Tape Drives

User Manual, TapePro 5000 Series SCSI Tape Drives

5612	Setup, configuration and operation
5614	Setup, configuration and operation
5212	Setup, configuration and operation
5214	Setup, configuration and operation

User Manual, Model 3210/3610 Tape Drive

3210	Setup, configuration and operation
3610	Setup, configuration and operation

User Manual, Model 3201/3601 Tape Drive

3201	Setup, configuration and operation
3601	Setup, configuration and operation

Table of Contents

Chapter 1 - Overview 1
About This Manual 1
Conventions 2
Where To Start 2
Services 3
Chapter 2 - About 9-track Tape 6
Physical Characteristics of 9-Track Tape 6
Tape Layout 9
Tape Data 14
Number Formats 16
Format Specifications 17
Chapter 3 - Hardware Installation & Setup 21
TX-8 Controller Card 22
TX*i*-16 Controller Card 31
XL/2 Controller Card 42
Parallel Port Interface 51
TMC Controller Card 53
Chapter 4 - Software Installation 69
Quick Software Installation 69
Detailed Software Installation 70
Chapter 5 - -DataTools 80
Overview 80
Using DataTools 80
Depot4 81
File Menu 85
Utilities Menu 87
Options Menu 142
Getting Around in Depot4 181
Depot2 185
FDUMP 202
TAPEUTL 212
DOS Backup (Flashbak) 286
Command Reference 313

Table of Contents

Appendix A: NovaBack A-1
Appendix B: (Controller Specifications) B-1
Appendix C C-1
Index i

Chapter 1 - Overview

About This Manual

This manual includes installation information and procedures for all Overland Data controllers and the DataTools™ (through release 1.07) software program. This manual also includes details for all DataTools™ utilities.

The manual is divided into four chapters.

Chapter 1

Describes how to use the manual and list the products available from ODI.

Chapter 2

Provides technical information on 9-track formats.

Chapter 3

Provides information on hardware installation and setup of the Overland Data controllers.

Chapter 4

Provides information on software installation and configuration.

Chapter 5

Gives you a detailed look at the Overland Data software, DataTools™.

Appendix A

Provides technical specifications for the tape controller and translation tables.

Note: Last-minute changes to hardware or software are documented as supplemental information. Refer to the end of the manual for addenda *and* the readme.doc file on the release disk *before* using the utilities.

Conventions

Throughout this manual, we use the following conventions:

Action	Description
input	Bold typeface is used to show character strings that must be typed from the keyboard.
prompt	This font is used to show prompts and other information displayed on-screen.
[ENTER]	Bold typeface, in brackets is used to indicate keyboard entries. For example: **[ENTER]** or **[CTRL]** or **[SPACE]** refer to keys on the keyboard.
[CTRL]+E	Keys shown with a plus (+) sign indicate that you need to press the two keys at the same time.
italics	Bold italics are used to indicate the name of a program, command, or title.

Where To Start

If you are installing a controller and DataTools for the first time, start with Chapter 3. For information installing and setting up your tape drive, refer to the manual that came with your drive.

After your hardware is installed and tested, go to Chapter 5 for detailed information on the DataTools program.

Services

Technical Support

Overland Data maintains a full-time professional staff of technicians to support our products. Our technicians are knowledgeable and available to assist users in resolving problems and answering application questions.

Before you contact technical support, we encourage you to read this manual. Many common questions and simple problems can be resolved quickly by using the manual. However, if you still require assistance contact technical support by calling...

(619) 571-5555

Monday through Friday
7:00 A.M. to 4:00 P.M. Pacific time

Before You Call

Before you call, have the following information ready for your technician.

- ✔ Model number of the controller and tape drive
- ✔ Version number of your software
- ✔ Serial number (or date stamp on some controllers) of the tape drive and controller
- ✔ Purchase date(s)
- ✔ For software installation problems, be at your computer. Have all the installation diskettes available and note the error (s) that occurred.
- ✔ For controller problems, have the results of XTEST (test software). If you do not have the results, note the error message or condition that is displayed.
- ✔ If your Overland Data tape drive fails, note the error code or condition. Have the tape drive manual available.
- ✔ If the condition is an operational problem, make sure the tape drive is loaded and online (if possible) and the software and tape drive manuals are available.

Bulletin Board System

Overland Data maintains a 24-hour bulletin board system (BBS) for customers at no charge. Customers may dial into the BBS at anytime and download any standard DOS software uatility or package.

For optional fostware, contact Overland Data technical support with proof-of-purchase information. You will be given the access codes to download the optional software.

BBS Access

(619) 571-3651	9600 baud, 8 bits, 1 bit stop	
	Protocols:	X-modem Y-modem

Chapter 2 - About 9-track Tape

Since 9-track drives use a standard format, software relies on consistent tape format and physical characteristics.

Physical Characteristics of 9-Track Tape

Standard tape reel diameters are 10.5, 8.5, or 7 inches. A 10.5-inch reel holds 2400 or 3600 feet of tape, depending on tape thickness. The 2400-foot tape length is the most common. An 8.5-inch reel holds 1200 feet of tape and a 7-inch reel holds 600 feet.

Each reel has two reflective strips on the back of the tape. The Beginning of Tape (BOT) marker is about 15 feet from the start of the tape. Tape drives "see" this marker and place the head just past it when loading or rewinding. The End of Tape (EOT) marker is about 25 feet from the tape's end. The EOT marker warns software that the tape end is near. Drives can record past this, but must quickly finish writing once the EOT marker is seen to keep the tape from coming off the reel.

Data is recorded in "blocks" separated by a small amount of blank tape called an inter-record gap (IRG). A file is one or more blocks separated from the next file by a special block called a filemark (or tapemark). Two filemarks signal the end of data (called logical EOT)—usually before physical EOT.

9-track drives have a safety feature that prevents writing over data on tape. Usually, a tape is "write-protected" if there is no *write-enable ring* placed on the back of the reel. The write-enable ring allows the drive to write data to the tape. A reel *without* the ring keeps the drive from writing.

Tape Format and Density

Density is the number of bits stored on a linear inch of tape. Density is measured in bytes-per-inch (bpi), but does not include inter-record gaps (IRGs).

Tape format describes the recording technique used by the drive. The three formats listed below are *standard* formats that correlate to specific densities:

- **NRZI** (*non-return to zero inverted*)

  The oldest and least common format. Density is 800 bpi.
- **PE** *(phase-encoded):*

  Recording density is 1600 bpi or 3200 bpi. 1600 bpi is the most common for data interchange. Because twice as much data fits on a 3200 bpi tape, it is used for backup more than data interchange.
- **GCR** *(group-coded recording):*

  GCR records at 6250 bpi. GCR is a common interchange format, especially between mainframes. Because GRC has a high data capacity, it is also used for backup.

Many drives support several densities. These are described as, double-, tri- or quad-density.

Data Bytes

9-track drives write nine parallel data tracks for *each* data byte. There is one track for each of the eight data bits and one track for parity. The parity track is used to detect and correct read and write errors.

Tape Capacity

Tape capacity depends on tape length, block size, IRG size and density. The formula for computing tape capacity is as follows:

$$\text{TAPE CAPACITY} = \frac{\text{TAPE LENGTH}}{\frac{\text{BLOCK SIZE}}{\text{DENSITY}} + \text{IRG}} \times \text{BLOCKSIZE}$$

TAPECAPACITY	tape capacity measured in bytes
TAPELENGTH	tape length measured in inches
BLOCKSIZE	block size measured in bytes
DENSITY	recording density measured in bytes per inch
IRG	length of the inter-record gap in inches.

This table below lists tape capacity for various block sizes, assuming a 0.6 inch IRG at 1600 and 3200 bpi, and a 0.3 inch IRG at 6250 bpi, and a 2400-foot reel of tape.

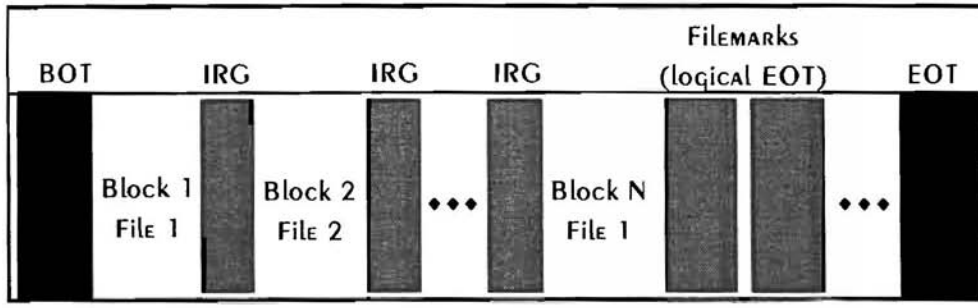
Density (bpi)	Block Size (bytes)					
	512	1024	3072	5120	10240	16384
	Capacity (shown in millions of bytes)					
1600	16.0	23.7	35.1	38.8	42.1	43.5
3200	19.4	32.0	56.7	67.0	77.6	82.4
6250	38.6	63.6	112	132	152	162

Tape Layout

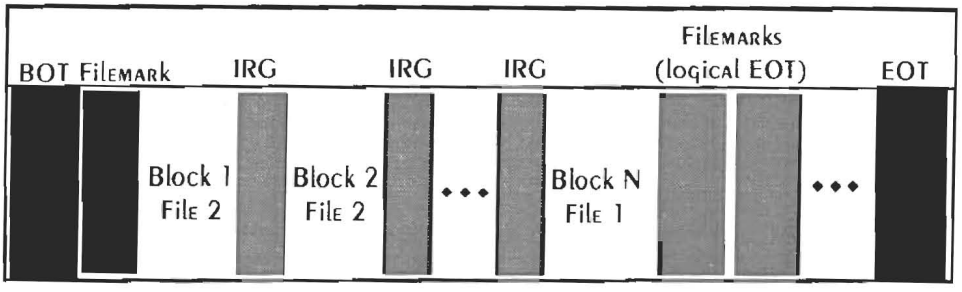
The combination of filemarks and data blocks on a tape is called the tape layout. Any combination of data blocks and filemarks is possible, except that data block size is generally limited to 32Kb or less and two consecutive filemarks must be written to denote logical EOT. Logical EOT is used to designate the end-of-valid data on a tape.

Unlabeled Tapes

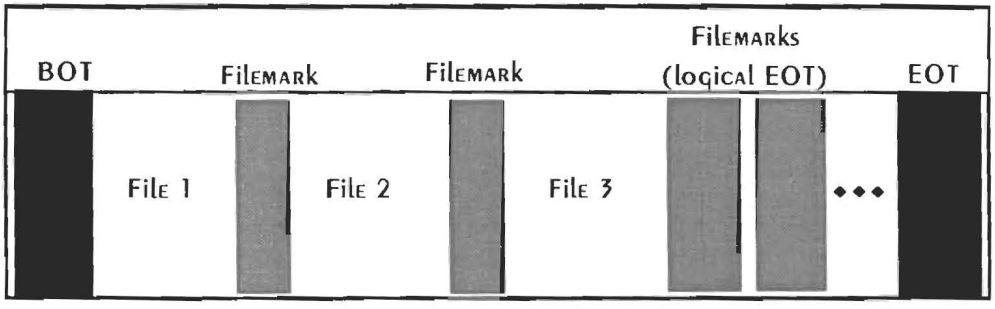
The generic layout is "unlabeled tape." For example, a tape with a single file. The file contains blocks separated by IRGs and ended by two filemarks. This diagram shows such a file, of ***N*** blocks.



Some tapes are written with an initial file mark, as shown below:



Tapes with two or more files use the same format. There is one filemark between files and two filemarks after the last file showing the logical EOT. The blocks in each file are separated by IRGs. The diagram below shows the layout for three files. IRGs are not shown.

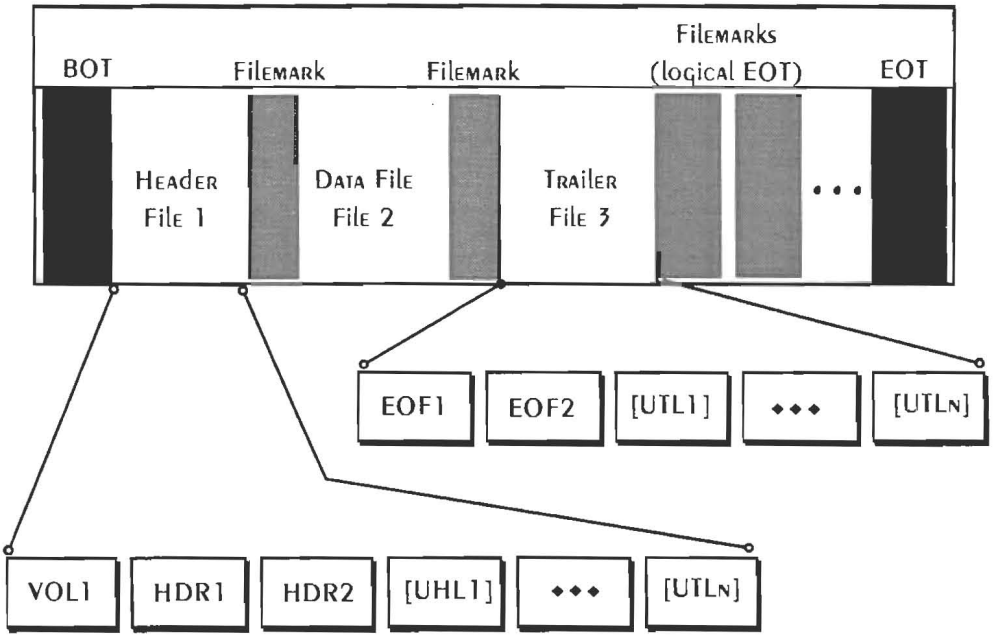


Labeled Tapes

In general, labeled tapes have at least three files:

- a header file describing the data
- data file
- trailer file

Other data files may follow with the same format: header, data, trailer. This diagram shows a labeled tape with a single data file.



There are two label standards in general use today, IBM and ANSI. When you select **IBM,** labels (header and trailer) are specified to hold only EBCDIC characters. If you select **ANSI,** labels use only ASCII characters.

Both label standards are 80-byte blocks and define similar information in the headers and trailers. For the IBM standard, refer to IBM Publication *OS/VS Tape Labels , GC26-3795-3* available from your local IBM branch office. For the ANSI standard, refer to *American National Standard Magnetic Tape Labels and File Structure for Information Interchange X3.27-1978.* Copies may be ordered from: American National Standards Institute, 1430 Broadway, New York, NY 10018

For both IBM and ANSI labeled tape, the label header is a header file of at least three 80-byte blocks. Further blocks are optional user-defined labels.

- These blocks are:
- volume label (VOL1)
- header 1 label (HDR1)
- header 2 label (HDR2).

Optional user labels (UHL1–UHL8) follow. The header file ends with a filemark.

The header and trailer labels use identical formats, except there is no VOL1 trailer. End of file trailer 1 (EOF1) follows the data file and filemark. It has the same format as the HDR1 label. The end of file trailer 2 (EOF2) follows EOF1. Any user trailers, called UTL1 through UTL8, follow EOF2. Two filemarks end the last trailer.

Tape Data

To understand how data is stored on a tape, think of the way information is kept in a filing cabinet. Each drawer of the filing cabinet is like a reel of tape. Like a tape, the cabinet contains one or more files. The files are marked with tabs, which are similar to the filemarks on a tape. The information in the tabbed files is then divided into manila folders. Each folder is like a block of data on tape. Each folder can have many pieces of paper in it, which is similar to having several logical records in a block on tape.

Data Blocks

The unit of data on a tape is a data block. Sometimes, people use the word "record" when referring to a block. In this manual, the term *block* is used. Each block of data is separated by an inter-record gap (IRG).

The term *record* is used in this manual for the logical grouping of the data. In many cases the record size is the same as the block size; that is, a block *contains* a single data record. More often, blocks contain *multiple* records (20, 100, 500, etc.). The number of records per block is the blocking factor. The block size is an even multiple of the record size. Typically, a program writes data to tape in a series of fixed-sized blocks, ending the data transfer with two filemarks.

Choosing an Appropriate Block Size

There is actually no "correct" block size, but a range of sizes appropriate for an application. If data is in fixed-size records, the block size will be a multiple of the record size.

For example, if you have a database of 10,000 records and each record is 230 bytes long, you can pick a block size that is an even multiple of 230 (such as 230, 4600, etc.). Remember: block size affects tape capacity and tape drive data rate. The larger the block size, the higher the tape capacity and the higher the tape drive data rate.

In the example, 4600 is a "medium-sized" block, offering the following:

- an acceptable tape capacity
- optimum tape drive data rate
- high compatibility within an unknown configuration.

Records

A record is a logical data grouping, which can be a fixed length or variable-length. Fixed-length records are convenient for writing to tape (accommodating blocking characteristics). Often, data is a fixed-size because it is stored that way in a database. If the data is not fixed-size (e.g., a text file), each line of text can be placed in a record (say, 100 bytes) and "blank-padded" to a fixed-length size.

In the IBM format, variable-size records use a 4-byte header (in zoned decimal) before the data, showing the record length (including header). This is used with the variable-size blocks which have a 4-byte block size indicator, a 4-byte record size indicator, and data for the first record. The next record appears with a 4-byte record size indicator followed by *that* record's data.

Data Fields

Data records can be broken into individual fields. For fixed-size records, each record has the same number and sizes of fields. Different fields, however, have various sizes and formats. If the entire record is character data, each field will consist of ASCII or EBCDIC values. It's possible for the fields to be in a number format, such as Binary Coded Decimal (BCD), binary 2-byte integer, binary 4-byte integer or one of several floating point formats. The next section discusses standard formats found in 9-track tape data.

Number Formats

This section provides a description of formats other than the standard ASCII or EBCDIC characters that may be encountered when reading or writing 9-track tapes. These formats can be translated by the OD TapeView program, which is available separately.

Format	Size (bytes)	Format	Size (bytes)
Signed Integer	1, 2 or 4	Unpacked BCD	open
Unsigned Integer	1, 2 or 4	Comp3	open
Microsoft Floating Point	4 or 8	IBM Comp3	open
IEEE Floating Point	4 or 8	Zoned Decimal	open
Packed BCD	open		

Format Specifications

Signed Integers

Signed integers all use the leftmost bit as a sign bit (1 for negative, 0 for positive).

- *One Byte Signed*

  Values range: -128 (0x80) to +127 (0x7F) and are 8 bits wide.

- *Two Byte Signed*

  Values range: -32,768 (0x8000) to +32,767 (0x7FFF) and are 16 bits wide.

- *Four Byte Signed*

  Values range: -2,147,483,648 (0x80000000) to +2,147,483,647 (0x7FFFFFFF) and are 32 bits wide.

Unsigned Integers

Unsigned integers do not use a sign bit.

- *One Byte Unsigned*

  Values range from 0 (0x00) to 255 (0xFF) and are 8 bits wide.

- *Two Byte Unsigned*

  Values range from 0 (0x0000) to 65,535 (0xFFFF) and are 16 bits wide.

- *Four Byte Unsigned*

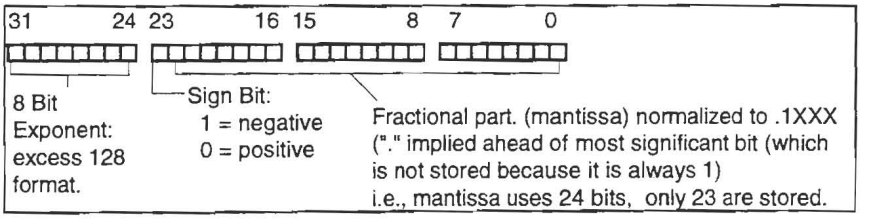
  Values range from 0 (0x00000000) to 4,294,967,295 (0xFFFFFFFF) and are 32 bits wide.

Microsoft Floating Point

Microsoft floating point numbers' range is 1.7E+38 to 2.9E-39.

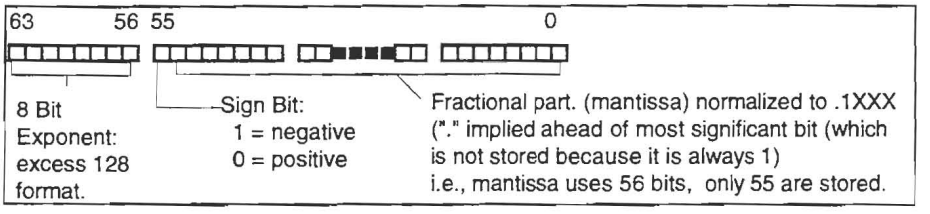
- *Four Byte Floating Point*

  The bias of the exponent is 128 and the mantissa is normalized to between .1 and .2. A zero is represented by an exponent of all zeros (the other bytes don't matter). The format is:



- Eight Byte Floating Point

  These have the same range as the 4-byte numbers, but 8-byte format's accuracy is 15 decimal places. The format is:

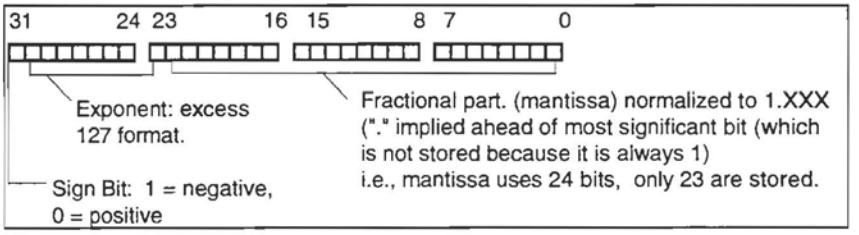


Note: the difference between 32 bit and 64 bit format is the number of bits allocated to the mantissa.

IEEE Floating Point

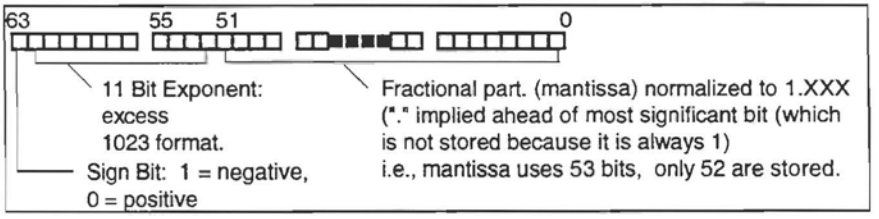
- Four Byte Floating Point

  Four byte IEEE floating point numbers range from about 3.4E-38 to 3.4E+38. The format of this 32 bit number has three fields: 1 sign bit, an 8 bit biased exponent, and a 23 bit mantissa. The bias of the exponent is 127 and the mantissa is normalized to between 1 and less than 2. The format is:



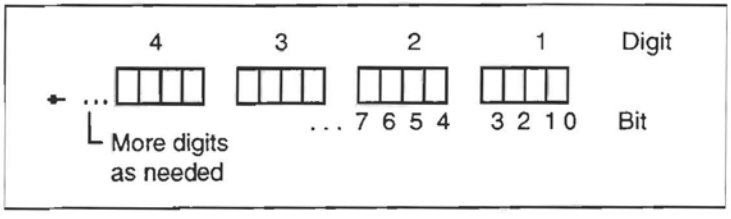
- Eight Byte Floating Point

  Eight byte IEEE floating point numbers range from about 1.7E-307 to 1.7E+308. The 64 bit number has 3 fields: 1 sign bit, an 11 bit biased exponent, and a 52 bit mantissa. The bias of the exponent is 1023. It looks like:



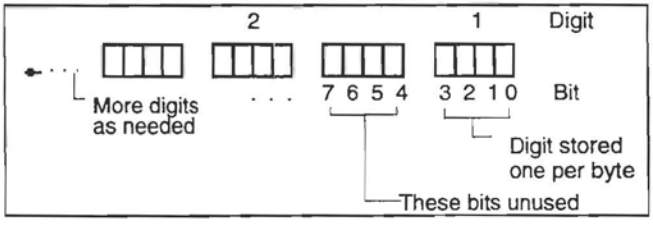
Packed BCD

For packed BCD (Binary Coded Decimal) each digit is stored in a 4-bit nibble (half a byte). Binary numbers 0000 to 1001 represent the digits 0 to 9. Digits are "packed," 2 per byte, for a number of any length. BCD *usually* represents an even number of digits, but the most significant digit can be discarded. BCD does not have a decimal point, but can imply one. The format is:



Unpacked BCD

BCD may also be unpacked, with only one digit per byte. This is not as commonly used as packed BCD (it wastes storage). Number representations are the same as packed BCD, but the high four bits in each byte are not used. The format is:



Chapter 3 - Hardware Installation & Setup

This chapter describes installation and setup procedures for Overland Data 9-track controllers: TX-8, TXi-16, XL/2, and TMC. Chapter 3 also tells you how to configure the ASPI (SCSI) and parallel* port interfaces.

All controller cards use programmed input/output (I/O) instead of direct memory access (DMA) for data transfer. The TX-8 and XL/2 both have 64kB on-board cache buffers, the TXi-16 has a 1MB cache buffer. The TXi-16 improves performance by using 16-bit I/O. The DOS software uses loadable drivers, making it device independent.

NOTE: The following warning applies to all Overland Data tape drive controller cards; the TX-8, TXi-16, XL/2, TMC-ISA and TMC-MCA . It also applies to the parallel interface on Overland Data tape drives. All have been tested and meet the standards set for Class A computing equipment.

WARNING: This equipment generates, uses and can radiate radio frequency energy, and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. The equipment has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of the FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

---

* Available on some Overland Data tape drives.

TX-8 Controller Card

This section describes the TX-8 and installation procedures. Before you install the TX-8, make sure you have everything listed in the inventory checklist and the documentation for your PC. Also make sure the TX-8 controller card is correctly configured and that you *understand* the procedures.

Standard Features

The standard TX-8 features are:

- 64kB cache buffer.
- Switch selectable I/O address (8 contiguous ports required).
- Modes: GCR, PE and NRZI.
- Densities: 800, 1600, 3200, and 6250 bytes per inch (bpi).
- Compatible with most Cipher/Pertec formatted tape drives.
- Burst transfer rate: 900kBs. Sustained rate: 632kBs. (System throughput depends on the PC, program, tape condition and hard disk attributes).

Inventory Checklist

Make sure your TX-8 package includes the following items:

- TX-8 controller board
- One shielded cable
- One P-clamp for grounding the cable to the drive
- 5.25 inch or 3.5 inch installation diskette (the number of disks varies)

NOTE: You *must* be familiar with DOS and your computer hardware to install the TX-8. Operating system commands and I/O port designations assume single-user, single tasking MS-DOS.

System Requirements

- Any IBM PC, XT, AT or fully compatible machine that operates under MS-DOS.
- A minimum of 512kB of RAM.
- No DMA channel is required for operation.
- The DOS device driver (Options Manual) is *NOT* required to use the DOS programs provided with the controller. Installed, the driver uses memory. Do not install the DOS device driver unless you plan to access the tape system using your own software.

Installation sequence for the TX-8

The following sequence should be followed when installing a TX-8 for the first time. You may not have to adjust the hardware configuration if no conflicts are encountered with the defaults* .

1. Configure the controller, if needed. (*Detailed TX-8 Installation*)
2. Install the controller. (Go to *Quick TX-8 Installation* or *Detailed TX-8 Installation*).
3. Install the software. (Go to *Quick Software Installation* or *Detailed Software Installation*).
4. Configure the software (*Depot4, Options Menu, Interface Menu*)
5. Test your subsystem. (Go to *Depot4* in Chapter 5, under the heading, *Utilities, Test the Tape Drive*).

* IRQ : 0 (timer-tick), I/O address : 360h

Quick TX-8 Installation

The following is a quick procedure for installing your TX-8 controller. The steps provide a brief description of a standard installation. This procedure assumes that your installation is normal and without problems. Refer to the heading, *Detailed Hardware Installation,* if you encounter problems.

1. Turn your PC off and unplug all peripherals.
2. Open the PC case.
3. Find an open slot and install the TX-8 controller card. The card *must* be inserted into a 16-bit slot (two plug-in, card edge connectors, one long, and one short).
4. Plug the cable into the single, D-type connector on the TX-8 controller, located at the rear of the PC. Tighten both screws to secure the connector and provide a good ground.
5. Reinstall the PC cover.

NOTE: The next step attaches the other end of the cable to the tape drive. Before connecting the cable, note that pin-1 is marked on both the tape drive, below the P1 and P2 designators, and on the cable connectors, J1 and J2.

---

6. Connect the cable to the tape drive. J1 attaches to P1 and J2 to P2. Make sure the pin-1 designators on J1 and J2 match the pin-1 designators on P1 and P2.
7. Install the P-clamp (supplied with cable) on the exposed metal shielding of the tape drive end of the cable. Attach the P-clamp to the tape drive grounding point (shown below) using the 8-32 X 3/8" screw supplied with the cable.
8. Go to the heading *Quick Software Installation* or *Detailed Software Installation* (Chapter 4).

Detailed TX-8 Installation & Configuration

Configuring the TX-8 Controller

1. It may not be necessary to perform this step if you do not have an I/O address conflict with another card in your system. However, if you know of an I/O address conflict between the TX-8 and another adapter in your system, see the table below to select a new I/O base address.

   You can change the I/O base port on a TX-8 controller without reconfiguring your software if you select a new base address that appears in the table below. Unlisted base-port addresses require you to set the correct DIP switches and reconfigure the software in *Depot4* under the *Options* menu.

   NOTE: Make a note of the new address for reference during configuration.

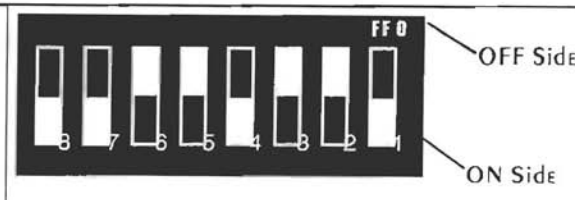
DIP Switch	1	2	3	4	5	6	7	8
Address Line	**A10**	**A9**	**A8**	**A7**	**A6**	**A5**	**A4**	**A3**
360 (default)	ON	OFF	OFF	ON	OFF	OFF	ON	ON
300	ON	OFF	OFF	ON	ON	ON	ON	ON
310	ON	OFF	OFF	ON	ON	ON	OFF	ON
320	ON	OFF	OFF	ON	ON	OFF	ON	ON
330	ON	OFF	OFF	ON	ON	OFF	OFF	ON
340	ON	OFF	OFF	ON	OFF	ON	ON	ON
350	ON	OFF	OFF	ON	OFF	ON	OFF	ON

NOTE: Step 2 references two types of TX-8 architecture. There is no electronic difference in the controllers, however, the physical architecture is different.

2. For the purposes of this procedure we will call the controllers *Type-I* and *Type-II*. The type can be identified by noting the DIP-switch orientation on the controller. The switch labels (numbered 1 through 8) on *Type-I* are up-side-down. *Type-II* switch labels are oriented normally. Use the following tables as a guideline to set the DIP switches.

Type-I TX-8 Architecture

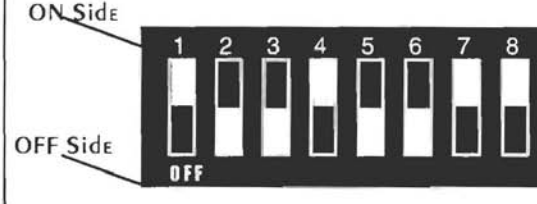
Switch	Setting
1	ON
2	OFF
3	OFF
4	ON
5	OFF
6	OFF
7	ON
8	ON



Note the position of the rocker switches. The rockers are DOWN on the active side. For example: switches 1, 4, 7 & 8 are ON; thus they are depressed TOWARD the Numbers.

Type-II TX-8 Architecture

Switch	Setting
1	ON
2	OFF
3	OFF
4	ON
5	OFF
6	OFF
7	ON
8	ON



Note the position of the rocker switches. The rockers are DOWN on the active side. For example: switches 1, 4, 7 & 8 are ON; thus they are depressed TOWARD the Numbers.

5. This completes the TX-8 controller configuration.
6. Go to *Installing the TX-8 Controller,* below.

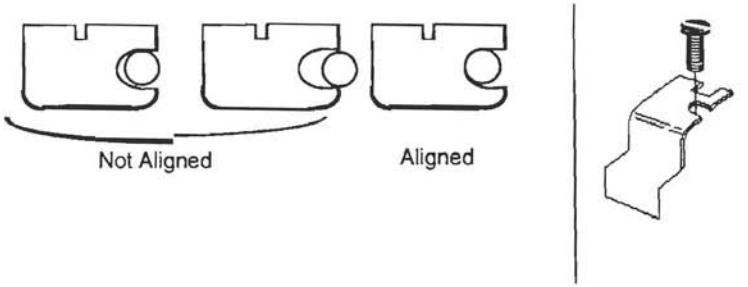
Installing the TX-8 Controller

1. Turn the computer off and unplug all peripherals.
2. Remove the computer cover.
3. Look at the computer motherboard to see if you have an available 16-bit slot. You should use the double slots (16-bit) when installing the adapter in an XT or AT. Do not install in this step.

Important Notes

- The TX-8 may be installed in an 8-bit slot. However, the controller will not have the use of extended interrupts.
- When you use an 8-bit slot, you must change the jumper on the TX-8 from the IRQ default (10) to an IRQ of 3, 4, 5, 6, or 7. The IRQ you select must be one that is *not* used by other peripherals on your system.
- If you install the TX-8 in an 8-bit slot, it is important that the extended connector does not touch (short against) anything!

4. Remove the screw that secures the card blank in place. Loosen the screw in a counterclockwise direction. Be sure to save the screw; it will be used to install the product later.
5. Install the TX-8 card by pressing it into the expansion slot. Align the bracket with the computer panel. Replace the screw.



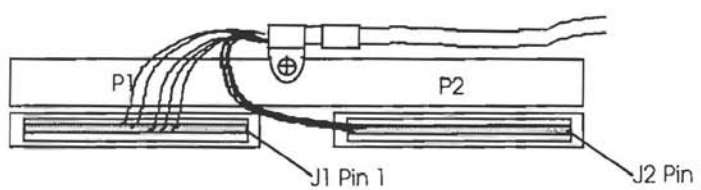
Aligning the TX-8 retaining bracket

6. Replace the cover and tighten screws.

5. This completes the TX-8 controller configuration and installation.
6. Go to *Installing the Cable,* below.

Installing the Cable

1. Starting at the PC end, plug the single, D-type connector into the TX-8 connector. Secure by tightening both connector screws.
2. Working from the tape drive end, locate the two card-edge connectors at the rear of the tape drive. These connectors are normally labeled as J1 and J2. Some drives have two sets of connectors(refer to your drive manual for more detail). Use either set as long as it is J1 and J2.
3. Before connecting the cable, note that pin-1 is marked on both the tape drive, below the P1 and P2 designators, and on the cable connectors, J1 and J2. Connect J1 to P1 and J2 to P2. Make sure the pin-1 designators on J1 and J2 match the pin-1 designators on P1 and P2.



Pertec Cable Orientation-(3210 & 3610)

4. This completes the TX-8 cable installation.
5. Go to *Grounding the Cable,* below.

Grounding the Cable

1. The tape drive cable must be grounded to the drive to meet FCC requirements for radio frequency emissions. Install the P-clamp (supplied with cable) on the exposed metal shielding of the tape drive end of the cable.

2. Attach the P-clamp to the tape drive grounding point using the 8-32 X 3/8" screw supplied with the cable.

3. This completes grounding the TX-8 cable.
4. Go to *Quick Software Installation* or *Detailed Software Installation* in this Chapter.

TX*i*-16 Controller Card

This section describes the TX*i*-16 controller installation and configuration procedures. Before you install the controller, read this entire section. Make sure you understand the procedures and that you also have all parts in the *Inventory Checklist*.

WARNING

This equipment generates, uses and can radiate radio frequency energy and if not installed and used in accordance with the instruction manual, may cause interference to radio communications.

The equipment has been tested and found to comply with the standards for a Class A computing device pursuant to Subpart J of Part 15 of the FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment.

Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

Standard Features

- 1MB cache buffer.
- Configuration data is stored in NOVRAM (non-volatile RAM).
- On-board microprocessor.
- 16-bit architecture.
- Supported densities[1] are: 800bpi (NRZI), 1600 bpi (PE), 3200 bpi (PE), and 6250 bpi (GCR).
- Compatible with most Cipher/Pertec formatted tape drives
- Burst rate: 2 MB/s. Sustained[2] transfer rate: 1.3 MB/s .

[1] Not all densities are available on all drive models.

[2] Throughput is dependent on the computer, the program, the condition of the tape, and attributes of the hard disk.

Inventory Checklist

Make sure your TX*i*-16 package includes the following:

- TX*i*-16 circuit board with bracket
- One round, shielded cable
- TX*i*-16 configuration tool (plastic rod), approximately (1/8" X 4").
- One P-clamp for cable grounding
- Distribution software on either 5.25-inch or 3.5-inch diskettes.

Installation sequence for the TXi-16

The following sequence should be followed when installing a TXi-16 for the first time:

1. Install controller. (Go to *Quick TXi-16 Installation* or *Detailed TXi-16 Installation*).
2. Place controller in *Configuration Mode*. (Go to *Preparing the TXi-16 for Configuration Mode*)
3. Install software. (Go to *Quick Software Installation* or *Detailed Software Installation*).
4. Configure software and controller. (*Depot4, Options Menu, Interface Menu*)
5. Test your subsystem. (Go to *Depot4* in Chapter 5, under the heading, *Utilities*, *Test the Tape Drive*).

Quick TXi-16 Installation (for experienced users)

The following is a quick procedure for installing your TXi-16 controller. The steps provide a brief description of a standard installation. This procedure assumes that your installation is normal and without problems. Refer to the heading, *Detailed Hardware Installation,* if you encounter problems.

1. Turn the computer off and unplug all peripherals.

2. Open the PC case.

3. Find an open card slot and install the TX-16i card. The card *must* be inserted into a 16-bit slot (two plug-in, card edge connectors, one long, and one short).

4. Plug the cable into the single, D-type connector on the TXi-16 controller card. Tighten both screws to secure the connector and provide a good ground.

5. Reinstall the PC cover.

The next step is used to attach the other end of the cable to the tape drive. Before connecting the cable, note that pin-1 is marked on both the tape drive, below the P1 and P2 designators, and on the cable connectors, J1 and J2.

6. Connect the cable to the tape drive. J1 attaches to P1 and J2 to P2. Make sure the pin-1 designators on J1 and J2 match the pin-1 designators on P1 and P2.

7. Install the P-clamp (supplied with cable) on the exposed metal shielding of the tape drive end of the cable. Attach the P-clamp to the tape drive grounding point using the 8-32 X 3/8" screw supplied with the cable.

NOTE: **Other 9-track tape drives have a similar configuration for connecting the interface cables.**

8. This completes the *Quick Hardware Installation.*

9. Go to *Quick Software Installation* or *Detailed Software Installation* (Chapter 4), then return here and go to step number 10.

10. Go to *Setting the TXi-16 Configuration Mode* in this chapter.

Detailed TXi-16 Installation

Installing the Controller

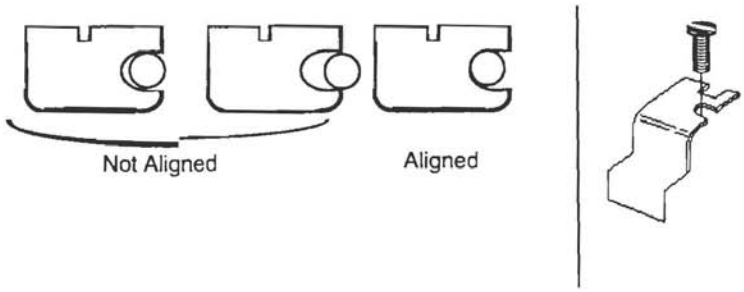
1. Turn the computer *off* and unplug all peripherals (note cable connections so that you can replace them later).
2. Move your keyboard and peripherals away from the work area. Place the PC in a position so you can work on it. Remove the cover
3. Select an open card slot.

NOTE: The TXi-16 must be inserted into a 16-bit slot (one long and one short slot). The board will not operate in an 8-bit slot.

4. Remove the screw holding the blank cover associated with the open card slot you selected in the previous step. Save the screw; it is used to install the card later.
5. Install the TX*i*-16 controller by grasping the top of the card then gently, but firmly, pressing it into the slot. The rear of the board must be aligned with the vertical card guide as it is installed. Align the board retaining bracket slot with the slot in the computer panel.

NOTE: Make sure the controller card is securely seated in both card slot connectors before continuing.

6. Reinstall the screw, and tighten securely in a clockwise direction.



Aligning the TX*i*-16 Retaining Bracket

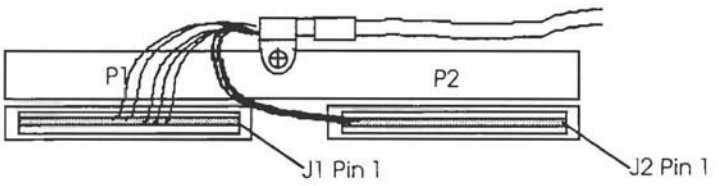
7. Reinstall the computer cover.
8. This completes the TXi-16 controller installation.
9. Go to *Installing the Cable*, below.

Installing the Cable

1. Starting at the PC end, plug the single, D-type connector into the TX*i*-16 connector. Secure by tightening both connector screws.

2. Working from the tape drive end, locate the two card-edge connectors at the rear of the tape drive. These connectors are normally labeled as J1 and J2. Some drives have two sets of connectors (refer to your drive manual for more detail). Use either set as long as it is J1 and J2.

3. Before connecting the cable, note that pin-1 is marked (see illustration below) on both the tape drive, below the P1 and P2 designators, and on the cable connectors, J1 and J2. Connect J1 to P1 and J2 to P2. Make sure the pin-1 designators on J1 and J2 match the pin-1 designators on P1 and P2.



Pertec Cable Orientation-(3210 & 3610)

4. This completes the TXi-16 cable installation.
5. Go to *Grounding the Cable,* below.

Grounding the Cable

1. The tape cable must be grounded to the drive to meet FCC requirements for radio frequency emissions. Install the P-clamp (supplied with cable) on the exposed metal shielding of the tape drive end of the cable.
2. Attach the P-clamp to the tape drive grounding point (shown above) using the 8-32 X 3/8" screw supplied with the cable.
3. This completes grounding the TXi-16 cable.
4. Go to *Quick Software Installation* or *Detailed Software Installation* in Chapter 4, then return here and go to step number 5.
5. Go to *Setting the TXi-16 Configuration Mode*.

TXi-16 Configuration—Overview

Proper controller configuration is essential for optimum performance. Configuring the controller is easy with the TXi-16 since it is done through the software. No hardware configuration (except setting the global configuration switch) is necessary. In all cases configuration should be done as part of your initial software and controller installation. However, if changes are necessary, the system can be reconfigured at any time.

When to Configure or Reconfigure

You should configure your controller :

- when you perform your initial installation.
- if you change tape drive types.
- if you encounter an I/O address conflict.

Setting the TXi-16 Configuration Mode

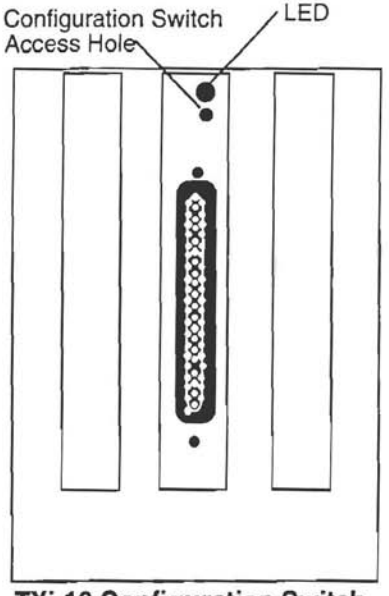
The TXi-16 has a hardware configuration switch that is used to set the controller in *Configuration Mode*. This switch, when activated, provides the capability to configure the controller while configuring the software in *Depot4*. The selected base I/O address and tape drive type are stored in the controller's NOVRAM (non-volatile RAM) after the configuration procedure is performed in *Depot4*.

The following procedure describes how to configure your controller:

1. Turn the computer off.

2. Position yourself so you can reach the front and rear panels[3] of the computer.

3. Locate the TX*i*-16 bracket on the rear of the computer.



TXi-16 Configuration Switch

4. Find the switch access hole (shown in the illustration above). The hole is located just below the LED.

[3] This is because you will have to turn on the power while holding a switch on the controller.

5. Use the configuration tool[4] or small rod to depress and hold the switch.

NOTE: Do not release the switch until the next step is complete.

The switch is recessed about an 1/8th inch behind the access hole and is not clearly visible. When you press the switch, you will feel slight pressure from the spring.

6. Depress and hold the switch, then turn the computer ON. In about 4 seconds, the LED will turn amber. This means the board recognizes that the controller is in *Automatic Configuration Mode*. Release the switch.

7. Go to *Configuring the TXi-16 Controller.*

The following table shows the different possible TXi-16 LED colors and states, and their meanings:

LED State & Color	Meaning
Steady Amber	The TXi-16 is in *Automatic Configuration* mode. The device driver or other software can now change the contents of the NOVRAM .
Steady Green	The checksum for the TXi-16 NOVRAM is valid– indicating a valid configuration* .
Steady Red	The checksum for TXi-16 NOVRAM is invalid . You must perform *Automatic Configuration* before the controller can be used.
Flashing Red	A fault was detected during the TXi-16 power-on self-test. Call Overland Data Technical Support.

[4] A clear Plexiglas rod supplied with the TXi-16 controller.

* When the TXi-16 is first installed, the default configuration, though valid, may not match the configuration parameters you selected during driver installation. Ensure that the controller has been configured to match your software if you changed the configuration in your software when you first install the TXi-16.

Configuring the TXi-16 Controller

For new installations, the following steps must be completed, in the order listed. You must complete these steps before attempting your initial configuration:

1. Make sure the computer and tape drive are ***off***.
2. Install the TXi-16 controller (see details, this chapter).
3. Connect the tape drive to the computer. (see details, this chapter).
4. Set and hold the *Hardware Configuration Switch* on the TXi-16 controller while you turn the computer ***on***.
6. Hold the switch until the LED turns an amber color then release.
7. Install the ODI software (see details, this chapter).

Perform the following steps to configure your controller:

1. If you are already in *Configuration Mode*, skip to step number 3.
2. Make sure the power to the computer and tape drive is off.
3. Set the hardware configuration switch as outlined, under the heading, *Preparing the TXi-16 for Configuration Mode*; then return here to step 3.
4. The controller should now in *Configuration Mode* (LED amber in color).
5. Access the *Depot4* program and select *Options* from the menu bar.
6. Choose *Interface* menu list.
7. Go to Chapter 5, and refer to the heading, *Interface–Selection and Configuration.*

8. Follow the procedure outlined under the sub-heading, *Selecting and Configuring the TXi-16 Cipher/Pertec Controller*; then go to step 9.
9. After the software configuration process is complete, the LED on the controller should change color from amber to green indicating that the controller accepted the configuration.
10. This completes the procedure for *Configuring the TXi-16 Controller*.
11. Go to Chapter 5, *Depot4*.
12. Test your subsystem by selecting *Test Tape Drive* from the *Utilities* menu.

XL/2 Controller Card

This section describes the XL/2 controller installation and configuration procedures. Before installing the card in your PC, read the entire XL/2 section. Be sure that you have all parts in the checklist, that you have the recommended system documentation, and that you understand the procedure.

Standard Features

The standard XL/2 features are:

- Automatic configuration
- Full hardware support of PS/2 Programmable Option Select
- Supports GCR, PE, and NRZI densities
- Densities: 800, 1600, 3200, and 6250 bytes per inch (bpi)
- Compatible with most Cipher/Pertec formatted tape drives
- Burst data transfer rate: 900kB/s. Sustained data transfer rate: 632kB/s. System throughput will depend on the computer, the program, the condition of the tape, and attributes of the hard disk

Inventory Checklist

Check that your XL/2 package includes the following:

- XL/2 controller board assembly
- One round, shielded cable
- One P-clamp for cable grounding
- Overland Data distribution software on a 3.5 inch diskette

NOTE: This manual assumes some familiarity with MS-DOS and the IBM PS/2 hardware. All references to operating system commands and I/O port designations in this manual assume operation under a single-user, single tasking version of the DOS operating system (OS/2 is not directly supported).

System Requirements

- The XL/2 works with any IBM PS/2 with the Micro Channel Architecture expansion bus (this includes Models 50, 60, 70, 75, 80).
- The supplied software requires a minimum of 512kB of RAM.
- No DMA is required.
- A copy of the IBM Reference disk is required to configure the XL/2 when it is installed.

Installation sequence for the XL/2

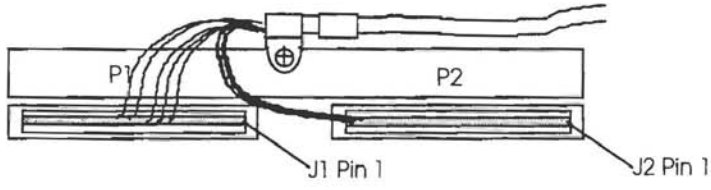
The following sequence should be followed when installing a XL/2 for the first time:

1. Install controller. (Go to *Quick XL/2 Installation* or *Detailed XL/2 Installation*, this chapter).
2. Install software. (Go to *Quick Software Installation* or *Detailed Software Installation*, in Chapter 4).
3. Configure the software by selecting the XL/2 interface. (*Depot4, Options Menu, Interface Menu*, in Chapter 5)
4. Configure the controller by using the working *Reference* diskette. (provided by your computer manufacturer)
5. Test your subsystem. (Go to *Depot4* in Chapter 5, under the heading, *Utilities, Test the Tape Drive*).

Quick XL/2 Installation (for experienced users)

The following is a quick procedure for installing your XL/2 controller. The steps provide a brief description of a standard installation. This procedure assumes that your installation is normal and without problems. Refer to the heading, *Detailed Hardware Installation,* if you encounter problems.

1. Turn your PC off and unplug all peripherals.
2. Open the PC case.
3. Find an open card slot and install the card. The card can be installed into a 16-bit slot or an 8-bit slot. You will have to configure your controller for the non-extended interrupts if you use an 8-bit slot.
4. Plug the cable into the single, D-type connector on the XL/2 controller, located at the rear of the PC. Tighten both screws to secure the connector and provide a good ground.
5. Reinstall the PC cover.
6. This step attaches the other end of the cable to the tape drive. Before connecting the cable, note that pin-1 (see the following illustration) is marked on both the tape drive, below the P1 and P2 designators, and on the cable connectors, J1 and J2. Connect the cable to the tape drive. J1 attaches to P1 and J2 to P2. Make sure the pin-1 designators on J1 and J2 match the pin-1 designators on P1 and P2.



Pertec Cable Orientation-(3210 & 3610)

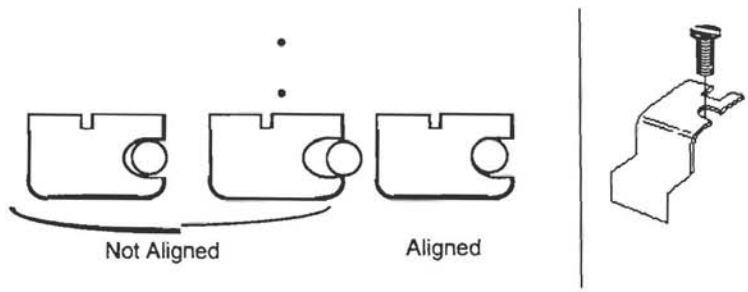
7. Install the P-clamp (supplied with cable) on the exposed metal shielding of the tape drive end of the cable. Attach the P-clamp to the tape drive grounding point (shown below) using the 8-32 X 3/8" screw supplied with the cable.

NOTE: Other 9-track tape drives have a similar configuration for connecting the interface cables.

8. This completes the *Quick Hardware Installation.*
9. Go to *Quick Software Installation* or *Detailed Software Installation* (Chapter 4).

Detailed xl/2 hardware Installation

1. Turn power off and unplug the computer and all peripherals. Place the PC in a position to gain access to the rear of the drive.
2. Remove all cover mounting screws (counter-clockwise) then remove the cover. Be sure to save all screws for the cover replacement.
3. Find an empty slot then remove the screw that secures the card blank. Loosen the screw in a counterclockwise direction. Be sure to save the screw; it will be used to install the product later.
4. Install the XL/2 by pressing it into the expansion slot. Align the bracket with the computer panel. Replace the screw.



Aligning the XL/2 retaining bracket

5. Replace the cover and tighten screws with a screwdriver in a clockwise direction.
6. This completes the XL/2 controller configuration and installation.

7. Go to *Installing the Cable,* below.

Installing the Cable

1. Working from the PC end of the cable, plug the single, D-type connector into the XL/2 connector . Secure by tightening both connector screws.
2. Working from the tape drive end, locate the two card-edge connectors at the rear of the tape drive. These connectors are normally labeled as J1 and J2. Some drives have two sets of connectors(refer to your drive manual for more detail). Use either set as long as it is J1 and J2.
3. Before connecting the cable, note that pin-1 is marked on both the tape drive, below the P1 and P2 designators, and on the cable connectors, J1 and J2. Connect J1 to P1 and J2 to P2. Make sure the pin-1 designators on J1 and J2 match the pin-1 designators on P1 and P2.
4. This completes the XL/2 cable installation.
5. Go to *Grounding the Cable,* below.

Grounding the Cable

1. The tape cable must be grounded to the drive to meet FCC requirements for radio frequency emissions. Install the P-clamp (supplied with cable) on the exposed metal shielding of the tape drive end of the cable.
2. Attach the P-clamp to the tape drive grounding point (shown above) using the 8-32 X 3/8" screw supplied with the cable.
3. This completes grounding the XL/2 cable.
4. Go to *Quick Software Installation* or *Detailed Software Installation* in Chapter 4, then return here and go to step number 5.
5. Go to Configuring the XL/2.

Configuring the XL/2

Configuration of the XL/2 Tape Controller is easy. There are no switches to set or jumpers to move. When the card installed in the computer, the software on the IBM Reference Diskette, uses the Automatic Configuration File from Overland Data (file @7E76.ADF). This file is installed with your standard *DataTools* software. Usually the Configuration Program automatically selects the appropriate settings. In some cases you may need to change them to solve a configuration conflict or to better match your environment.

Configuration Options for the XL/2

The following section describes all the configuration settings.

Configurable XL/2 options and default settings

Option	Default Setting
I/O Address	360 hexadecimal
Interrupt Level	10 decimal
Drive Address	0
Tape Density/Tape Speed	Normal
Start/Stop or Streaming	Streaming

Address

The XL/2 uses 8 contiguous I/O ports and can address any base address on even boundaries from 0000 to FFF8 hex. Much of the address space below 400 hex is used by IBM.

The default address is 360 hex. If this conflicts with another device*, make another selection.

* A conflict is signified with a * next to the selection.

Interrupt Level

The Micro Channel bus has 11 interrupt levels available. This option allows you to select any one of these or to disable the hardware interrupts

The default hardware interrupt level is 10 (decimal). If this conflicts with another device, select a different interrupt. Overland Data software uses the DOS system clock interrupt (IRQ 8) if the hardware interrupt is disabled (IRQ 0).

Drive Address

This selection allows you to specify which drive address the XL/2 uses when the unit is first powered-on. Normally you should specify drive address 0.

Tape Speed Select

Pin number 50 on the J2 connector is used on some drives to select either one of two speeds. The default setting is *Normal.*

Streaming or Start/Stop

Selecting *Streaming* or *Start/Stop* mode alters the commands sequence issued by the software to the drive. Streaming is the normal choice (for streaming tape drives). Some older drives have trouble with commands issued before the formatter-busy (IFBSY) line indicates that the formatter is not busy. This problem sometimes causes the tape drive to quit functioning until it is reset. In these rare cases, try the start/stop setting.

If you select start/stop on a streaming drive, you may experience degraded performance such as an increase in tape repositioning.

Configuring the XL/2 with the Reference diskette

After a new adapter card is installed, the micro channel BIOS detects the change in its hardware configuration. This results in an error message and a request for the Reference diskette. As the computer use a *Reference Diskette* (supplied by the computer manufacturer) and an *ADF* file (supplied by the controller manufacturer) to configure controller.

NOTE: To protect your master *Reference Diskette*, make a working copy before you start the configuration process.

The following steps are used to configure the PS/2 for the XL/2 :

1. Start the working copy of the Reference diskette.
2. Copy the XL/2 configuration file to the Reference disk.
3. Activate the XL/2 configuration file.

1 Start Reference Diskette

Locate the section in the *IBM Quick Reference* booklet that explains how to start the *Reference* diskette.

1. Insert the *Reference* diskette into drive A. Make sure the diskette clicks into place.
2. If the PC is off, turn it on. If on, press **[CTRL]** + **[ALT]** + **[DEL],** then release. This will reboot the system into DOS.
3. Wait for the Reference disk sign-on message. It will ask if you want to run auto-configure. Answer no (press N).

2 Copy XL/2 Configuration File to Reference Diskette

1. Select Copy an option diskette from the menu. Wait for the prompt for the option diskette to be placed in drive a:.
2. Place the DOS DataTools disk #1 in drive a: or b: and press W. Wait for the prompt to put Reference diskette in the drive.
3. Place the Reference diskette in drive a: and press W. Wait until the menu appears.

3 Activate XL/2 Configuration File

1. Select Set Configuration from the Main Menu.
2. Select Run Automatic Configuration from the Set Configuration Menu. Remove the Reference Diskette.
3. Press **[CTRL]** + **[ALT]** + **[DEL]** at the same time, then release them. This will reboot the system into DOS.

Test the System

These steps outline the subsystem test procedure.

1. Load a tape onto the drive: Load a scratch, write-enabled tape onto the tape drive. See your tape drive Operations Manual for the correct loading procedure.
2. This procedure will write to tape, so make sure the tape you choose has nothing important on it.
3. Working from *Depot4*, select *Utilities*.
4. Test your subsystem by selecting *Test Tape Drive*.
5. This completes the procedure for *Configuring and Testing the XL/2 Controller*.

Parallel Port Interface

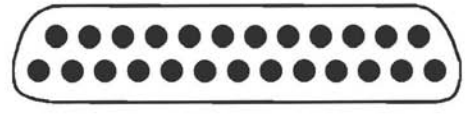
There are only three steps to installing your parallel port tape drive:

- Attach the parallel port cable to the drive and to your computer.
- Install and configure the software.
- Test the subsystem.

Parallel Port Setup

WARNING! **The parallel port cable must be grounded (see your tape drive manual) to the tape drive, using the P-clamp provided, and must remain grounded at all times. Failure to do so could cause your PC to malfunction or damage your PC's parallel port!**

1. The tape drive uses a parallel port to communicate with the computer. This provides the flexibility of sharing the tape drive with several computers. To determine if a printer port is available, examine the back of the computer for a female, 25 pin (connector has sockets, not pins) 'D' type connector (see illustration below).



Parallel Port Connector

If a port is not available, you need to purchase a parallel printer board from a computer dealer. These are common and inexpensive. Follow the instructions for setting up the parallel port card. Another alternative is to purchase one of the Overland Data controllers for 9-track tape drives.

2. Connect the supplied interface cable from the parallel port, on the computer, to J3, on the back of the tape drive.

Important Notes

- The tape drive cable *must* be the only device connected to the PC's parallel port. Some software vendors supply devices to protect their programs from unauthorized use which use the parallel port.

- If your system uses a software protection device in the parallel port, an alternate parallel port must be used for the tape drive. Your system may have already have an alternate parallel port or one can be added.

- If you are using a combination of Cipher/Pertec interface and a parallel port interface, commands from the Cipher/Pertec interface (J1 and J2) will be ignored once the cable is connected to the drive. We recommend that the drive be turned OFF when switching between controller card and parallel port operation.

3. This completes the *Parallel Port Setup Procedure.*
4. Go to the heading, *Parallel Port Configuration*, below.

Parallel Port Configuration

1. Working from the *Depot4* menu, configure your subsystem by selecting *Interface* from the *Options* menu. Refer to Chapter 5, *DataTools* for more information.
2. When configuration is complete stay in *Depot4*; test your subsystem by selecting, *Test Tape Drive*, from the *Utilities* menu. Refer to Chapter 5, *DataTools* for more information.
3. This completes the procedures for installation, configuration, and testing of the Parallel port interface.

TMC Controller Card

This section provides instructions for installing the TMC-ISA and TMC-MCA controller boards.

Make sure you know which controller you have. The TMC-ISA is marked TMC-1660 and the TMC-MCA is marked MCS-700. You should also review the correct instructions for your board. The instructions for the TMC-ISA (Industry Standard Interface) start below. The instructions for the TMC-MCA (Micro Channel Architecture) follow that.

The TMC-ISA requires an IBM AT or 386 or 486 computer, or a compatible computer, and that you run DOS 3.0 or later.

The TMC-MCA requires an IBM PS/2 or other Micro Channel-compatible computer, and that you run DOS version 3.0 or later.

The following procedures assume that you are using DOS operating system. If you use another operating system refer to the documentation that came with the operating system for questions.

TMC-ISA

Host Resources Required

- The TMC-ISA host adapter uses no DMA channels and no interrupts under PC-DOS and MS-DOS.
- The standard default configuration:
- Requires 8kB of memory space
- ROM address CA000 (reserved for ROM BIOS)
- Alternate memory addresses, beginning at C8000, CE000 and DE000 can be configured. Sixteen bytes of I/O space, beginning at hex address 0140, 0150, 0160, or 0170 are also used.

Factory (Default) Configuration

The TMC-ISA is factory configured as follows:

- BIOS/Host Adapter Address: CA000 (occupies CA000–CBFFF)
- I/O ports used: 0140 hex (occupies 0140–014F hex)

TMC-ISA Installation

Installation of the TMC-ISA requires the use of either a screwdriver or $^{3}/_{16}$-in. nut driver.

Opening Your PC

1. Turn off power to your PC.
2. Turn off all external equipment and unplug the PC power cord.
3. Remove the PC cover, following the instructions in the manuals provided by the PC manufacturer.
4. Find an unused 16-bit expansion slot in the PC. Using the screwdriver or $^{3}/_{16}$-in. nut driver, remove the screw holding the card blank by turning it counter-clockwise. Remove the card blank.

Installing Your Controller Board

1. Align the TMC-ISA with the expansion slot and firmly plug it in. Replace and tighten the screw to secure the board.
2. Plug in the high density connector of the shielded cable (ODI model TMC-112, TMC-113, or equivalent) into the connector at the rear of the TMC-ISA. Check that he connector securely clicks into place on both sides.
3. Connect the other end of the cable to the connector on the back of the drive.

Closing Your PC

1. Reassemble your PC, reviewing the instructions in the PC's manual as necessary.
2. After reassembling your PC, turn the power on. Your system is now ready for checkout and use.

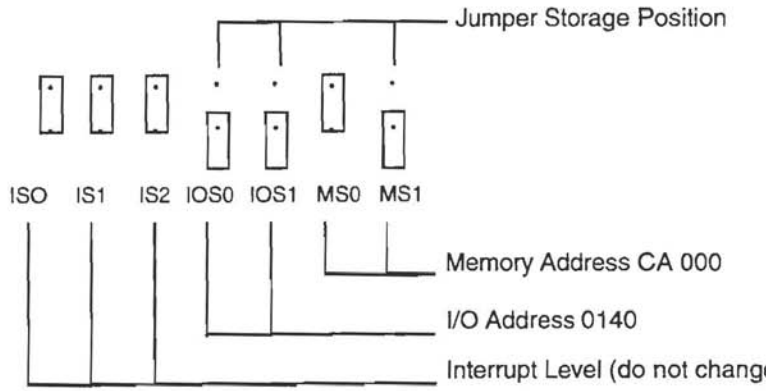
TMC Configuration and Testing

1. Working from the *Depot4* menu, configure your subsystem by selecting *Interface* from the *Options* menu. Refer to Chapter 5, *DataTools* for more information.
2. When configuration is complete stay in *Depot4*; test your subsystem by selecting, *Test Tape Drive*, from the *Utilities* menu. Refer to Chapter 5, *DataTools* for more information on the test procedure. If a conflict occurs, refer to the heading, *Address Conflicts—Changing ROM and/or I/O Address(s)* below.
3. This completes the procedures for installation, configuration, and testing of the TMC SCSI interface controller.

Address Conflicts—Changing ROM and/or I/O Address(s)

NOTE: If a conflict occurs, change the address as described below. If you are not sure if there will be a conflict, we suggest that you install the board using the standard (default) settings.

The jumper settings at W1 change both the ROM BIOS and/or the I/O address. To find W1, hold the board with the cable connection facing to your right and the board connectors facing down. The W1 shunt jumper set is now located on the lower right-hand side of the board.



Jumpers located at W1, Showing the Factory Settings

If your system encounters conflicts, or if you know that your system will conflict with another controller, alternate settings are shown below.

WARNING: **Do not change jumpers that are not shown or discussed. Doing so can degrade the operation of your system or cause conflicts with other controllers.**

Changing the ROM BIOS Address

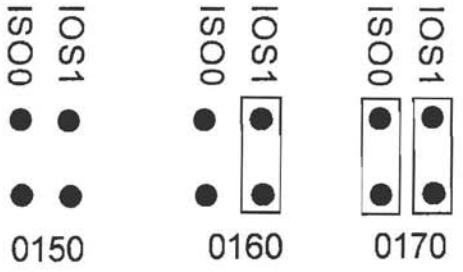
If you experience a conflict with the TMC-ISA, the problem is most likely the ROM BIOS address rather than an I/O address conflict. In such a case the problem will exhibit an error associated with the video card, network card, hard disk controller, or the like. The result of such a conflict is that both the TMC as-well-as the other device will not work. If you install the TMC-ISA and the PC fails to boot or fails to boot normally, you should change the TMC ROM BIOS address).

If you have a ROM BIOS conflict between the TMC-ISA and another ROM BIOS in your system, try the configurations as shown below:

Changing the I/O Port Address

If, after installing the TMC-ISA, your system operates strangely, it may be an I/O port address conflict (symptoms produced by conflicts with I/O address are more difficult to predict than ROM BIOS conflicts).

Change the jumper settings on IOS0 and IOS1 to a new address, as shown below. Try each of the settings, sequentially until the problem is corrected.



TMC-MCA

The TMC-MCA is an advanced design that uses the Micro Channel automatic configuration feature, eliminating, in most cases, any need for user configuration. The TMC-MCA host adapter uses no DMA channels and no interrupts under PC-DOS and MS-DOS.

The standard configuration uses 8kB of memory space and 16 bytes of I/O space. The standard configuration puts the TMC-MCA BIOS at memory address CA000 (hex), and the TMC-MCA I/O registers at 0140 (hex).

Address Conflicts

Conflicts are resolved automatically during configuration with the system Reference diskette.

TMC-MCA Installation & Configuration

Installation of the TMC-MCA requires the use of a medium-size, flat blade screwdriver and the IBM Reference Disk (we STRONGLY encourage you to use a backup copy of this disk).

Opening Your PC

1. Turn off power to your PC.
2. Turn off all external equipment and unplug the power cord.
3. Remove the PC cover, following the instructions in the manuals provided by the PC manufacturer.

Installing Your Controller Board

1. Find the expansion slots, numbered 1, 2 and 3 at the rear of the PC. Find a slot that is empty.
2. Remove the expansion card blank by loosening the screw and remove the expansion card blank.
3. Align the TMC-MCA board with the expansion slot and firmly plug it in. Replace and tighten the screw to secure the board in the slot.
4. Plug in the high density connector of the shielded cable (ODI model TMC-112, TMC-113, or equivalent) into the connector at the rear of the TMC-MCA. Check that he connector securely clicks into place on both sides.
5. Connect the other end of the cable to the connector on the back of the drive.

Closing Your PC

Reassemble your PC, reviewing the instructions in the PC's manual as necessary.

System Configuration

Configuration is a two step process. First you must tell the Overland Data software that the TMC-MCA is the active interface. Then you must configure the system software using the Reference diskette provided by your computer manufacturer. You should use a working copy of the Reference diskette. The following procedure describes how to configure your system using the Reference diskette. This information is also provided by your computer manufacturer.

1. Working from the *Depot4* menu, configure your Overland Data software by selecting *Interface* from the *Options* menu. Refer to Chapter 5, *DataTools* for more information on the *Interface* menu, then return here and go to step 2.

2. Insert your working copy of the IBM Reference Diskette in Drive a:.

NOTE: You should use a backup copy of the IBM Reference Disk. You will be copying files from the Option Disk supplied with your TMC-MCA to your backup Reference Disk. You cannot copy files to the original Reference Disk.

NOTE: Your system may vary from the system described in the screens below. This should have no effect on installation.

2. Apply power to the system. The cursor should flash within 30 seconds (if not, reseat the card). You should see:

```
1024KB OK

165
```

3. The system should boot from the Reference diskette. The screen will indicate a 165 error; *option not set up in the system*. Do *not* execute automatic configuration. Instead, type N to return to the main menu. The following screen should appear:

```
IBM Personal System/2 model 90/60
        Reference Diskette
            Version 1.0
(C) Copyright IBM Corp. 1981...1987
  Press Enter (↵) to continue...
```

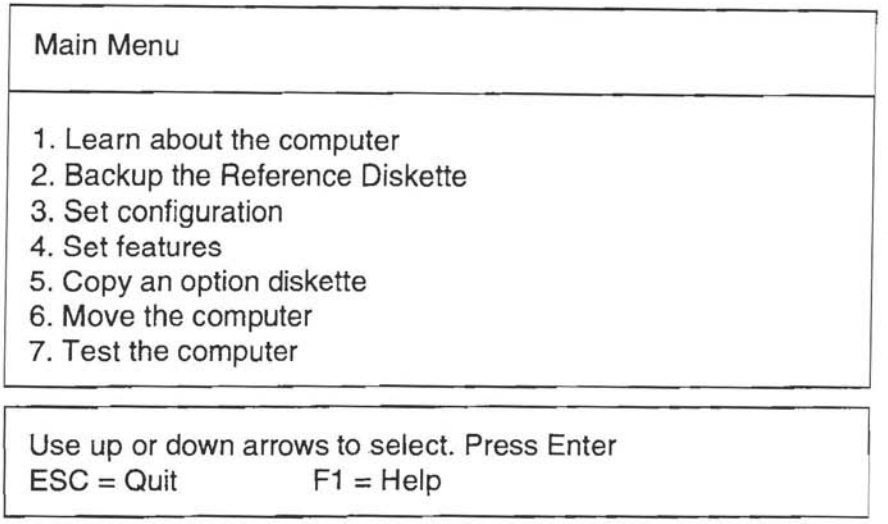
4. Press **[ENTER]**. The following screen should appear:

Adapter Configuration Error - 00165 Page 1 of 2
The Computer's internal self-tests found an option adapter that is different from the option adapters indicated in the computer's configuration. This error occurs if option adapters are added, removed, or are not working properly. If you have added or removed an adapter, run automatic configuration. To view or change the results of automatic configuration, go to the Main Menu of this diskette and select "Set configuration."
Page Down

5. Press **[PGDN]** to continue.

Adapter Configuration Error - 00165 Page 2 of 2
Select "View configuration" or "Change configuration" from the Set Configuration menu. If you have added or removed an adapter, do not run automatic configuration. Go to the Main Menu of this diskette. Select "Test the computer" to determine the cause of the error and what action to take.
Automatically configure the system? (Y/N) Page Up

6. Answer N to automatic configuration.



7. If you have not made a copy of the Reference Disk, do so now by selecting 2, and pressing **[ENTER]**. Follow the instructions on the screen. Return to this when you are done.
8. Select 5, Copy an option diskette.

9. Put the DOS DataTools disk #1 in drive A. If you cannot find this disk, contact Overland Data. You must have this disk to configure the adapter. Press **[ENTER]**. The system will read the disk and the following screen will appear:

Main Menu
1. Learn about the computer 2. Backup the Reference Diskette 3. Set configuration 4. Set features 5. Copy an option diskette 6. Move the computer 7. Test the computer

Information
Insert your backup of the Product 2 Diskette in Drive A:
Press Enter to continue ESC = Quit

10. Put the BACKUP Reference Disk in Drive A:.

NOTE: IBM calls the Backup Reference Diskette the Product 2 Diskette on some screens. These are the *same* disk.

11. Press **[ENTER]**. You will see the following:

Main Menu
1. Learn about the computer 2. Backup the Reference Diskette 3. Set configuration 4. Set features 5. Copy an option diskette 6. Move the computer 7. Test the computer

Information
Update of the Product 2 Diskette complete.
Press Enter to continue.

12. Press **[ENTER]** to return to the Main Menu. Select 3, *Set Configuration* and press **[ENTER]**. You will see the following:

Set Configuration
1. View configuration 2. Change configuration 3. Backup configuration 4. Restore configuration 5. Run automatic configuration

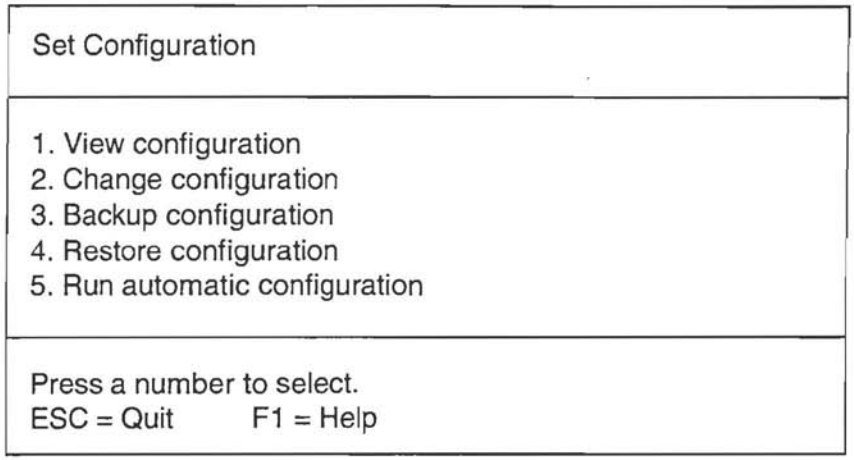
Press a number to select. ESC = Quit F1 = Help

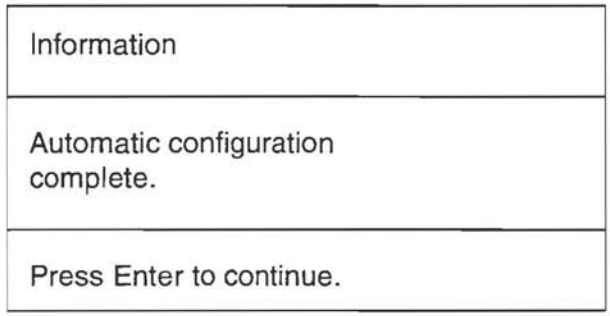
13. Select 5, *Run automatic configuration*. You may see a warning (shown below). Press **[ENTER]** to continue.

Set Configuration
1. View configuration 2. Change configuration 3. Backup configuration 4. Restore configuration 5. Run automatic configuration
Press a number to select. ESC = Quit F1 = Help

Warning:
If changes were made to the configuration in the Change configuration screen, they will be reset to their normal settings.
Press Enter to continue. ESC = Quit

14. Press **[ENTER]** to continue.





15. To check the configuration, select 1, *View configuration*. Press **[ENTER]**. The following screen should appear, showing the TMC-MCA host adapter configuration.

View Configuration

Total System Memory	
Installed Memory	1024 kB (1.0 MB)
Usable Memory	1024 KB (1.0 MB)
Built in Features	
Installed Memory	1024 KB (1.0 MB)
Diskette Drive A Type	1.44MB 3.5"
Diskette Drive B Type	Not Installed
Math Co-processor	Not Installed
Serial Port	SERIAL_1
Parallel Port	PARALLEL_1
Slot 1 - ODI TMC-MCA	Host adapter
Adapter Memory Location	Segment CA00
Adapter I/O Location	0140
Select Interrupt Line	Interrupt 5
(Reserved)	

16. Return to the Main Menu. Remove the working Reference Diskette and store it in a safe place. Follow the instructions on the screen to reboot the system.
17. TMC-MCA installation and configuration is now complete. Consult the Standard Software Manual (and any others) for the software you will use with your tape drive.
18. When configuration is complete stay in *Depot4*; test your subsystem by selecting, *Test Tape Drive*, from the *Utilities* menu. Refer to Chapter 5, *DataTools* for more information on the test procedure.

Chapter 4 - Software Installation

Quick Software Installation

The following information is provided for quick Overland Data software installations. The information is in an abbreviated form and will work for most installations. If you require more detailed information, refer to the heading, *Detailed Software Installation*.

1. Make sure the subsystem hardware has been installed.
2. Turn the tape drive on first, then the computer.
3. Insert the diskette marked, *DATATOOLS, Disk 1* in drive A: or B:.
4. Do as follows:

From the hard drive prompt type:
a:\install or b:\install*
[ENTER]

5. Follow the prompts on the screen to complete installation.
6. This completes the *Quick Software Installation*.
7. If you were referred here from another procedure, return to that procedure and step; if not, go to the next step.
8. To configure your subsystem, go to Chapter 5, *Depot4*, under the *Interface* menu.

* a: or b: depends on the source drive you are installing from.

Detailed Software Installation

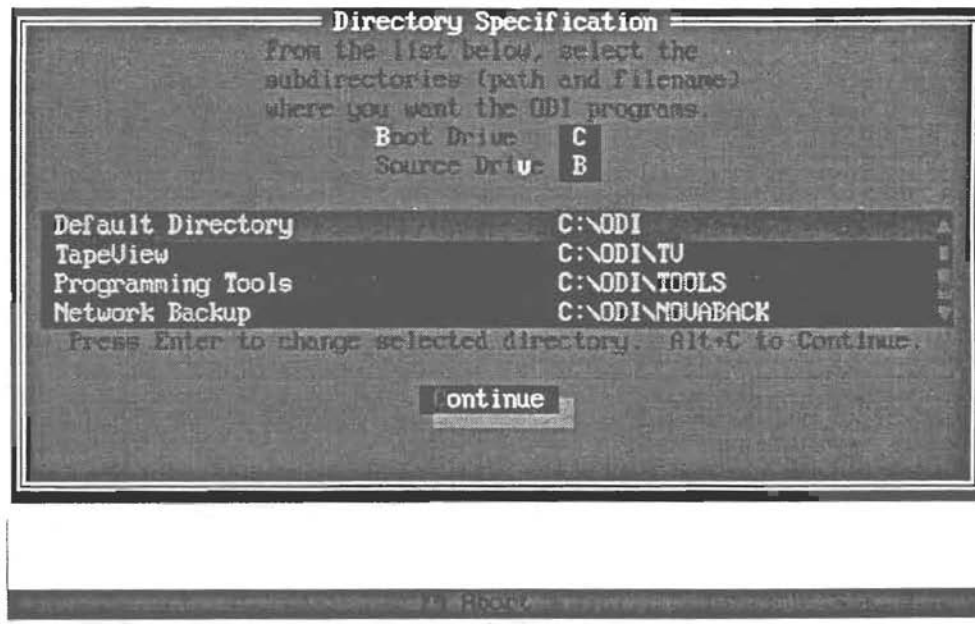
1. Make sure the subsystem hardware has been installed. Turn on the tape drive, then the computer.
2. Insert the diskette marked *DataTools, Disk 1* in drive A: or B:.
3. From your hard drive prompt, type the drive letter of the source disk (A: or B:), space, then type INSTALL; as follows:

From the hard drive prompt type:*
a:\install or b:\install
[ENTER]

Note: Press [F4] at any time to quit the installation and exit to DOS.

Opening Window

4. Select OK to open the next window.
5. The Directory Specification window shows the path and directories where each ODI program will be written. If you are satisfied with the default path, select C (or Alt+C) to continue the installation.



Path Defaults

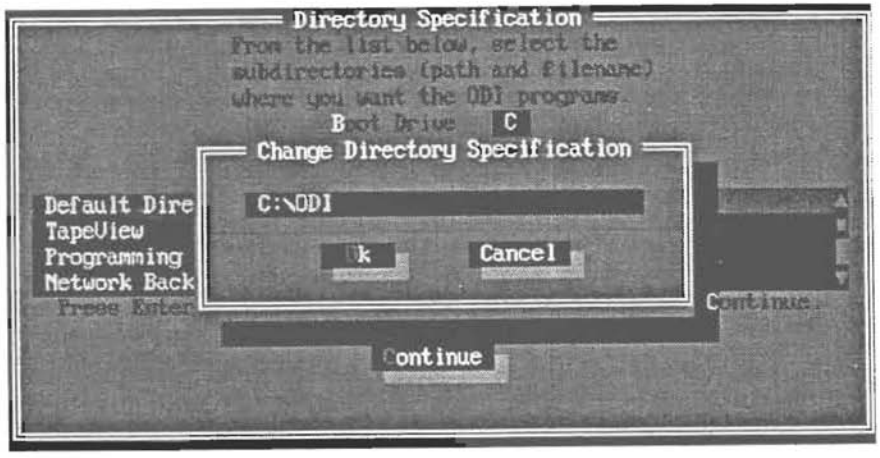
6. Use the following steps to change the default path:

   a) To select the path you want to change, use the [↓] or [↑] keys or click on the line.

   b) Press [ENTER] or click on Continue.

   c) The next window has a text field that lets you change that path and directory.

   d) If you change the main default directory (ODI), each path changes to the new directory name. However, if you want to change any of the subdirectories, you must highlight them individually and repeat the process.



Change Directory Popup Window

The next screen displays a list of programs you can install. Look at the labels on each diskette to see which programs you purchased.

7. Use the [↓] [↑] keys to move the cursor to the program you want, then press the [SPACEBAR]. A check mark appears in the box indicating that the program has been selected. When you finish selecting the programs, press [F8] to continue the installation.

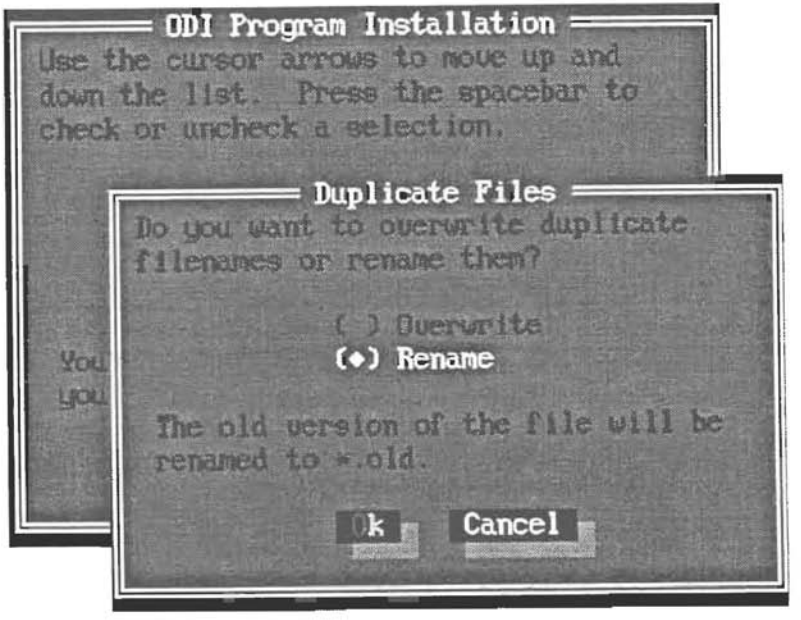
Program Selection Window

8. The following warning appears regardless of previous installations. Three conditions are possible:

   a) If a previous copy of the ODI utilities is on your hard drive, you can choose to have the existing files renamed to *.OLD. The installation program installs the new utilities under their standard names.

   b) If a previous copy of the ODI utilities is on your hard drive and you don't care if you overwrite them, select *Overwrite*.

   c) If you do not have any previous copies of the ODI utilities, simply select *Overwrite* and OK to continue the installation.

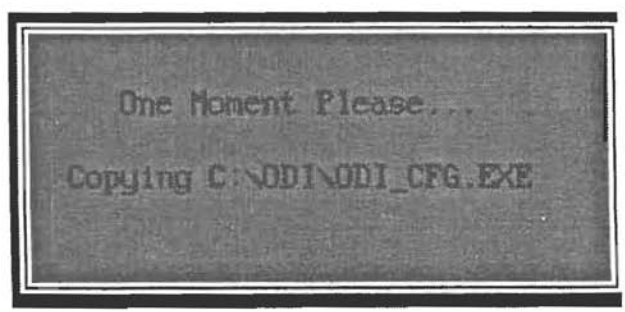


Duplicate File Warning

9. After accepting one of the previous selections, a second window displays instructions for continuing the installation.

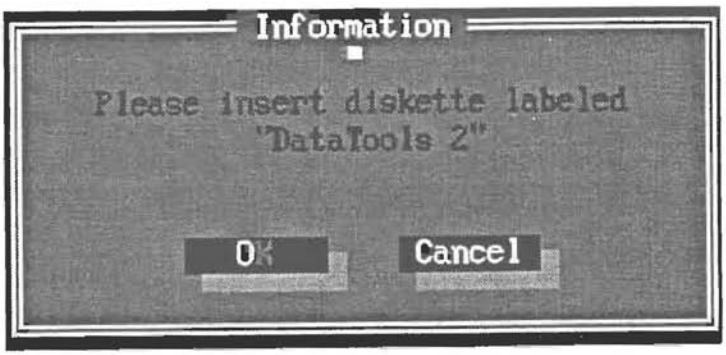
Instruction Window

10. When you return to the installation window, press [F8] to continue. While the files are being copied, the following window appears.



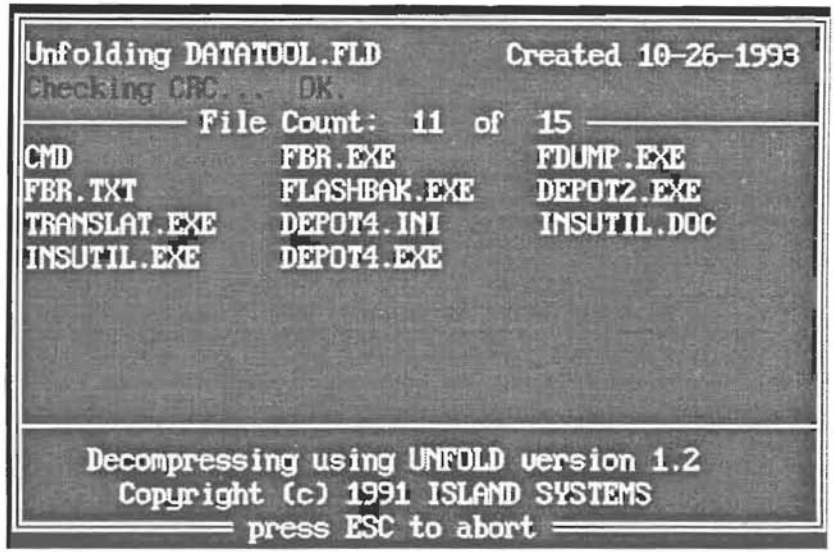
The DataTools software package is contained on a single high-density disk. However, software installation may require up to four diskettes. The number of diskettes varies–depending on the number of options you purchased.

11. You are prompted when the next disk should be inserted. The name on the screen should match the label on the diskette. Insert the next diskette then press [ENTER] to continue.



Insert Diskette Prompt

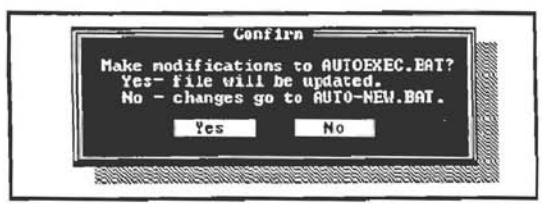
12. For each program, a screen appears showing the decompressed files that are being loaded onto your hardrive.

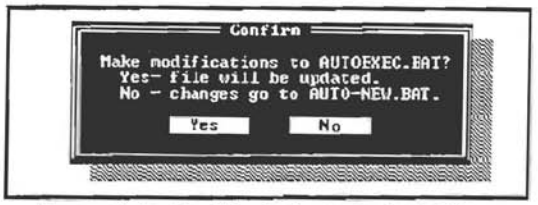


Decompressed Filennames

13. After all selected programs are installed, the next screen asks if you would like to modify your Autoexec.bat file. Choose [YES], to update the PATH statement. The program also inserts a SET command to point to the ODI software. If you select NO, you must manually modify your Autoexec.bat file with the DOS Set command. This command line allows the application to find the ODI.RLB file.

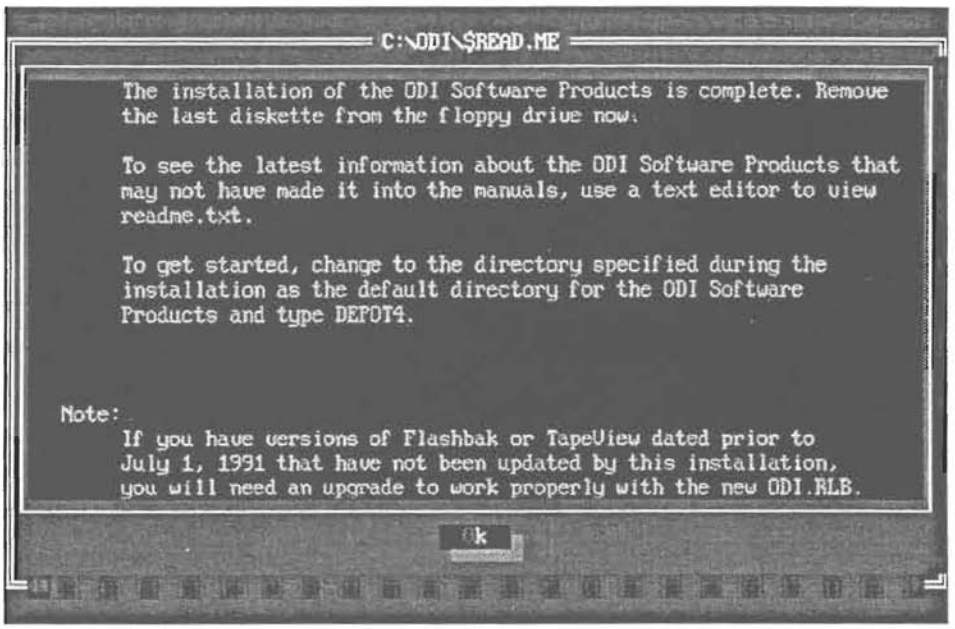
SET ODI=:\ODI





Prompt to Modify Autoexec.bat File

14. The following message appears after a successful installation.



Final Window

15. If you were referred here from another procedure, return to that procedure and step. Otherwise, go to the next section.

Testing the Tape Drive

Once the software and hardware installation is complete, be sure to test your subsystem. This will ensure data integrity before you begin processing data. To test your drive:

1. Load a good quality, write enabled tape and put the drive online.

Note: This test destroys all data on the tape.

2. Type, *Depot4*, and select Utilities from the main menu bar.
3. Select, *Test Tape Drive...*, from the Utilities Menu
4. When the window appears, select, *Do It...* This starts the test using the default settings.
5. If the test does not report an error, your system is ready for your applications.

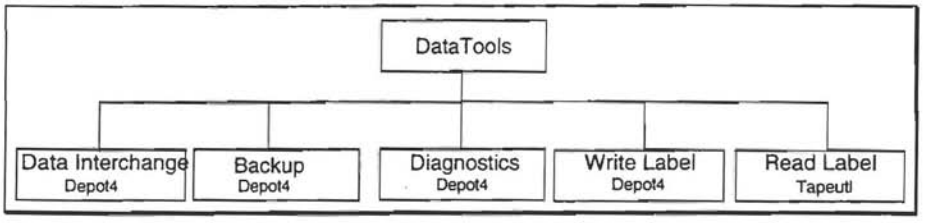
   If the test returns an error, try to resolve the problem by following the instructions on the screen and retesting. If you are still having problems, call ODI Technical Support.
6. Press [ESC] to exit the test program and return to the main menu bar.
7. From the File menu, select Exit and OK to return to the DOS prompt.

Chapter 5 - DataTools™

Overview

DataTools™ is a comprehensive set of utilities used for all your 9-track applications. The software package makes it easy for you to, read or write unlabeled tapes, read or write labeled tapes, view a tape, or to test your subsystem. All these capabilities are under one umbrella called *DataTools*™. Most of the *DataTools*™ software can be accessed through one utility called *Depot4*. *Tapeutl* is the only stand-alone

DataTools™ utility. The illustration below shows you the capabilities available and the utility needed to perform the job. As you can see *Depot4* can perform most of your data interchange needs.



Using DataTools

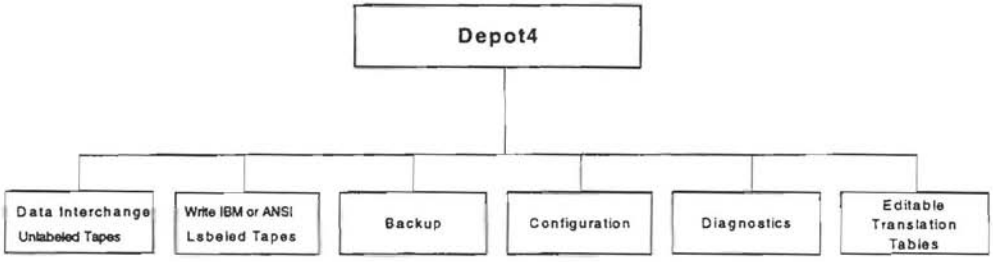
This section describes the utilities available for data interchange, testing, and configuration. The section also includes *Depot4* which provides easy user interface for most the ODI utilities.

Note: **There are differences between the *Tapeutl* translation tables and the *Depot4* translation tables. *Tapeutl* uses a different set of tables than *Depot4*. Use *Translation Tables* in the *Options* menu to change *Depot4* and *insutil.exe* to change *Tapeutl*.**

Depot4

Depot4 Overview

Depot4 is a window and mouse (or keyboard)-oriented program that provides intuitive access to your DataTools™ For DOS. Depot4 gives you a feel similar to Microsoft®Windows™. You will be able to launch familiar ODI utilities with a click of the mouse or stroke of the keyboard. To make data interchange and labels easier, we have incorporated a front-end window that allows you to select your options. The utility also incorporates an innovative feature called Heads-up Display (HUD). HUD allows you to see the data format before actually executing the program. The following is a list of DataTools™ capabilities provided from Depot4:



Depot4, The Shell Program

Depot4 is a shell program that invokes other programs. This section of the manual covers the operation of Depot4 only. To find detailed information on the actual utility that Depot4 launches, refer to the following list of Depot4 menu commands then determine the program that is actually run. Refer to that heading in this chapter. This symbol (🖱) indicates that the program can use a mouse.

Menu Selection	Program Name
Read Unlabeled Tape	Depot4
Write Unlabeled Tape	Depot4
Write Labeled Tape	Depot4
Network Backup	NovaBack
DOS Backup	Flashbak
View Tape	Fdump
Test Tape Drive	Depot4

Setup Depot4

Depot4 does not require an environment variable to function. It looks for the DataTools™ software in the following places (in order) after a command is invoked:

1. The current directory
2. The directory where Depot4 is located
3. All the directories in the DOS PATH
4. The path and directory in the ODI environment variable (if set)

Start Depot4

1. Depot4 is executed from the DOS command line.
2. Type **Depot4** then **[ENTER]**.

A Quick Summary of Menus

Depot4 menus have four main categories: File, Utilities, Options and Drive. The name and a brief description of each menu selection under the four categories are provided in the following tables. More detailed information on each item and its options are available in this chapter under the heading *Menus and Descriptions*.

FILE MENU	
MS-DOS SHELL...	Opens a door to DOS.
EXIT	Exits Depot4 to the DOS command line.
ABOUT...	Provides information about the program and the hardware.

UTILITIES MENU	
READ UNLABELED TAPE...	Transfers unlabeled data from your tape drive to your computer disk drive. The most commonly used interchange utility.
WRITE UNLABELED TAPE...	Transfers unlabeled data to the tape drive from your hard drive.
WRITE LABELED TAPE...	Write an ANSI or IBM standard labeled tape. Supports level 2 conventions.
NETWORK BACKUP...	This utility enables you to backup and restore your network disk drive(s). Selecting this invokes a separate utility called NovaBack. The utility is provided as an optional package and is not active unless the package has been installed.
DOS BACKUP...	This utility is used to backup and restore your local disk drive to tape. Selecting this invokes a utility called FLASHBAK.
VIEW TAPE...	Choose View Tape to examine the contents of a 9-track tape. When invoked, the shell immediately executes a program called Fdump.
TEST TAPE DRIVE...	Choose, *Test Tape Drive*, to quickly verify your subsystem. After selecting your options, the program then executes a program called XTEST.EXE.

OPTIONS MENU	
INTERFACE ...	Provides options to change the interface driver. Available drivers are: ASPI, Parallel, TMC, TX-8, TXi, and XL/2.
TRANSLATION TABLES...	Provides the capabilty to modify the ASCII to EBCDIC or EBCDIC to ASCII translation tables. Also provides the capability to create custom translation tables.
COMPRESSION...	This option is active only if your tape drive is ICRC capable. Available selections make it possible to turn on compression for all utilities and/or to prompt when performing archive routines.
DISPLAY...	Provides the capability to turn on a help feature called Heads-up Display (HUD). Another option sets your screen black and white for monochrome displays.

DRIVE MENU	
REWIND	Rewind the tape to the beginning.
REWIND & UNLOAD	Rewind the tape to the beginning then unload.

Help!!

Depot 4 offers online Help which is displayed as a one line message at the bottom of the screen. The message changes as you tab, click, or choose different fields or options. More detailed information on specific fields and options are also provided in this manual.

The Heads-up Display (HUD) lets you preview your data format before actually transferring the data to your disk or tape. Short Help messages are displayed as you pass the mouse pointer over each function in the HUD window.

Menu Selections & Descriptions

The following section provides a detailed description and use for each option under a menu item. This section also provides step-by-step examples of some common uses. If you are not sure how to move around or access options with a mouse or keyboard, refer to the heading, *Getting Around In Depot4*.

File Menu

Figure 1,

MS-DOS Shell

Choose this selection to open a door to the DOS command line. This option provides the capability to execute DOS commands such as, DIR, COPY, or even the capability to execute another program. To return to Depot4, type EXIT, then **[ENTER]**.

About

Choose this selection for information on the software version, serial number, free memory, disk space, and the type of interface (controller) detected by the software.

Exit

This option is used to exit the Depot4 program and return to the DOS command line. There are three ways to exit the Depot4 program:

Utilities Menu

The Utilities menu provides most of the programs you will need for data interchange and archiving. This section describes each menu selection and the options provided in the Main Window.

READ THESE IMPORTANT NOTES

Some of the programs under the *Utilities* menu have a window and mouse oriented screen. This window makes it easy to set parameters in the program. Some utilities, however, do not have this capability. It is for this reason that some menu items will produce a window to handle parameters, while others invoke a utility directly.

The following is a list of menu selections that allow mouse support:

- ***Read Unlabeled Tape***
- ***Write Unlabeled Tape***
- ***Write Labeled Tape***
- ***Network Backup***

The next section describes each menu item and its options:

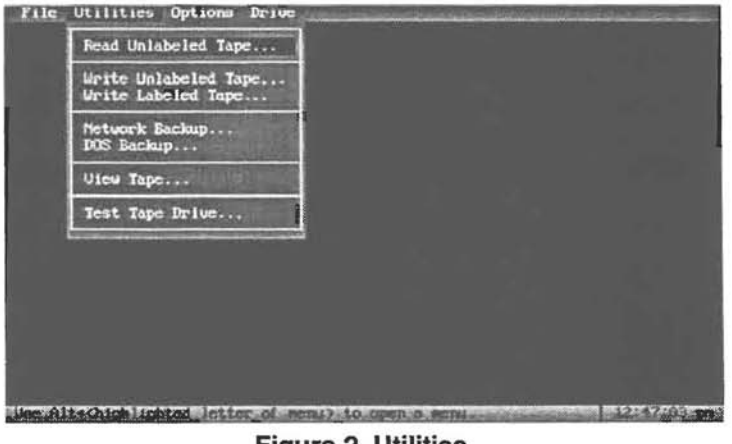


Figure 2, Utilities

Read Unlabeled Tape...

Choose this item to read ASCII, EBCDIC, or Binary data from an unlabeled[1] 9-track or 3480 tape. *Read Unlabeled Tape* processes fixed length data and has the ability to insert CR/LF characters.

Note: **To read labeled tapes automatically, refer to the heading, *Tapeutl* in the Table of Contents.**

The *Read Unlabeled Tape* window lets you select program options. Selecting these options invokes a secondary program called Depot2. Most of your needs are handled within the Depot4. If you have special needs, refer to *Depot2* in this manual.

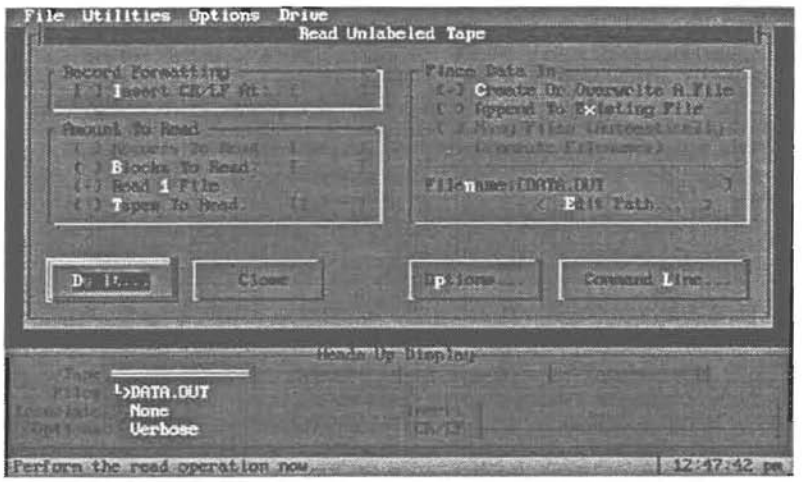


Figure 3, Window To Read Unlabeled Tape

Record Formatting

This pane is used to format your records with carriage returns and line feeds (CR/LF). CR/LF characters are inserted in fixed length records only. Variable length data does not have the option to insert CR/LF. If your data is variable length, do not select this option.

[1] The utility does not process labels. However, you still have the ability to read the data from a labeled tape by skipping over the header.

Note: **Use *Tapeutl* to read IBM or ANSI labeled tapes with variable length data.**

Insert CR/LF At: [n]

Enter the size of your record.

If this option is not selected, the data is transferred without delimiters.

AMOUNT TO READ

This pane is used to limit the read operation. The options in this group are selected with radio buttons, therefore, when one is selected the other options are deselected.

*R*ecords To Read [n]

Choose this radio button to limit the read operation to a specific number of records. Enter the number of records you wish to read.

This option becomes active only when the, *Insert CR/LF At [n]*, is selected. When the *Records To Read [n]* option is selected, the alternate group selection, *Blocks To Read [n]*, becomes inactive.

*B*locks To Read [n]

Choose this radio button to limit the read operation to specific number of data blocks. Enter the number of blocks you wish to read.

Data blocks can contain one or more records. This option does not insert carriage returns and line feeds at the end of these records. It is for this reason that the option becomes inactive when CR/LF is not selected.

Read *1* File

Select this option to read one file from one tape.

This option reads data until a filemark is detected.

Tapes To Read [n]

Enter the number of tapes you wish to read.

When this option is selected, the read operation will continue through the number of tapes (volumes) you entered in the text field. It is used for five situations:

1. To read a single file that spans more than one tape.
2. To read and concatenate multiple files from multiple tapes.
3. To read and concatenate multiple files from a single tape.
4. To read multiple files from multiple tapes and generate filenames automatically for each file as it is transferred to disk[2].
5. To read multiple files from a single tape and generate filenames automatically for each file as it is transferred to disk.

Place Data In

This pane provides options that affect the output file to your disk drive.

Create Or Overwrite A File

Choose this button to create a new file on your disk or overwrite an existing file.

If a file with the same name exists on your disk, this option *does* warn of an impending overwrite.

Append To Existing File

Choose *Append* to concatenate to a file existing on your disk. Enter the name of the file you wish to append to in the *Filename* field.

---

[2]Refer to the heading, Many Files, for information on the following condition:

Downloading one file that spans more than one tape. The tapes also contain more than one file.

*M*any Files (Automatically Generate Filenames)

Choose this option to generate file names automatically.

The *Many Files* selection is very powerful tool that enables you to download up to 999 files to your disk without any further user intervention. The filename is always DATA unless you change it in the *Filename* field. The extension [nnn] will be an incrementing number, starting from 001 up to 999. For example, if you transfer 99 files using this method, 99 files will appear on your disk. The first filename will be DATA.001 and the last filename will be DATA.099. This option is active only if the *Tapes To Read* option has been selected.

A report is generated when this option is selected. The filename entered in the *Filename*: field is used as the root filename for the report file with the extension DP2. The report's default name is DATA.DP2.

The report is used to find files that need to be concatenated on your disk. The software does not have the capability to handle a file that spans more than one tape, as a single file. This means that files that span more than one tape, using the above options, will be divided into two or more files .

The following is a sample of the report generated when using the *Many Files* option:

```
Reading from tape 1.
        File C:\depot2\data.001 Size      22
        File C:\depot2\data.002 Size      2291
        File C:\depot2\data.003 Size      58762
        File C:\depot2\data.004 Size      18860
        File C:\depot2\data.005 Size      8590
        File C:\depot2\data.006 Size      42103
        File C:\depot2\data.007 Size      41404
        File C:\depot2\data.008 Size      208488
Changing to tape 2.
        File C:\depot2\data.009 Size      16947
        File C:\depot2\data.010 Size      13891
        File C:\depot2\data.011 Size      50
        File C:\depot2\data.012 Size      21
        File C:\depot2\data.013 Size      9753
        File C:\depot2\data.014 Size      6386
        File C:\depot2\data.015 Size      78375
        File C:\depot2\data.016 Size      246864
```

Figure 4, Sample: Generate Filenames Report

Note: **The operator must determine whether files surrounding a tape change must be concatenated or not.**

Note the point where the tape was changed: There are two files you should pay close attention to, DATA.008 (read before the tape was changed) and DATA .009 (read after the tape was changed). You may need to concatenated (combine) these files. Determine if the original file spanned more than one tape. If so, the two files must be concatenated on your disk drive to reconstruct the original file. To concatenate, use the following DOS command:

```
COPY file1 + file2 fileout
```

Where:

- file1 and file2 are the files that span more than one tape.

- fileout is a new filename that *you* assign as the single file containing the contents of file1 and file2.

Filename:[ ]

Enter the disk, path, and name of the file you wish to create on disk. Wild cards are not accepted. If no path is entered the file will be created in the current directory. You also have the option of using the default name (DATA.OUT).

If *Many Files* has been selected, it generates filenames using the name you entered with sequential extensions. It *also* generates a report with the same name. See *Many Files* for more information on generating a report and filenames.

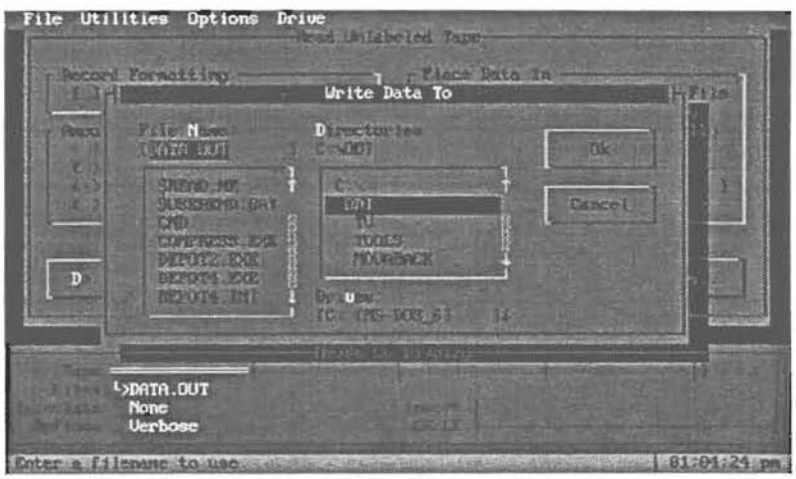


Figure 5, Read Unlabeled Tape, Edit Window

Select *Edit* as an alternative method to selecting a file to overwrite or append to.

Perform the following steps to select your file with a mouse:

1. Select the disk drive that the directory or file resides on by clicking on the drive box directly below the *Directory* box.
2. Click on the arrow to the right of the box to produce a list of drives.
3. Scroll to the desired drive then click.
4. Move to the Directory pane then scroll through the list of directories.
5. Double click the desired directory.

Note: **A list of files from the selected directory are displayed in the *File* Name box.**

6. After the path has been selected, click on the *File* Name box.
7. If you know the path and filename, enter it here. Then press **[ENTER]**. You may *not* use wildcards when you enter your file selection. If you need to select your file from a list, go to the next step.
8. Use the scroll bar to scroll located to the right of the *File Name* box, then scroll through the list of files. Select the desired file by double clicking that file. This action returns you to the *Read Unlabeled Tape* window.
9. Verify your selection by checking the *Filename* field.

Perform the following steps to select your file(s) with a keyboard:

1. Select the drive that the directory or files reside in by using the **[TAB]** key to step through the main selections until you highlight the disk drive box.
2. Use the [↑] or [↓] to scroll through the list of available disk drives.
3. Highlight the desired drive. No need to press **[ENTER]** because the directory and file lists are automatically updated as each drive is highlighted.
4. Select the ***Directory*** pane by pressing the **[ALT]** + **[I]**.
5. Use the [↑] or [↓] to highlight the directory.
6. Press **[ENTER]** the path is shown above the box.

7. Repeat steps 5 and 6 for each subsequent directory level.

Note: **A list of files from the selected directory are displayed in the *File Name* box.**

8. Move to the *File Name* field by pressing **[ALT]** + **[N]**.
9. If you know the path and filename, enter it here. Then press **[ENTER]**. You may also use wildcards in your file(s) selection. The default **[*.*]** will select all files in the immediate directory. If you need to select your file(s) from a list of files, go to the next step.
10. Use the **[TAB]** key to move to the list box.
11. Use **[↑]** or **[↓]** to highlight the desired file.
12. Press **[ENTER]** to select that file and go back to the Main Window. Your selection(s) are shown in the *Get Data From...* pane and in the Heads-up Display.

Options

A popup dialog box will appear when this button is selected. The Options window is used to; set the position to start reading, select translations, echo commands, and handle errors. The defaults for this window are:

- skip zero files and zero blocks (begin reading from the beginning of tape)
- no translations
- ignore errors (off)
- echo commands (on).

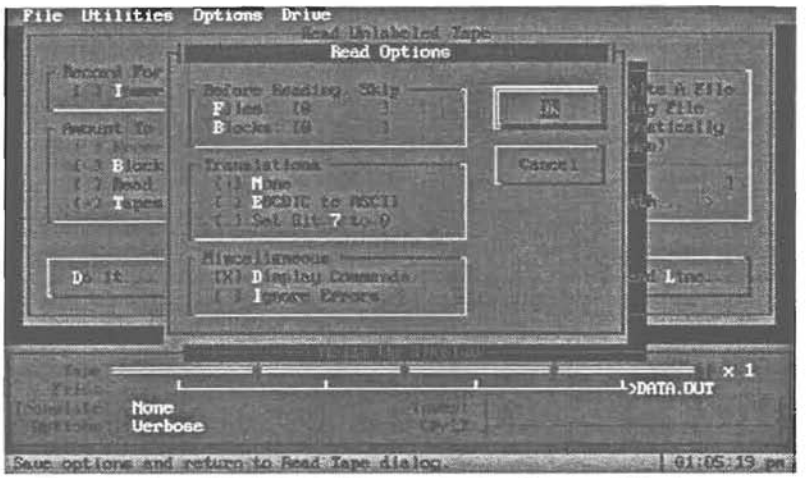


Figure 6, Read Unlabeled Tape, Option Popup Window

Before Reading, Skip...

This pane is used to set the position to start reading a tape. Two text fields are used to set the desired position *Files* and *Blocks*. The two fields can be used together or independently to fix the position to start reading.

The tape is always rewound to the beginning before it is read, so skipping *Files* is referenced to BOT. Skipping *Blocks* is referenced to the specified file position.

Files [n]

Enter the number of files to skip before reading.

Blocks [n]

Enter the number of blocks to skip (after a specified file position) before reading.

Translations

This pane is used to select the translation type. Three types of translations are available NONE, EBCDIC to ASCII, and Set Bit 7 to 0.

NONE (default)

Choose NONE when no translation is desired. Turn this option *on* when transferring binary or ASCII data to disk.

EBCDIC to ASCII

Choose this option to translate EBCDIC data from your tape to ASCII data on your disk. The EBCDIC to ASCII translation table can be changed using the Translation Tables dialog box in the Options menu. The translation program can also be accessed from the DOS prompt using the ***translat.exe*** command.

Set Bit 7 to 0

Select this option to set the high order bit in ASCII data to zero.

Some ASCII data uses the high order bit to pass control characters. This format is usually evident in printer files. If the data appears to be correct when using the View function (FDUMP), but is corrupted when the actual transfer has been completed, turn this switch on as a possible solution.

Miscellaneous

The options in this pane are used to display or inhibit the display of commands, operations, or messages.

*D*isplay Commands

Check this box to echo (display) commands as they are executed by DEPOT2. With this option selected, Depot2 displays the block count as data is transferred from tape to disk. It is usually used to monitor the progress of the job.

*I*gnore Errors

Choose this option to inhibit any interactive prompts from being displayed.

This feature is useful when creating command lines that will be used in batch files. For example, if this option is *on*, and you are about to read a file to your disk, and the filename already exists, then the file is overwritten without warning. The same holds true for any warning or error messages. The software continues to transfer data and tries to handle any situation without prompting the operator. In the case of the error message, "Abort, Retry, Continue Beyond Error", the software automatically continues beyond the error until it physically can not.

Command Line...

Since this utility uses a Main Window (see the Important Note under the heading *Utilities*), choosing this button, gives you an opportunity to see the actual DEPOT2 command line. The Depot2 command is executed when the *Do It* ... button is selected.

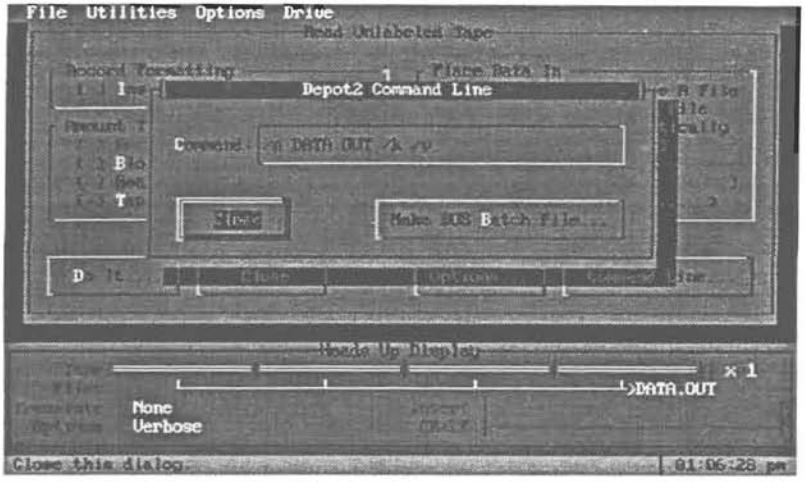


Figure 7, Read Unlabeled Tape, Command Line Popup Window

Depot2 Command Line

This pane provides a view of the Depot2 command line created from the options as you have specified with Depot4. The command line cannot be edited in this field. You must return to the Main Window and change the options.

As you select options from the Main Window, Depot2 parameters are inserted into the actual command line. Since Depot2 *is* a command line, switch driven, program, it offers another dimension to its capabilities. A batch file can be created reflecting the options selected in the Main Window.

Make DOS *B*atch File...

Enter the path and filename of the batch file you wish to create.

This button is used to create a batch file that includes the Depot2 command line. Once the batch file has been created, you do not need to use Depot4 to transfer data with the current tape format. You need only to load your tape and execute your batch file from the command line. It is important to note that tapes requiring different formats *must* have a different batch file setup to transfer the data properly.

After the batch file has been created, you can add other Depot2 command lines, Depot2 switches, or even DOS commands to the batch file by using a DOS text editor. Refer to the heading, DEPOT2, in this manual, for more information on using Depot2 on a command line.

Examples- Read Unlabeled Tape

Before You Try An Example:

The following examples must be performed from a known configuration, therefore, it is presumed that unless instructed to set an option in a step, all options are at default. If you are not sure if defaults are set, exit the dialog box then re-enter it. This will reset all options back to their default values.

Read And Translate An EBCDIC File And Insert Line-End Characters In The Data

1. Select *Read Unlabeled Tape* from the Utilities Menu.
2. Choose *Insert CR/LF At: [n].*
3. Enter the length of your record or line.
4. In the *Amount to Read* pane, select *Read 1 File.*
5. Choose *Options*, then select *EBCDIC to ASCII* translation.
6. The result is a file, in the current directory, using the default naming convention. The file is in ASCII format with carriage returns and line feeds at the interval you entered in step #3.

Read One ASCII Formatted Tape With Multiple Files And Generate Filenames Automatically

1. Select *Read Unlabeled Tape* from the Utilities Menu.
2. Choose the *Tapes to Read* option.
3. Leave the number of tapes at one.
4. Select *Many Files.*
5. Select *Filename* and enter the [DRIVE:] [PATH] [FILENAME (no extension)] of the files to be created on the disk.
6. Choose *Do It...*
7. The result is a series of ASCII files written to your disk drive, bearing the filename you selected, with incrementing extensions. If the filename is left at default, files named DATA.* are created in the current directory. The report file is also created in the same directory using the default name, DATA.DP2.

Read Some Data Blocks From An ASCII File On Tape And Concatenate Those Blocks To A File That Already Exists On Disk

1. Select *Read Unlabeled Tape* from the *Utilities* menu.
2. Choose *Blocks To Read.*
3. Enter the desired number of data blocks to be read from the tape and added to the file on disk.
4. Choose *Append To Existing File.*

5. Select *Edit* or enter the path and filename of the file to concatenate the data blocks to.
6. Choose *Do It...*
7. The result is a file already existing on your disk that has had data added to it from the tape.

Read One ASCII File that Spans Multiple Tapes

1. Select *Read Unlabeled Tape* from the *Utilities* menu.
2. Choose *Tapes To Read*, and enter the number of tapes to be read.
3. Choose *Create Or Overwrite A File.*
4. Select *Do It...*
5. The result is one file on disk, read from multiple tapes.

Read a Data File from a Labeled Tape

The following steps describe how to read a labeled tape, without actually processing the label. It is used to quickly process one file.

1. Select *Read Unlabeled Tape* from the *Utilities* menu.
2. Choose *Options.*
3. Go to the *Before Reading Skip...* pane, and select *Files.*
4. Enter 1 in the text field. You are skipping over the label with this option.
5. Go to the *Translations* pane and select *EBCDIC to ASCII* if the tape is written in EBCDIC. If the tape is written in ASCII, no translation is necessary.
6. Select *OK.*
7. Choose *Do It...*
8. This example skipped over the label (ANSI or IBM), then processed the data file. The result is a data file on your disk, using the default filename, DATA.OUT.

Read One ASCII File that Spans Multiple Tapes with Multiple Files

Note: **The program generates two or more independent files, from a single tape file. This happens if that file spans more than one tape, and you are using the options, *Tapes To Read* and *Many Files*.**

1. Select *Read Unlabeled Tape* from the *Utilities* menu.
2. Choose *Tapes To Read* then enter the number of tapes to be read.
3. Choose *Many Files*.
4. Select *Do It...*
5. The result is several files on disk, read from multiple tapes.
6. Since we did not change the default filename, the files should appear on your disk, as follows:

```
data.001
data.002
data.003
data.004
data.005
data.006
data.007
data.008
data.009
data.010
data.011
data.012
data.013
data.014
data.015
data.016
```

6. Working with the tape, note the position of the file you wish to work with. The position number will be used to identify that file on your disk.

7. A report is created automatically when the Many Files button is selected. The report is located with the data files and bears the extension, DP2. We will use this report to find your file. Print or view the document file.

8. Working with the report, [FILENAME.DP2], locate your file by finding the same extension number as its sequence number on the tape. For example, if the file on tape was the third file, then the default filename on disk will be DATA.003.

9. The report also shows when a tape has been changed. This information is important because it is very unlikely that the end of your data file will correspond with the end of tape boundary. If the file you need is at a point where the tape has been changed, it is probable that the rest of the file is under the next filename generated. For example, using the sample below, it is likely that DATA.008 should be concatenated with DATA.009 because that is the point where the tape was changed.

```
Reading from tape 1.
        File C:\depot2\data.001 Size      22
        File C:\depot2\data.002 Size      2291
        File C:\depot2\data.003 Size      58762
        File C:\depot2\data.004 Size      18860
        File C:\depot2\data.005 Size      8590
        File C:\depot2\data.006 Size      42103
        File C:\depot2\data.007 Size      41404
        File C:\depot2\data.008 Size      208488
Changing to tape 2.
        File C:\depot2\data.009 Size      16947
        File C:\depot2\data.010 Size      13891
        File C:\depot2\data.011 Size      50
        File C:\depot2\data.012 Size      21
        File C:\depot2\data.013 Size      9753
        File C:\depot2\data.014 Size      6386
        File C:\depot2\data.015 Size      78375
        File C:\depot2\data.016 Size      246864
```

Figure 8, Sample of Report Generated, Many Files Option

10. Downloaded files that spanned more than one tape must be concatenated on your disk to reconstruct the original tape file. This is done with the DOS COPY command. Refer to your DOS User's Guide and Reference Manual for more information on concatenating.

Write Unlabeled Tape...

Select this item from the Utilities Menu to write to an unlabeled[3] tape, overwrite an existing file on a tape, or append a file to a tape. The utility writes fixed length data, translates ASCII to EBCDIC, and has the ability to strip CR/LF characters.

The *Write Unlabeled Tape* selection provides a Main Window for easy selection of program options. Since it *is* a Main Window (see above Important Notes), a secondary program called Depot2 is invoked after your Main Window selections are complete. Most of your needs are handled within the Depot4 Main Window, however, if you have special needs, more detailed information on Depot2 can be found under the heading *Depot2* in this manual.

Note: **The tape reel must have a write ring inserted in order to activate the write circuitry in your tape drive. For 3480 users the write enable is a thumb wheel at the front end of the cartridge. Make sure the white dot on the wheel is not showing.**

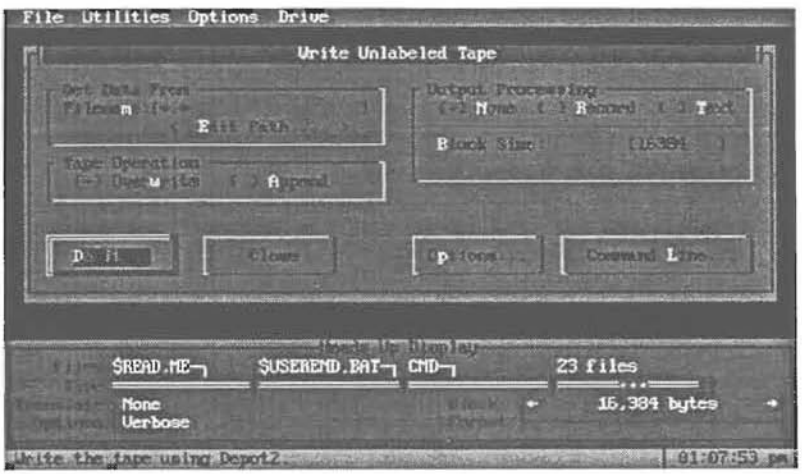


Figure 9, Write Unlabeled Tape Window

[3]Use the *Write Labeled Tape* menu item to write ANSI or IBM labels.

Write Unlabeled Tape

This pane is used to setup the format of the file to be written to tape.

Filename [disk:] [path] [filename]

Enter the disk, path, and filename of the file to write to tape. DOS wildcards can be used in the filename. If no path information is entered, the program looks in the current directory for the file.

Edit

Use this as an alternative method for entering the path and filename. When *Edit* is used, a dialog box appears for selecting the disk, path, and filename. The filename entered also appears in the *File* text field in the Main Window.

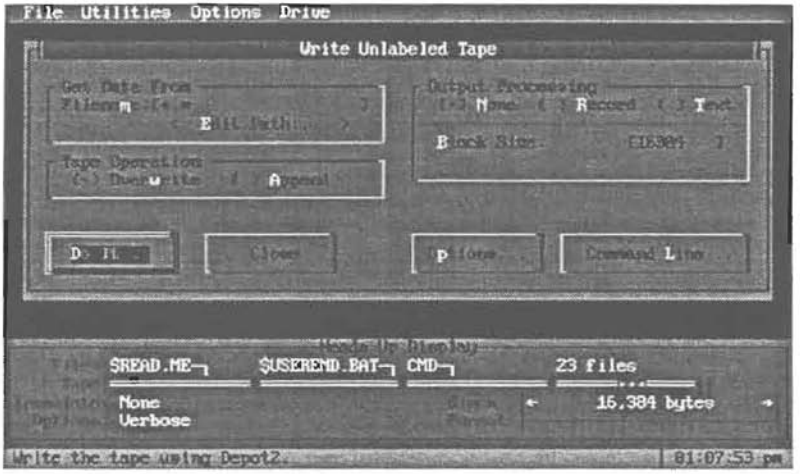


Figure 10, Write Unlabeled Tape, Edit Window

Perform the following steps to select your file(s) with a mouse:

1. Select the disk drive that the directory or files reside on by clicking on the *Drives* box directly below the *Directory* box.
2. Click on the arrow to the right of the box to produce a list of drives.
3. Scroll to the desired drive then click.
4. Move to the Directory pane then scroll through the list of directories.
5. Double click the desired directory.

Note: **A a list of files from the selected directory will be displayed in the *File Name* box.**

---

6. After the path has been selected, click on the *File Name* box.
7. If you know the path and filename, enter it here. Then press **[ENTER]**. You may also use wildcards when you enter your file(s) selection. The default, **[*.*]**, will select all files in the immediate directory. If you need to select your file(s) from a list of files, go to the next step, if not, go on to step 9.
8. Use the scroll bar to scroll to the right of the *File Name* box to scroll through the list of files. Select the desired file by double clicking that file. This action returns you to the Main Window.
9. Verify your selection by checking the *Get Data From...* pane or by checking the *Heads-up Display*.

Perform the following steps to select your file(s) with a keyboard:

1. Select the drive that the directory or files reside on by using the **[TAB]** key to step through the main selections until you highlight the disk drive box.
2. Use the **[↑]** or **[↓]** to scroll through the list of available disk drives.
3. Highlight the desired drive. There is no need to press **[ENTER]** because the directory and file lists are automatically updated as each drive is highlighted.
4. Select the *Directory* pane by pressing the **[ALT]** + **[I]**.

5. Use the [↑] or [↓] to highlight the directory.
6. Press **[ENTER]**. The path is shown above the box.
7. Repeat steps 5 and 6 for each subsequent directory level.

Note: **A list of files from the selected directory is displayed in the *File Name* box.**

8. Move to the *File Name* field by pressing **[ALT]** + **[N]**.
9. If you know the path and filename, enter it here. Then press **[ENTER]**. You may also use wildcards in your file(s) selection. The default **[*.*]** will select all files in the immediate directory. If you need to select your file(s) from a list of files, go to the next step.
10. Use the **[TAB]** key to move to the list box.
11. Use [↑] or [↓] to highlight the desired file.
12. Press **[ENTER]** to select that file and go back to the Main Window. Your selection(s) are shown in the *Get Data From...* pane and in the Heads-up Display.

Tape Operation

This pane is used to direct the software to either, begin writing at the beginning of tape (thus overwriting existing data), or append a file after the last file on tape.

Overwrite

Choose this button if the data on tape does not need to be saved.

This selection will overwrite any existing data. If the tape has never been used, it simply creates a new file on tape. No initialization is needed. A double filemark (logical end of tape) is written when this operation is completed.

Append

Choose this button to append a file after the last file on tape.

The function does not concatenate to an existing file. Instead the utility spaces forward until a double filemark (logical end of tape) is detected. The drive then positions the tape between the filemarks. This positioning separates the first file from the next. The next command is a write function which overwrites the second filemark and starts writing the new file. A double filemark is written at the completion of this operation.

Output Processing

This pane contains all necessary selections to format the data. All options in this pane are mutually exclusive.

None

Choose this option to move data without stripping the carriage returns and line feed characters.

This option is usually used to move binary files. The utility will not strip carriage returns and line feeds, therefore, will not change the format of the data. The option, *Block Size*, becomes active when this button is selected.

Block Size

Enter the size of the block to be written to tape. The size can range between 1 byte to 65280 bytes.

If the file size is not a multiple of the block size you enter, a short block will be written to tape as the last block. DEPOT2 does not have the capability to pad out to the selected block size. The default block size is 16384 bytes.

Record

Select this option to strip carriage returns and line feeds from the data. Most main frames process data without the CR/LF characters, therefore this option is useful for those applications. Two text fields appear below the *Record* option when the button is selected, *Record Length* and *Records Per Block*.

Record Length

Enter the length of the record to be transferred. The length should be the data only. Do not add in the characters used for the carriage returns and line feeds. For example, if your data record on disk is 120 bytes plus another 2 bytes for the CR/LF characters. You would enter only the 120 bytes as your record length.

The record length is used by the program to calculate where the line end characters are to be stripped from.

Records per Block

This will determine the block length to be written. Insert the number of records you desire in each block. The number of records per block is known as the blocking factor (BF). The record length (RL) multiplied by the blocking factor is equal to your block size (BL).

[BF] x [RL] = [BS]

The block size must not exceed 65280 bytes.

Text

This option provides the same capabilities as the *Record* option. The difference is the terminology. The choice of *Text* or *Record* has been provided as an aid to you. If you are more comfortable using the Record Length and Blocking Factor terminology, then select the *Record* radio button. If you are more comfortable thinking in terms of line length and lines per block, then check the *Text* radio button.

Line Length

This field is functionally the same as the *Record Length* field; refer to the heading, *Record Length* for an explanation of requirements.

Lines per Block

This field is functionally the same as the *Records Per Block* field; refer to the heading *Records Per Block*, for and explanation of the requirements for this field.

Options

Choose this command button to access other write options such as, translation, echo commands, and error handling.

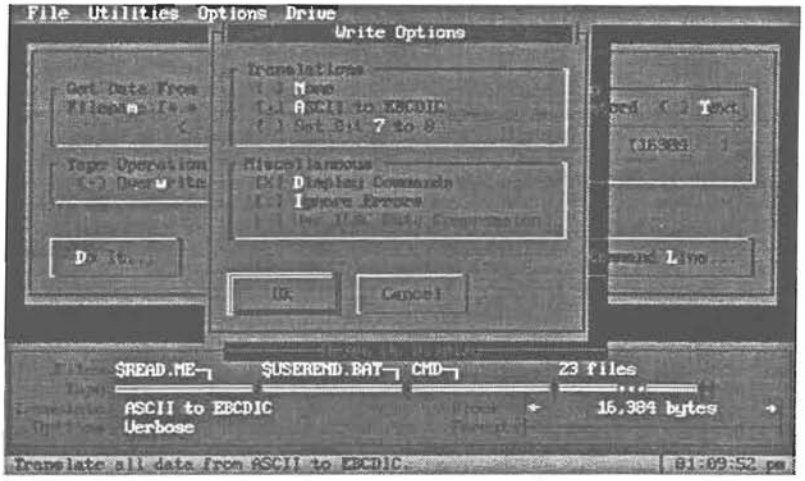


Figure 11, Write Unlabeled Tape, Options Popup Window

Translations

This pane provides three types of translations: *None*, *ASCII to EBCDIC*, and *Set Bit 7 to 0*.

NONE (default)

Select NONE to move data without translating it. Turn this option *on* when transferring binary or ASCII data to tape.

ASCII to EBCDIC

Choose this option to translate ASCII data, from your disk, to EBCDIC data on the tape.

EBCDIC is the standard character set used by IBM and other mainframes. The ASCII to EBCDIC translation table can be changed using the translation tables feature in the options menu.

Set Bit 7 to 0

Select this option to set the high order bit in your ASCII data to zero.

Sometimes the high order bit in ASCII data is used as a control bit. When it is set, the data will appear corrupted after transfer. Use this option to correct that situation.

Miscellaneous

This pane is used for miscellaneous options. These options are used to display commands, error information, and to select compression. They do not affect the format of the data.

Display Commands

Check this box to echo (display) commands as they are executed by DEPOT2. This option also displays the block count as data is transferred from tape to disk. It can be used to monitor the progress of the operation.

Ignore Errors

Choose this option to inhibit any interactive prompts from being displayed.

This feature is useful when creating command lines that will be used in batch files For example, in the case of the error message, "Abort, Retry, Continue Beyond Error", the software automatically continues beyond the error until it physically can not.

Use ICRC Data Compression

Select this option to enable the compression capability on your tape drive.

Note: **This option is active only if your tape drive is equipped with compression capabilities.**

Command Line...

Choosing this button, gives you an opportunity to see the actual DEPOT2 command line that will be executed when the *Do It ...* button is selected.

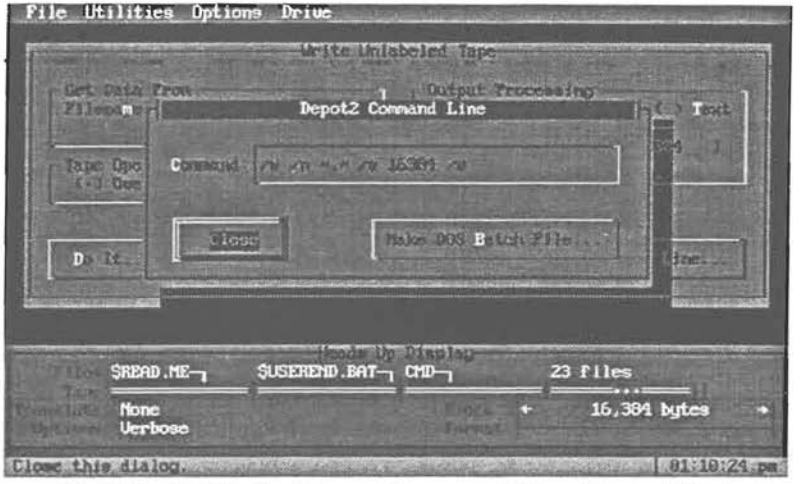


Figure 12, Write Unlabeled Tape, Command Line Popup Window

Depot2 Command Line

This pane provides a view of the DOS command line as it appears with the Depot2 command. The command line cannot be edited in this field. You must return to the Main Window and change the options.

As you select options from the Main Window, parameters are inserted into the actual Depot2 command. Since Depot2 *is* a command-line, switch-driven program, it offers another dimension to its capabilities. A batch file can be created reflecting the options selected in the Main Window.

Make DOS *B*atch File...

Enter the path and filename of the batch file you wish to create.

This button is used to create a batch file that will include the Depot2 command line. Once the batch file has been created, you do not need to use Depot4 to transfer data with the current format. You need only to load your tape and execute your batch file from the command line. It is important to note that files with different formats *must* setup a different batch file to transfer the data properly.

After the batch file has been created, you can add other Depot2 command lines, Depot2 switches, or even DOS commands to the batch file by using a DOS text editor. Refer to the heading, DEPOT2, in this manual, for more information on using Depot2 on a command line.

Examples- Write Unlabeled Tape

Before you try an example:

The following examples must be performed from a known configuration, therefore, it is presumed that unless instructed to set an option in a step, all options are at default. If you are not sure if the defaults are set, exit the dialog box then re-enter it. This will reset all options back to their default values.

Write a Tape Using the Default Settings

1. Select *Write Unlabeled Tape* from the Utilities Menu.
2. Choose either *Edit*, to select and input filename, or type the path and filename in the *Filename* text field.
3. Select *Do It...*
4. The result is an ASCII file written with a block size of 16384. The data or format has not been changed in any way.

Append a File to a Tape, Translate to EBCDIC, and Strip Carriage Returns and Line Feeds

1. Select *Write Unlabeled Tape* from the Utilities Menu.
2. Choose *Filename* or *Edit* and enter the path and filename of the text file to be written.
3. Select *Append.*
4. Choose *Record* or *Text* to strip the line-end characters.
5. Enter the length of the record or line.
6. Go to *Records/Lines Per Block* and enter the number of records or lines to put into one block.
7. Choose the *Options* button, then select *ASCII to EBCDIC* translation.
8. Select *Do It...*

9. The result is an EBCDIC file appended after the last file on tape. The file is written in a block size equal to the record/line length multiplied times the number of records/lines per block. The file also had the carriage returns and line feeds, originally at the end of each record, stripped out of the data.

Write a Labeled Tape

Notes About Labeled Tapes

This utility provides the capability to write an IBM or ANSI standard label. It supports up to ANSI level 2 and the IBM equivalent.

The Label program supports the following standard capabilities:

- Single files on a single volume.
- Single files on multiple volumes.
- Multiple files on a single volume.
- Multiple files on multiple volumes.

Attributes Unique to IBM Labels

- Header information is in EBCDIC.
- The Owner ID Field is 10 bytes long.
- An option is provided for translating the data to EBCDIC.
- HDR2, EOF2, and EOV2 labels are required.
- End of volume processing does not require two filemarks after the final End of Volume (EOV) trailer. (*Label* does write two filemarks)

Attributes unique to ANSI Labels

- Header information is in ASCII.
- The Owner ID Field is 14 bytes long.
- No option is provided for translating the data to EBCDIC.
- End of volume processing *requires* two filemarks after the final End of Volume (EOV) trailer.
- HDR2, EOV2, and EOF2 labels are optional.

For more information on ANSI Standard Labels, refer to the *ANSI Spec. For Magnetic Tape Labels And File Structure For Information Interchange, ANSI X3.27-1987.*

For information on IBM Labels, refer to the *IBM Spec. for OS/VS Tape Labels, GC26-3795-3.*

Options For ANSI or IBM Standard Labels

The following section describes the options available to write an IBM or an ANSI standard label:

Note: **Options for one label type may not be available for the other label type. Such selections are inactive when not available.**

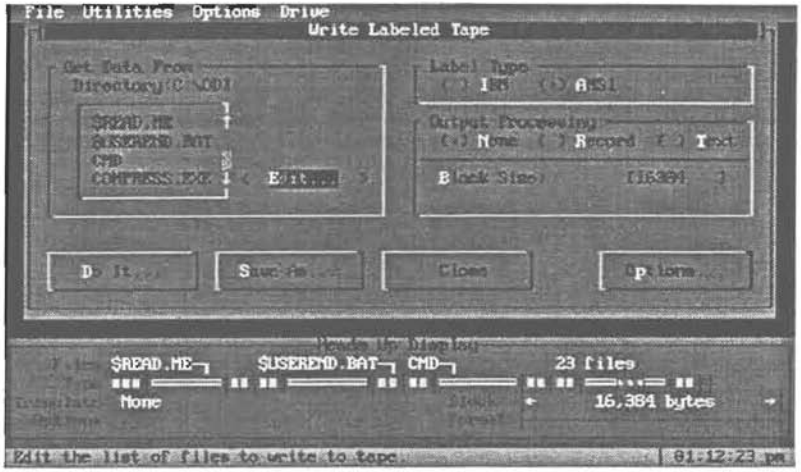


Figure 13, Write Labeled Tape Window

Get Data From...

This pane provides the means to select the file or files that will be written to tape. File selection is done with the *Edit* command.

Edit

Select the *Edit* button to choose the file or files to be written to tape. Perform the following steps to select your file(s):

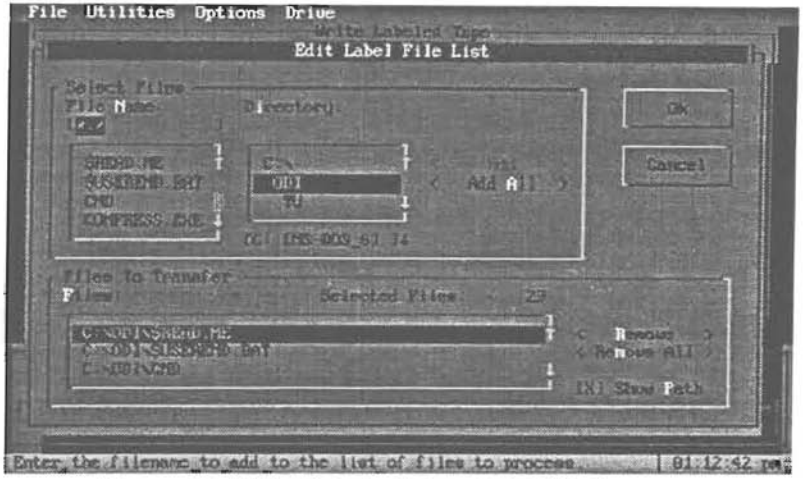


Figure 14, Write Labeled Tape Edit Window

Perform the following steps to select your file(s) with a mouse:

1. Select the disk drive that the directory or files reside on by clicking on the drive box directly below the *Directory* box.
2. Click on the arrow to the right of the box to produce a list of drives.
3. Scroll to the desired drive and click on it.
4. Move to the Directory pane then scroll to the desired directory.
5. Double click the desired directory to display a list of files in the *File Name* box.
6. After the path has been selected, click on the *File Name* box.
7. If you know the name and path and filename, enter it here. Then press **[ENTER]**. You may also use wildcards in the *File Name* text box. The default, **[*.*]**, will select all files in the immediate directory. If you need to select your file(s) from a list of files, go to the next step, otherwise go on to step 9.

8. Use the scroll bar to find the *File Name* in the list of files. Select the file by double clicking on that file. Repeat for each subsequent file you need to write to tape. A list of selected file(s) are shown in the ***Files*** pane.

Note: **The number of files you select can be limited by the amount of free disk memory in your computer.**

---

9. Refer to the *Files To Transfer* pane in Figure 14, the files selected are displayed in that pane. Also notice in the *Selected Files* field, that the number shown corresponds to the number of files selected. Use the scroll bar to view and verify your selections.
10. If the *Files To Transfer* are correct, then click OK to return to the main screen.

Perform the following steps to select your file(s) with a keyboard:

1. Select the drive that the directory or files reside on by using the **[TAB]** key to step through the main selections until you highlight the disk drive box.
2. Use the [↑] or [↓] to scroll through the list of available disk drives.
3. Highlight the desired drive. There is no need to press **[ENTER]** because the directory and file lists are automatically updated as each drive is highlighted.
4. Select the ***Directory*** pane by pressing the **[ALT]** + **[I]**.
5. Use the [↑] or [↓] to highlight the directory.
6. Press **[ENTER]** the path is shown above the box.
7. Repeat steps 5 and 6 for each subsequent directory level.
8. Move to the *FILE NAME* field by pressing the **[ALT]** + **[N]**.
9. If you know the name and path and filename, enter it here. Then press **[ENTER]**. You may also use wildcards in your file(s) selection. The default **[*.*]** will select all files in the immediate directory. If you need to select your file(s) from a list of files, go to the next step.
10. Use the **[TAB]** key to move to the list box.
11. Use the [↑] or [↓] to highlight the desired file.

12. Press **[ALT] + [D]** to add that file to a list of selected files. Repeat steps 11 and 12 for each file.

    or

    [ALT] + [A] to add all the files in that subdirectory.

Note: **Files may come from different paths and the number of files are not limited.**

---

13. If you change your mind, just press **[ALT] + [M]** to remove all the files...

    or

    press **[ALT] + [F]** to move to the ***Files*** pane. Then use the [↑] or [↓] to highlight the file you wish to delete. Press **[ALT] + [R]**.

14. Refer to the *Files To Transfer* pane in Figure 15. The files selected with the *Add* Button are displayed in that pane. Also notice in the *Selected Files* field, that the number shown corresponds to the number of files selected. **[ALT] + [F]** to move to the *Files* box then use the [↑] or [↓] to view and verify your selections.

15. Once file selection(s) are complete, press **[TAB]** to move to the *OK* button.

16. Press **[ENTER]** to go back to the Main Window. Your selections are shown in the *Get Data From...* pane and the Heads-up Display.

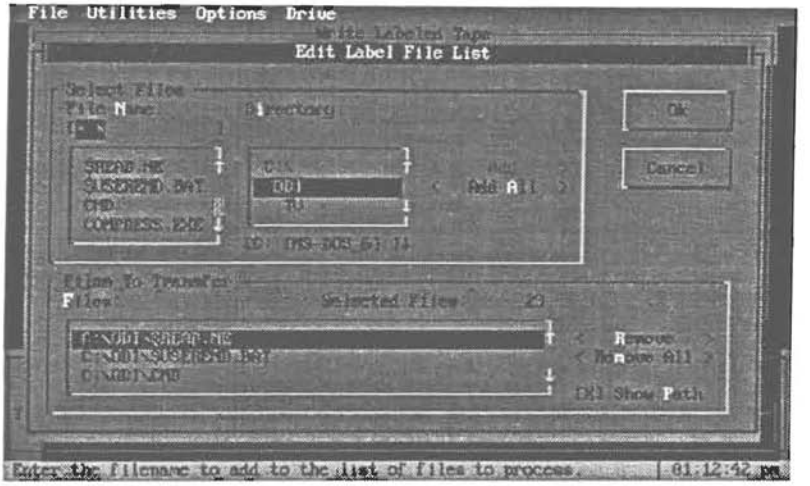


Figure 15, Write Labeled Tape, Edit Window

Label Type

IBM or ANSI Standard Labels

Use this option to select the type (see Figure 13) of label (IBM or ANSI) you wish to write to tape.

Output Processing

This pane contains all necessary selections to format the data with CR/LFs and set a block size. All options in this pane are mutually exclusive.

None

Choose this option to transfer data without changing the format.

When this option is selected, the program will not strip carriage returns and line feeds. The *Block Size* field, becomes active when this button is selected.

Block Size

Enter the size of the block to be written to tape. The size of the block can range between 1 byte to 32760 bytes. The default block size is 16384 bytes.

Record

Select this option to strip carriage returns and line feeds from the data. Two text fields appear when this button is selected, Record Length and Records Per Block.

Record Length

Enter the length of the record to be transferred.

This is a text field that requires the length of the record to be transferred. The record length is used, by the program, to calculate where the line end characters are to be stripped from.

Records per Block

Enter the number of records to be written per block.

This field is used to set the block length to be written. Insert the number of records you want in each block (known as the blocking factor). The record length (RL) multiplied by the blocking factor (BF) is equal to the block size (BS).

[BF] x [RL] = [BS]

The block size must not exceed 32760 bytes.

Text

Select this option to strip carriage returns and line feed out of the data file.

This option provides the same capabilities as the *Record* option. The difference is the terminology. The choice of *Text* or *Record* has been provided as an aid to you. If you are more comfortable using the *Record Length* and *Records Per Block* terminology, then select the *Record* radio button. If you are more comfortable thinking in terms of *Line Length* and *Lines Per Block*, then check the *Text* radio button.

Line Length

Enter the length of the line.

Lines per Block

Enter the number of lines to write per block.

Options

Selecting the *Options...* button takes you to the Label Options window. This window displays the volume serial number, volume owner ID, and provides the capability to enter optional data into some fields. The *Label Options* window also provides translations from ASCII to EBCDIC (IBM only). The following information describes the options available and their function:

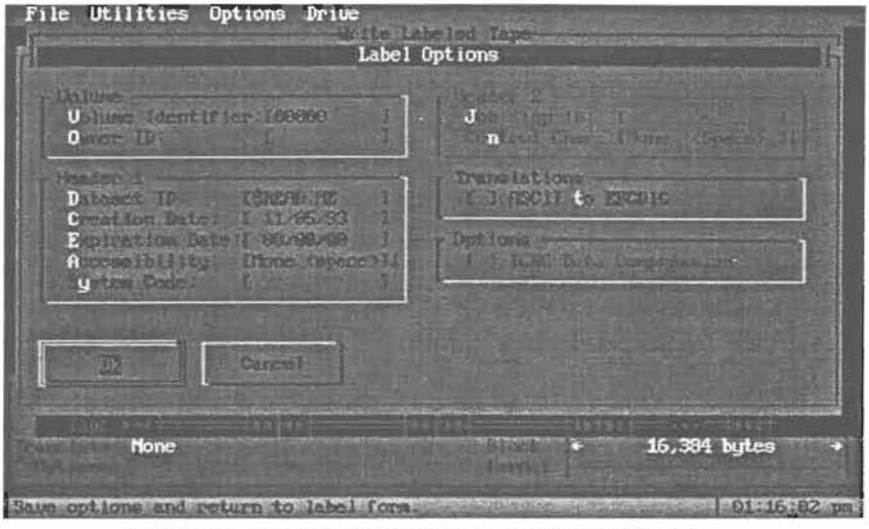


Figure 1, Write Labeled Tape, Options Window

Volume

Serial Number

The Serial Number is a unique code assigned by the operator. It consists of *up to* six alphanumeric characters that range between 000001 and 999999.

Owner ID

The Owner ID indicates a specific customer, person, business, department, etc., to which the volume belongs. Any ten character code or name is acceptable.

Header 1

The Header is used to identify and describe data sets and to protect them from unauthorized use.

*D*ata Set ID

The *Data Set ID* consists of a 17 byte alphanumeric field. If 17 characters are not used, it left justifies the characters and fills the remaining field with spaces.

The *Data Set ID* field is used to ensure the correct data set is being used. It is also used in conjunction with password protection to determine if the existing data can be overwritten.

*C*reation Date

This field is automatically updated, but can be edited. It documents the date the data set was created. The format of the *Creation Date* is the month, day, then year (mm/dd/yy). The date that the tape was written is default date.

*E*xpiration Date

Specify the month, day, and year (mm/dd/yy) that the data can be destroyed. If no date is entered, the field will show 000000.

This date indicates an end of life for the data on the tape. The data on the tape may be scratched or overwritten after the designated expiration date. Select this field to provide security protection for the tape.

Accessibility

Note: **Changing this field from the default (*space*) causes the files to be inaccessible by main frames unless password information is provided along with the tape. Do not change this setting unless you are versed in security procedures.**

Click on the arrow to the right of the field. A drop down box will appear with options.

From the keyboard, first press **[ALT] + [A]** to move to the *Accessibility* field then use **[↓]** or **[↑]** to make your selection.

Options for *Accessibility* are listed below:

Code	Definition
Space	No Password Protection
1	Password protection. Additional identification of the data set is required before it can be *read, written or deleted.*
3	Password protection. Additional identification of the data set is required before it can be *written, or deleted.*
Other Character	Protected Volume. No access is possible under the operating system.

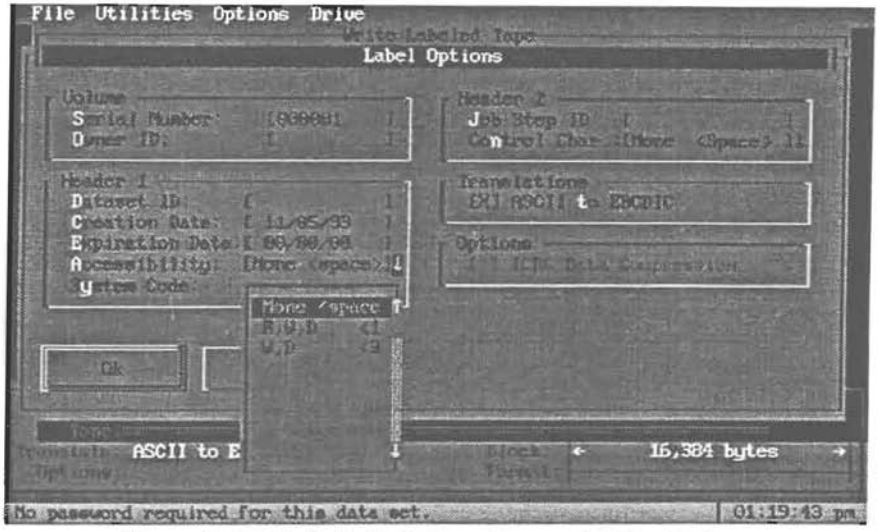


Figure 2, Write Labeled Tape, Accessibility Code

System code

The System Code is a 13 byte number that is unique to your system. The default for this field is all spaces. Each mainframe system has an identity. This identity is entered here as a means of identifying the system that originated the tape. The *System Code* is verified by the system.

Header 2

Header 2 options are used with IBM labels and are not available for ANSI labels. It is used to show structure of records in a file.

Job Step ID

This is a 17 byte field used as identification of the job and job step that created the data set. The first 8 bytes contain the name of the job; the ninth byte is a (/), and the last 8 bytes contain the name of the job step.

Control Character

A Control Character is a 1 byte code indicating whether a control character set was used to create the data set, and the type of control characters used. The default to this field is none.

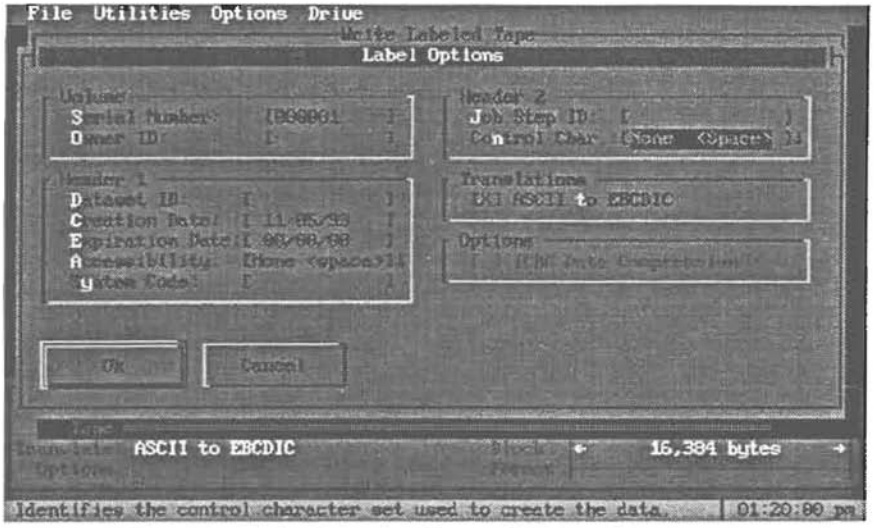


Figure 3, Write Labeled Tape, Control Character

Click on the arrow to the right of the field. A drop down box will appear with options.

From the keyboard, first press **[ALT] + [N]** to move to the *Control Char* field then use **[↓]** or **[↑]** to make your selection.

Options for *Control Character* are listed below:

None	Inserts a space that is interpreted as no Control Characters.
A	Used to write ANSI Control Characters.
M	Used to write Machine Control Characters.

Translations

Not available for ANSI Standard Labels.

ASCII *to* EBCDIC

Select this check box to translate the data from ASCII text to EBCDIC.

Options

This selection is active only if you have a 3480 Tape Drive that supports data compression.

/CRC Data Compression

Select this option to switch compression *on* or *off*.

Examples- Write Labeled Tape

Before You Try An Example:

The following examples must be performed from a known configuration; therefore, it is presumed that unless instructed in a step, all options are at default. If you are not sure if defaults are set, exit the dialog box then re-enter it. This will reset all options back to their default values.

Note: **It is not necessary to change the options under the *Options* button to write a labeled tape. These are optional parameters.**

Write an IBM Standard Label on a Tape and Translate to EBCDIC

1. Select *Write Labeled Tape* from the *Utilities* menu.
2. Choose *Edit* then select and *Add* the file(s) to be written.
3. Select OK.
4. Choose *Do It...*
5. The result is an IBM Standard Label written to a tape with one or more data files, and translated to EBCDIC with a block size of 16384.

Write an ANSI Standard Label on a Tape

1. Select *Write Labeled Tape* from the Main Menu.
2. Choose *Edit* then select and *Add* the file(s) to be written.
3. Select *OK*.
4. Choose *ANSI* from the *Label Type* pane.
5. Select *Do It...*
6. The result is an ANSI Standard Label written to a tape with one or more data files. The data file is written in ASCII text and is blocked at 16384 bytes.

NETWORK Backup (optional software)

Selecting this item from the *Utilities Menu* provides you with network backup capabilities. When selected, Depot4 launches a program called *NovaBack*. The *NovaBack* manual has been included with your product. Differences between the *Novaback* manual and the Overland Data version of *NovaBack* have been documented in Appendix A of this manual. Please read to Appendix A before performing a network backup.

DOS Backup

Flashbak is a window oriented local archive utility. It provides the capability to perform a complete backup or restore of your local disk drive. You can also tag individual files, or read from a file list to backup or restore. Flashbak also provides the capability to create a report tailored to your needs.

Selecting this option from the *Utilities* menu invokes the program directly. No Main Window is provided from the Depot4 program, therefore no mouse capabilities are provided. All options are selected from the actual Flashbak program. For more information on Flashbak, refer to the heading Flashbak in this manual.

View Tape

Selecting this option from the *Utilities* menu invokes a program called FDUMP. There is no Main Window, therefore no mouse capabilities. All options are selected from the actual program.

FDUMP provides the capability to view the contents of any 9-track or 3480 tape. Of course density is a function of your tape drive. Make sure the tape drive is capable of reading the density that the tape has been written in.

The utility is used to:

- Determine tape format
- Sample tape data

View Tape should not be used to troubleshoot a subsystem error condition. The utility does not have the same error trapping capabilities as the other utilities and may lead to an incorrect assumption. Use XTEST (*Test The Tape Drive*) to troubleshoot any error condition.

View Tape Features Allow You to :

- View the data from tape in ASCII, EBCDIC, and/or Hex.
- Scroll through the data using page down or [↑] [↓] keys.
- Step through the data one block at a time.
- Position to a specific block.
- Position to a specific file.
- Position to a specific byte.
- Continuously monitor the position.
- Monitor the tape drive status word.

Running View Tape

View Tape (FDUMP) does not have a Main Window. When you select this option from the *Utilities* menu, you go directly to the FDUMP program. The program lists the functions and keys used to operate the program on the screen. For more detailed information on the program, refer to the heading FDUMP in this manual.

Test the Tape Drive

This menu selection provides a Main Window used to select options for a program is called XTEST. XTEST provides the capability to verify the integrity of your subsystem. Since your system *is* an integrated system, it is difficult to narrow a problem down to a single component (controller, cable, tape drive). XTEST does, however, provide diagnostic messages that can help to find the cause of a fault.

Requirements

- Make sure the head and tape path are clean.
- Load a good quality scratch tape (test writes to the tape).
- Tape must be write enabled.
- Tape drive must be online.
- Tape must be at the beginning.

You should run this test to:

- verify a new installation.
- check out an error detected during normal operation.
- verify subsystem capability prior to important jobs.

Main Window Options

Block size

Fixed *A*t: [n]

Choose this radio button to input a block size other than the default of 16384. Enter a number ranging from 1 byte to 65280 bytes.

Rando*m*

Selecting this radio button will produce blocks that range in size from 1 byte to 65280 bytes.

Repeat Test

Times: [n]

Select this radio button to enter the number of times you wish to repeat the test. No *end of test* prompt is displayed. The program simply restarts the test for the number of times you requested. The default parameter is one time.

*F*orever

Select this radio button to continuously test the subsystem until you press **[ESC]** or cycle the power on your tape drive to *off*. This is useful to verify an intermittent error condition.

Data

*I*ncrementing

Choose this radio button to write an incrementing pattern of characters that uses the standard and extended ASCII character set.

*R*andom

Select this radio button to write a random pattern of data, using the standard and extended ASCII character set.

Blocks/*P*ass: [n]

Enter the number of blocks to write.

Options

Read *O*nly

Place a check in this check box to perform a read only operation. The tape must have been written previously with XTEST and loaded in the tape drive. The pattern selection must match that of the test tape. This option is useful in checking interchange problems especially if the test tape is written with at different tape drive. It can also be useful as a step saver when you know that the tape drive is able to write.

Test Stacker

This option is only available for cartridge tape drives with a magazine loader or stacker. The test must be performed with the stacker loaded with the maximum number of tapes. It loads the first tape into the drive, the drive then does a complete load then unloads the tape into the stacker. The stacker is then stepped to the next tape for the same procedure. This is done until the last tape has been loaded then unloaded. An error is displayed on both the tape drive and on the screen if the test fails.

Examples- Test The Tape Drive

Before You Try An Example:

The following examples must be performed from a known configuration; therefore, it is presumed that, unless instructed in a step, all options are at default. If you are not sure if defaults are set, exit the Depot4 dialog box then re-enter it. This will reset all options to their default values.

Test Your Tape Drive Using the Default Settings

1. Select *Test Tape Drive* from the Utilities Menu.
2. Choose *Do It...*
3. The results are described below:

How the Default Test Runs

The default test writes 250 blocks of data in incrementing patterns. When 250 blocks have been written, the program then writes two filemarks (Logical End of Tape), then rewinds to the beginning of the tape. The program then initiates a read/verify operation on the 250 blocks previously written. When the two filemarks are detected, XTEST then issues a rewind command. Test results are listed in the information pane. If the test fails, it stops at the point of failure and displays the error along with possible solutions.

The following is a list of the default settings:

Test Defaults	
Number of Times to Run Test:	1
Number of Blocks to Write:	250
Block Size:	16384
Data Pattern:	Incrementing

Test Your Drive Continuously and Write Random Block Sizes with Random Data

1. Select *Test Tape Drive* from the Utilities Menu.
2. Go to the *Block Size* pane and select *Random*.
3. Go to the *Data* pane and select *Random*.
4. Go to the *Repeat Test* pane and select *Forever*.
5. Choose *Do It...*
6. The test will run continuously, writing and random blocks of random data until you press **[ESC]**.

Options Menu

The *Options Menu* offers four selections (Figure 19); *Interface, Translation Tables, Compression,* and *Display*. The *Interface* selection provides the ability to change and configure the type of interface you will be using with your 9-track tape drive. *Translation Tables* gives you the capability to change the default table, edit a table, copy a table, or create a table. *Compression* is tool used to switch the compression feature on the T480 cartridge tape drive on or off. *Display* enables you to toggle on or off the *Heads-up Display* (HUD) provided in the *Depot4* program.

Each of these selections are explained in detail, in they order they appear, in the following section.

Note: **You can also run the configuration program or translation program from the DOS promt using the commands:** *odi_cfg.exe* **(for configuration) and** *translat.exe* **(for translation).**

Figure 19, Depot4 Options Menu

Interface—Selection and Configuration

Overview—Selecting and Configuring an Interface

All supported controller interfaces and configurations are accessed through the *Interface* selection in the *Options* menu. Interface selection can be a one or two step process. The need for an extra step depends on the need to setup your interface for anything other than the default.

The first step in setting-up your interface is to select an interface that matches the controller interface you have installed. Available interfaces are listed below and shown in Figure 21.

- ASPI (Advanced SCSI Programmable Interface)
- Parallel Port
- TMC (Microchannel and ISA)
- TX-8 (Cipher/Pertec Interface)
- TXi-16 Cipher/Pertec Interface)
- XL/2 (Cipher/Pertec Interface)

Once an interface is selected, configure that interface[1] (if necessary) by using the *Configuration* pane located on the bottom of the window.

Configuration changes made in the *Configuration* pane can be done and saved for multiple interfaces in a single session. This provides the capability to switch to different pre-configured interface types in one computer.

[1] You must configure the XL/2 controller using the reference diskette provided by your computer vendor.

Interface Default Settings						
	ASPI	Parallel	TMC	TX-8	TXi-16	XL/2
I/O Address	N/A		N/A	360 hex	340 hex	[2]
Tape Drive Address	N/A	N/A	N/A	0	N/A	*
IRQ	N/A	N/A	N/A	10	N/A	*
Start/Stop	N/A	N/A	N/A	Normal	N/A	*
Drive Type	N/A	N/A	N/A		32XX/36XX	N/A
Other	no stacker	N/A	no stacker	N/A	N/A	N/A

Installing ASPI

Overland Data software applications are compatible with most ASPI drivers that talk to SCSI controllers[3]. ASPI (Advanced SCSI Programming Interface) was developed by Adaptec and has become the industry standard for the PC market.

Note: **For information on installing your ASPI driver, refer to the manual that came with your controller**

ASPI drivers support a maximum of four controllers and allow the user to dynamically change to any drive (connected and online) on the SCSI bus.

[2] Configure with Reference Diskette

[3] The SCSI controller manufacturer provides the drivers for ASPI compatible controllers.

IMPORTANT NOTES

☞ **You must install the ASPI manager that accompanied the host adapter (e.g., *ASPI4DOS.SYS* provided by Adaptec). If the driver is not installed, you will see a message like the one shown in Figure 20. This message is also displayed if you are using an old version of the ODI.RLB file.**

☞ **All SCSI peripherals connected to an ASPI compliant host adapter must be powered-on before applying power to the computer. The message shown in Figure 21 is associated with this condition.**

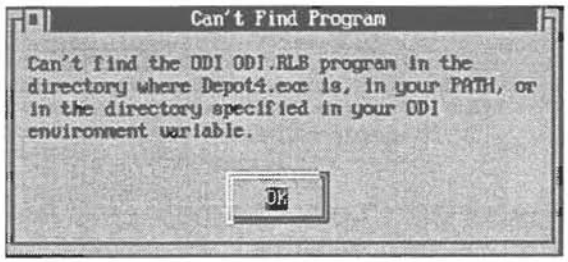


Figure 20, ASPI Interface Selection—Old ODI.RLB File

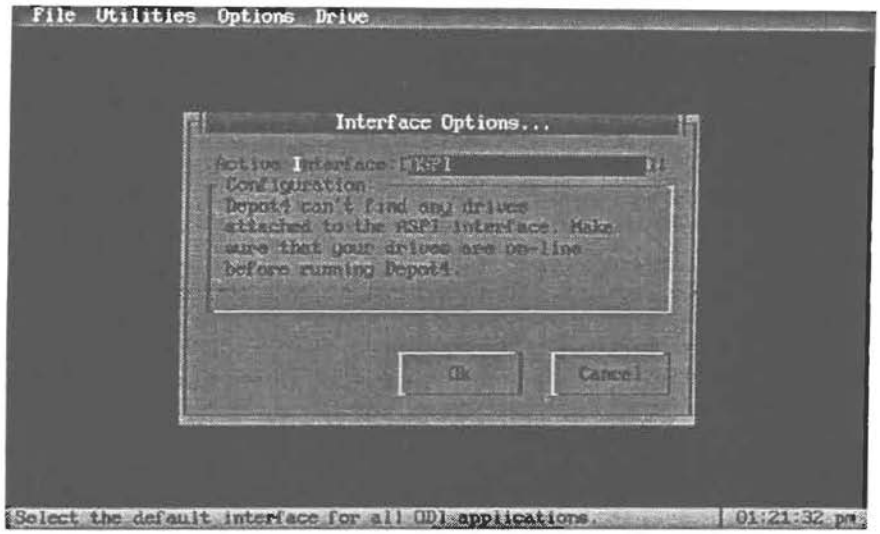


Figure 21, ASPI Interface Selection—No Device Found

Overland Data provides connectivity (ASPI Module) to ASPI through a selection in the *Interface* window. Selecting *ASPI* from the list of interfaces provided in this window offers the capability to communicate with an ASPI compatible host. The tape drive is automatically configured for three conditions (i.e., <0, 5, 0>):

- Host Bus Adapter (HBA): select from 0 to 3 in the drop-down box.
- Target: automatic from 0 to 7
- Logical Unit Number (LUN): automatic from 0 to 7

The above list of numbers and valid parameters are presented respectively (i.e., HBA 0, Target 5, LUN 0).

The following procedure selects and configures for an ASPI interface:

1. Make sure the *ASPI Manager* is installed in CONFIG.SYS and that all hardware installation is complete before attempting to configure. Refer to the reference manual provided by your controller manufacturer for more information on driver and hardware installation.

2. Working from the *Interface* pane, click the arrow to the right of the active interface label to open a drop-down box with a list of available interfaces.

3. If *ASPI* is not already highlighted (see Figure 22) select it now.

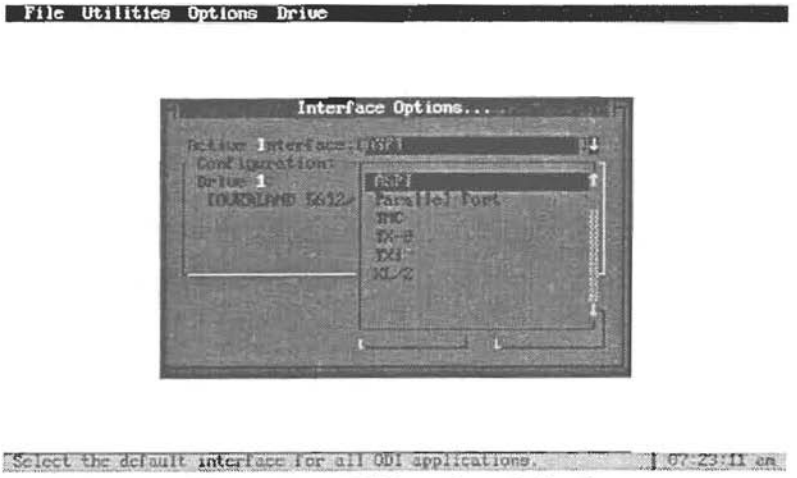


Figure 22, ASPI Interface Selection

4. The *Interface Option* pane provides the opportunity to select a tape drive from a list (if more than one drive is available) of tape drives. This list is accessed by clicking the arrow. Select the desired *drive* (see Figure 25) to be used for *Drive 1*.

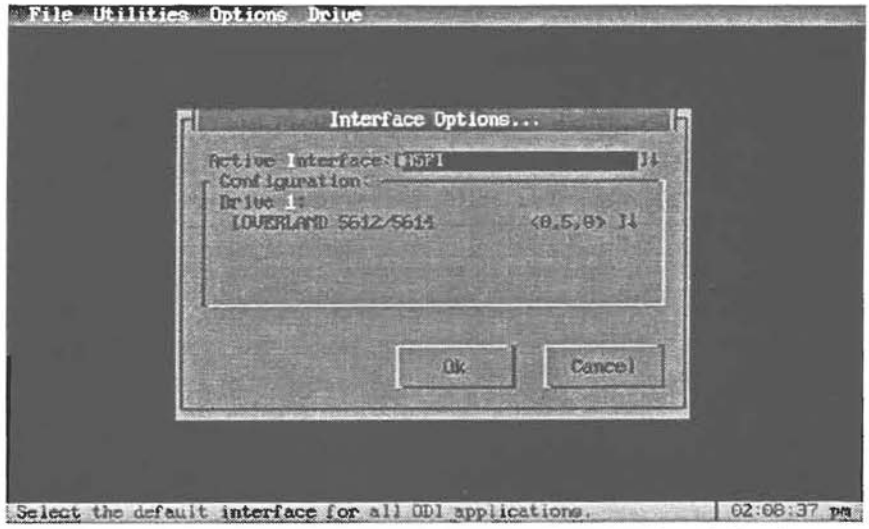


Figure 23, ASPI Interface —Source & Destination Selection

Figure 24, ASPI Interface —Tape Drive Selection

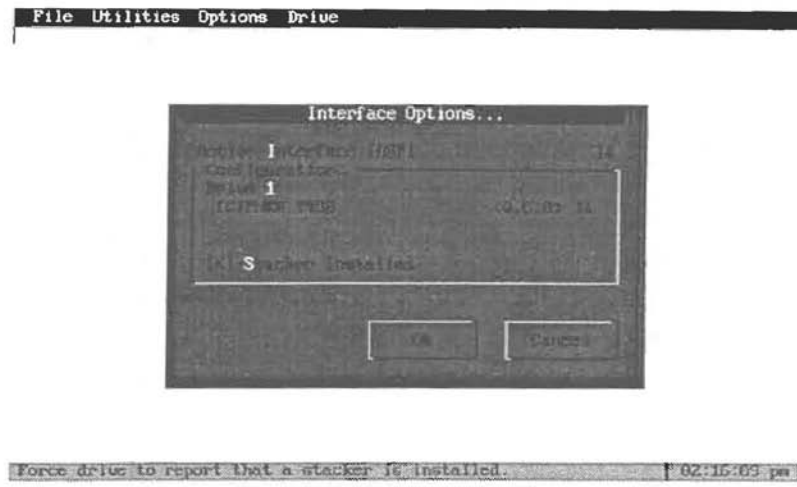


Figure 25, ASPI Interface —After Tape Drive Selection

5. Check *Stacker Installed* if you are using an Overland Data T480 tape cartridge product with a stacker. If you are using an STK 3480 cartridge tape drive, this option is automatically updated. The *Stacker Installed* option is not editable for STK but is displayed as status. This option is not available for any other drive type.

6. This completes the ASPI selection and configuration procedure. Before you run an actual application, test the subsystem by selecting *Test Tape Drive* from the *Utilities* menu.

Selecting and Configuring the Parallel Port Interface

1. Make sure all hardware installation is complete before attempting configuration. Refer to the heading, *Parallel Port Interface* in Chapter 3 for more information on hardware installation.

2. Working from the *Interface Options* pane, click the arrow on the right of the pane to open a drop-down box with a list of available interfaces.

3. Highlight *Parallel Port* to select Parallel Port (Figure 26) as the active interface.

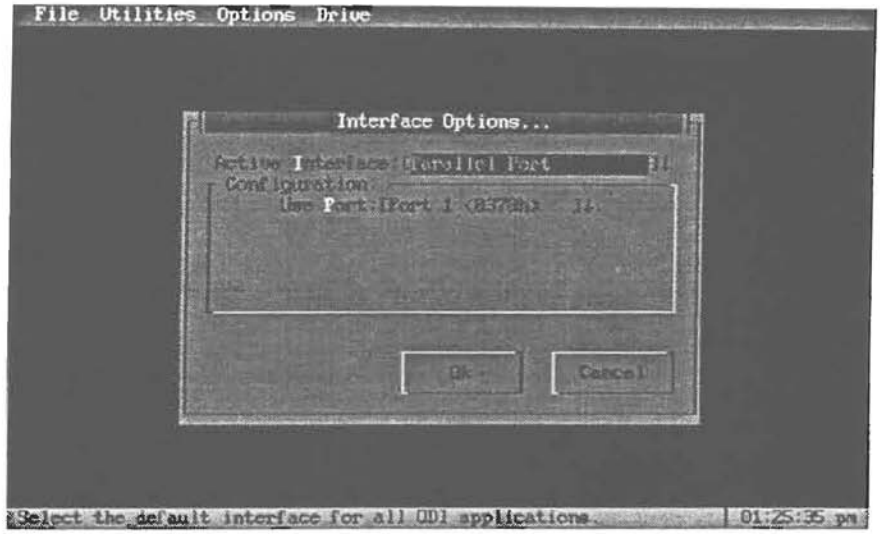


Figure 26, Parallel Port Interface Selection

4. Configure the interface by selecting the port and address from the *Configuration Options* pane. If more than one port is available they will be listed in a drop-down box. To access the list of available ports, click on the arrow to the right of the currently selected port and address. Make your selection by highlighting, then click *OK* to accept. Overland Data uses the default configuration of LPT1 at I/O address 378 hex.

Note: **Some computers use a different address for the printer port (such as a monochrome adapter printer port). Before making a decision, verify available ports from the list the ports provided then make your selection.**

---

5. This completes the Parallel Port selection and configuration procedure. Before you run an actual application, test the subsystem by selecting *Test Tape Drive* from the *Utilities* menu.

Selecting and Configuring the TMC SCSI Interface Controller

1. Make sure all hardware installation is complete before attempting configuration. Refer to the heading, *TMC Controller Card* in Chapter 3, for more information on hardware installation.

2. Working from the *Interface Options* pane, click the arrow located to the right of the *Active Interface* pane to open a drop-down box with a list of available interfaces.

3. Highlight *TMC* to select TMC as the active interface controller.

4. Check *Stacker Installed* (see Figure 27) if you are using an Overland Data tape cartridge product with a stacker. If you are using an STK 3480 cartridge tape drive, this option is automatically updated. The *Stacker Installed* option is not editable for STK. It is displayed only as status. This option is shown in grey if you are using an STK without a stacker.

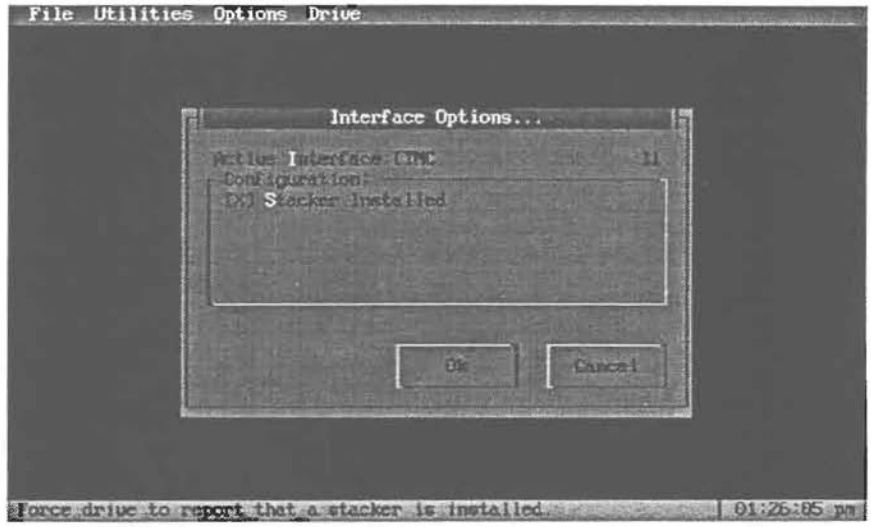


Figure 27, TMC Interface Controller—Stacker Selection

5. Click *OK* to accept the configuration.

Note: **If you are installing a TMC-MCA (microchannel) controller, you must configure your subsystem using the reference diskette provided by your computer manufacturer.**

6. This completes the TMC selection and configuration procedure. Before you run a data application, test the subsystem by selecting *Test Tape Drive* from the *Utilities* menu.

Selecting and Configuring the TX-8 Cipher/Pertec Interface Controller

1. Make sure all hardware installation is complete before attempting configuration. Refer to the heading, *TX-8 Controller Card*, in Chapter 3 for more information on hardware installation.

Note: **If you change the the controller hardware configuration, make sure you make a notation of the new settings for reference when configuring the software.**

---

2. Working from the *Interface Options* pane, click the arrow to the right of the *Active Interface* pane to open a drop-down box with a list of available interfaces.

3. Highlight *TX-8* to select the TX-8 as the active interface controller.

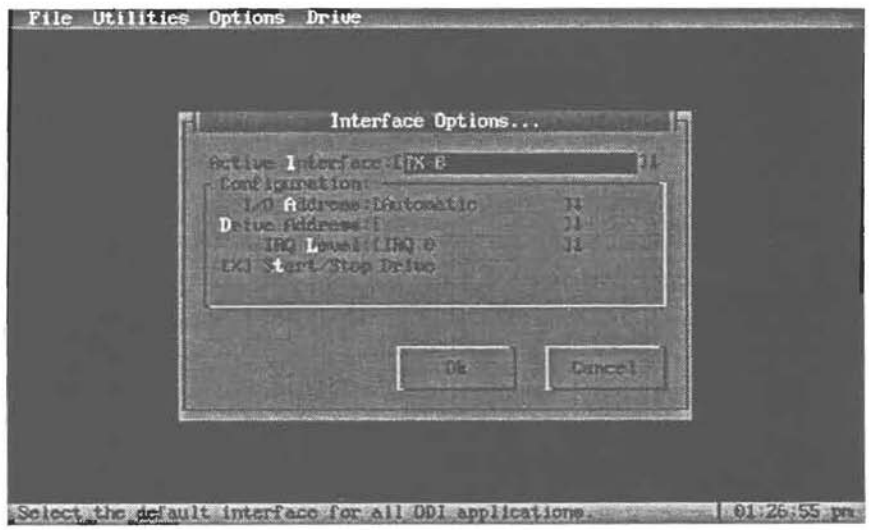


Figure 28, TX-8 Interface Controller Selection

4. If you made no changes to the hardware settings skip to step five.

5. If you changed the hardware settings to avoid a conflict, you must change your software to match the hardware configuration. The following options are available:

I/O Address

a. Click the arrow on the right side of the *I/O Address* option to display a list of available addresses in the drop-down options box. A complete list of addresses is provided below:

300	340
310	350
320	360 Default
330	

Note: **Normally the controller address can remain in *Automatic* if your controller is configured for one of the addresses listed in step a (above). The software automatically searches the listed range of addresses. Setting an exact address is usually done if you are operating with two or more controllers or to avoid a conflict.**

b. Highlight the I/O address that matches the address you set on your controller.

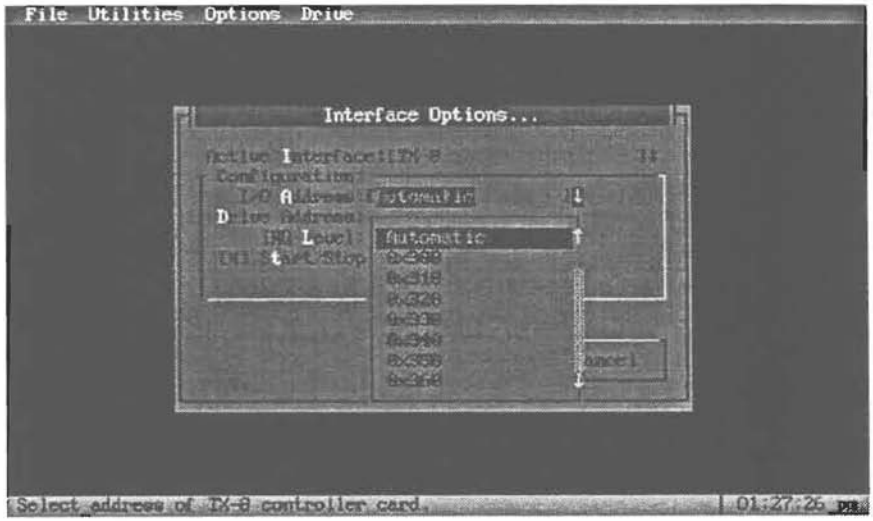


Figure 29, TX-8 Interface Controller—Address Selection

Drive Address

a. Click the arrow on the right side of the *Drive Address* pane to display a list of available tapedrive addresses. Default is drive *0*. Addresses are selectable from 0 to 7.

Note: **Normally this option is not changed. The default for most 9-track tape drives is 0 (zero).**

b. Highlight the tape drive address that matches the address you set on your tapedrive.

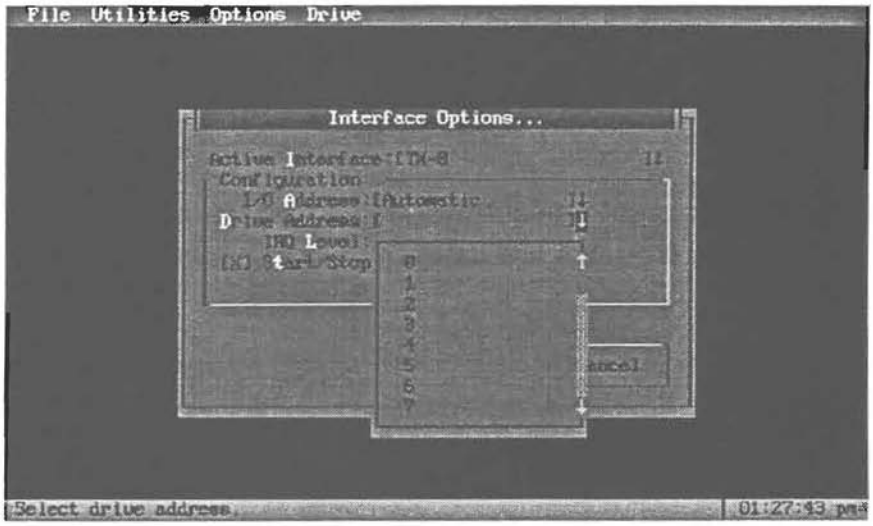


Figure 30, TX-8 Interface Controller—Drive Address Selection

IRQ Level

a. Click the arrow on the right side of the *IRQ Level* pane to display a list of available IRQs in the drop down options box. The default is IRQ 0. IRQ 0 is the timer tick provided by the computer. The range for extended interrupts range from 3 to 15.

Note: **If you are using a computer without extended interrupts (PC or XT), your range of IRQ options are 3, 4, 5, or 7. Refer to your computer manual for information on interrupts dedicated for other purposes.**

b. Highlight the address that matches the IRQ you set on your controller.

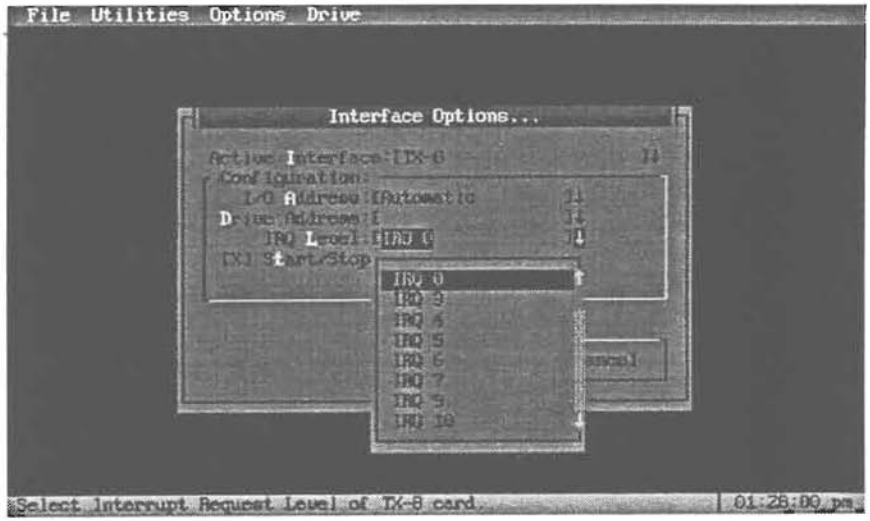


Figure 31, TX-8 Interface Controller—IRQ Selection

Start/Stop Drive

Check this box only if your tape drive is a Start/Stop tape drive. Refer to your tapedrive manual if you are not sure.

Note: **Overland Data does not sell a start/stop tape drive.**

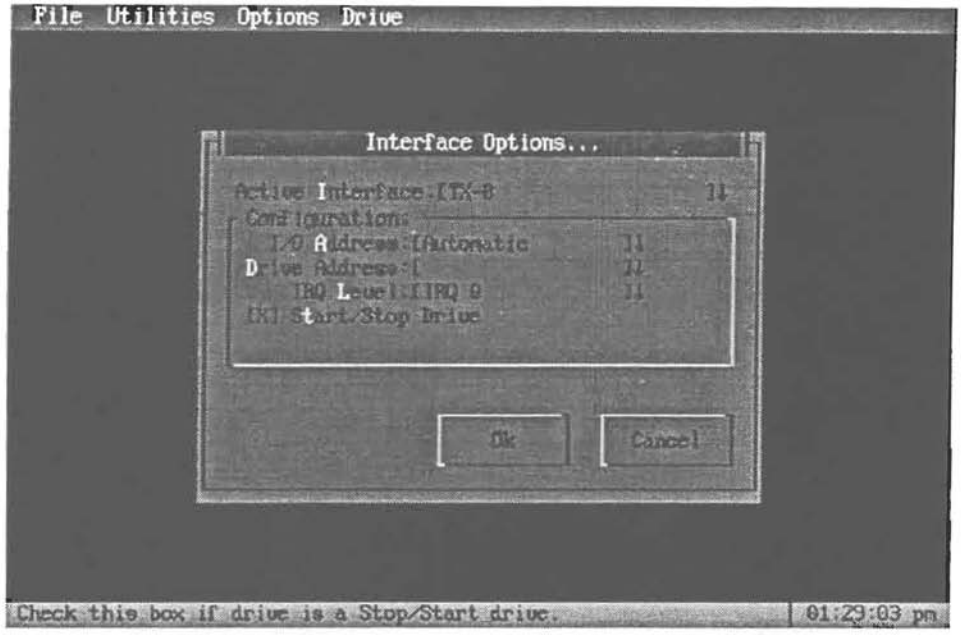


Figure 32, TX-8 Interface Controller—Start/Stop Drive Selection

6. Click *OK* to accept this configuration.

7. This completes the TX-8 interface selection and configuration procedure. Before you run an actual application, test the subsystem by selecting *Test Tape Drive* from the *Utilities* menu.

Selecting and Configuring the TXi-16 Cipher/Pertec Interface Controller

Note: **The software configuration settings (other than *Interface* selection) do not become effective unless the controller is in *Configuration Mode*. If you are changing the tape drive type or the I/O address, make sure the TXi-16 controller is in *Configuration Mode*. Refer to the heading, *TXi-16 Configuration* in this manual.**

1. Make sure all hardware installation is complete before attempting configuration. Refer to the heading, *TXi-16 Controller Card*, in Chapter 3 for more information on hardware installation.

2. Working from the *Interface Options* pane, click the arrow to the right of the pane to open a drop-down box with a list of available interfaces.

3. If you are selecting the TXi-16 interface without changing its configuration, then you do not have to place the controller in *Configuration Mode*. You need only to perform step three then skip to step five. If you *do* need to configure your controller for a different drive type or I/O address, perform all steps. Default settings are as follows:

   Drive Type: 32XX/36XX

   I/O address: 340h

4. Highlight *TXi-16* to select the TXi-16 (Figure 33) as the active interface controller.

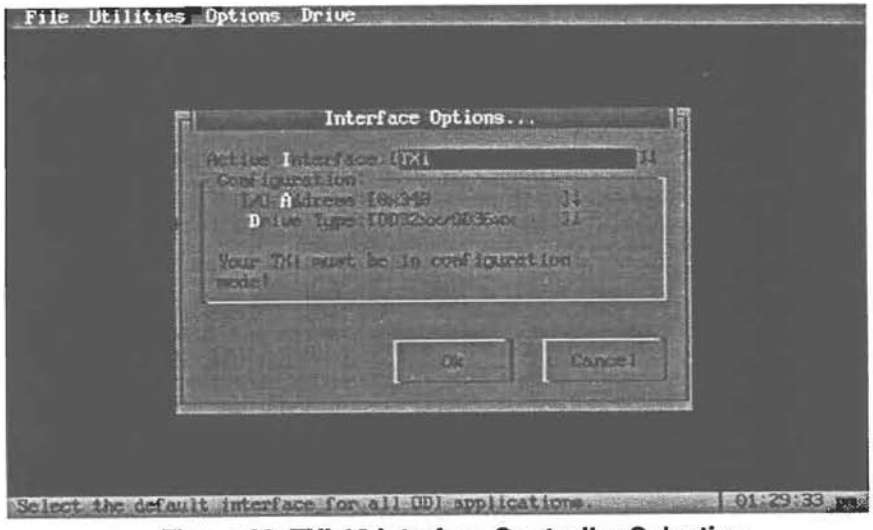


Figure 33, TXi-16 Interface Controller Selection

5. If you need to change the default settings (I/O 340, tape drive 32XX/36XX), the following options are available:

I/O Address

a. Click the arrow to the right side of the *I/O Address* label to display a list of available addresses (see Figure 34) in the drop down options box. The following is a list of available I/O address:

100	300
110	310
220	320
250	330
260	340 Default
280	350
290	360
2A0	3E0

b. Highlight the I/O address that matches the address you configured your controller to.

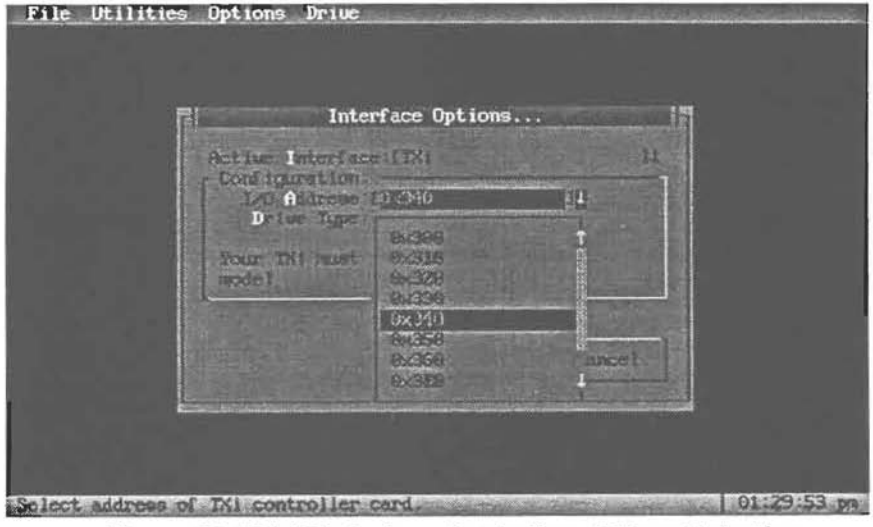


Figure 34, TXi-16 Interface Controller—Address Selection

Drive Type

Click the arrow on the right side of the *Interface Options* pane to display a list of available drive types (see Figure 35) in the drop-down options box. A complete list of available drives is provided below:

Anritsu 2620	Overland 64XX
Cipher F880	Overland 32XX/36XX
Cipher M990	Pertec FS2000
Cipher M995	Qualstar 1052
HP 88780B	Qualstar 1260
Kennedy 9600	Qualstar 34XX
Kennedy 9610	PE Streamer
M4 9905	Start/Stop Drive
M4 9914	

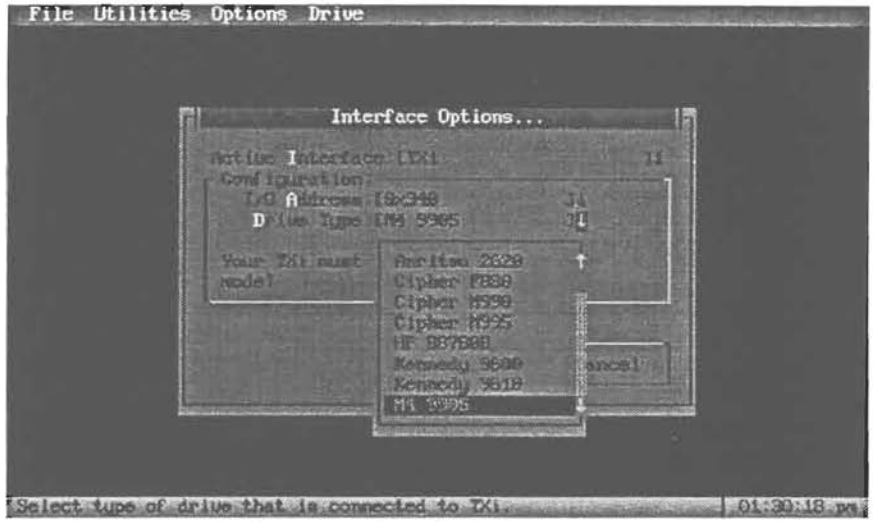


Figure 35, TXi-16 Interface Controller—Tape Drive Selection

6. Click *OK* to accept this configuration.

7. This completes the TXi-16 interface selection and configuration procedure. Before you run an actual application, test the subsystem by selecting *Test Tape Drive* from the *Utilities* menu.

Selecting and Configuring the XL/2 Cipher/Pertec Interface Controller

The following procedure describes how to select the XL/2 as the interface controller. Options for the XL/2 are not configurable with this software. Configuration must be done by using the reference disk provided by your computer vendor.

1. Make sure all hardware installation is complete before attempting configuration. Refer to the heading, *XL/2 Controller Card*, in Chapter 3 for more information on hardware installation.

2. Working from the *Interface Options* pane, click the arrow to the right of the *Active Interface* label to open a drop-down box with a list of available interfaces.

3. Highlight *XL/2* to select the XL/2 as the active interface controller.

4. Click *OK* to accept this configuration.

5. If you need to configure your controller for anything other than the defaults (see below), use the *Reference* diskette (provided by computer vendor) to configure the system. Refer to the heading, *XL/2 Controller Card* or the manual that was provided with your computer for information on configuring with the reference diskette. Default settings follow:

   I/O Address ..... 360 (hex)
   Interupt ..... 10 (decimal)
   Drive Address..... 0
   Density/Speed.... Normal
   Start/Stop... ..... Streaming

6. This completes the XL/2 interface selection and configuration procedure. Before you run a data application, test the subsystem by selecting *Test Tape Drive* from the *Utilities* menu.

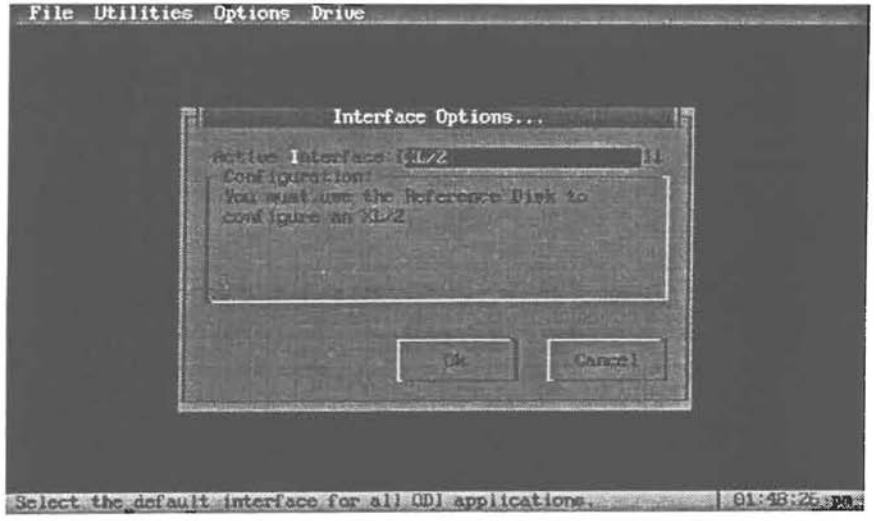


Figure 36, XL/2 Interface Controller Selection

This Page Blank

Translation Tables

Translation tables are provided as a utility to translate data from ASCII to EBCDIC when writing a tape or EBCDIC to ASCII when reading a tape. These tables are called automatically when the*Translate* parameter is used in any of the data transfer applications.

The *Translation Tables* selection from the *Options* menu provides the capability to select the table(s) you wish to use as default, to edit existing tables, to delete tables, or to create new tables. The following section explains how to accomplish these tasks.

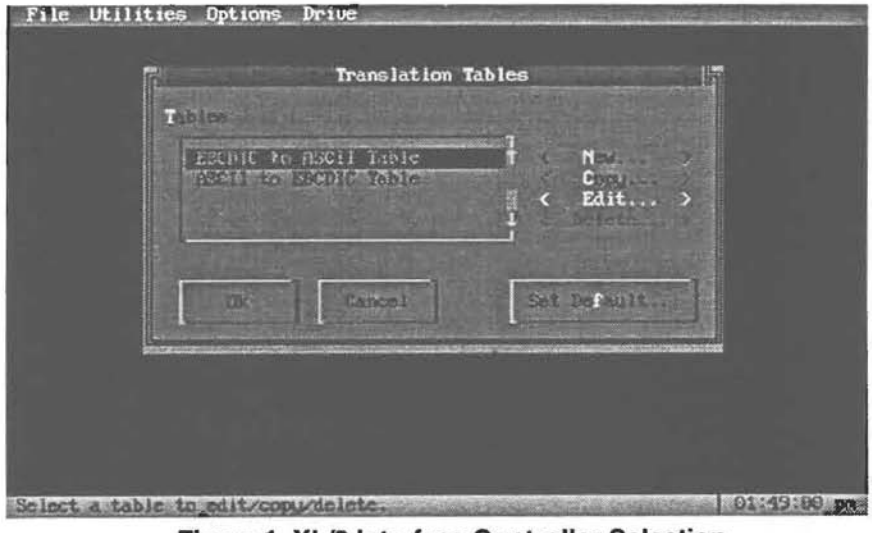


Figure 1, XL/2 Interface Controller Selection

Changing to a Different Default Table

This option provides the ability to change the standard default table or a previously selected table to a different one. Normally the default, when reading from a tape is to translate from EBCDIC to ASCII, and during a write operation is to translate from ASCII to EBCDIC.

Note: **Tables other than the standard default tables must exist to prior to using the *Set Default* option. The standard default can be changed but is normally not done except for special translations.**

The *Set Default* option allows you to select from the list of tables, a different table to be used as the default for a read or write operation. The following section provides the procedure to change the default:

1. Select *Set Default*, from the *Translation Tables* window.
2. The *Default Translations* window opens with two selections available, *Reading From Tape*, and *Writing To Tape*.

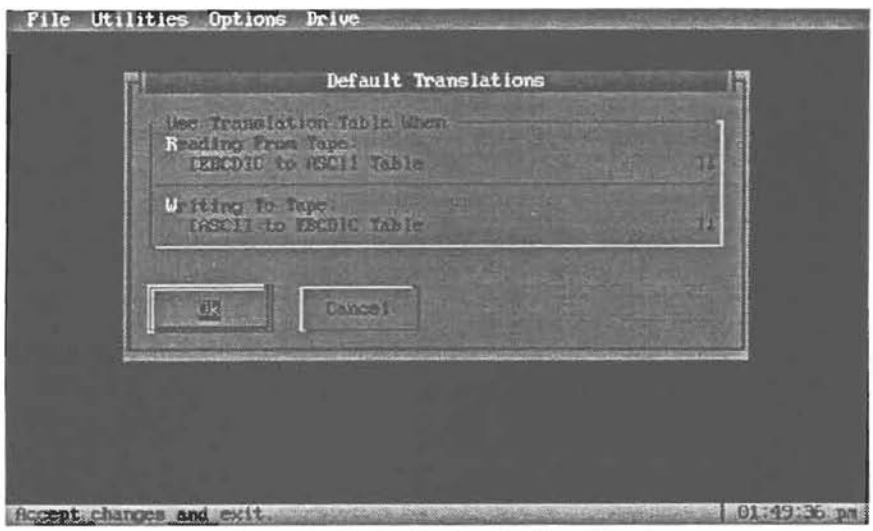


Figure 2, Default Translation Window

3. Select the small arrow to the right of the operation you wish to change. This will produce a list of pre-existing tables.

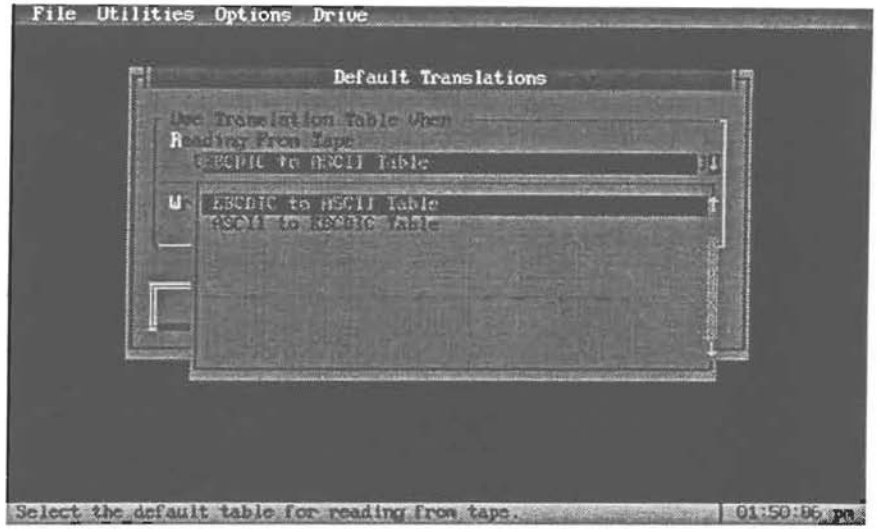


Figure 3, Sample of Default Table Selections

4. Highlight the name of the table you want to use as the default, then select **Ok**[1].

Creating a New Translation Table

This option provides the capability to create a new translation table from a template. This template provides a list of 256 hex or decimal values listed in a one-to-one incrementing pattern. The following procedure explains how to create a new table from this template:

1. Working from the *Translation Tables* window select *New*. The following screen will be displayed:

[1]Select **Cancel** to undo the changes you made and return to the *Translation Tables* window.

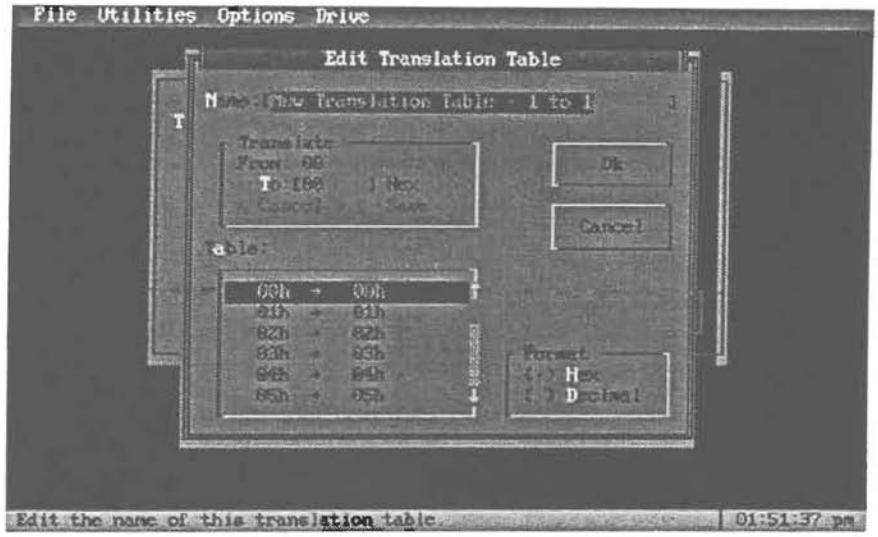


Figure 4, Edit New Translation Table

2. The screen opens with the *Name* field highlighted. Enter a unique name to identify the new table.

Note: **You cannot create a table with the same name as an existing table.**

3. Go to the *Table* pane and highlight the first set of values to be changed.

4. Go to the *Translate* pane and edit[2] the *To* field as needed.

5. Press **[ENTER]** to accept the change[3] and return to the translate pane. You may cancel an entry at any time before exiting this field. The entry will return to its original value.

---

[2] This process can be done in hex or decimal by selecting the desired system in the *Format* field.

[3] You may abort the operation by selecting *Cancel*. Selecting this option returns you to the original window. If you cancel you loose all changes.

You also have the option restore the original values by selecting *Restore*. Selecting this option changes all entries you made back to their original values.

6. Highlight the next value to change and repeat steps 1 to 4 until all values have been entered.

7. After all values have been entered select **Ok** to save the table and return to the main *Translation Tables* screen.

8. The new table will be listed along with the other previously existing tables.

9. If you wish to make this on of the default tables, refer to the heading, *Changing to a Different Default Table*.

Copying an Existing Translation Table

This option allows you to create a clone of an existing table. This is useful when making small changes to a table that you may want to use again. Adjustments to the table can be made by highlighting the new name then selecting *Edit*. Use the following procedure to copy an existing table:

1. Working from the *Translation Tables* window select *New*. This opens the *Edit Translate Table* window.

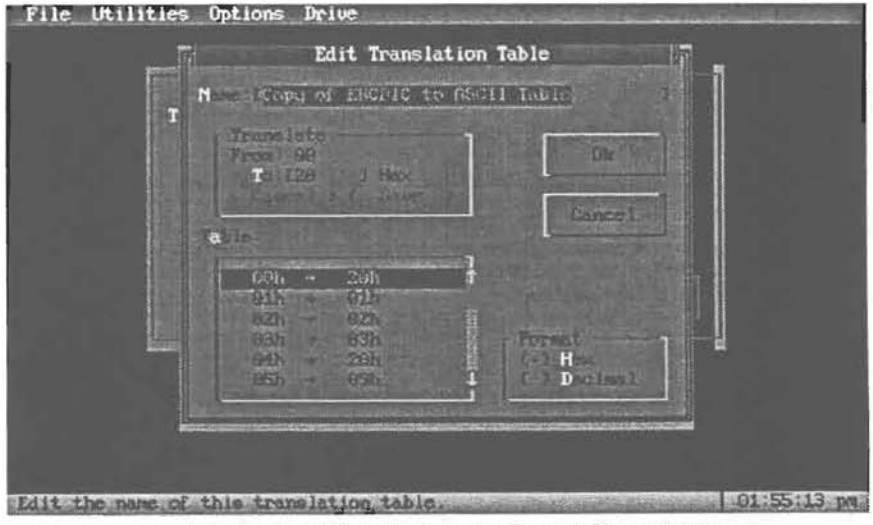


Figure 5, Edit a Copy of a Tanslation Table

2. The window opens with the *Name* field highlighted. Enter a unique name to identify the copy.

Note: **You cannot create a table with the same name as an existing table.**

3. If you need to make changes to the table, perform steps 3 to 7 under the heading *Creating a New Table*.

4. If you wish to make this on of the default tables, refer to the heading, *Changing to a Different Default Table*.

Editing an Existing Translation Table

This selection is used to change the values contained in an existing table.

Note: **If you edit a default table, a** RESTORE **button is displayed. This button *resets* all translation table edits back to the factory default. You *cannot* delete the two default translation tables.**

1. Highlight the table you wish to edit, then select *Edit*. The *Edit Transation Table* window will open.

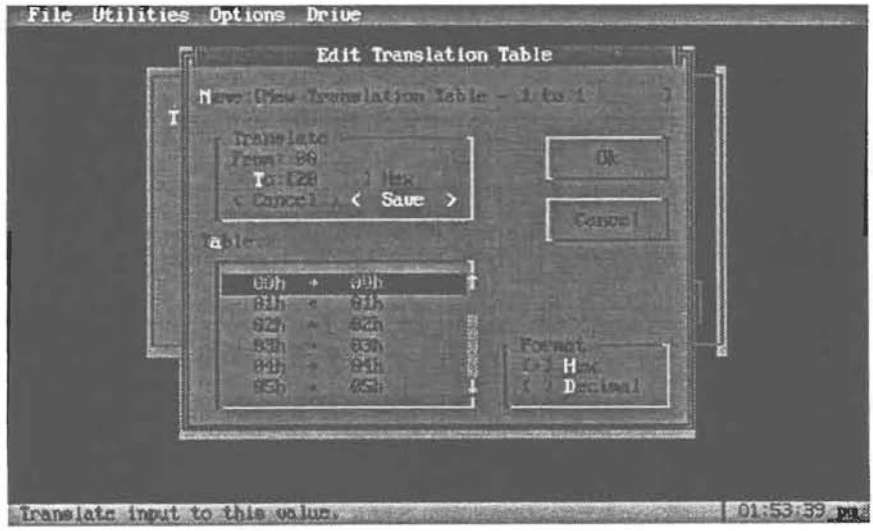


Figure 6, Edit an Existing Translation Table

2. To make changes, perform steps 3 to 7 under the heading *Creating a New Table*.

Deleting a Table

This option allows you to remove outdated, duplicate, or unwanted tables. Once removed, they cannot be recovered.

1. Working from the *Translation Tables* window, highlight the table to be deleted.

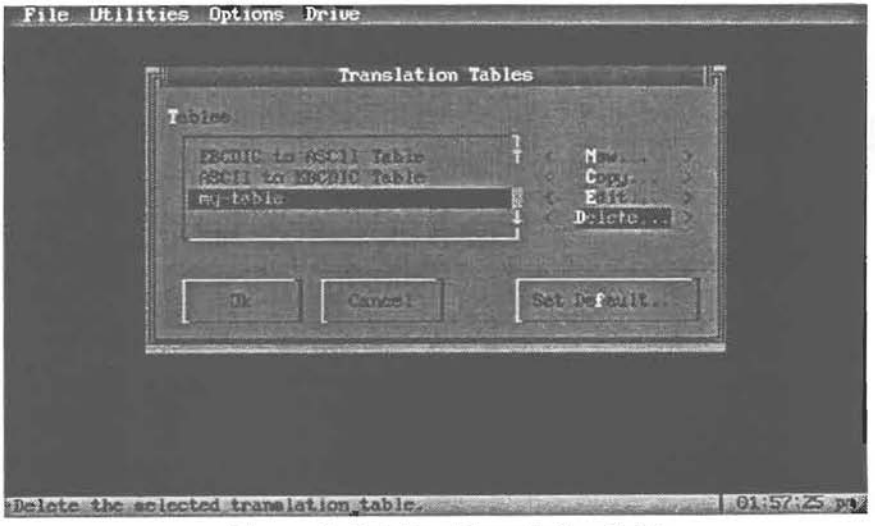


Figure 7, Delete a Translation Table

2. The following popup window will appear confirming the name of the table that will be deleted.

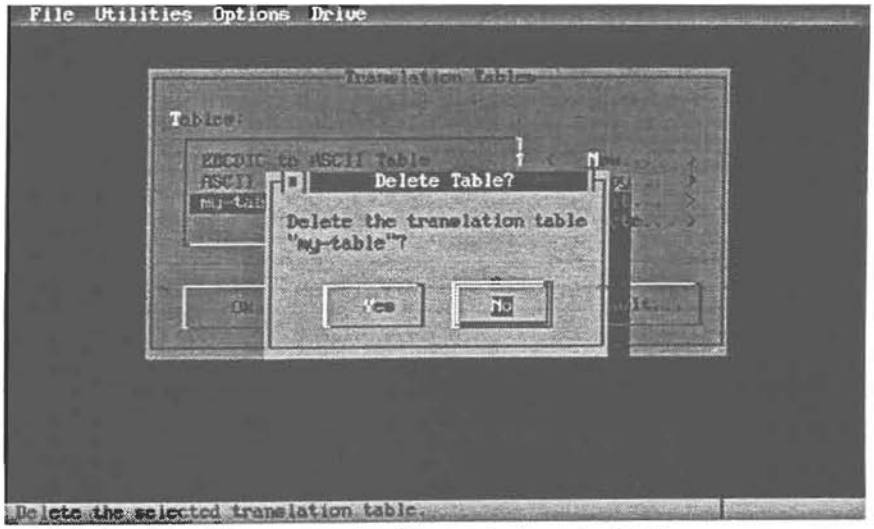


Figure 8, Confirm Deletion of a Table

3. Verify the name of the table then select **Yes** to delete.

Compression

This utility provides the capability to activate compression. It is only available if you have an STK 4220 tape drive equipped with compression. The software activates compression for write operations only. When reading a tape, compression is handled automatically by the tape drive.

The following section explains the options available:

Use ICRC Data Compression

Choose this check box to turn compression on for all the utilities. If compression is not selected here, the option is available in each of the write utilities.

Figure 9, Compression Window, Selected From Options

Prompt Before Running Backup Programs

Choose this box in addition to the *Use ICRC Data Compression* option if you wish to be notified prior to backups that ICRC is on.

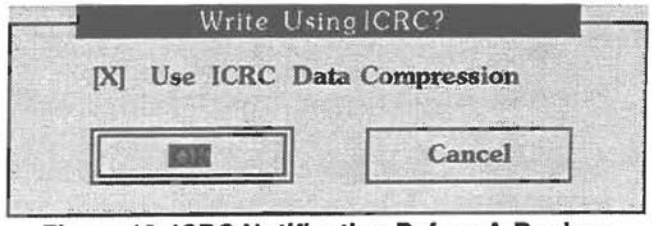
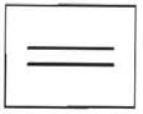
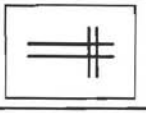


Figure 10, ICRC Notification Before A Backup

Display

Show Heads-Up Display

The Heads-Up Display (HUD) option lets you view the tape activity and status of the read/write functions: *Read Unlabeled Tape, Write Unlabeled Tape*, and *Write Labeled Tape*. HUD appears in a separate window in the lower part of your screen. The following table describes a list of conventions used in the Heads-up Display:

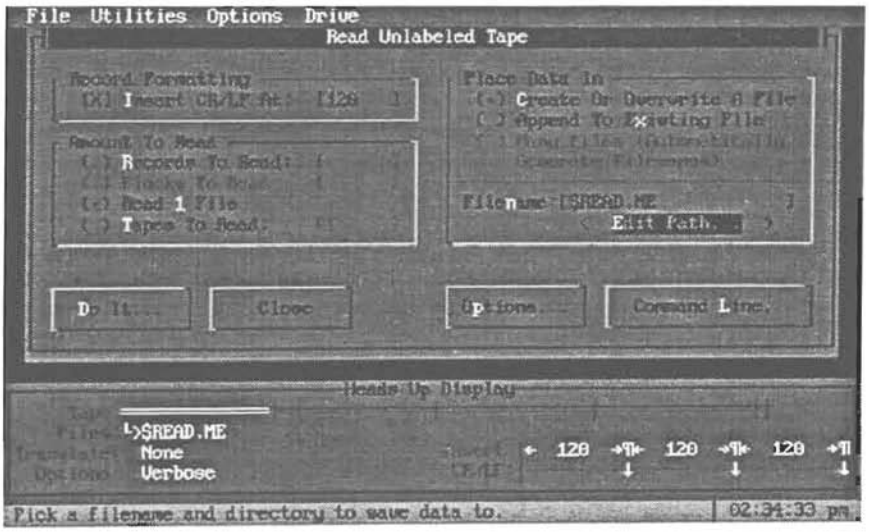
Heads-up Display Symbol Conventions	
Symbol	Description
	This symbol is used only when writing a labeled tape. It indicates a single block within a header.
	This symbol represents the 9-track tape.
	This symbol represents a single filemark.
	This symbol represents two filemarks or Logical End of Tape (LEOT).
	This symbol is used in Disk Format to represent the end of a record. It is displayed when inserting CR/LF.

Examples- Heads-up Display

The following Heads-up Display shows options made from the *Read Unlabeled Tape* selection. All options except the following choices remain at their default values:

- Insert CR/LF At: [120] Inserts carriage returns and line feeds into the data every 120 bytes. This defines the record.

- Tapes To Read [1] To concatenate all the files on tape into one file as shown in the Files: line. The name of the single output file is DATA.OUT.



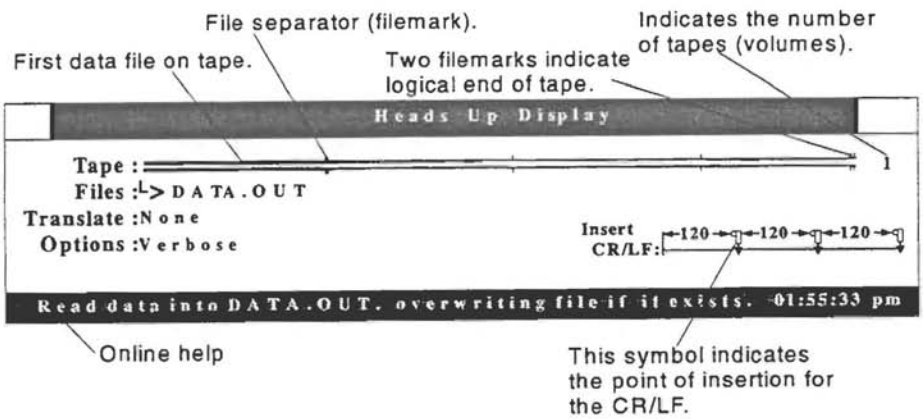


Figure 1,

The following Heads-up Display shows options made from the Write Unlabeled Tape selection. All options except the following choices remain at their default values:

Record Select this option to strip carriage returns and line feeds from the output. The sample in Figure 20 shows the following entries:

- Record Length This field sets the point that the carriage returns and line feeds are to be stripped from the data. In this case carriage returns and line feeds are stripped every 120 bytes.
- Records Per Block This field sets the size of the block to be written to tape. You can determine your block size with the following simple calculation. Multiply the record length (rl) times the number of records you wish to write per block or blocking factor (bf). The product is the block size (bs).

  (rl) [n] X (bf) [n] = (bs)

  In our example:

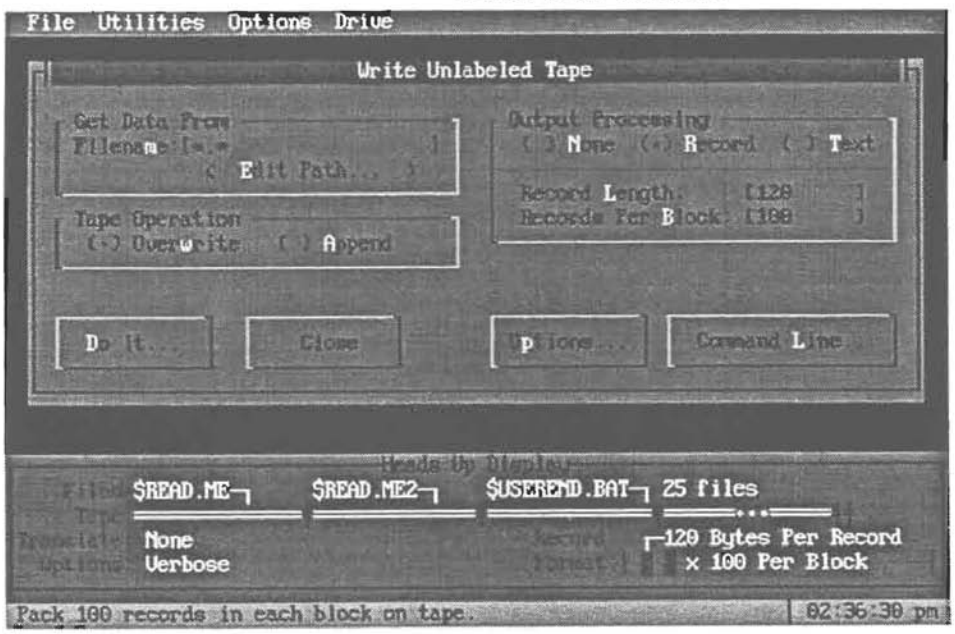
  120 X 100 = 12,000

  Where:

  120 is the record length

  100 is the blocking factor

  12,000 is the block size



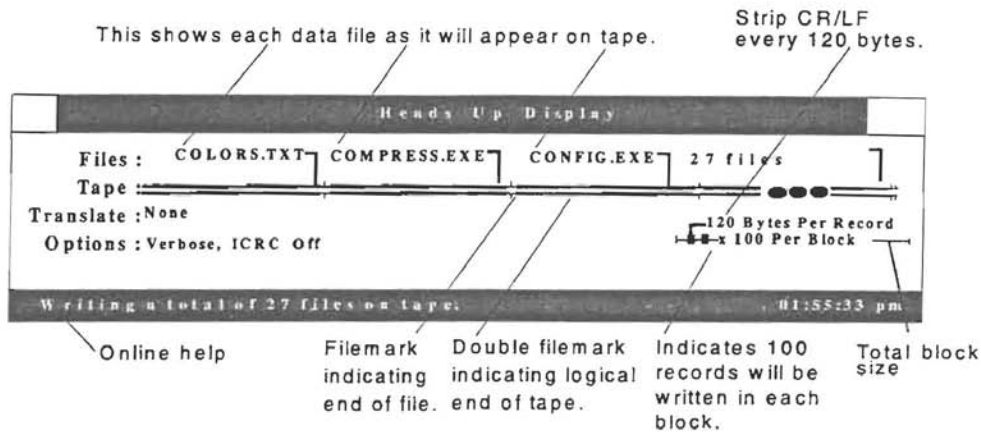


Figure 2, ,The following Heads-up Display shows options made from the *Write Labeled Tape* selection. All options except the following choices remain at their default values:

Record Select *this* option to strip carriage returns and line feeds from the output. The sample in Figure 20 shows the following entries:

- Record Length This field **sets the** point that the carriage *returns* and line feeds **are** to be stripped from *the data.* In this **case carriage returns and** line **feeds are stripped every** 120 bytes.
- Records Per *Block* This field sets the size of the block to be written to tape. You can determine your block size with the following simple calculation. Multiply the record length (rl) times the number of records you wish to write per block or blocking factor (bf). The product is the block size (bs).

  (rl) [n] X (bf) [n] = (bs)

  In our example:

  120 X 100 = 12,000

  Where:

  120 is the record length

  100 is the blocking factor

  12,000 is the block size

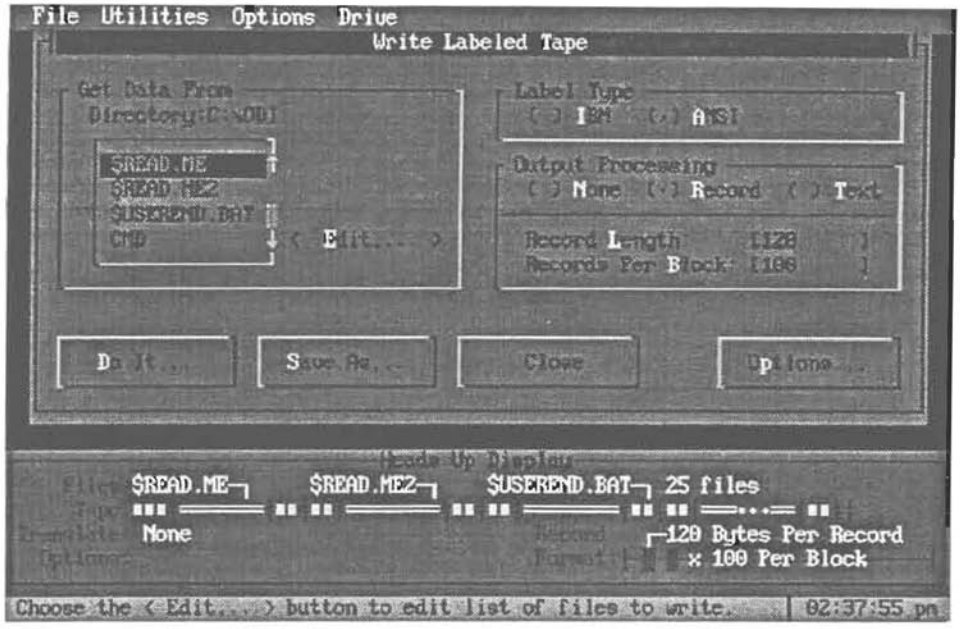
Display Summary

Working from the *TAPE* line in the following display, we see that two files have been selected to write to tape. This is indicated by the small rectangles or blocks. These blocks represent blocks of data used to create labels.

In this case, an IBM Standard Label has label at the beginning of tape with volume information and two headers. This is represented by three blocks at the beginning of the tape.

Next is a length of tape representing the data file, followed by two more blocks or the trailer. The trailer contains information similar to the two header blocks. This is where labeling would end if there was only one file to be transferred. Since there are two files to be transferred the next two blocks are the headers for the that file, followed by the data and finally another trailer.

IBM Standard Label



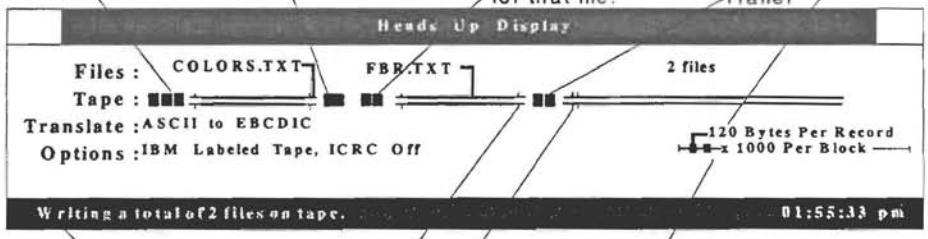
The first three rectangles represent three 80 byte blocks in the Volume & Header Label.

These two rectangles represent two 80 byte blocks in the trailer.

If another file follows, the next two rectangles represent the header for that file.

Strip CR/LF every 120 bytes.

Trailer



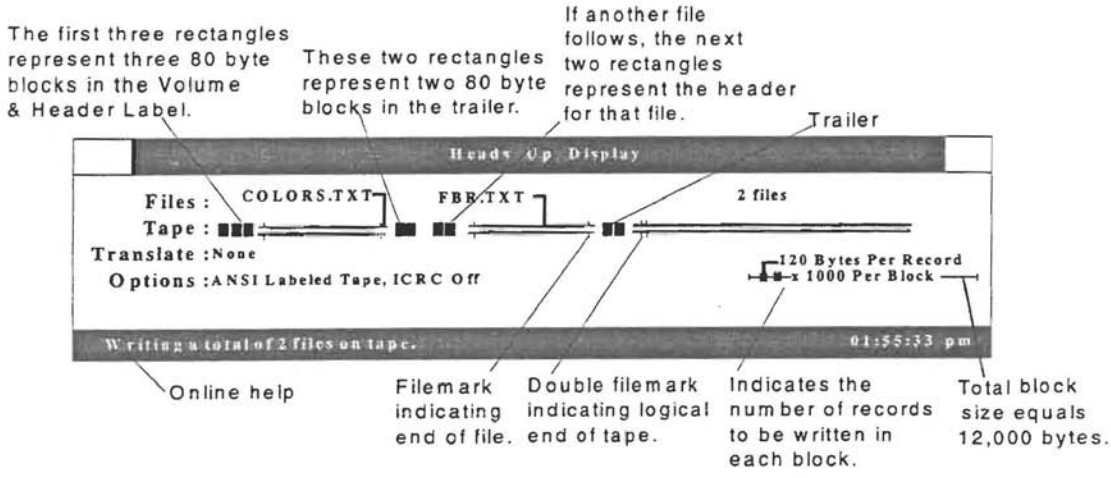
Online help

Filemark indicating end of file.

Double filemark indicating logical end of tape.

Indicates the number of records to be written in each block. Total block size equals 12,000 bytes.

As you can see in the following illustration, the standard format remains the same between IBM and ANSI Standard Labels. The differences are in some of the requirements within the labels and IBM labels use EBCDIC character set.ANSI Standard Label



Drive

Getting Around in Depot4

The following section describes the mechanics of Depot4 by showing you how to: move within a window, select options, and interpret conventions. This section also provides a brief overview of features.

Mouse Conventions

Make sure your mouse driver has been installed. ODI does not provide drivers for this capability. Refer to your product manual for more information on installation and configuration of the mouse for DOS programs. The following is a list of action terms that are associated with mouse movements and the results of that action:

Name	Function
Clicking or Click	This means to press the left button on your mouse one time.
Double Clicking or Double Click	This means to press the left button on your mouse twice in rapid succession. It has a similar function to the **[ENTER]** key on your computer.
Pointer	This is the cursor that moves when you move your mouse. It is different than your keyboard cursor in that , the mouse cursor can be moved to areas other than the affected area. The Mouse Cursor affects an area only after it is moved to that area and you click or double click. Clicking or Double Clicking moves the keyboard cursor to that point.
Dragging	Clicking and holding the button down as you move the pointer. This action can be used to block text.

Keyboard Conventions

Name	Key(s)	Function
Shortcut Keys	[ALT] + [*letter*]	Use the [ALT] key in combination with the highlighted letter to choose a menu or option. These shortcut keys are the easiest way, from the keyboard, to access or change an option.
Move Cursor	[TAB]	The [TAB] key allows you to move to different options or parameters in a window.
Move Cursor	[↓]	The down arrow key allows you move down to an option within a pane.
Move Cursor	[↑]	The up arrow key on allows you move up to an option within a pane.
Move Cursor	[→]	The right arrow key on allows you move the keyboard cursor right.
Move Cursor	[←]	The left arrow key on allows you move the keyboard cursor left.
Block Text (EDIT)	[SHIFT] + [ARROW]	Use this combination of keys to select a block of text for editing.
Cut (EDIT)	[CTRL] + [X]	Cuts the selected text from the text field and stores it for the purpose of pasting it to another text field. The text is stored until Cut is invoked again or the Copy function is used.
Copy (EDIT)	[CTRL] + [C]	Stores a copy of the selected text for pasting to another text field.
Paste (EDIT)	[CTRL] + [V]	Pastes text from a Cut or Copy function

Miscellaneous Conventions

Name	Function
Menu	A menu is selectable using either a mouse or keystrokes. When selected, menus display a list of tape available utilities.
Option	After a utility is selected from a menu, some utilities display a Main Window with several selectable items; these items are called options.

Window Conventions

The following table provides a list of names and functions for items used in a Depot4 window. This information will help you understand the logic to any window or pane used in Depot4. Refer to Figure 24 for samples of each convention.

Name	Function
Command Button	Command buttons invoke some action. Buttons come in two types, Standard Buttons and Small Buttons. The center of these buttons display the function that is invoked when the button is selected. Buttons can be selected three ways: 1. by moving the mouse pointer to the desired button, then clicking 2. by pressing **[ALT]** + ***[letter]*** of the command 3. by tabbing to the button and then pressing **[ENTER]**.
Radio Button	Radio Buttons are mutually exclusive. There is always more than one Radio Button, this is called a group. When one button is selected, all other button(s) in a group, are toggled off.
Text Field	Text Fields are used to enter text. The edit commands, listed in Keyboard Conventions, are active for this field only.
Check Box	Check Boxes are either on or off. They are toggled on or off by moving the mouse pointer to the box and clicking or by pressing **[ALT]** + ***[letter]***.
Drop Down Combo	This is used to display a list of items. Click on the arrow to display a List Box.
List Box	A List Box contains a list of items. The items can extend beyond the boundaries of the List Box; click the scroll bar on the right or use the [↓] or [↑] keys to scroll through a list.
Label	A label is used to describe the contents, function, or input request of any field within a window.

DEPOT2

DEPOT2 offers increased speed and the convenience of command file execution. DEPOT2 transfers data between system disk and magnetic tape, either in its binary form or with optional translation. The program assumes that data to be translated to EBCDIC is ASCII and data to be translated to ASCII is EBCDIC.

Features

DEPOT2 has the following features:

- Copies data from tape to disk
- Copies data from disk to tape
- Converts data: ASCII to EBCDIC (optional during tape writes)
- Converts data: EBCDIC to ASCII (optional during tape reads)
- Positions to an arbitrary tape location prior to read
- Permits user specified record sizes with a blocking factor for inserting or stripping carriage-return line-feeds
- Reads options from the command line
- Optionally reads commands from a file, permitting batch files for multiple tape operations.
- Has enhanced error handling of tape errors and disk errors.
- Has enhanced support for writing multivolume tapes.

overview

Depot2 is used in circumstances when Depot4 cannot be used. For most of your data transfer needs Depot4 is all you need- so if you are not sure where you should start, try Depot4.

Depot2 performs two basic functions, read or write, hence this chapter is divided into read and write sections.

Depot2 uses command line switches to invoke options or functions. These switches are always letters preceded with a forward slash (/). The slash indicates to the program that an option will follow.

Note: **Switches *are* case sensitive. For example the /M means to read until a filemark is detected, however, a /m means to skip a filemark. Make sure the correct case has been used.**

Some switches require arguments, for example; the record length (/r) requires the length of the record to follow the letter; i.e., /r120. Each section provides a list the switches, its argument (if needed), and a brief description. Also included are examples of some common operations.

Note: **Switches can be placed in any order on the command line.**

Reading from Tape

The *absence* of option /w and /a specifies a read operation.

The /r and /b switches are normally only used with ASCII text files, as the automatic padding. Do not use these switches to transfer binary or executable files. *The /s switch is the normal switch which you will use.* Under special circumstances, where you want records of fixed length, use /r and /b.

Use the /T switch to indicate that more than one tape will be read. Specify the destination file name, such as /n outfile, and the stopping criteria (such as reading until **DEPOT2** reads a filemark: /M). You use the same switch, /n, for both the source and destination file. The tape operation (read or write) dictates how **DEPOT2** uses the file. If the file already exists for a tape write operation, **DEPOT2** asks for permission before overwriting. If /i (ignore) is used, **DEPOT2** assumes permission to overwrite.

If there are several files on tape and you wish to generate filenames automatically, use the /g switch instead of the /n switch.

On read, you may also specify a deblocking factor, a record size, files or blocks to skip, and translation (EBCDIC to ASCII).

Record Size and Blocking Factor (handling CR/LF delimiters)

Record size and blocking factor are intended for use with ASCII disk files where records are delimited by carriage returns and line feeds. While writing to tape, lines in the source file which are longer than record size will be truncated and lines which are shorter than record size will be padded with spaces; in either case, the CRLF characters are *not* written to tape. When reading a tape which was written with record size specified, that same record size should be specified again to restore the file to its original format (this assumes the written record size was large enough to prevent truncation of lines); the CRLF characters are inserted in the disk file.

Disk files which are in EBCDIC or binary may contain certain characters which are interpreted as control codes. This can cause these files to be modified when record size and blocking factor are specified. You may write EBCDIC or binary disk files to tape by specifying the blocksize (/s).

Carriage-Returns & Line-Feeds

Function	Operation	Depot2 Switch	Depot4 Option
Insert CR/LF	Read Tape	/r /b	Record Size & Blocking Factor
Strip CR/FL	Write Tape	/r /b	Record Size & Blocking Factor
No Change to CR/LF	Read Tape	/s or no block switch	Record Size & Blocking Factor Set to Zero
No Change to CR/LF	Write Tape	/s	Block Size

Use

When using **DEPOT2** to read a tape, remove the write enable ring for extra protection and load the tape onto the tape drive. For **DEPOT2** to write a tape, put a write enable ring on the reel and load the tape onto the drive.

Note: **If you are writing to tape, make sure the tape you choose has nothing important written on it.**

When the tape is loaded, place the drive online. Invoke **DEPOT2** from the command line, specifying any options, as follows

C:\> depot2 options

followed by **[ENTER]**, where options is one or more of the option switches specified below. Use option /c filename to tell **DEPOT2** to read filename instead of the command line for its program options. **DEPOT2**'s options consist of one or more switches separated by spaces. A switch is / (forward slash) followed by a letter. If a switch needs an argument, put the argument after the letter with a space. (The space is optional if the argument begins with a number.) For example, the following option specifies that the tape block size is 1024 bytes:

/s 1024

or

/s1024

You may combine toggle options (ones not requiring arguments) with a single slash, as in:

/tvM

This specifies options translate, verbose, and stop reading at filemark detection.

The following table describes the **DEPOT2** read tape options.

DEPOT2 Read Options

Switch	Argument	Description
/A	disk filename	Provides the capability to append to a file already existing on disk.
/B	val	Read val data blocks.
/b	val	Use blocking factor val. The default is 1.
/c	filename	Reads filename for **DEPOT2** commands. If it is present, **DEPOT2** will ignore *any* other switches on the command line.
/d	val	Skip val data blocks.
/g		Generate filenames automatically.
/i		Ignore errors (but print error messages).
/k		Ignore single filemarks, allowing files being read to be concatenated into a single DOS file. It must be asserted for *each* command line
/M		Read until **DEPOT2** detects a filemark.
/m	val	Skip val filemarks.
/n	name	Use name for destination on tape read operations. name must include the extension
/p		Set data bit 7 to 0.
/R	val	Read val records.
/r	val	Use record size val.
/s	val	Use val block size.
/T	val	Enter the number of volumes (tapes) to be processed.
/t		Translate data from EBCDIC to ASCII on tape read.
/v		Verbose.
/W		Suppresses the 'wait for user approval to exit keystroke' (normally the program waits for a user keystroke before exiting, to allow you to see what has been done; this switch tells it not to wait).

Note: **The r and b switches cause a CRLF (carriage-return line-feed) to be inserted after each record during disk to tape operations.**

Writing to Tape

Use option /w (write) or /a (append) to write to tape. Specify a block size (such as /s 1024) and a source file (such as /n infile) *or* you may specify a record size (/r) and a blocking factor (/b; the default is 1). Use *either* s *or* r and b. Do not use s *with* r and b. Setting r and b defines the delimiter stripping, setting s defines the size of the block. The /r and /b switches should be used only with ASCII text files. When writing to tape, using the /r and /b switches automatically remove the carriage-return, line-feeds (CRLF) from the disk file. This may cause partial data transfer. You can also specify translation (/t) (ASCII toEBCDIC on tape write), the number of filemarks (/m) or data (/d) blocks to skip.

Record Size and Blocking Factor

Record size and blocking factor are intended for use with ASCII disk files in which records are (or need to be) delimited by carriage returns and line feeds. While writing to tape, lines in the source file which are longer than record size will be truncated and lines which are shorter than record size will be padded with spaces. When reading a tape which was written with record size specified, that same record size should be specified again in order to restore the file to its original format (this assumes the written record size was large enough to prevent truncation of lines).

Note: **Disk files which are in EBCDIC or binary may contain certain characters which are interpreted as control codes. This can cause these files to be modified when record size and blocking factor are specified. You may write EBCDIC or binary disk files to tape by specifying the /s option.**

Note: **It is unnecessary to use /s when you use /r and /b, or vice versa. Use either, as you choose.**

The following table describes the **DEPOT2** write tape commands.

DEPOT2 Write Options

Switch	Argument	Description
/A	disk filename	Provides the capability to append to a file already existing on disk.
/a		Append to tape.
/B	val	Read val data blocks.
/b	val	Use blocking factor val. The default is 1.
/c	filename	Reads filename for **DEPOT2** commands. If it is present, **DEPOT2** will ignore *any* other switches on the command line.
/d	val	Skip val data blocks.
/g		Generate filenames automatically.
/i		Ignore errors (but print error messages).
/k		Ignore single filemarks, allowing files being read to be concatenated into a single DOS file. It must be asserted for *each* command line
/M		Read until **DEPOT2** detects a filemark.
/m	val	Skip val filemarks.
/n	name	Use name for destination on tape read operations. name must include the extension
/p		Set data bit 7 to 0.
/R	val	Read val records.
/r	val	Use record size val.
/s	val	Use val block size.
/w		Write new tape.
/t		Translate data from EBCDIC to ASCII on tape read.
/v		Verbose.
/W		Suppresses the 'wait for user approval to exit keystroke' (normally the program waits for a user keystroke before exiting, to allow you to see what has been done; this switch tells it not to wait).

Multiple File Operations

The /c ("filename") switch is very powerful, allowing you to create a **DEPOT2** command file, running a series of **DEPOT2** commands without supervision. An example of this, with more detailed explanation, follows, under *Example Command File*.

Warning!!Results are unpredictable if you use /s and /r and /b. Do *not* use both.

Options

This summary table describes *all* the **DEPOT2** read and write command options.

Depot2 Read & Write Options

Switch	Argument	Read and/or Write operation	Description
/a		W	Append to tape.
/B	val	R	Read val data blocks.
/b	val	R/W	Use blocking factor val. The default is 1.
/c	filename	R/W	Reads filename for **DEPOT2** commands. If it is present, **DEPOT2** will ignore *any* other switches on the command line.
/d	val	R	Skip val data blocks.
/g		R	Generate filenames automatically.
/i		R/W	Ignore errors (but print error messages).
/k		R	Ignore single filemarks, allowing files being read to be concatenated into a single DOS file. It must be asserted for *each* command line.
/M		R	Read until **DEPOT2** detects a filemark.
/m	val	W	Skip val filemarks.
/n	name	R/W	Use name of disk file on tape write operations. Use name for destination on tape read operations. name must include the extension
/p		R/W	Set data bit 7 to 0.
/R	val	R	Read val records.
/r	val	R/W	Use record size val.
/s	val	W	Use val block size.
/T	val	R	Enter the number of volumes (tapes) to read.
/t		R/W	Translate data from EBCDIC to ASCII on tape read. Translate data from ASCII to EBCDIC on tape write.
/v		R/W	Verbose.
/W		R	Suppresses the 'wait for user approval to exit keystroke' (normally the program waits for a user keystroke before exiting, to allow you to see what has been done; this switch tells it not to wait).
/w		W	Write new tape.

Exit Codes

When **DEPOT2** finishes and returns to DOS, it returns a status byte which a batch file can examine to determine if an error has occurred. A value of zero indicates no error. Each bit indicates a type of error. More than one bit may be on at a time. If an error is detected and a retry is successful, then no error codes bit will be set. This table shows the exit codes and their meanings:

Decimal	Hex	Error Indication
128	80	Tape write
64	40	Tape read
32	20	Tape position (skipped too many files or blocks, etc.)
16	10	Disk write (path not found, write to "read only", etc.)
8	8	Disk read (file or path not found, etc.)
4	4	Disk other than read or write
2	2	No command switches, no controller, lacks memory
1	1	Miscellaneous (block too big, file not closed, etc.)
0	0	No Error

Multiple Volume Tape Writes

If your write operation requires more than one tape to successfully write all the data you have specified, **DEPOT2** will write to the end of the tape, terminate the current write, write two filemarks, and prompt you for a new tape to be mounted. Once the new tape is mounted, **DEPOT2** will continue the incomplete write operation.

Error Handling

If an error occurs during reading or writing of a tape, **DEPOT2** automatically retries the operation 10 times. If the failure persists, you will be warned that an error has occurred. The message will tell you what happened and what you may be able to do about it, if it's not too obvious. You choose how to proceed, but sometimes there is only one choice. If you choose not to continue, **DEPOT2** will stop running the present command line. If the /i (ignore) switch is used, **DEPOT2** will continue with the next line of the command file (if any). The /i switch causes most errors to be ignored, but you will always be warned of any serious error. Some errors, such as disk failures, will prevent **DEPOT2** from continuing. There is no guarantee that a retry will be successful. You can never do additional retries on errors that occur while writing to tape. If you see the write error message, the automatic retries have already happened.

This table shows the possible choices when an error has occurred:

Key	Message	Meaning & Effect
S	Stop Trying	Discontinue present operation. Usually returns to the main menu.
T	Try Again	Another attempt is made on the failing operation.
C	Continue Beyond Error	DEPOT2 attempts to ignore the error. No additional retries are attempted on this error.

NOTE: Some selections may not be available if the error condition warrants it.

If the physical end of file marker is detected while reading or positioning tape, you will be asked whether to continue the operation. This warning will normally appear only once (in command file operation, the warning will appear for all command lines which do not use the /i switch).

If the physical end of tape marker is detected while writing or appending to a tape, **DEPOT2** will finish writing all data in the cache, write two filemarks, and unload the tape. You will be warned that EOT was reached and asked to load a new tape to continue writing the unfinished file. If you select verbose (/v), you will be told the name of the file which spans two tapes.

Command Line Examples

Example 1:

This command sets the verbose option and writes the contents of disk file, input.txt, to tape, translating from ASCII to EBCDIC and using a 1024 byte block size. **DEPOT2** will ignore any errors (except hard errors).

C:\> depot2 /vwti /n input.txt /s 1024

Example 2:

To read this file, we instruct **DEPOT2** to read the first file (stop reading at first filemark), translate it back to ASCII. **DEPOT2** writes the data to file out.txt.

C:\> depot2 /tM /n out.txt

You do not have to specify block size (it reads a block at a time from tape), or an explicit read operation (the absence of /w and /a implies read).

Example 3:

The following command appends a file to tape, writing 80 byte records with a blocking factor of 100 (100 records per block). It writes the contents of disk file cardfile.dat:

C:\> depot2 /av /r80 /b100 /n cardfile.dat

Example 4:

To read and write binary or executable files, do not specify translation. For example, the following command copies disk file output.bin to tape, using a 2048-byte block size:

C:\> depot2 /wv /n output.bin /s 2048

Similarly, to read a file from tape (again, without translation), use a command such as:

C:\> depot2 /vM /n input.bin

This creates disk file input.bin. Again, we use option /M to stop reading at a filemark.

Example 5:

This command line enables you to read tapes that contain several files without entering the output filename for each file.

C:\> depot2 /vM /g [DRIVE:] [PATH] [NAME (no extension)]

Each file is transferred to disk with a name automatically generated as an incrementing extension up to 999.

Example Command File

We include a sample command file, CMD, on the distribution disk.

IMPORTANT NOTE: **This example shows how you can set DEPOT2 to perform multiple and/or repetitive tasks. Note that each DEPOT2 task requires a separate line in the command file you create, although the operations that make up a task can (and should) be kept together on one line.**

Here are the contents of the sample file:

```
; This is a sample command file that illustrates
; how multiple tape operations may be specified
; for repetitive tasks. Please note that comment
; lines MUST begin with a semi-colon
; in the FIRST column of text.
; Write read.me file with fixed length records,
; 50 per block. The "w" switch indicates writing
; from the beginning of the tape.
/wv /r78 /b50 /n read.me
; Write it again as a straight image.
; The remaining write operations are specified
```

```
; with an "a" switch set to append these files
; to the existing data on tape.
/av /s 1000 /n read.me
; Write it this time with short records,
; truncating lines.
/av /r40 /b50 /n read.me
; Write it this time with long records, and
; larger block factor.
/av /r120 /b100 /n read.me
; Write it again with short records, and smaller
; block factor.
/av /r40 /b30 /n read.me
; Write one last time in EBCDIC, with 80 byte
; records and short blocks
/avt /s10 /n read.me
; Read the first file, deblock it and write it
; back to disk.
; Stop when a filemark is reached.
/vM /r78 /b50 /n test
; Read and reproduce the straight image.
/vM /n test1 /m 1
; Read file written with truncated lines, but
; only the first thirty lines.
/v /r 40 /b 30 /m 4 /n test2 /R 30
; Read the first file, deblock it and write it
; back to disk.
; This time stop after reading only one data
; block.
/v /r78 /b50 /n test3 /B 1
; Read the EBCDIC version. Note that if you use
; the DOS "TYPE" command to examine this file, it
; will appear to be double-spaced, due to
; the auto-wrap function of the display. Viewing
; the file with a text editor will eliminate
; this illusion.
/vtM /s10 /m5 /n test4
```

DOS Batch File Examples

Example 1:

This batch file will show a warning if there is ANY error:

```
echo off
depot2 /c cmdlist
if errorlevel 1 echo ** DEPOT2 RETURNED AN ERROR **
rem If DEPOT2 returns a status byte equal or ...
rem greater than 1 the "echo" message will be ...
rem printed on the screen
```

Example 2:

This batch file will warn only if there is a tape error:

```
echo off
depot2 /c cmdlist
if errorlevel 32 echo ** DEPOT2 RETURNED A TAPE ERROR **
rem If DEPOT2 returns a status byte equal or greater than ... rem 32 the "echo" message
will be printed on the screen
```

Limitations

- **DEPOT2** is limited to a 65,280 byte block size.
- **DEPOT2** accepts block sizes less than 1, but greater than zero.
- During retries on a damaged tape, **DEPOT2** may lose track of the tape position.
- The /r and /b switches should normally only be used with ASCII text data files, as the automatic padding will make executable files useless.
- When using DEPOT2 with text files which include nulls, using switches /a or /w with /r will cause problems.

  If you use DEPOT2 to write (option /w) to tape, or to append (/a) to an existing tape file, and you use the /r option for a text file that includes nulls (the "Ø" character) you will experience problems.

  The difficulty arises from the use of nulls, causing the write procedure to incorrectly format the data. The reason is that the standard definition of ASCII text files disallows nulls.

  To avoid this problem:
  Avoid putting nulls into text files.
  If your file includes nulls; instead of using /r, use the /s option (with either /a or /w). DEPOT2 will correctly write (and read) these files in such cases.

FDUMP

Use **FDUMP** to examine the contents of a tape. **FDUMP** makes no assumptions about the format or block size of the tape's data. **FDUMP** is especially suited to:

- Determine the format of a tape
- Quickly sample a tape's data visually
- Troubleshoot a tape problem
- Debug a custom tape utility

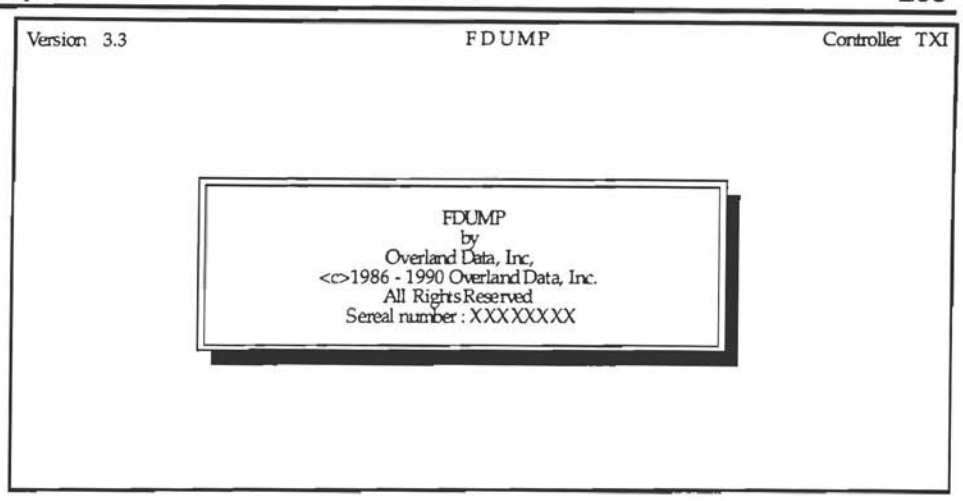
Features

FDUMP is a screen-oriented, interactive tape reading utility. **FDUMP** allows you to select various functions, including reading the next block, positioning to a specific file, reading a specific block within the file, displaying a block of data in hexadecimal format (and EBCDIC or ASCII), scrolling or paging through the block, and jumping to a specific byte in a block. **FDUMP** continuously updates the file number, block number, tape drive status word, block size, and whether the data is displayed in ASCII or EBCDIC. Finally, **FDUMP** displays the meaning of the status word by highlighting the description of each bit that is *on* or enabled.

Use

Choose a tape with data on it and load the tape onto the drive. **FDUMP** does not write to tape; it only reads data from the tape.

Boot DOS and type: C:> **fdump [ENTER]**



(Your **FDUMP** version number, date, or Controller model may differ.)

At this point the first block is displayed. You may instruct **FDUMP** to read the next block by pressing function key **[F3]**. To position the tape head at another file, press **[F5]** or to read another block within the current file, press **[F6]**. After each function, **FDUMP** updates the status information indicating the file number, block number, tape drive status word, block size, and data mode (ASCII or EBCDIC). Press function key 1 to view a description of the status word.

Once **FDUMP** reads and displays a data block, you may scroll (press **[↑]**, or **[↓]**) or page (press **[PGUP]** or **[PGDN]**) through the screen display, or display a specific byte address at the first line of the display (press **[F2]**). **FDUMP** displays the data in hexadecimal notation and either ASCII or EBCDIC. Press the **[F4]** function key to toggle between ASCII and EBCDIC display modes. Press the **[F3]** function key to read and view successive blocks.

Exit **FDUMP** at any time by typing **[ESC]**.

FDUMP's function keys are summarized in the following table:

FDUMP Function Keys

Key	*Command Description*
[F1]	Display Status Help
[F2]	Jump to Byte
[F3]	Read Next Block
[F4]	ASCII—EBCDIC Toggle
[F5]	Jump to File Number
[F6]	Jump to Block Number
[↑]	Scroll Display Up
[↓]	Scroll Display Down
[PGUP]	Page Display Up
[PGDN]	Page Display Down
[ESC]	Exit to DOS

1*:* Status ***Help.***

FDUMP continuously updates the drive's status word in hexadecimal notation. Pressing **[F1]** displays a description of the drive's status word. **FDUMP** highlights the description of each bit that is enabled, allowing you to easily interpret its value. For example, if the status word is 5015 (hex) then bits 0, 2, 4, 12 and 14 are enabled. **FDUMP** highlights the descriptions of IFBY (bit 0), IRDY (bit 2), IONL (bit 4), DBYT (bit 12), and IDENT (bit 14).

The tape drive's status word (16 bits) has the following meaning.

Bit 0 IFBY

The drive is busy executing or completing a command. This bit is enabled during and just after a command.

Bit 1 IDBY

The drive is in DMA phase of a command. This bit is enabled during direct memory access to and from the tape drive.

Bit 2 IRDY

This bit is true unless the drive is not selected, not loaded, offline, or rewinding.

Bit 3 IFPT

The tape is write-protected (the reel lacks a write-enable ring).

Bit 4 IONL

The drive is online.

Bit 5 ISPD

The last command was high speed (for dual-speed drives only).

Bit 6 IBOT

The tape head is at the Beginning of Tape (BOT) marker.

Bit 7 IEOT

The tape head is at the End of Tape (EOT) marker. There is a few feet of usable tape beyond EOT.

Bit 8 IHER

A hard error (uncorrected) occurred on read or write.

Bit 9 ICER

A hardware correctable error occurred on read.

Bit 10 IFMK

The drive detected a filemark on the last command.

Bit 11 IRWD

The drive is rewinding.

Bit 12 DBYT

The drive dropped the data busy signal indicating that a command was completed. This bit is always zero on TX*i*-16 controllers.

Bit 13 NRZI

The drive returned NRZI (Non-Returned to Zero Inverted) status. NRZI is an encoding mechanism common in 800 bpi (bytes per inch) drives. Bit 13 may also indicate the alternate density in a dual-density tape drive (for example, 6250 bpi or 3200 bpi on a 1600 bpi dual-density tape drive).

Bit 14 IDENT

Either the drive detected an ID (identification) burst for PE (Phase Encoded) tape or CCG (Check Character Gate) status for NRZI tape. PE is an encoding mechanism common in 1600 bpi drives. The ID burst occurs only after a read at the Beginning of Tape. CCG status occurs with every data block read on NRZI blocks.

Bit 15 PAR

The controller detected a parity error. This indicates bad cables, controller, or tape drive. This bit is always zero on TX*i*-16 controllers.

2*: Jump to Byte.*

Press **[F2]** to display the data from a specific block address at the top line of the screen. prompts for the address in hexadecimal notation as follows:

Enter byte position in HEX: 0b43 <Enter>

Here we specify address 0b43. **FDUMP** now displays the data beginning at address 0b40 at the top line of the screen. If you specify an address larger than the block size, FDUMP simply displays the last portion of the data block.

3*: Next Block.*

Press **[F3]** to read the next block of data. If the tape is at BOT, **FDUMP** reads the first data block. Pressing **[F3]** multiple times will read successive data blocks. After a read, **FDUMP** reports the current file number, block number, tape drive status word, block size, and display mode. **FDUMP** can read any block up to 64kB.

When **FDUMP** detects a filemark, it responds with this message:

>>> FILEMARK DETECTED <<<

hit **[F3]** to read a block

Now press **[F3]** to read the first block of the next file.

4*:* ASCII-EBCDIC.

Press **[F4]** to toggle the display mode between ASCII and EBCDIC. You can press **[F4]** any time during the **FDUMP** session, so you do not need to know ahead of time the character code mapping of your tape.

5: Jump to File.

This function allows you to position the tape head at the beginning of an arbitrary tape file. **FDUMP** defines a file to include all data blocks between filemarks. The first file is file 0. To read an arbitrary file, press **[F5]**. FDUMP prompts:

Enter file position in decimal: 3 <Enter>

Here we specify the fourth file (file 3). **FDUMP** will position the tape at the beginning of this file and display this message:

>>> FILEMARK DETECTED <<<

hit **[F3]** to read a block

Press **[F3]** to read the first data block. If you specify the first file (file 0), FDUMP responds with:

>>> BEGINNING OF TAPE <<<

hit **[F3]** to read a block

If you specify a file number larger than the number of files on tape, **FDUMP** will attempt to find it and respond with:

>>> FILEMARK DETECTED <<<

>>> LOGICAL EOT DETECTED <<<

> PROCEED FORWARD WITH CAUTION <

indicating that the specified file does not exist. This occurs when FDUMP reads two consecutive filemarks.

Use this to move both forwards and backwards to particular tape files.

6: Jump to Block.

The Jump to Block function allows you to read any block within the current file (or position the tape head at the filemark separating it from the next file). **FDUMP** defines a block to include all data separated by IRGs. The blocks may be of any size (up to 64kB), and **FDUMP** will report the size of the most recently read block on the display. **FDUMP** counts blocks beginning at 1. For example, to read the third block of the current file, press **[F6]** and responds with the following prompt:

Enter file position in decimal: 3 **[ENTER]**

Here we respond with 3 for the third block. (Remember, blocks count from 1.) **FDUMP** now displays the data from block 3.

If you specify a block number larger than the number of blocks in the current file, **FDUMP** detects a filemark, positions the tape at the first block of the next file, and displays the following message:

>>> FILEMARK DETECTED <<<

hit **[F3]** to read a block

Press **[SHIFT]+[F3]** to read and display the first block of this (that is, the next) file.

If you specify a block number larger than the number of blocks in the current file and there are no more files on the tape, **FDUMP** detects the filemark as before. However, when you attempt to read the first block of this next file, **FDUMP** detects another filemark indicating the Logical End of Tape (EOT), and responds with:

>>> FILEMARK DETECTED <<<

>>> LOGICAL EOT DETECTED <<<

> PROCEED FORWARD WITH CAUTION <

Scroll Up, Scroll Down

Use [↑], and [↓]. to scroll the display a line at a time.

Page Up, Page Down

Use **[PGUP]** and **[PGDN]** to page the display up or down.

Examples

To illustrate **FDUMP**'s use, we describe the following examples:

- Reading a tape containing an unknown format, character mapping, or density
- Checking the integrity of a tape (quick visual inspection)
- Troubleshooting a tape problem
- Debugging a custom tape utility program

Example 1:

Reading a tape containing unknown data.

Remove the write enable ring from the back of the tape reel and mount the tape on the drive. Boot DOS and call **FDUMP**:

C:>fdump [ENTER]

FDUMP immediately reads the first data block. Note the **FDUMP** status line and the data on the screen. Answering the following questions may help you determine the format and character mapping of the tape's data.

1. Did **FDUMP** read good data?

   Inspect the data, scrolling or paging as needed. Toggle the ASCII EBCDIC switch (**[F4]**). Press **[F1]** to examine the tape drive's status word.

   If Bit 8 (IHER) is true, you may have a bad spot on the tape or the tape may be written in a density incompatible with your tape drive (in this case, FDUMP can't read it at all). Use 6 to select another block. If this block also returns abnormal status, then the density is probably incompatible.

   If Bit 9 (ICER) is true, the data is probably good (as long as status Bit 8 is false).

   If Bit 15 (parity error) is set, the data is probably bad even if **FDUMP** returned data.

2. What's the format of my tape?

   Although programs write tapes in the general format of multiple files separated by filemarks and blocks separated by tape gaps, the data in the blocks may have meaning about how the rest of the data is formatted. For example, the first file may be a directory file with names and file locations of the following files. Or the entire tape may contain only 1 file, with directory information stored in the initial block or blocks.

   The files themselves may have intrinsic formats. Card images may contain null-padded or blank-padded 80 byte records.

   Use the scrolling and paging facilities and the arbitrary block and file reads to examine the tape's data. After determining format, you can use **DEPOT2** to read and translate it.

Example 2: Checking the integrity of a tape.

Using **FDUMP** is a quick way to check the contents of a tape. Again, remove the write enable ring, mount the tape in the drive, boot DOS, and call program **FDUMP**. **FDUMP** immediately reads the first data block. If the data is in ASCII, you do not need to toggle the ASCII-EBCDIC switch. Page through the block, visually inspecting the data for correctness. Read subsequent blocks and files as needed.

This method is not meant to verify data correctness. It is simply a quick method for checking overall contents.

Example 3: Troubleshooting a tape problem.

If you use another utility to read a tape, but the data is not correct or the tape does not seem to be readable, **FDUMP** may help locate the problem. Load the tape and call **FDUMP** as before. **FDUMP** immediately reads the first data block. If the tape is readable, but the data is not correct, press **[F1]** to examine the status word. If Bit 15 (parity error) is true, check cables between the controller and the drive. Bit or bits in error (in your data) may contain clues about which cable lines or channels are bad. The least significant bit corresponds to tape channel 7.

Mount another tape (remember to remove the write enable ring) to determine if the tape itself is faulty. If other tapes also have problems, run the **XTEST** program (using a scratch tape) to gather further diagnostic information.

Example 4: Debugging a custom tape utility program.

You can use **FDUMP** to help debug your own tape utility program. For example, when you're ready to test your program, mount your test tape and write your testing information to tape. Now call **FDUMP** to examine it (block by block if desired). **FDUMP** will tell you the length of each block, how many blocks in a file, and the positions of the filemarks.

Use **[F3]** to read successive blocks. Note their length and the block number from the beginning of file. Also note the file number. As **FDUMP** reads each block, you may page through its contents as necessary. At the end of the file, **FDUMP** clearly indicates that it read a filemark, and the file number and block number reflect this status. Continue through the tape, examining its contents until you're convinced your tape utility is operating correctly.

TAPEUTL

This section contains complete instructions for running all the programs included in **TAPEUTL**. Each program is covered separately. The manual follows the main menu order (unlabeled tape, labeled tape, etc.) with each submenu choice covered in sequence within each section (disk-to-tape, tape-to-disk, etc.).

The DOS filter function allows you to translate the data stream during transfers between disk and tape according to statements in a user-created translation file.

Automatic-mode execution of **TAPEUTL** under control of a command file allows you to capture a **TAPEUTL** session and 'replay' the same sequence of commands later.

TAPEUTL

The main program in the Tape Utility package is TAPEUTL.EXE. To run this program, simply type its name at the DOS command prompt:

C:> tapeutl [ENTER]

When the program is loaded the first of a number of help screens will be displayed. At this point you can either start the program proper by hitting any alphanumeric key or you can page through the help screens by using the **[PGUP]** and **[PGDN]** keys. You may want to do this the first time you run **TAPEUTL** to familiarize yourself with the help facility.

The first thing **TAPEUTL** does before displaying the first screen is find the amount of system memory available to determine if there is enough space for its own buffers. If the available memory is insufficient **TAPEUTL** will display a message saying so and stop. If this happens you will at the *minimum* need to reconfigure the system software on your machine. If your system needs more memory you may have to install more physical memory to run **TAPEUTL**. Check the following:

- Do you have memory-resident programs installed? Such programs reduce the amount of memory available to other programs: reboot without them and retry **TAPEUTL**.

- Some print spoolers and RAM-disks such as VDISK also use system memory: remove the device's statement from your system's CONFIG.SYS file, reboot, then retry TAPEUTL. (**Note:** if the device uses extended memory on an auxiliary plug-in board then it will not interfere with **TAPEUTL**.)

When you start **TAPEUTL** the main menu screen is displayed with a list of submenus you can choose from (Figure 1). Note that all **TAPEUTL** screens (except the tape block display screen) follow the general format: current submenu stays on the top half of the screen while particular options for the operation currently being done appear on the bottom half. The tape display screen is described below.

To perform a tape function choose the submenu from the main menu (type its letter) then choose the desired function from the submenu. Many functions (e.g., file transfer to/from tape) will present fields on the lower hall of the screen (e.g., filenames and record lengths) which you fill in; when all needed fields have been filled in press **[ENTER]** to start the function. To move from field to field use **[TAB]** to move forward and **[SHIFT]+[TAB]** to move backward. To cancel a function, **[ESC]** will always put you back at the menu you came from without performing the function (think of **[ESC]** as a user's panic button). If you are at the MAIN menu **[ESC]** will end the program and return you to DOS.

```
Flagstaff Engineering Tape Utility Program
A) Process unlabeled tapes
B) Process labeled tapes                    >>>> Main <<<<
C) Reposition tape or display records         Unlabeled Tape
D) Change tape system parameters              Labeled Tape
E) Initialize DOS filter                      Position/Display
                                              Change Parameters
                                                   Command:
                                              Initialize Filter

----------<F10 = Help ESC = Exit Routine>----------
```

Figure 1. Main Menu

Processing Unlabeled Tapes

When the unlabeled tape option is chosen from the main menu the following submenu is displayed:

```
Flagstaff Engineering Tape Utility Program

A) Copy from disk to tape
B) Copy from tape to disk                                    Main
C) Reposition tape or display records    >>>>Unlabeled Tape <<<-
D) Copy multiple files from tape to disk          Labeled tape
E) Summarize Tape Data                          Position/Display
                                                Change Parameters
                                                        Command:
                                                Initialize Filter

----------<F10 = Help ESC = Exit Routine>----------
```

Figure 2. Unlabeled Tape Submenu

Although this option is primarily intended for handling unlabeled tapes it can be used to transfer data from labeled tapes if the user uses caution. This option should not be used to transfer data to labeled tape as no labels are written by the transfer routines.

The exception to this rule is if certain system specific labels need to be written on a tape; in which case the labels are written as separate files preceding and/or following the actual data file. In such a case we recommend that the user consult a system programmer familiar with the target system label format.

Copy from Disk to Tape

Use this option to copy data from a DOS (disk) file to unlabeled tape at the current tape position. Unlike the labeled tape disk-to-tape transfer this operation is always carried out at the current tape position. The tape must be correctly positioned before this operation is started; use the Reposition tape or display records option on this submenu if the tape is not where you want it.

Initially the following three or four fields will appear at the bottom half of the screen:

Input file name: this is the fully-qualified name of the DOS disk file. A wildcard DOS file name may be given for unlabeled tapes. If just a normal wildcard file name is given each input file will create a separate output file on tape. *(On labeled tapes each tape file name will be the same as input DOS file name.)* If a "+" is placed immediately after the wildcard name then all wildcard files will be concatenated into one file on the tape.

Translate data? Three choices are possible:

N - no translation

A - data will be translated from EBCDIC to ASCII

E - data will be translated from ASCII to EBCDIC.

Data type: this specifies how the incoming data is to be treated. Since data on tape is always written in blocks the entry in this field will control how the disk file data is broken up into these tape blocks. There are three possibilities:

1. **Fixed-length records:** data is read from the disk file as records of a specified length (given by the user). The disk data is not scanned for line-end characters or other delimiters but is assumed to be divided into logical records of the user-specified length.

2. **Variable-length records:** the disk file data is assumed to be in logical records of varying length; each record terminated by an identical character sequence. With this type of transfer data is read from disk as variable-length but written to tape as fixed-length records (short records are padded).

3. String data**:** this option is used in the case where the disk data file is not divided into logical records at all (e.g., an executable program file that consists of binary data). The disk data is treated as a continuous string of data. The tape file is divided into blocks but the blocks are simply fixed-length chunks of the input file with a possible short last block.

Filter **on (N/Y):** this field will only appear if the translation filter has been initialized from the main menu. If Y(es) is specified, data from the input file is translated according to statements in the filter file before being written to tape. The default (predicted) value here is N(o). See the section on the DOS filter function for more information.

Depending on the data type chosen for the transfer additional fields will appear Complete screens for each data type are shown next with a description of the remaining fields.

Fixed-Length Data Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------

<Disk0Tape>

Input file name \customer\sales.dta
Translate data? (N=no A=EBCDIC to ASCII E=ASCII to EBCDIC) N
Data type: (F)ixed (V)ariable (S)tring f
Record size 1024
Records per output tape block 20
Records to skip O
Records to transfer ALL

File size 189952 bytes
There is a partial record of 512 bytes: transfer it? y/n y
```

Figure 3. Disk to unlabeled tape-fixed-length transfer fields.

Four additional fields appear for fixed-length transfer operations:

1. Record size: this specifies the length of each logical record in the disk file. Remember that TAPEUTL doesn't do any kind of checking on the input records but assumes that the input records are in fact this length. If you're not sure what length the file records are look at the file first with a file dump/display utility (e.g., the DFILE program) to determine the proper record size.

2. **Records per output tape block:** this is the blocking factors to be used on tape. This number multiplied by the record length gives the tape block size in bytes which can be a maximum of 65,536 bytes. If this size is exceeded the program will display a warning message and you will be asked to change either the record size or the number of records per block.

3. Records **to skip:** this field pre-filled with zero specifies an optional number of records from the input file to be skipped before the data transfer to tape starts.

4. **Records to transfer:** this field, pre-filled with ALL, specifies an limit on the number of records to be transferred to tape. If only a certain number of records are to be copied, enter the number in this field.

When a fixed-length transfer is started TAPEUTL gets the size of the input file from the disk directory displays it and compares this size to the record size to determine whether there will be a short record at the end of the file if there is a message will appear at the bottom of the screen asking if you want to transfer this partial record. Answering Y will cause this last record to be padded with binary zeros (0) at the end of the file; answering N will simply drop this last record from the tape file.

There will usually be a short block at the end of a fixed-length blocked record file (when writing out fixed length records and if file size is not a multiple of the record size the operator is still asked if he/she wants to truncate the partial record).

After the transfer is complete the number of tape blocks, number of logical records and number of bytes in the tape file will be displayed. A prompt will appear at the bottom of the screen asking if you want to append another file to this one; see *Appending Files on Tape,* below, for an explanation of this choice. To transfer another file simply hit **[ENTER]** at this point. This will close the tape file by writing filemarks after the last block and prepare you for writing another new file after this one. Press **[ESC]** to return to the unlabeled tape submenu.

Variable-Length Data Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------

<DiskTape>

Input file name \text\letter1.doc
Translate data? (N=no A=EBCDIC to ASCII E=ASCII to EBCDIC) N
Data type: (F)ixed (V)ariable (S)tring v
Record size 60
Records per output tape block 1000
Records to skip O
Records to transfer ALL
Line delimiter 0DOA
Pad character 20

File size 48182 bytes
Data transfer in progress...
```

Figure 4. Disk to unlabeled tape-variable-length transfer fields.

Six additional fields appear when a variable-length transfer is chosen:

1. Record size: this is the size of the tape (output) record which is the size of the largest record in the file not including line delimiter characters. Remember that variable-length only applies to the disk (input) file; the tape file is written as fixed-length records. If a record is shorter than the given record size it is right-filled with pad characters (see below). if a record is bigger than this size it will be split—the excess part will be written as a new record—and a warning message will be displayed after the transfer:

WARNING! nn records exceeded specified length

If this happens try the transfer again with a larger record size. No data will be lost in any case but the file may need to be reformatted to restore lines to their original form.

2. **Records per output tape block:** this is the blocking factor to be used on tape. This value multiplied by the record length gives the tape block size in bytes which can be a maximum of 65,536 bytes. If this size is exceeded the program will display a warning message and you will be asked to change either the record size or the number of records per block.

3. **Records to skip:** this field pre-filled with zero specifies an optional number of records from the input file to be skipped before the data transfer to tape starts.

4. **Records to transfer:** this field pre-filled with ALL specifies an optional limit on the number of records to be transferred to tape. If only a certain number of records are to be copied enter the number in this field.

5. **Line delimiter:** this specifies which character or string of characters are to be taken as an end-of-line marker in the input file. This entry must correspond to the actual line delimiters in the file for the transfer to be successful. The characters (up to 10) are specified as 2-digit hexadecimal values with the most common values pre-filled: 0D 0Aor ASCII carriage-return and line-feed characters.

CAUTION! See the section on the DOS filter function for an explanation of pitfalls to avoid if you are using a translation filter.

6. **Pad character:** the character which will be used to fill short records is specified here as a 2-digit hexadecimal value. This field is pre-filled with a commonly-used fill character the ASCII space (hex 20). The EBCDIC space character is hex 40.

The variable-length data transfer works like this: as the input file is read from disk TAPEUTL continually scans the data for the specified line delimiter characters. When the delimiter string is found the record is filled out to the specified record length with pad characters if necessary, then written to tape. The line delimiters themselves are not transferred but they can be restored by the corresponding tape-to-disk transfer routines. If a record exceeds the specified length it is split as explained above in the Record size field description and a count of all such records is kept and displayed after the transfer operation.

After the transfer is complete the number of tape blocks number of logical records and number of bytes in the tape file will be displayed. A prompt will appear at the bottom of the screen asking if you want to append another file to this one; see *Appending Files on Tape* below for an explanation of this choice. To transfer another file simply hit **[ENTER]** at this point; this will close the tape file just written by writing filemarks after the last block and prepare you for writing another new file after this one. Press **[ESC]** to return to the unlabeled tape submenu.

String Data Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------

<Disk0Tape>

Input file name \binary\prog1.exe
Translate data? (N=no A=EBCDIC to ASCII E=ASCII to EBCDIC) N
Data type: (F)ixed (V)ariable (S)tring S
Output tape block size 60000
Characters to skip 0
Characters to transfer ALL
Terminate on EOF (X'1A') character N

4 blocks 189952 bytes copied
                Do vou wish to append another DOS file? (n/y) n
```

Figure 5. Disk to unlabeled tape-string transfer fields.

When a string-data transfer is chosen four additional fields appear on the lower half of the screen:

1. **Output tape block size:** this specifies the actual tape block size to be written up to 65 536 bytes.
2. **Characters to skip:** this field pre-filled with zero specifies an optional number of characters to be skipped before the data transfer to tape starts.
3. **Characters to transfer:** this field pre-filled with ALL specifies an optional limit on how many characters to transfer to tape. If the limit is needed enter the number of characters in this field.
4. **Terminate on** EOF **(X'1A') character?** If the data to be transferred is in a normal ASCII text file this option can be chosen (by entering Y) to end the transfer on the conventional ASCII end-of-file character (hexadecimal 1 A decimal 26 also known as Control-Z). This option should not be chosen (by entering N) if so-called binary data is to be transferred (i.e. executable programs binary image data etc.) because such data is likely to contain many hex 1 A characters which would falsely be interpreted as end-of-file.

Copy from Tape to Disk

Use this option to copy a file from tape at the current tape position to a disk file. Unlike the labeled tape tape-to-disk transfer this operation is always carried out at the current tape position. The tape must be correctly positioned before this operation is started; use the Reposition tape or display records option on this submenu if the tape is not where you want it.

Initially the following three or four fields will appear at the bottom half of the screen:

Output DOS file name: this is the fully-qualified name of the DOS disk file which will be written which must include any necessary drive letters and pathnames to identify the file (for example B:\text\letter1.doc).

Translate data? Three choices are possible:

N - no translation

A - data will be translated from EBCDIC to ASCII

E - data will be translated from ASCII to EBCDIC.

Data type: this specifies how the incoming data from tape will be treated when transferring it to disk. There are four choices:

1. **Fixed-length** records**:** the tape file is assumed to be divided into equal-length records of a size given by the user. These logical records are simply copied from the tape blocks to the disk file as they are read. No line delimiters are read or written.

2. **Variable-length records:** this really means variable-length physical blocks on tape. Each block in the tape file is treated as a separate logical record and these blocks can vary in size. In transferring these blocks to disk file records line-end delimiters (e.g. carriage-return/line feed pairs) can be added to the end of each record to produce a valid ASCII-type file—that is, a file that can be successfully displayed using the TYPE command or printed on a line printer.

3. IBM **Variable-length records:** this is a special type of variable-length record where each logical record has a 4-byte prefix which contains a binary count of characters in the record. This count is contained in the first word (2 bytes) of the prefix and the second word of the prefix is usually binary zero. This option is generally only used when transferring files from tapes written on IBM mainframe computer systems.

4. String data: this option is used in the case where the tape file is not divided into logical records at all (for example an executable program file which consists of binary data). The tape data is treated as a continuous string of data. The tape file is divided into blocks but the blocks are simply fixed length chunks of the input file with a possible short last block. All tape data is transferred to disk contiguously (no breaks).

Filter **on (N/Y):** this field will only appear if the translation filter has been initialized from the main menu. If Y(es) is specified data from the tape is translated according to statements in the filter file before being written to the disk file. The default (pre-filled) value here is N(o). See the section on the DOS filter function for more information. Regardless of the type of data transfer chosen TAPEUTL will search for the given output disk file to see if one already exists. If it does the following message will appear:

DOS file already exists -do you want to erase old file (Y/N)?

If you answer Y at this point the existing file will be replaced and all data currently in the file will be lost. If you answer N the transfer will not start until you hit **[ENTER]** again giving you an opportunity to change the output file name.

If the target volume (fixed disk or diskette) that the file is being copied to becomes full during the transfer operation the transfer will be suspended and the following message will appear at the bottom of the screen (**Note:** Output breaks at a record boundary when the output DOS volume becomes full):

DOS volume full-new file name (ESC to abort)

If you want to continue the transfer you will need to enter the name of the file which will contain the next part of the tape file including any needed drive letter and pathname specifiers. The transfer will resume with the next (1 byte (character) of the tape file on the new disk file. To end the transfer hit **[ESC]**; the data on the full disk volume will be retained and you will return to the unlabeled tape submenu.

Depending on the data type chosen for the transfer additional fields will appear below these fields. The complete screens for each data type are shown next with a description of the remaining fields.

1. **Pad character:** used in conjunction with the preceding option this specifies the pad character to be (optionally) stripped from the output record. This entry is specified as a 2-digit hexadecimal value corresponding to a single character and is pre-filled with the most common pad character the ASCII space (hex 20).

2. **Insert line end?** this field and the next set options that allow a tape file of fixed-length records to be written to disk as a file of variable-length records with line-end delimiters. If Y is chosen here then each logical record from tape will have line-end delimiters appended in the output record. By appending delimiters the file will be essentially a normal ASCII text-type file (assuming that the data is ASCII text).

3. **Line end characters** (In hex): in conjunction with the preceding field this specifies the line-end characters to be appended to each output record it Insert line end is chosen. These characters (up to 10) are specified as 2-digit hexadecimal values. The field is pre-filled with the most common line delimiters (carriage-return, line-feed characters (hex OD-OA decimal 13-10)).

4. Records **to skip:** this field pre-filled with zero specifies an optional number of records from the input file to be skipped before the data transfer to tape starts.

5. **Records to transfer**: this field pre-filled with ALL specifies an optional limit on the number of records to be transferred to tape. If only a certain number of records are to be copied enter the number in this field.

After the transfer is completed a message will appear stating the number of records transferred. Alter clearing this message you can start another tape-to-disk transfer by filling in the appropriate fields and hitting **[ENTER]**; the tape will be positioned at the start of the next file. Hit **[ESC]** to exit to the unlabeled tape submenu in order to reposition the tape etc.

Variable-Length Data Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------
Maximum block 16384

                                                <Tape0Disk>

output DOS file \temp\junk1.out
Translate data? (N=no A EBCDIC to ASCII E=ASCII to EBCDIC) N
Datatype: (F)ixed (V)ariable (I)BM variable (S)tring v

Record length 1000
Strip trailing pad characters (Y/N) N  Pad character 20
Insert line end (Y/N) y
Line end characters (in hex) 0D 0A
Records to skip 0
Records to transfer ALL

Copy complete - 60 records, 57313 bytes - press any key to continue.
```

Figure 7. Unlabeled tape to disk-variable-length transfer fields.

When transferring data from tape to disk, "variable-length record" has a different meaning than the usual sense of variable-length logical records in a disk file. Remember that all data on tape is written in blocks. All the blocks in a tape file do not have to be the same size. To the tape, these blocks look like records since they must be treated as contiguous chunks (the drive cannot stop reading or writing in the middle of a block). Hence variable-length records for tape really means variable-length blocks.

All of this is to caution that it is easy to become confused about tape blocks and records. The variable-length unlabeled tape-to-disk transfer option is provided for unusual data transfer needs where each tape record (block) needs to be written to disk as a separate logical record with the options of stripping trailing pad characters and adding line-end delimiters to each output record.

If you are transferring a tape file to disk that was originally written by TAPEUTL (or some other source) from a file of variable-length records you probably need to use the fixed-length transfer option above. This can reconstruct the file in its original form including proper record length and line-end delimiters (assuming that no records were split during the disk-to-tape transfer).

IBM Variable-Length Record Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------
Maximum block 16384

                                                <Tape0Disk>

Output DOS file \temp\junk2.out
Translate data? (N=no A=EBCDIC to ASCII E-ASCII to EBCDIC) N
Data type: (F)ixed (V)ariable (I)BM variable (S)tring i

Record length 50
Strip trailing pad characters (Y/N) N Pad character 20
Insert line end (Y/N) y
Line end characters (in hex) 0D 0A
Records to ship 0
Records to transfer ALL

Copy complete - 99 records 3997 bytes - press any key to continue:
```

Figure 8. Unlabeled tape to disk-IBM variable-length transfer fields.

Files containing IBM variable-length records consist of blocks which are prefixed by a 4-byte *block length* header. This header contains the total number of bytes in the block *including the header.* Each record is prefixed by a 4-byte record length header containing the number of bytes in the record *including the header.* The file and header layouts are shown below.

IBMVARIABLE-LENGTH RECORD LAYOUTS

block length header	record 1 header	record 1 data	record 2 header	record 2 data

Figure 9. IBM Variable-length Record File Block Structure.

length count MSB	length count LSB	binary zero	binary zero

Figure 10. IBM Variable-length Record Block/Record Header Fields.

Both block and record headers are 4 bytes (2 words) long with the block or byte count in the first word with the most-significant byte (MSB) first least-significant byte (LSB) second. This is the opposite of how the IBM PC stores binary words in memory. The second word of the header should always be zero (NOT the ASCII character to !). Remember that both the block and record length headers include their own length so that these counts are actually 4 greater than the actual block or record length.

The additional fields that appear on the bottom half of the screen for this type of transfer are the same as for simple variable-length record transfers. See the preceding section for a detailed description of these fields and of the transfer operation.

String Data Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------

Maximum block 16384
                                                <Tape0Disk>

Output DOS file \temp\string.dta
Translate data? (N=no A=EBCDIC to ASCII E=ASCII to EBCDIC) N
Datatype: (F)ixed (V)ariable (I)BM variable (S)tring S

Characters to skip 0
Characters to transfer ALL

Copy complete - 189952 bytes transferred - press any key to continue:
```

Figure 11. Unlabeled tape to disk-string data transfer fields.

The unlabeled tape-to-disk transfer operation works the same way as a string transfer going from disk to tape data is simply transferred byte-for-byte from tape and written out to the output disk file. Differing block sizes have no effect on the data transfer and data is neither read nor written as logical records but as one continuous string of data.

When a string tape-to-disk transfer is chosen two additional fields appear on the bottom half of the screen:

1 **Characters to skip:** this field pre-filled with zero specifies an optional number of characters to be skipped before the data transfer to tape starts.

2 **Characters to transfer:** this field pre-filled with ALL specifies an optional limit on how many characters to transfer to tape. If needed enter the number of characters in this field.

Reposition Tape or Display Records

This option chosen from the unlabeled tape submenu is **exactly the same** as the option of the same name on the main menu.

Copy Multiple Files from Tape to Disk

Option D allows you to *concatenate* multiple tape files into a single disk file. This is similar to what happens in the opposite direction—disk to tape--when you append a disk file to a file that has been copied to tape but not yet closed. The specified number of files are read from tape and all data goes in one disk file.

The multiple-file tape-to-disk transfer operations use the same additional fields as the single-file tape-to-disk transfers; for a particular type of transfer (for example variable-length records) see the appropriate section above. When multiple-file transfer is specified there is one additional field that appears near the bottom of the screen for all data types:

- **Number of files to transfer:** this field pre-filled with ALL specifies the number of files to be transferred from the current tape position. If the logical end-of-tape (two filemarks) is found before this number of files is transferred the operation ends and the disk file closed. When replacing the value in this field be sure to overtype all 3 characters in the field (use spaces if less than 3 digits).

Summarize Tape Data

This option has been added to the unlabeled menu which allows users to read from the beginning of the tape to the logical end of the tape and summarize each file. A file is defined as data delimited by filemarks. These items are displayed for each file a) minimum block size found b) maximum block size found c) the number of blocks of data in the file and d) the total number of bytes in the file.

Processing Labeled Tapes

Before using the labeled tape operations you should be familiar with the basic concepts of labeled tape formats and processing. Read Chapter 1 of this manual for more information.

When the labeled tape option is chosen from the main menu the following submenu appears and stays on the top half of the screen throughout all labeled tape operations:

```
Flagstaff Engineering Tape Utility Program

A) Copy file from disk to tape
B) Copy file from tape to disk                    Main
C) List files on the tape                         Unlabeled Tape
D) Initialize volume label                        >>>Labeled Tape <<<<
E) Switch OS/DOS label type                       Label type OS/DOS
                                                  Change Parameters
Command: d                                        Initialize Filter

----------<F10.= Help ESC.= Exit Routine>----------
```

Figure 12. Labeled tape submenu.

About Tape Positioning During Labeled Tape Operations

All the labeled tape operations whether disk-to-tape or tape-to-disk rely on reading the HDR and EOF labels written before and after each file. This may cause delays during transfer operations due to the time necessary for TAPEUTL to find the correct position on the tape. This section is meant to explain how TAPEUTL finds its way around a labeled tape so that you'll understand what is going on when nothing seems to be happening. DO NOT attempt to write files on a tape that has no volume label!

The program does not attempt to read the volume label until option A B or C is chosen. If no label exists the position is not underlined—it simply gives an error message and rejects the request. After entering labeled routines the current logical position is the first data file for either read or write.

When you first enter the labeled tape submenu the tape is automatically rewound if it's not already at the load point Then the volume label is read and displayed on the bottom of the screen. The tape is then positioned just before the first file on the tape if there is one. If there are no files on the tape the tape is positioned properly to begin writing the first file. If there is no volume label the tape position will be undefined. Again, **DO NOT** attempt to write files on a tape that has no volume label (if you need to do so you should be in the Unlabeled Tape-section of this manual)!

If you choose to transfer from the current tape position ("C") **then no** repositioning need be done: TAPEUTL simply assumes that the file you want is the next one on the tape.

Two more items of special note:

- unlike the directory on a disk or diskette it is possible to have non-unique filenames on a tape. If there is more than one file with the same name on a labeled tape and you specify that file by name TAPEUTL will always find the first file and never find the second file. To position to files with identical names you must use the position-by-sequence number option.
- whenever you exit the labeled tape submenu and later re-enter it TAPEUTL always reinitializes the tape position by rewinding if necessary then positioning just past the volume label. Be aware of this if you need to go to another submenu (for example if you want to display a tape block) because the current position may no longer be where you think it should be.

Copy from Disk to Tape

```
Flagstaff Engineering Tape Utility Program

A) Copy file from disk to tape
B) Copy file from tape to disk                    Main
C) List files on the tape                         Unlabeled Tape
D) Initialize volume label                        >>>>Labeled Tape <<<<
E) Switch OS/DOS label type                       Label Type OS/DOS
                                                  Change Parameters
Command: a                                        Initialize Filter

                                   <F10 . Help ESC Exit Routine>

                                                  <Disk0Tape>
Tape position (C)urrent/file#/filename/(E)nd C
Input file name \tape\file001.dta                 Tape name File # 1
Translate data? (N=no A=EBCDIC to ASCII E=ASCII to EBCDIC) N
Data type:(F)ixed (V)ariable (S)tring S Filter on (N/Y) N
```

Figure 13. Labeled tape submenu-common disk-to-tape fields.

This option is used to create a tape file from data read from a disk file or files with header and EOF labels automatically written before and after the tape file (Chapter 2 for a discussion of tape labels).

When this option is first chosen from the labeled tape submenu the tape will automatically be positioned correctly to write the first file on the tape (just after the volume label). This is true whether or not there are any files on the tape. When there are existing files on the tape that need to be preserved, you must reposition the tape to avoid overwriting this first file: see the Tape position field description below for an explanation. DOS and OS labels as well as multivolume labeled input and output files are supported.

DOS labels are essentially the same as OS labels except HDR2/EOV2/ EOF2 records are not present. Therefore format record size and block size are unknowns. DOS labels also have a corresponding ANSI (ASCII) format.

Initially the following five or six fields will appear on the bottom half of the screen:

1. **Tape** position**:** this allows the tape to be repositioned before the disk-to-tape transfer starts. There are four possible options:

2. **(C)urrent position: as** one might guess this is simply where the tape already is. Keep in mind that this current position is at the beginning of the tape when the disk-to-tape option is first chosen from the labeled tape submenu (see above). C is the pre-filled default option.

3. **File** #: by entering a number here **TAPEUTL** will look tor that file by sequence number on tape starting at the first file. If the file is found the tape will be positioned at the start of that file ready to start overwriting the file. If the number given is greater than the total number of files on the tape a File Not Found message will be displayed.

4. Filename**:** if the name of a file is entered in this field **TAPEUTL** will attempt to locate a file with that name written in its HDR1 label. If the file is found the tape will be positioned at the start of the next file as described above; if no file by that name is found a File Not Found message will be displayed.

5. **Input file name:** this is the fully-qualified name of the DOS disk file which must include any necessary drive letters and pathnames to identify the file (e.g., B:\text\letter1.doc).

Translate data? Three (3) choices are possible:

N - no translation

A - data will be translated from EBCDIC to ASCII

E - data will be translated from ASCII to EBCDIC.

6. **Data type:** this specifies how the incoming data is to be treated. Since data on tape is always written in blocks the entry in this field will control how the disk file data is placed into the tape blocks. There are three (3) possibilities:

- **Fixed-length records:** data is read from the disk file as records of a specified length (given by the user). The disk data is not scanned for line-end characters or other delimiters but is assumed to be divided into logical records of the user-specified length.
- Variable-length records: the disk file is assumed to be in logical records of varying length, each record terminated by an identical sequence of character(s). WIththis type of transfer, data is read from disk as variable-length but written to tape as fixed-length records (short records are padded out with a fill character).
- String data: this option is used in the case where the disk data file is not divided into logical records at all (e.g., an executable program file which consists of binary data). The disk data is treated as a continuous string of data. The tape file is divided into blocks, but the blocks are simply fixed length chunks of the input file, with a possible short last block.

7. Filter **on (N/Y)**: this field will only appear if the translation filter has been initialized from the main menu. If Y(es) is specified, data from the input file is translated according to statments in the filter file before being written to tape. The default (pre-filled) value here is N(o). See the section on the DOS filter function for more information.

   Depending on the data type chosen for the transfer, additional fileds will appear below these fields. The complete screens for each data type are shown next, with a description of the remaining fields.

Fixed-Length Data Transfer

```
----------<F10 = Help ESC = Exit Routine>----------


                                                <Disk0Tape>

Tape position (C)urrent/file#/filename/(e)nd e
Input file name \onfile\onfile.exe              Tape name ramboloon
Translate data? (N=no, A=EBCDIC to ASCII, E=ASCII to EBCDIC) N
Data type: (F)ixed, (V)ariable, (S)tring f
Record size 256
Records per output tape block 100
Records to skip 0
Records to transfer ALL

8 blocks 742 records 189952 bytes copied
                    Do you wish to append another DOS file? (n/y):
```

Figure 14. Disk to labeled tape-fixed-length transfer fields.

Four additional fields appear for fixed-length transfer operations:

1. Record size: this specifies the length of each logical record in the disk file. Remember that TAPEUTL doesn't perform any checking of the input records, but assumes that the input records are, in fact, this length. If you are not sure what length the file records are, look at the file first with a file dump/display utility (for example, Flagstaff Engineering's DFILE program) to determine the proper record size.

2. **Records per output tape block:** this is the blocking factor to be used on tape. This number multiplied by the record length gives the tape block size in bytes, which can be a maximum of 65,536 bytes (65,519 for the OD3210). If this size is exceeded, the program will display a warning message and you will be asked to change either the record size or the number of records per block .

3. **Records to skip:** this field, pre-filled with zero, specifies an optional number of records from the input file to be skipped before the data transfer to tape starts.

4. **Records to transfer:** this field, pre-filled with ALL, specifies an optional limit on the number of records to be transferred to tape. When only a certain number of records are to be copied, enter the number in this field.

When a fixed-length transfer is started, TAPEUTL gets the size of the input file from the disk directory, displays it, and compares this size to the record size to determine whether or not there will be a short record at the end of the file. When there is, a message will appear at the bottom of the screen asking if you want to transfer this partial record. Answering Y will cause this last record to be padded with binary zeros (O) at the end of the file; answering N will simply drop this record from the tape file. There will usually be a short block at the end of a fixed-length, blocked record file.

After the transfer is complete, the number of tape blocks, number of logical records and number of bytes in the tape file will be displayed. A prompt will appear at the bottom of the screen asking if you want to append another file to this one; see *Appending Files on Tape* below for an explanation of this choice. To transfer another file, simply hit **[ENTER]** at this point. This will close the tape file by writing EOF labels after the last block and prepare you for writing another, new file after this one. Press **[ESC]** to return to the labeled tape submenu.

Variable Length Data Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------



                                                <Disk0Tape>
Tape position (C)urrent/file#/filename/(e)nd C
Input file name \doc\taputil2.asc                Tape name databait!!!
Translate data? (N=no, A=EBCDIC to ASCII, E=ASCII to EBCDIC) N
Data type: (F)ixed, (V)ariable, (S)tring v
Record size 80
Records per output tape block 200
Records to skip 0
Records to transfer ALL
Line delimiter 0D 0A
Pad character 20

2 blocks 238 records 19040 bytes copied
                WARNING! 6 records exceeded specified length
                                                          —press
any key to continue:
```

Figure 15. Disk to labeled tape-variable-length transfer fields.

Six additional fields appear when a variable-length transfer is chosen:

1. Record size: this is the size of the tape (output) record, which is the size of the largest record in the file *not including line delimiter characters.* Remember that variable-length only applies to the disk (input) file; the 2 blocks 238 records 19040 bytes copied tape file is written as fixed-length records. If a record is shorter than the given record size, it is right-filled with pad characters (see below). If a record is longer than this size, it will be split--the excess part will be written as a new record--and a warning message will be displayed after the transfer:

WARNING! nn records exceeded specified length

If this happens, try the transfer again with a larger record size. No data will be lost in any case, but the file may need to be reformatted to restore lines to their original form.

2. **Records per output tape block:** this is the blocking factor to be used on tape. This value multiplied by the record length gives the tape block size in bytes, which can be a maximum of 65,536 bytes (65519 bytes for the OD3210). If this size is exceeded, the program will display a warning message and you will be asked to change either the record size or the number of records per block.

3. **Records to skip:** this field, pre-filled with zero, specifies an optional number of records from the input file to be skipped before the data transfer to tape starts.

4. **Records to transfer:** this field, pre-filled with ALL, specifies an optional limit on the number of records to be transferred to tape. When only a certain number of records are to be copied, enter the number in this field.

5. **Line delimiter:** this specifies which character or string of characters are to be taken as an end-of-line marker in the input file. This entry must correspond to the actual line delimiters in the file for the transfer to be successful. The characters (up to 10) are specified as 2-digit hexadecimal values, with the most common values pre-filled: 0DOA, or ASCII carriage-return and line-feed characters.

CAUTION! See the section on the DOS filter function for an explanation of pitfalls to avoid if you are using a translation filter.

6. **Pad character:** the character which will be used to fill short records is specified here as a 2-digit hexadecimal value. This field is pre-filled with a commonly-used fill character, the ASCII space (hex 20). The EBCDIC space character is hex 40.

The variable-length data transfer works in this manner: as the input file *is read* from disk, TAPEUTL continually scans the data for the specified line delimiter characters. When the delimiter string is found, the record is filled out to the specified record length with pad characters if necessary, then written to tape. The line delimiters themselves are not transferred, but they can be restored by the corresponding tape-to-disk transfer routines. When a record exceeds the specified length, it is split as explained above in the record size field description, and a count of all such records is kept and displayed after the transfer operation. For a variable-length transfer, data remaining at the end of a file will always be written as a short block on tape if necessary.

After the transfer is complete, the number of tape blocks, number of logical records and number of bytes in the tape file will be displayed. A prompt will appear at the bottom of the screen asking if you want to append another file to this one; see Appending Files on Tape below for an explanation of this choice. To transfer another file, simply hit **[ENTER]** at this point. Press **[END]** to return to the labeled tape submenu.

String Data Transfer

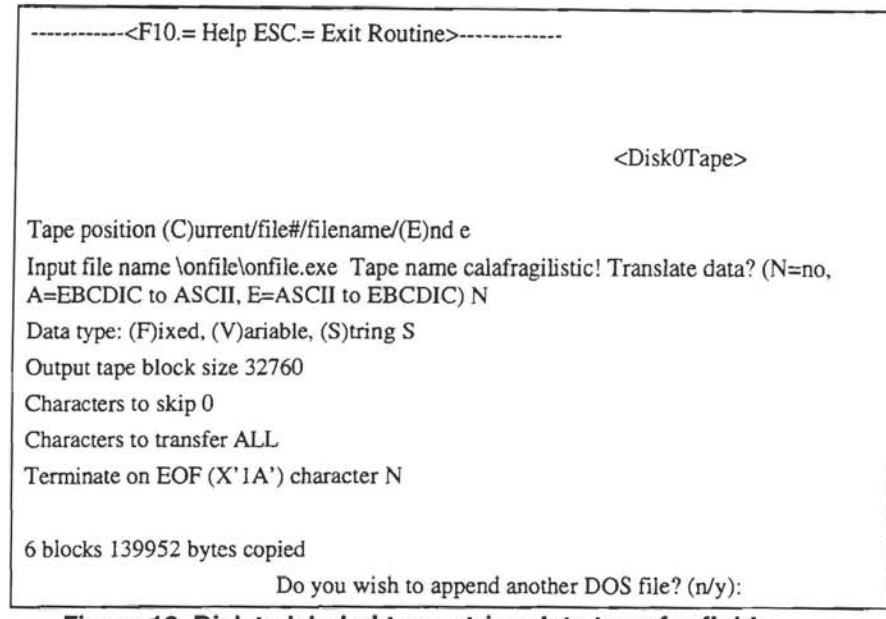


Figure 16. Disk to labeled tape-string data transfer fields.

When a string-data transfer is chosen, four additional fields appear on the lower half of the screen:

1. **Output tape** block size: this specifies the actual tape block size to be written, up to 65,536 bytes (65,519 bytes for the 3201 and 3210 drives).
2. **Characters to skip:** this field, pre-filled with zero, specifies an optional number of characters to be skipped before the data transfer to tape starts.
3. **Characters to transfer:** this field pre-filled with ALL specifies an optional limit on how many characters to transfer to tape. If such a limit is needed enter the number of characters in this field.

4. **Terminate on EOF (X'1A') character?** If the data to be transferred is in a normal ASCII text file this option can be chosen (by entering Y) to end the transfer on the conventional ASCII end-of-file character (hexadecimal 1A decimal 26 also known as Control-Z). This option should not be chosen if so-called binary data is to be transferred (i.e. executable programs image data, etc.) because such data is likely to contain many hex 1A characters which would falsely be interpreted as end-of-file.

A string data transfer will simply copy all the data from the disk file byte-for-byte to the tape file with no processing or alteration unless translation between ASCII and EBCDIC is chosen or if the data is filtered (see the section on the DOS filter function). When there is any question about the form of the disk file this option is the safest way to transfer data to assure that no data is lost.

After the transfer is complete the number of tape blocks and number of bytes in the tape file will be displayed. A prompt will appear at the bottom of the screen asking if you want to append another file to this one; see *Appending Files on Tape* below for an explanation of this choice. To transfer another file simply hit **[ENTER]** at this point Press **[ESC]** to return to the labeled tape submenu.

Copy from Tape to Disk

```
Flagstaff Engineering Tape Utility Program
A) Copy file from disk to tape
B) Copy file from tape to disk                          Main
C) List files on the tape Unlabeled                     Tape
D) Initialize volume label                  >>>>Labeled tape <<<<
E) Switch OS/DOS label type                 Label Type OS/DOS
                                              Change Parameters
                                                Initialize Filter

 Command: b

------------<F10.= Help ESC.= Exit Routine>------------
Maximum block = 16384                                <Tape0Disk>
Tape position (C)urrent/file#/filename) C
Output DOS file
Translate data? (N=no, A=EBCDIC to ASCII, E=ASCII to EBCDIC) N
Data type: (F)ixed, (V)ariable, (I)BM variable, (S)tring Filter on (N/Y) N
```

Figure 17. Labeled tape Submenu-common tape-to-disk fields.

This option on the labeled tape submenu allows tape files to be transferred to disk. When this option is first chosen from the labeled tape submenu, the tape will automatically be positioned to read the first file on the tape, as indicated by a message that appears at the bottom of the screen. The first field that appears on the lower half of the screen asks you where you would like the tape to be positioned when the transfer starts. Filling in this field and hitting **[ENTER]** will cause this positioning to take place; to choose the pre-filled option of C(urrent), just hit **[ENTER]**. The positioning options are described below.

After the tape has been positioned, the other 4 or 5 fields common to all tape-to-disk transfers will appear (the 5th field only appears if you have initialized the filter: see the section on the DOS filter function). Fill in these fields, then press **[ENTER]** to start the transfer operation.

1. **Tape** position: specifies the position of the tape before the transfer takes place. There are three possible choices:

- **(C)urrent:** this leaves the tape in its current position. When you first choose this transfer option from the labeled tape submenu the tape will be positioned at the start of the first file. After a transfer operation the tape will be positioned at the start of the next file on the tape.

- **File #:** this positions the tape at the start of a selected file by sequence number on tape. If the requested file is not found before the logical end of tape is found an error message will be displayed.

- Filename: this positions the tape at the start of a selected file by tape file name. If the named file is not found on the tape (by reading the tape's HDR1 labels) a message will be displayed.

2. **Output DOS file:** this is the fully-qualified name of the DOS disk file which will be written which must include any drive letters and pathnames needed to identify the file (e.g., B:\text\letter1.doc). If the drive and pathnames are omitted the file is written in the current directory on the current default drive.

   Translate data? three choices are possible:

   N - no translation

   A - data will be translated from EBCDIC to ASCII

   E - data will be translated from ASCII to EBCDIC

3. **Data type:** this specifies how the incoming data from tape will be treated in transferring it to disk. There are four possible choices:

- **Fixed-length records:** the tape file is assumed to be divided into equal-length records of a size given by the user. These logical records are simply copied from the tape blocks to the disk file as they are read. No line delimiters are read or written.

- **Variable-length records:** this really means variable-length physical blocks on tape. Each block in the tape file is treated as a separate logical record and these blocks can vary in size. In transferring these blocks to disk file records line-end delimiters (e.g. carriage-return line feed pairs) can be added to the end of each record to produce a valid ASCII-type file--that is a file that can be successfully displayed using the TYPE command or printed on a line printer.

- IBM **Variable-length records:** this is a special type of variable-length record where each logical record has a 4-byte prefix which contains a binary count of characters in the record. This count is contained in the first word (2 bytes) of the prefix and the second word of the prefix is usually binary zero. This option is generally only used when transferring files from tapes written on IBM mainframe computer systems.

4. String data**:** this option is used in the case where the tape file is not divided into logical records at all (for example an executable program file which consists of binary data). The tape data is treated as a continuous string of data. The tape file is divided into blocks but the blocks are simply fixed-length chunks of the input file with a possible short last block. All tape data is transferred to disk contiguously (with no breaks).

5. Filter **on (N/Y):** this field will only appear if the translation filter has been initialized from the main menu. If Y(es) is specified, data from the tape is translated according to statements in the filter file before being written to the disk file. The default (pre-filled) value here is N(o). See the section on the DOS filter function for more information.

Regardless of the type of data transfer chosen TAPEUTL will search for the given output disk file to see if one already exists. If it does the following message will appear:

DOS file already exists
do you want to erase **old file (Y/N)?**

If you answer Y at this point the existing file will be replaced and all data currently in the file will be lost. If you answer N, the transfer will not start until you hit **[ENTER]** again giving you an opportunity to change the output file name.

If the target volume (fixed disk or diskette) that the file is being copied to becomes full during the transfer operation the transfer will be suspended and the following message will appear at the bottom of the screen:

DOS volume full
- new file name (ESC to abort)

If you want to continue the transfer, you will need to enter the name of the file which will contain the next part of the tape file, including any needed drive letter and pathname specifiers. The transfer will resume with the next byte character) of the tape file on the new disk file. To end the transfer, hit **[ESC]**; the data on the full disk volume will be retained, and you will return to the unlabeled tape submenu.

Fixed-Length Data Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------

Maximum block = 16384                          <Tape0Disk>
Tape position (C)urrent/file#/filename) jazzercise
Output DOS file \temp\jazz
Translate data? (N=no A=EBCDIC to ASCII, E=ASCII to EBCDIC) N
Data type: (F)ixed, (V)ariable, (I)BM variable, (S)tring F Filter on (N/Y)N

Record length 50
Strip trailing pad characters (Y/N) N Pad character 20
Insert line end (Y/N) N
Line end characters (in hex) 0DOA
Records to skip 0
Records to transfer ALL

                    Copy complete -1310 records transferred
                                  press any key to continue:
```

Figure 18. Labeled tape-to-disk fixed-length transfer fields.

Seven additional fields will appear for a fixed-length tape-to-disk transfer operation:

1. **Record length:** this specifies the length of one logical tape record. When the tape file is located, the record length given in the HDR2 label is put in this field. To override this length, type in the new length here.

2. Strip **trailing pad characters?** If Y is chosen here, TAPEUTL will remove all trailing pad characters in the output record (up to the first non-pad character from the end). Usually if this option is chosen the pad character will be a space and this option is chosen to eliminate wasted file space. If N is chosen, no characters are removed from the output record. In either case only the output (disk) file is affected, not the input (tape) data.

3. **Pad character:** used in conjunction with the preceding option this specifies the pad character to be (optionally) stripped from the output record. This entry is specified as a 2-digit hexadecimal value corresponding to a single character and is pre-filled with the most common pad character, the ASCII space (hex 20).

4. **Insert line end?** this field and the next set options that allow a tape file of fixed-length records to be written to disk as a file of variable-length records with line-end delimiters. If Y is chosen, each logical record will have line-end delimiters appended in the output record. By appending delimiters the file will be essentially ASCII text-type file (assuming ASCII text data).

5. **Line end characters (In hex):** used with the preceding option, this specifies the line-end characters to be appended to each output record if Insert line end is chosen. These characters (up to 10) are specified as 2-digit hexadecimal values. The field is pre-filled with the most common line delimiters the ASCII carriage-return and line-feed characters (hex OD-OA decimal 13-10)

6. Records **to skip:** this field pre-filled with zero specifies an optional number of records from the input file to be skipped before the data transfer to tape starts.

7. **Records to transfer:** this field pre-filled with ALL specifies an optional limit on the number of records to be transferred to tape. If only a certain number of records are to be copied enter the number in this field.

After the transfer is completed a message will appear stating the number of records transferred. After clearing this message you can start another tape-to-disk transfer by filling in the appropriate fields and hitting **[ENTER]**; the tape will be positioned at the start of the next file. Hit **[ESC]** to exit to the labeled tape submenu.

Variable-Length Data Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------

Maximum block 16384                          <Tape0Disk>
Tape position (C)urrent/file#/filename) 4
Output DOS file \temp\slaboom
Translate data? (N=no, A=EBCDIC to ASCII, E=ASCII to EBCDIC) N
Data type: (F)ixed, (V)ariable, (I)BM variable, (S)tring v Filter on (N/Y)N

Record length 256
Strip trailing pad characters (Y/N) N Pad character 20
Insert line end (Y/N) N
Line end characters (in hex) 0DOA
Records to ship 0
Records to transfer ALL

                                                        Copy
complete - 8 records, 266241 bytes
                                                        press
any key to continue:
```

Figure 19. Labeled tape to disk-variable-length transfer fields.

Seven additional fields will appear for a variable-length tape-to-disk transfer operation:

1. **Record length:** for notation purposes only. If the tape file was formatted with fixed-length records, the record length from the file's HDR2 label will be prefilled here; otherwise, this field is blank. This field is ignored by TAPEUTL, except that there must be a non-zero value entered here.

2. Strip **trailing pad characters?** If Y is chosen here, then TAPEUTL will remove all trailing pad characters in the output record (up to the first non-pad character from the end). Usually, if this option is chosen, the pad character will be a space, and this option is chose to eliminate wasted file space. If N is chosen, no characters are removed from the output record. In either case, only the output (disk) file is affected, not the input (tape) data.

3. **Pad character:** with the preceding option, this specifies the pad character to be stripped from the output record (optionally). This entry is specified as a 2-digit hexadecimal value corresponding to a single character, and is pre-filled with the most common pad character, the ASCII space (hex 20).

4. **Insert line end?** this field and the next set options that allow a tape file of fixed-length records to be written to disk as a file of variable-length records with line-end delimiters. If Y is chosen here, then each logical record from tape will be have line-end delimiters appended in the output record. By appending delimiters, the file will be essentially a normal ASCII text-type file (assuming that the data is ASCII text).

5. **Line end characters (In hex):** used with the preceding option, this specifies the line-end characters to be appended to each output record if Insert line end is chosen. These characters (up to 10) are specified as 2-digit hexadecimal values. The field is pre-filled with the most common line delimiters, the ASCII carriage-return and line-feed characters (hex 0D-0A, decimal 13-10).

6. **Records to skip:** this field, pre-filled with zero, specifies an optional number of records from the input file to be skipped before the transfer to tape starts.

7. Records **to transfer:** pre-filled with ALL, this specifies a limit on the number of records transferred. You can enter a number in this field.

After the transfer is done, a message shows the number of records transferred. After clearing this, you can start another tape-to-disk transfer by filling in the fields and pressing **[ENTER]**; the tape move to the start of the next file. Hit **[ESC]** to exit.

IBM Variable-Length Record Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------
Maximum block . 50000                          <Tape0Disk>
Tape position (C)urrent/file#/filename) C
Output DOS file \temp\ibmvar
Translate data? (N=no A=EBCDIC to ASCII E=ASCII to EBCDIC) N Data
type: (F)ixed (V)ariable (I)BM variable (S)tring i Filter on (N/Y)N

Record length 50
Strip trailing pad characters (YIN) N Pad character 20
Insert line end (Y/N) N
Line end characters (in hex) 0DOA
Records to skip 0
Records to transfer ALL

                    Copy complete - 3 records 65757 bytes -
                                  press any key to continue:
```

Figure 20. Labeled tape to disk-IBM variable-length transfer fields.

IBM variable-length record files consist of blocks prefixed with a 4-byte block length header containing the number of bytes in the block and each record is prefixed with a 4-byte record length header containing the number of bytes in the record. The layout of these blocks and headers is shown in Chapter 1.

The additional fields that appear on the bottom half of the screen for this type of transfer are the same as for simple variable-length record transfers. See the preceding section for a detailed description of these fields and of the transfer operation.

This Page Blank

String Data Transfer

```
----------<F10.= Help ESC.= Exit Routine>----------
Maximum block 50000                          <Tape0Disk>
 Tape position (C)urrent/file#/filename) C
Output DOS file FROWN
Translate data? (N=no, A=EBCDIC to ASCII, E=ASCII to EBCDIC) N
Data type: (F)ixed, (V)ariable, (I)BM variable, (S)tring s Filter on (N/Y)N

Characters to skip O
Characters to transfer ALL

                    Copy complete - 13250 bytes transferred
                                  press any key to continue:
```

Figure 21. Labeled tape to disk-string data transfer fields.

The labeled tape-to-disk string data transfer operation works the same way as a string transfer going from disk to tape:data is simply transferred byte-for-byte from tape and written out to the output disk file. Differing block sizes have no effect on the data transfer, and data is neither read nor written as logical records, but as one continuous string of data.

When string data is written to a labeled tape, the record type in the HDR2 label is set to U for Undefined, and no record length (a length of zero) is recorded in the label. Two additional fields appear on the bottom half of the screen:

1. **Characters to skip:** this field, pre-filled with zero, specifies an optional number of characters to be skipped before the data transfer to tape starts .
2. **Characters to transfer:** this field, pre-filled with ALL, specifies an optional limit on how many characters to transfer to tape. If needed, enter the number of characters in this field.

List Files on the Tape

This option gives a directory listing of the tape. By reading each file's header labels (HDR1 and HDR2), TAPEUTL lists each file in sequence and displays the following information:

Sequence number: the number of the file, starting at 1.

Name: the name of the file recorded in the HDR1 label, up to 17 characters long.

Type: F for fixed-length records V for IBM variable-length records U for undefined type.

Block Size: the physical size of the tape blocks (maximum size—file may contain short blocks at the end or in the middle).

Record size: the logical record length recorded in the file's HDR2 label (always zero for type "U" files).

Number of blocks: number of physical tape blocks in the file.

If the screen fills up with entries and there are more files to display, the display will stop and the following message will appear:

More - press any key to continue:

Hit any key to display the next screenful of entries. When the last file on the tape has been found, the message:

Directory list completed

will appear at the bottom of the screen. The display will remain on screen until you hit any alpha/numeric key to return to the labeled tape submenu.

```
----------<F10.= Help ESC.= Exit Routine>----------

seq. no.   name      type   block size   record size   blocks
1          GORILLA   U      20000        0             1
2          SETDN     U      200          0             1
3          ZRAY1     U      20000        0             4
4          TURGIL    F      6000         60            11
5          SCUMBY99  F      12800        128           2
Directory list completed - press any key to continue:
```

Figure 22. Directory Listing for Labeled Tape

Initialize Volume Label

This submenu option prepares a tape for writing labeled files.This is analogous to formatting a diskette by using the DOS FORMAT command. A standard ANSI or IBM label is written at the beginning of the tape, and the tape is positioned to write the first file. You can also choose to completely erase the tape after the volume label to eradicate any existing data; however, this operation takes several minutes.

Because this operation writes a new volume label on the tape, any data on the tape should be considered destroyed even if you choose not to erase to the end of the tape. This is because volume labels may not be written in precisely the same place by different systems, even by different runs on the same system. If the first file's HDRI label is overwritten by the VOL1 label, the data can be considered "trashed." If a tape is accidently initialized, you may still be able to recover files from it. Use the Reposition Tape and Display Records option from the unlabeled tape submenu to examine the existing data on the tape.

When initializing a labeled tape, four fields appear on the bottom hall of the screen:

1. **Type of label:** (this can be one of two types):

A — writes an ANSI label (in ASCII).

B — writes an IBM label (standard OS/VS) encoded in EBCDIC.

2. **Volume ID:** this is a 1 to 6-character name which will be written as the tape volume name. This is the name by which the tape would be identified to a minicomputer or mainframe operating system. When entering alphabetic characters, TAPEUTL changes all lowercase letters to uppercase. TAPEUTL does no checking on names, but the host system may impose rules for valid names.

3. **System Name:** A system name may be entered when initializing tapes. This field may be up to ten characters in length on IBM volumes and up to fourteen characters long on ANSI volumes. This system name is placed in volume VOL1 record. If no name is entered here, the ID field in the VOL1 label will be filled with spaces.

4. Erase **to end of tape?** This field is pre-filled with N. If Y is entered, the tape will be physically erased to the end marker to eradicate any existing data. This operation can take several minutes to complete. If this option is chosen and you decide to stop while the tape is being erased, hold down both **[SHIFT]** keys simultaneously to stop the erase operation. This may cause an error message to be displayed, but hitting any key (or **[ESC]**) should clear this message and return you lo the labeled tape submenu.

After these fields are filled in and you press **[ENTER]** to start the initialization, the message:

Tape volume label written

should appear at the bottom of the screen. If the erase to end option was used, the tape will run for several minutes, after which you will be back at the labeled tape submenu.

Positioning the Tape and Displaying the Records

The tape positioning and display submenu looks like this:

```
----------<F10.= Help ESC.= Exit Routine>----------

A)Forward space by files        G)Rewind and unload
B)Back space by files           H)Write filemark
C)Forward space by blocks       I)Display tape record on creen
D)Backspace by blocks           J)Erase to end of tape
E)Space to logical end of tape
F)Rewind

Command:
```

Figure 23. Tape positioning and Display Submenu.

All the operations on this submenu will work with either unlabeled or labeled tapes. When using these functions however TAPEUTL doesn't know whether the tape is labeled or not. Therefore tape labels are treated the same as separate data files when the space forward or backward and display block functions are used.You can display a tape's VOL HDR and EOF labels just as you can look at any other data on tape.

Note: Any repositioning done here on labeled tape is lost when you return to the labeled tape menu! When processing a labeled tape the rewind function on this submenu should be used only to reposition the tape for the purpose of displaying tape data (using the display option on the submenu) or when doing transfers to or from unlabeled tape. Do not use this rewind function in conjunction with labeled tape transfer operations (tape-to-disk or vice versa); when the labeled tape submenu is re-entered after leaving this submenu the tape will automatically be repositioned anyway. See the section on processing labeled tape for further explanation.

How the Positioning Command Works

On this submenu notice that there are two types of spacing (repositioning) operations: by blocks or by files (filemarks). Each can work either forward or backward. All spacing operation quantities (blocks or files) are counted from the *current* tape position. The following explains how positioning works.

Remember that a tape consists of data blocks separated by interrecord gaps, and that files are separated by file marks, or filemarks. Remember that labeled tapes have special records (labels) which are used to record tape and file information. Refer to Chapter 1 for an explanation of tape data layouts.

Spacing Forward/Backward by Files

When spacing by files, TAPEUTL scans the tape and counts filemarks encountered until the specified number of filemarks has passed (the default, if no number is entered with this command, is one filemark). The tape will then be left positioned just after the last filemark found.

If the reposition by filemarks operation is backwards, the first filemark counted is the one at the start of the file the tape is currently positioned at regardless of what the current position is within that file. If the current position is at the beginning of the file, then a backspace-by-one operation will position the tape at the beginning of the previous file, unless the current file is the first one on the tape.

Some examples of spacing by files: (note that the tape is always positioned at an IRG between blocks or files)

Spacing example 1.

BEFORE:

	file 1 block 1	file 1 block 2	... ...	T M	file 2 block 1	file 2 block 2	T M	file 3 block 1	... ...

old position ↑

AFTER: (command given was "backspace 2 files"):

	file 1 block 1	file 1 block 2	... ...	T M	file 2 block 1	file 2 block 2	T M	file 3 block 1	... ...

↑ new position

Spacing example 2.

BEFORE:

file 1 block 1	file 1 block 1	T M	file 2 block 1	file 2 block 2	T M	file 3 block 1	T M	file 4 block 1	T M

↑ old position

AFTER: (command given was "forward space 3 files"):

file 1 block 1	file 1 block 1	T M	file 2 block 1	file 2 block 2	T M	file 3 block 1	T M	file 4 block 1	T M

new position ↑

Spacing Forward/Backward by Blocks

When spacing in either direction by blocks, TAPEUTL scans the tape from the current position and counts the number of blocks that have passed. When this number matches the number of blocks given in the command, the spacing operation is stopped and the tape is positioned at the start of a tape block. (If the tape is labeled, this could be either a data block or a tape label.)

If a filemark is found during the space-by-blocks operation, the operation is halted and the message File Mark Found will appear at the bottom of the screen. The tape will then be positioned at the start of the file after the filemark that was found. If the space operation was in the backwards direction, this will be the same file. If the space direction was forward, this will be the start of the next file. Unless a filemark is found, the space-by-blocks operation always stays within the same file. To space within another tape file, you must first get to that file using the space-by-files option (see previous section).

Remember that a tape consists of data blocks separated by interrecord gaps, and that files are separated by file marks, or filemarks. Keep in mind also that labeled tapes have special records (labels) which are used to record tape and file information. Refer to section one for a basic explanation of tape data layouts.

Some examples of spacing by blocks:

Spacing example 3. (Note that TAPEUTL jumps over filemarks in the forward direction.)

BEFORE:

	file 1 block 1	file 1 block 2	file 1 block 3	file 1 block 4	T M	file 2 block 1	file 2 block 2	T M	... ...

↑ old position

AFTER (command given was "forward space 6 blocks"):

	file 1 block 1	file 1 block 2	file 1 block 3	file 1 block 4	T M	file 2 block 1	file 2 block 2	T M	... ...

new position ↑

Spacing example 4.

BEFORE:

	file 1 block 1	file 1 block 2	file 1 block 3	file 1 block 4	T M	file 2 block 1	file 2 block 2	T M	... ...

old position ↑

AFTER (command given was "backspace 4 blocks"):

	file 1 block 1	file 1 block 2	file 1 block 3	file 1 block 4	T M	file 2 block 1	file 2 block 2	T M	... ...

↑ new position

Note that the tape is positioned at the start of the file if a filemark is found before the given number of blocks are skipped.

Space to Logical End of Tape

This function reads the tape until the logical end-of-tape is found. This is written on the tape as two consecutive filemarks following the last file on the tape. Note that this is not the same as the physical end-of-tape. The logical end of tape may be very close to the start of the tape if there is not much data on the tape.

If there is no logical end-of-tape, the drive will continue to run until the tape runs off the supply reel. If this happens, wind the tape back on the supply reel and wind the tape onto it by hand at least 30-40 turns, then use the drive's Load control to re-load the tape.

Rewind Tape

This function simply issues the rewind command to the tape controller. While the tape is rewinding, a message will be displayed at the bottom of the screen, and TAPEUTL will wait until the tape is actually rewound before returning control to you. If you want to do something else before a long rewind operation is done, hitting **[SHIFT]+[SHIFT]** will cause the message:

Rewind stopped

to appear, and you can choose another function. However, this doesn't stop the rewind operation itself, and you must wait until the tape is rewound before starting any tape operations. You can use such functions as changing the tape parameters, which don't directly send commands to the tape drive.

Rewind and Unload Tape

This function is identical to the rewind function (above), except that the tape is physically unloaded from the drive following the rewind. After this command executes, you must reload a tape in the drive to do further tape processing .

Write Tapemark

This function will write a filemark at the current tape position, then space the tape after the filemark. *THIS FUNCTION SHOULD BE USED WITH CAUTION!!* It is provided only for special circumstances, when you are certain about the tape's current position and need one or more filemarks written there. It you write two filemarks in a row, you create a logical end-of-file marker. This can be used when re-writing a tape that already has data on it, so that you can be sure that no data will be read past the new logical end-of-tape.

Note: All disk-to-tape transfer functions create double filemarks at the end of the file, which leaves the tape wi!h a valid logical end-of-filemarker.

Displaying Tape Records

This function can be used with both labeled and unlabeled tapes to view the contents of any tape records, including tape labels. Tape data is displayed in both hexadecimal and character (ASCII or EBCDIC) formats, in lines of 16 bytes on the screen. When you choose this option, data is displayed starting at the current tape position. It the tape is positioned at a filemark (end of tape file), the first record after the filemark will be displayed. Remember that a tape *record is* actually a tape *block.*

The tape display function has a different screen display than the rest of TAPEUTL. The top part of the screen shows 20 lines (320 bytes) of data in hex/character formats, with the displacement (distance from the start of the block) shown in decimal at the left. The current character representation (ASCII or EBCDIC) is shown at the top right. The bottom line shows the block number and block length, and contains a command entry field.

While in the display function, you can space forward by tape blocks, displaying data from any part of each block, until a filemark is read. When the end of a file is hit, TAPEUTL displays the following message:

> Tape EOF (filemark) found -
> press any key to continue:

The current screen will remain until you hit a key, after which you will be returned to the tape reposition/display submenu. It you choose the display option again without repositioning the tape, the first block of the next file (if any) will be displayed. There are several special keys you can use to select tape blocks to display and to select the position within the block. They are:

[PGUP]—displays the previous screen of data from this block.

[PGDN]—displays the next screen of data from this block.

[HOME]—displays the previous block from the tape.

[END]—displays the next block from the tape.

[↑]—scrolls the screen up one line (displays next 16 data bytes).

[↓]—scrolls the screen down one line (displays previous 16 bytes).

[F1]—toggles the character display between ASCII and EBCDIC.

[ESC]—exits the display routine.

In addition,there are two positioning commands that can be entered on the display command line:

nnn—display data starting at decimal offset nnn into the current tape block

Bnnn—display tape block number nnn.

(type these commands, then hit **[ENTER]**)

For the second command, the operation is the same as the space by blocks options from the reposition/display submenu. It a filemark is read before the specified tape block is read, the operation will be ended. Also note that the display routine may not keep track of the correct block location in the file it you forward space to a filemark, then backspace by blocks within the file.

SWITCHING BETWEEN ASCII & EBCDIC

The **[F1]** key toggles the display of characters in the file between ASCII and EBCDIC representations. This affects only the right-hand part of the display (the hex display stays the same). It you are displaying data that should be readable text and you can't see any recognizable alphabetic characters, you may be using the wrong character set. ASCII characters displayed as EBCDIC usually appear as "garbage," and vice versa. Binary data (binary data files, executable programs, etc.) will always appear as meaningless bytes, although you should be able to read embedded program prompts, messages, and other text.

TAPE RECORD DISPLAY SCREENS

Display screen 1.

This screen shows the HDR1 label from an ANSl-labeled tape file. Notice that the data is encoded in ASCII, and that the label is 80 bytes

DISP	HEX DATA	ASCII
0	48 44 52 31 47 49 52 41 46 46 45 20 20 20 20 20	HDR1 GIRAFFE
16	20 20 20 20 20 59 4F 46 4F 4F 4C 30 30 30 31 30	YOFOOL00010
32	30 30 31 30 30 30 31 30 30 20 38 36 32 39 30 20	001000100 86290
48	30 30 30 30 30 20 30 30 30 30 30 30 20 20 20 20	00000 000000
64	20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20	

COMMAND: BLOCK 2 LENGTH 80 Press ESC to exit

Display screen 2.

This screen shows a data block from an ASCII text file. Notice that each line of text is ended by the bytes 0D 0A (carriage return-line feed).

DISP	HEX DATA	ASCII
0	0B 2E 74 3A 70 63 77 32 63 6F 72 6F 2E 70 72 6D	..t:pcw2coro.pr
16	0D 0A 09 2E 77 3A 34 37 0D 0A 0C 00 0A 0D 0A 0D	.....w:47.....
32	0A 0D 0A 0D 0A 20 20 20 20 20 20 20 20 15 46 4C	... FLAGSTAFF
48	41 47 53 54 41 46 46 20 45 4E 47 49 4E 45 45 52	ENGINEERING
		,INC
64	49 4E 47 0D 0A 0D 0A 0D 0A 20 20 20 20 20 20 20	.....
80	20 20 20 20 20 20 15 54 41 50 45 20 55 54 49 4C	..TAPE UTIL
96	49 54 59 0D 0A 0D 0A 0D 0A 20 20 20 20 20 20 20	ITY.....
112	20 20 20 20 20 15 55 53 45 52 27 53 20 4D 41 4E	..USER'S
		MAN
128	55 41 4C 0D 0A 20 0D 0A 0D 0A 0D 0A 0D 0A 0D	UAL .....
	0A	
144	0D 0A 0D 0A 00 0A 0D 0A 0D 0A 0D 0A 00 0A 0D	I
	0A	
160	0D 0A 0D 0A 0D 0A 0D 0A 0D 0A 0D 0A 20 20 20	.....
	20	
176	20 20 20 20 20 20 20 20 20 20 20 20 20 1C 43 6F	.....Co
192	70 79 72 69 67 68 74 20 31 39 38 36 0D 0A 2C 20	pyright 1987..
208	20 20 20 20 20 20 20 20 20 20 20 IC 46 4C 41 47	.....FLAG
224	53 54 41 46 46 20 45 4E 47 49 4E 45 45 52 49 4E	STAFF
		NGINEERIN
240	47 00 0A 20 20 20 20 20 20 20 20 20 20 20 20 20	G.....
256	20 20 20 20 20 20 20 1C 42 6F 78 20 31 39 37 30	1120KAIBAB
		LANE
272	0D 0A 20 20 20 20 20 20 20 20 20 20 20 20 20 20	.....
288	1C 46 6C 61 67 73 74 61 66 66 2C 20 41 5A 20 38	.FLAGSTAFF,
		AZ8
304	36 30 30 32 0D 0A 20 20 20 20 20 20 20 20 20 20	6001.....

COMMAND: BLOCK 1 LENGTH 7081 Press ESC to exit

Changing the Tape System Parameters

The change tape parameters submenu lets you change a number of system values. Some are hardware-related and change the way the tape drive and controller operate, others are software parameters.

They are changed by entering the right letter in the command field and pressing **[ENTER]**. Press **[ESC]** to exit this menu, with no changes.

```
----------<F10.= Help ESC.= Exit Routine>----------

A) Set low tape speed (default)
B) Set high tape speed
C) Set short interrecord gap for write (default)
D) Set long interrecord gap for write
E) Set error retry count (default = 16)
F) Set tape address (O = default)
G) Set 1600 bits/inch density (default)
H) Set 3200 bits/inch density
l) Set maximum buffer size for read

 Command:
```

Figure 24. Change Tape Parameter Submenu.

Each of the parameters on this menu are explained in the following sections.

Set low tape speed (default)

This option selects the low tape speed for drives that have software selectable speeds (not all drives do). Speed selection is highly drive-dependant, and this option may not be usable on your drive. On drives that have 2 software-selectable speeds, low speed is usually 25 ips (inches per second). Other smart drives have several speeds, but their speed is a function of the data transfer rate and is not selectable by this option. Refer to your drive's operating manual to determine it you can select tape speed.

Low speed is the default set by TAPEUTL at startup time.

Set high tape speed

For drives with software-selectable speeds, this option sets the high speed. This option may have no effect, since speed selection is highly drive-dependent (see above). If high speed is available, it is usually 100 ips (inches per second).

Caution! If your computer is an IBM AT or compatible (not a PC or XT), you may not be able to use high speed tape transfer. The AT's highest possible data transfer rate is slower than that of the PC or XT (testing has found the effective upper limit to be near 100 kBs). The data transfer rate is a function of both tape speed and tape density:

transfer rate = density X speed

For example, at 100 ips and at a density of 1600 bpi, the data transfer rate would be 160 kBs, which is too fast for an AT. You would need to use low tape speed (25 ips), which would bring the transfer rate down to 40 kBs.

Set Short Interrecord gap for write

This option sets the default IRG, approximately 0.6 inches. The IRG setting affects tape write operations only.

Set long Interrecord gap for write

You can set your drive to write with a long IRG between blocks. The long IRG is about 1.2 inches, twice as long as the standard short IRG. The only reason to use a long IRG would be to keep the tape drive in streaming mode during write operations, where the drive is operating continuously rather than starting and stopping between writes. With the current implementation of TAPEUTL, no increase in speed is gained with this option.

Set error retry count

This option sets the number of retries that TAPEUTL will attempt before reporting a hard tape error. The default number of retries is 16. You should increase this number if you are trying to read or write a marginal quality tape. Keep in mind that doing so will increase the time required for all transfer operations, as multiple reads and writes may be required before a good operation occurs.

To set a new number of retries, follow the option letter by the new count. For example, **e 24** sets the error retry count to 24. Typing the option letter only will display the current value of the retry count.

Set tape address

The tape address is the logical unit number of a particular tape drive, O through 7. This address is selected at the drive by either switches or jumpers, and the address used by TAPEUTL must correspond to this address. For most users with a drive, this address will be 0; the default set by TAPEUTL at startup.

By changing the tape address, you can use more than one tape drive in a single session with TAPEUTL. Once the address is set, it remains set until reset or until you exit TAPEUTL.

Set 1600 bits/inch density

This option sends a command to the tape drive to set the tape density for both reads and writes to 1600 bpi, the default set by TAPEUTL at startup. Note that some tape drives will not respond to a software density-select command.

Set 3200 bits/inch density

When the drive in use will operate at 3200 bpi, this option will select this higher density for both reads and writes. Otherwise, this option will have no effect.

Set maximum buffer size for read

This option sets the size of the buffer used by TAPEUTL for tape reads. The default size at startup is set to 16 kB (16,384 bytes). The maximum size selectable is 64 kB (65,536 bytes).

To set the buffer size to a new value, follow the option letter by the new buffer size—for example:

i32000

will set the buffer size to 32,000 bytes. Typing the option letter only will display the current maximum buffer size.

If you receive a Tape block exceeds maximum error message during a tape operation, increase the buffer size with this option, then retry the operation.

The buffer size should not be set much larger than necessary, as this will slow down the average data transfer rate.

The DOS Filter Function

TAPEUTL has a data filter that can convert characters in the data stream, during both tape-to-disk and disk-to-tape transfers. The filter consists of translation statements in a file. Each statement specifies that a certain character or string of characters is to be changed to another character or string of characters each time it appears in the data stream. This file is an ASCII text file that can be created or modified using a text editor DOS Edit or a word processor that handles ASCII files (DisplayWrite 3, Wordstar in non-document mode, EDLIN, etc.).

Initializing the DOS Filter

The filter function is set up from the main menu by choosing option E from the main menu. The following single prompt will appear on the bottom half of the screen:

Code translation file:

Simply enter the name of the file containing the filter translation statements, including any needed drive letter or pathname qualifiers. Press E to submit the file to TAPEUTL. After the file is found, it will be read into TAPEUTL's workspace.When the file is read with no errors the following message will appear:

Code translation filter initialized:

If the given file can't be found, or if the translation file contains invalid statements, an error message will appear, and you can re-enter the filename or exit this option by use of **[ESC]**.

How and When the Filter Functions

Whether the data filtering function operates before or after the other processing functions (handling line-end characters and ASCII/EBCDIC translation) depends on whether the transfer is from disk to tape or vice versa. The following illustrations show the flow of processing for both types of transfer.

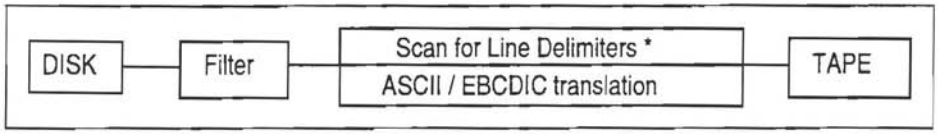


Figure 25. Disk-to-Tape Processing Flow

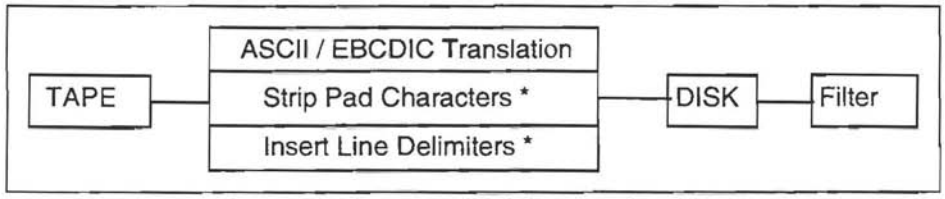


Figure 26. Tape -to- Disk Processing Flow

* —these operations do not apply to string-type data transfers

The filter function is optional.

Understanding when the filter function is applied to the data flow is extremely important to insure the proper results. You can easily get into trouble with filtering, especially if any changes are made to line-end delimiter characters or if filtering changes the length of data records.

It is safest to use filtering on string-type data, since no scanning for (or insertion of) line-end delimiters occurs. If you use variable-length records, care must be taken in altering data with the filter. There are three situations in particular to be aware of:

1. If line-end delimiters are altered during a disk-to-tape transfer, then the new line-end characters are the ones which must be specified on the transfer screen, since the filter alters the data before the line-end processing routines.
2. If the filtering process has the potential for changing the length of data records, the maximum record length should be given as the new longest record length, after any filtering has been done. Otherwise, records may be incorrectly split.
3. If any characters which would be treated as pad characters are altered by filtering (e.g., trailing spaces), the new character must be given as the pad character to strip during the transfer operation.

Filter Translation Statements

All statements must begin in the first position on the line, and all string specifiers must be separated by at least one space. All statements can include a comment field, indicated by square brackets below (the brackets themselves are not required), separated from the end of the statement by at least one space.

The maximum number of translation statements in a file is 2,000, not counting single character (1-to-1) statements. The maximum number of bytes in translation strings is 12,288. The command file text can be a total of 24,576 bytes long.

Each translation statement can be one of three possible types:

- simple string replacement
- toggle string replacement
- wildcard string replacement

The first two types (character strings) are indicated in one of three ways:

1. ASCII character string:

A'... ... ...'

The periods indicate any number (up to 64) of characters. The apostrophes are delimiters that mark the beginning and end of the string. Any character can be used for the delimiters, so long as it doesn't occur within the string itself. The delimiter character must be the first character after the A . To include an apostrophe in the string, use another character as a delimiter:

A/doesn't/

2. EBCDIC character string:

E'... ... ...'

EBCDIC strings are specified the same as ASCII strings (above), except that the EBCDIC equivalents of the characters will be searched for or replaced in the data stream.

3. Hexadecimal byte string:

X'xx xx ... xx'

Each "xx" indicates a byte as a pair of hexadecimal digits (0-9 and A-F). The string can include up to 64 bytes of data. The delimiter must be the first character after the X and there must be one space between each pair of hex digits. This form of string specifier can be used where the desired bytes for searching or replacement cannot be written as characters in the statement itself; for example, carriage returns, tabs, line feeds, etc.

FILTER TRANSLATION STATEMENT TYPES

1. Simple String Translation Statements

This is a line in the file in the form:

old string new string [comment]

where old string and new string are each one of the string specifier forms given above. The simple replacement statement simply does a one-or-one replacement: each occurrence of old string in the data stream is replaced by new string. This statement can also be used to delete old string by simply omitting new string in the statement.

2. Toggle String Replacement Statements

The toggle statement has the form:

T old string new string 1 new string 2 [comment]

The T must be the first character on the line. Each of the 3 strings in the toggle command can be any of the 3 string specifier types given above. The toggle replacement statement does alternating replacement of character strings. When old string is first found in the data stream, it is replaced by new string 1. The next time it is found, it is replaced by new string 2. the next by new string 1, and so on. This function is useful tor translating printer control codes (boldface, underline, etc.) which must be turned on at the start of some text, then turned off at the end of the text.

3. Wildcard String Replacement Statements

The wildcard statement has the form:

W old string new string [comment]

For the wildcard statement only, special string specifiers are used. Each statement starts with the character W. These are the same as the other specifiers described above, except that the question mark character has a special meaning within these strings. Whenever a ? is given in a wildcard string specifier, any character in the data stream will match at this position. The new string specifier can also contain ? characters. If present, they will be replaced by characters from the data stream at the corresponding position in the string. Using wildcard translation, you can convert strings where only part of the string must match exactly.

For example, the translation statement:

WA Dave ?????? A Chris??????

will cause the following translations:

Old String		New String
Dave Johnson	→	Chris Johnson
Dave posters	→	Chris posters
Dave # 12345	→	Chris # 12345

You can use ASCII, EBCDIC or hexadecimal string specifiers with the wildcard statement. To specify a character in a hex string, use the hex equivalent of the ASCII question mark, 3F. The only limitation is that you cannot specifically search for or replace a question mark with this type of statement. The question mark character itself, however, will match if it is found in the data stream where there is a question mark in the string specifier.

Special Note on ASCII vs. EBCDIC Character Strings

When the terms "ASCII character" and "EBCDIC character" are used, these refer to characters in the data stream to or from tape—NOT characters in the translation statement file. The text in the translation file itself must be all ASCII, which is the native character set of the IBM PC.

If a string specifier statement refers to an EBCDIC "e", this means that it will match or replace with the character that corresponds to the EBCDIC "e" character. The "e" in the translation file is in ASCII, always.

Translation Statement Examples

A/"/ A/'/

This statement replaces all ASCII quotation marks in the data stream with ASCII apostrophes.

X'0D 0A'X'0D 25'

replaces the hexadecimal byte string 0D 0A (ASCII carriage return-line feed) with the byte string 0D 25 (EBCDIC CR/LF).

TX'02'X'1B6531'X'1B6530'

alternately replaces the hexadecimal byte value 2 with the second and third byte strings (these strings are toggle printer control codes).

WA'file.???' A'file???.XYZ'

searches for the ASCII text string file. followed by any three (3) characters. The three characters are then moved to a new position in the string.

Special Note on the Order of Translations

The filter function first performs all multiple-character translations, then all single-character (one-to-one) translations. Therefore, it makes no difference which order any one-to-one translation statements appear in the translation file. However, the character strings to be replaced will be searched for in the order that they appear in the translation file. To make sure the proper translations are made, you should realize that this order is important. When the same string of characters are being searched for in two translations of differing lengths, the longer string must be specified first. Otherwise, the shorter string will be found and translated first, and the longer string will never be found.

For example, if the following two statements are in the translation file:

A'ab' E'ab'

A'abnormal' E'unusual'

the ASCII string abnormal will never be found because the string ab will have already been translated to EBCDIC. Reversing the order of these statements would allow the longer string to be found and translated. The following example shows an application of filtering, illustrating the translation file, and the input and output streams:

Translation file:

```
A'bad' A'good'
```

A'good' A'bad'

A'dogs' A'people'

a"" a""

TA'flip' A'flip' A'flop'

WA'Dave ???????' A'Chris ???????'

Input data: Some dogs are very bad, even on good days. Dave Johnson had one once that wouldn't even fetch his paper, so he sold it to Dave Oliver (who then gave it to Dave Jones). Like most dogs, his tail would go 'flip flip flip flip flip flip'.

Output data: Some people are very good, even on bad days. Chris Johnson had one once that wouldn't even fetch his paper, so he sold it to Chris Oliver (who then gave it to Chris Jones). Like most people, his tail would go "flip flop flip flop flip flop".

Automatic Operations of TAPEUTL

TAPEUTL has a facility to allow you to create command files which can later be used to run the program in automatic mode. This feature can be used to relieve the PC user of the task of entering commands manually for repetitive tape transfer jobs. For instance, if you process a certain kind of tape regularly, you can create an automatic command file for the processing session, then simply run TAPEUTL under the control of this command file whenever those tapes are processed.

The command file is created by running TAPEUTL in a special record mode. In this mode, all commands and keystrokes entered from the keyboard are stored and written to the command file. When the program is later run under control of the command file, all these commands and keystrokes are executed in their original sequence.

The automatic-mode feature includes options which allow pauses in automatic execution, during which the user can file in fields in screens before continuing in automatic mode.

The Automatic Mode Status Marker

When you are in automatic mode, either during recording or playback, an additional status marker appears at the top left of the screen. There are four different markers:

Rec—current session is being recorded

Man—manual mode has been entered during either record or playback

Auto—session is being run in automatic execution mode

****—error was encountered during automatic execution; reverts to manual mode.

Recording Your TAPEUTL Session

To create an automatic execution command file, give the name of a file which doesn't currently exist on the DOS command line, as shown here:

C:**tapeutl newfile.cmd**

where newfile.cmd is the name of the file where the automatic execution commands will be written (the reason the file cannot already exist is that an existing file will be taken by TAPEUTL as containing commands to run in automatic mode).

When TAPEUTL is invoked this way, every keystroke from that point is recorded and saved in the command file. Therefore, you should plan your session carefully so that no extraneous or unnecessary command paths are taken. When TAPEUTL plays back a session, it just blindly follows your original path.

Providing for User Input in Automatic Mode

When you are recording a session, you can cause the automatic execution to be temporarily suspended, to allow the user to enter information in screens (for example, file names). You do this by pressing **[F3]** at the point where you want automatic mode to be suspended. When you do this in record mode, the automatic status marker will change to Man, to indicate that manual mode has been entered. During recording, you can continue filling in fields, but these entries will no longer be written to the command file. This manual mode will continue up to the point where you hit the **[ENTER]** key during both record and playback. When **[ENTER]** is pressed, the program will revert back to record mode, or back to automatic execution during playback.

When **[F3]** is used to enter manual mode, the cursor position at that point is saved. When the command file is executed later, this becomes the initial cursor position during playback. You can use this fact to position the cursor to any desired field when recording your session.

Ending Automatic Execution Recording

There are two ways to end the recording of a session:

• by normal termination of TAPEUTL (using **[ESC]** to exit back to DOS from the main menu). This closes the command file, and will cause an exit to DOS at the end when the command file is later used to run TAPEUTL.

• by using the **[F4]** key during record mode. This closes the command file, and will cause TAPEUTL to enter manual mode (as indicated by the marker Man) during later automatic execution. At this point, the user has complete control, just as under normal program operation.

Running TAPEUTL from a Command File

To cause automatic execution of TAPEUTL, simply enter the name of a previously recorded command file on the DOS command line:

C:**tapeutl autofile.cmd**

This command will cause TAPEUTL to be run under the control of the keystrokes stored in the given command file. The status marker Auto, will appear at the upper left, and you will see screens flash by as the stored commands are executed. The program may stop during automatic execution tor one of two reasons:

- if the marker Man appears at the upper left, that means that manual mode was entered when the command file was recorded, either temporarily (using **[F3]**), or permanently (using **[F4]**). During playback, you now have control and must fill in the necessary fields and press **[ENTER]** to continue. If you still have control after **[ENTER]** was pressed, then the original session was ended with **[F4]** and you are now permanently in manual mode. Otherwise, control will revert back to automatic execution.
- if the marker **** appears, that means that an error was encountered during the current session. Either an actual error occurred (file not found, tape error, etc.), or your session took a path that diverged from the original recorded session. In either case, TAPEUTL reverts back to manual mode permanently, and you now have control over the program from the keyboard.

ASCII Batch Command File

TAPEUTL commands may be placed in an ASCII batch file created with any text editor or generated by a program. Each command entered when running the program must be entered in the batch file with the exception of the initial **[ESC]** for paging past the initial copyright screen (the copyright screen is automatically skipped when running from a batch file).

Special keys such as **[TAB]**, **[ESC]**, **[F1]** through **[F10]**, **[PGUP]**, **[PGDN]**, etc. may be assigned by entering the key code preceded and followed by tilde sign (~). As an example, the escape key may be entered as ~ESC~. Either upper or lower case letters may be used.

The following special key codes are recognized:

ESC	Escape key
TAB	Tab key (right tab)
STAB	Shifted tab (left tab)
F1 - F10	Function keys
SF1 - SF10	Shifted function keys
CF1 - CF10	Control function keys
AF1 - AF10	Alternate function keys
A0 - A9	Alternate numeric keys
A-	Alternate hyphen
A=	Alternate equal
Ax	Alternate any key (x any letter key)
UPAR	Up arrow
DNAR	Down arrow
LFAR	Left arrow
RTAR	Right arrow
HOME	Home key
END	End key
PgUp	Page up
PgDn	Page down
INS	Insert
DEL	Delete
CLAR	Control left arrow
CRAR	Control right arrow
CPGU	Control page up
CPGD	Control page down
CHOM	Control home
CEND	Control end

The old batch routines for TAPEUTL are still implemented. When the program is initiated and a batch file name is entered on the command line, it is processed as before. If the file does not exist, the program goes into automatic mode and tries to execute a file which was created before learn mode. If it does not exist, the program goes into auto mode and tries to execute a file which was created before in the learn mode. To execute an ASCII batch file, the file name must be preceded by an equal sign, such as:

C:>**TAPEUTL =FILE.ASC**

INSUTIL.EXE

(COLOR AND PARAMETERS INSTALLATION PROGRAM)

This program allows the user to change screen colors and parameters used by the Flagstaff Engineering TAPEUTL or EXTRACTD programs. In response to the request for NAME OF PROGRAM, enter the full program name, for example, TAPEUTL.EXE . Colors are specified for three types of display attributes:

NORMAL	used for displaying most prompts, text, etc.
ENHANCED	used for operator entry fields and error messages
REVERSED	used to highlight selected data (EXTRACT only)

When a new color is requested, pressing the return leaves it unchanged. The following color values should be used to indicate the desired display colors:

display colors:

FLAGSTAFF ENGINEERING - INSTALL PROGRAM OPTIONS

Program name: TAPEUTL.EXE

Screen display colors

0= Black	8= Dark Grey
1= Blue	9= Light Blue
2= Green	A= Light Green
3= Cyan	B= Light Cyan
4= Red	C= Light Red
5= Magenta	D= Light Magenta
6= Brown	E= Yellow
7= Light Grey	F= Bright White

Normal Background 1 Foreground 7
Enhanced Background 1 Foreground F
Reversed Background 7 Foreground 1

Foreground

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
3	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
4	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
5	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
6	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
7	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
8	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
9	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
B	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
C	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
D	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
E	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
F	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Press <ESC> to exit without update

Color Select Screen

After colors and parameters have been chosen, the same screen will display lines using the colors selected in this manner:

This is a normal line (x, the color you selected)

This is a bright line (x, the color you selected)

This is a reverse line (x, the color you selected)

You can also set default parameters of your choosing with Tape Utility. The options chosen from this screen will be the default values that the program will use to pre-fill these fields whenever TAPEUTL is run.

```
FLAGSTAFF ENGINEERING - INSTALL PROGRAM OPTIONS

Program name: TAPEUTL.EXE

System parameters ----------------------------------------
Read buffer length 16384                    Tape retries 15    Tape
address 0
Tape to disk defaults ----------------------------------
Translate data? (N=no, A=EBCDIC to ASCII, E= ASCII to EBCDIC) N
Data type: (F)ixed, (V)ariable, (I)BM variable, (S)tring S
Record length 0                  Strip pad characters (YIN) N    Insert line
end (Y/N) N
Pad character Line end characters 0D 0A
Disk to tape defaults ----------------------------------
Translate data? (N=no, A=EBCDIC to ASCII, E- ASCII to EBCDIC) N
Data type: (F)ixed, (V)ariable, (I)BM variable, (S)tring S
Block size 4096                             Record size 128    Records
per block 32
Pad character Line end characters 0D 0A
Press <ESC> to exit without update
```

Parameters Installation Screen

At the end, this screen displays verified updates were made thus:

```
Color defaults updated

Taputl defaults updated

C>
```

DOS Backup

Flashbak: High Speed Backup & Restore

Important Compatibility Note

Newer versions of Flashbak will restore backups written by older versions. However, the reverse is not true. Older versions of Flashbak cannot be expected to accurately restore tapes written by newer versions. In some cases, attempts to use old Flashbak versions to restore new version tapes have resulted in restored files being written incorrectly to disk. Make sure that you use the same version, or a newer version, of Flashbak when reading a tape. If your business uses Flashbak at different locations, make certain that they all are using the same version, and that updates are made to all installations.

Flashbak will not restore archives (backups) made using tar, earlier versions of ntar, or the current version of ntar—*when the backup was made using the command line version of this module.*

Important Note

Use the verify feature after making a backup to make sure the data written is valid *before* continuing!

Overview

Flashbak is an interactive, window-oriented tape backup and restore utility. With Flashbak you can select files (by various criteria) for backup or restore operations.

Flashbak can:

- Examine the directory structure of a disk.
- Examine the file names in any directory.
- Select files that match a set of file names for backup or restore.
- Select files that have been recently modified.
- Select files that reside within a particular subtree.
- Create a backup to tape.
- Restore a backup from tape.

In addition, Flashbak performs file management operations such as finding a file or directory; deleting one or more files, or a directory; printing a listing of files that it backed up.

Flashbak has a contextual help system to guide you through its operation. It has been optimized to perform tape operations quickly.

Note: **If you have expanded memory or emulate it with an expanded memory manager, Flashbak will automatically use it and will provide greater speed.**

Before Using Flashbak

Make sure your subsystem is correctly installed by executing the basic subsystem test: XTEST (see the Installation and Tape Utilities manual).

After running XTEST, test FLASHBAK:

- Load a tape into your tape drive.

Note: **Flashbak overwrites the tape during this test.**

- type: C:\ODI\FLASHBAK **[ENTER]**
  When the start up screen comes up, press any key.
- Press **[SPACE]**. This will tag (select) all the files in the root directory.
- Press **[ALT] +B**. This brings up the Backup menu.

- Press **T**. This selects and executes the backup option (from the menu) to backup only tagged files—a dialog box will pop up to remind you to load the tape drive, in case you haven't already.
- The Backup will take several seconds (depending on the size of your root directory). When complete, a message will indicate that backup was successful.
- Load the backup directory (press **[CTRL]+L**).
- Now press **[ALT]+R**. This opens the Restore Menu.
- Press **V**. This will cause all the backed up files to be verified against the files that are actually in your root directory, again you will get a dialog asking you to be sure that the correct tape has been loaded.
- When this is complete, you will get a message that all files were verified successfully. Press **[F2]** to return to the System Directory Window.

Important: If the *Verify* reports no problems, you can use Flashbak for archiving.

Window-Oriented

Flashbak is window-oriented, so you execute commands by opening a menu, selecting an item, and pressing **[ENTER]**. It also lets you execute most commands quickly by pressing a "Quick Key" sequence (e.g., **[CTRL]+B** backs up your entire file system).

Flashbak also recalls the last menu used (press **[INS]** to open it).

Tagging

Flashbak lets you select (tag) files for backup or restoration. You can tag by file name matching, directory selection, single file, modification date, or file attribute. Once you tag a file or set of files, a command can back up, restore or delete all these tagged files.

Tag Specification File

Flashbak provides a way to remember which files it tagged by storing the file names in a Tag Specification File. Flashbak can then tag all files according to a given Tag Specification File.

Scanning the Directory Structure

Examine your disk file system through the window displays (use cursor commands). The "Directory Window" shows your file system's structure. The "File Window" shows file names in the current directory. Function keys let you view one or both windows.

Cursor Control Commands

The ⇧ and ⇩ keys are used to scroll the cursor up and down. **[PGUP]**, **[PGDN]**, **[HOME]** and **[END]** let you move around in the directory tree or file list.

Loading the Tape Directory

You can examine a tape (from Flashbak or an earlier version of Flashbak) by telling Flashbak to load the tape's directory. You can scan the tape's directory structure just like a disk's.

Limitations

The number of files in a directory is limited to 32 k files.

The maximum number of directories is limited to 100 k directories.

Flashbak requires a minimum of 512 kB of memory.

Flashbak typically requires no more than 500 kB of disk swap space. For very large systems, this might go as high as 1 MB.

Keyboard Mapping

Remapping the keyboard with SuperKey or a similar utility may prevent Flashbak from interpreting keyboard input correctly.

Tutorial

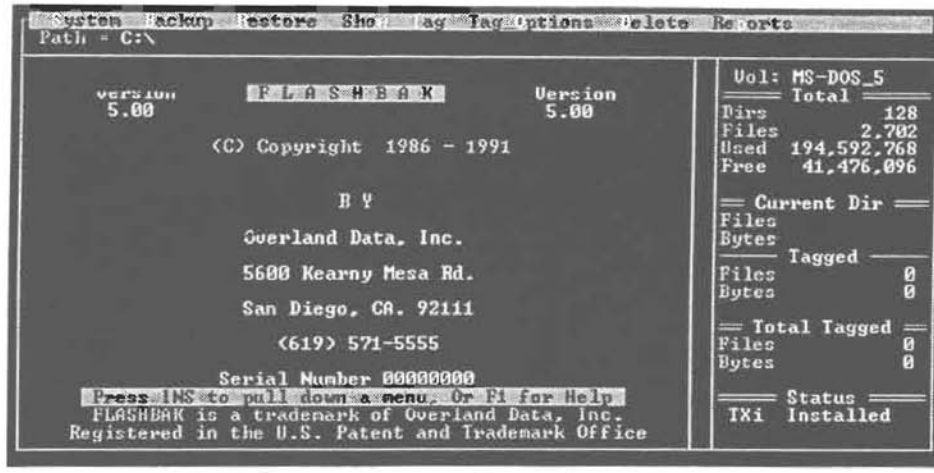
This section guides you through several commonly performed tasks. You monitor what it is doing through its display windows. You issue commands by moving the highlight or by using the quick commands. We'll start with some of Flashbak's basic procedures.

Getting Started

Write enable your tape (insert the write enable ring for a 9-track tape or for the 3480 tape cartridge, make sure the white dot on the write enable wheel is not showing), load the tape, and make sure the drive is ONLINE.

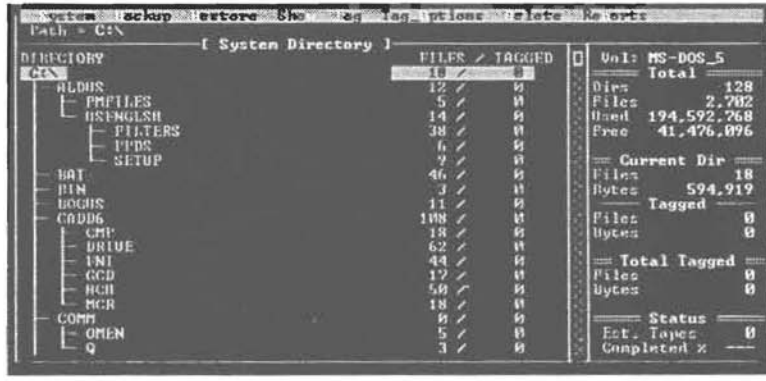
Warning! The tutorial writes to tape, *destroying* any data already on tape. Use a tape that does not hold important data!

Boot DOS and start Flashbak. This can be done in two ways. The preferred method is to select DOS BACKUP from Depot4 or you can type Flashbak then **[ENTER]**.



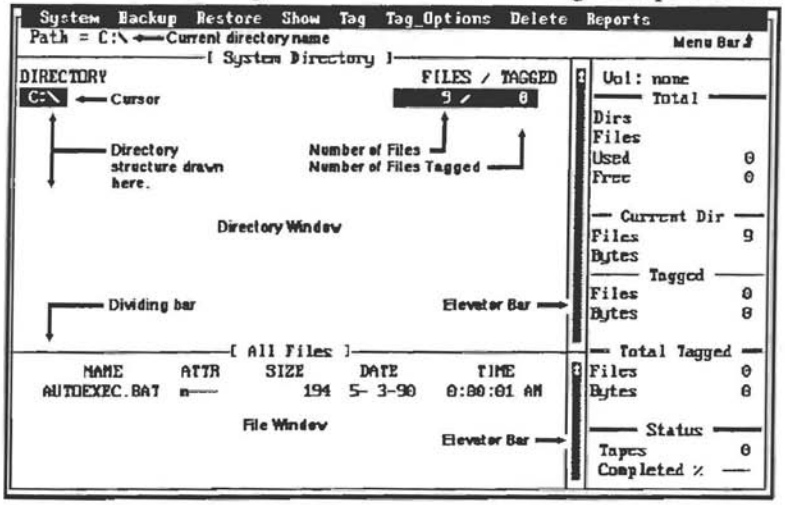
Flashbak reads the active DOS volume (this can take several minutes, particularly with very large directories), reporting statistics such as used and free bytes; number of directories, files and bytes used by the volume; and the number of tagged files. Flashbak displays these in the Information Window. Press any key.

While the directories are loading, Flashbak displays "Reading Directory Information..." in the top line of the screen. Note the status display in the lower right-hand corner. The ODI controller type detected by Flashbak is shown there.



[F6] splits the display, showing the Directory and File Windows. See Modify Config then configure the program to *your* needs.

The following screen shows this, labeling each part of the screen.

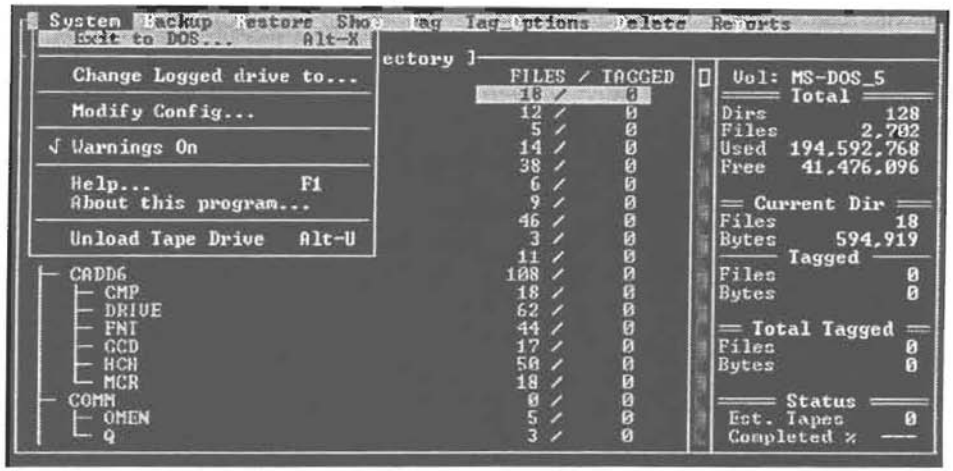


The display shows a menu bar with the available commands: the Path Window with the current directory path name, the Directory Window with the structure of your directory system, the File Window with the file names in this directory and the Information Window with statistics on your disk volume.

There is an elevator bar for both the Directory Window and File Window, indicating how far through the file system the currently highlighted item is. Press **[F4]** to remove the split screen.

Using Flashbak's Menus

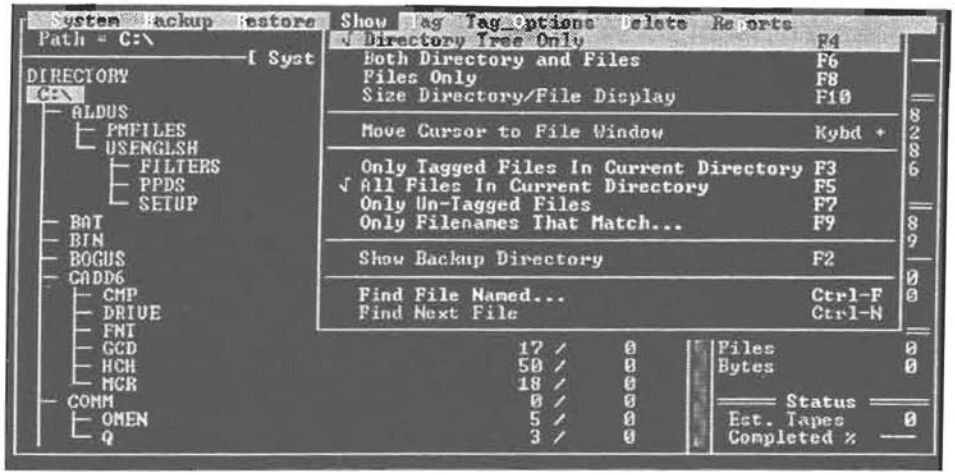
Press **[INS]** to open Flashbak's last active menu. Here, **[INS]** activates the System menu. You will see the following menu under the highlighted menu title, System:



Press **[F1]** at *any* time for Help. Help is context sensitive, so it displays information relative to the currently selected item.

To exit Flashbak you would press **[ALT]+X** We call such key commands "Quick Commands," since you may use them at any time.

Press **[➔]** to move to the next menu. The cursor will move off the right side and reappear on the left side of the menu, so Flashbak will return to the System Menu. You may select a menu by pressing **[ALT]** and the highlighted letter of the menu name. For example, press **[ALT]+W** to get the Show menu:



Note the items shown with a check (✓), meaning these options are the current settings. The highlight bar indicates the cursor position in the menu. Press ⇨ or ⇨ to highlight other items. Type **[ENTER]** to execute the highlighted item. Press the highlighted letter of a menu item to execute it directly (i.e., "o" for Only Tagged Files Current Directory. Use **[ESC]** to back out of a menu.

The quick commands allow you to use commands without using the menus, (e.g., **[F6]** for Both Directory and Files).

Here are guidelines for using Flashbak's menus:

- Press **[INS]** to open the most recently accessed menu.
- Press **[ALT]** and the highlighted letter of a menu name to open it (e.g., **[ALT]+S** for the System menu). The highlight bar shows the selected item. Flashbak marks current settings with a "_".
- Use **[➔]** and **[←]** (while a menu is open) to open other menus. Use ⇨ or ⇨ to highlight items within the open menu.
- Press **[ENTER]** to execute the highlighted item. Use the quick key to execute an item at *any* time.

- Press **[ESC]** to exit a menu without executing any commands.

Some menu items are not currently selectable (not shown bold), as they don't make sense based on the status of the program (e.g., backing up only tagged files if no files are tagged. Flashbak does not allow you to select this menu item until you tag at least one file.

Looking At Your Disk

Flashbak uses four main windows to display information: the Information Window shows the used and free bytes on disk, the number of directories, and the number of files. It also shows the files and bytes in the current directory and the number of files and bytes tagged in the current directory. This window also shows the total number of tagged files and bytes in all directories, and the number of tapes needed to back up all tagged files.

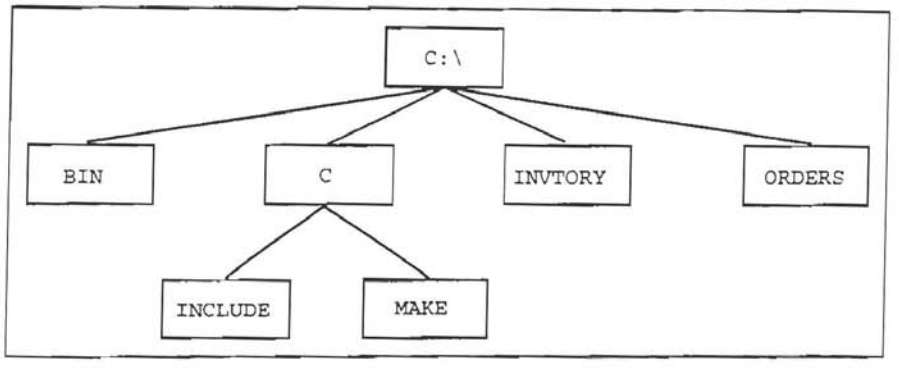
The Path Window shows the current directory's path. We will see shortly how to change the current directory.

The Directory Window displays your disk's (or tape's) directory structure. Indenting shows the relative depth of each directory in your file system.

The File Window shows the file names in the current directory, and can be "hidden," if desired. We examine the Directory Window in this section.

Use the arrow keys to select the menu item Directory Tree Only and press **[ENTER]** (or type **[F4]**, the quick command).

Flashbak will display your disk's directory structure in the Directory Window. Suppose, for example, that your file system consists of the following directory hierarchy:

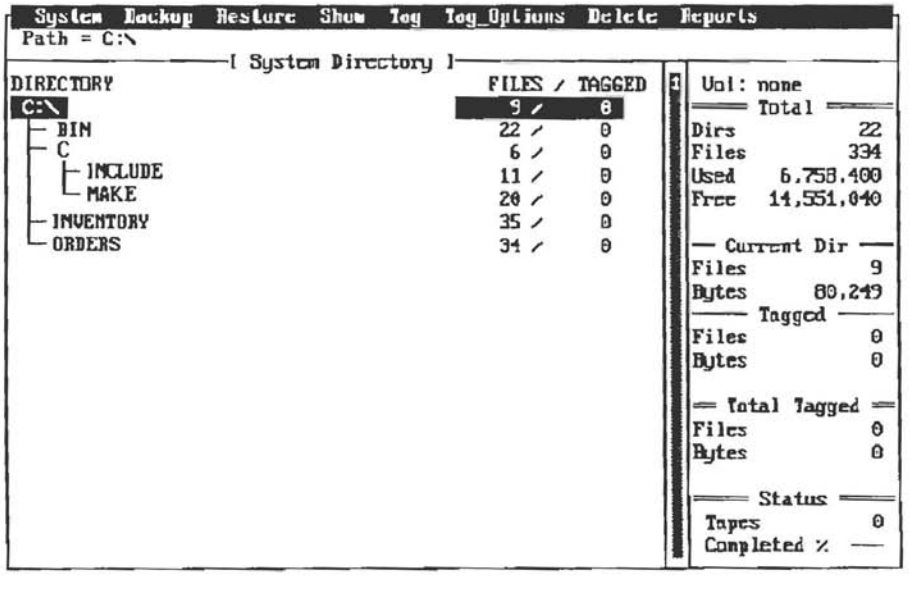


C:\ is the top level or root directory.

Directories BIN, INVTORY and ORDERS appear at the next level.

Directories INCLUDE and MAKE are below directory C (INCLUDE and MAKE are subdirectories of C; C is a subdirectory of the root.)

There are no subdirectories in BIN, INVTORY, ORDERS, INCLUDE or MAKE. Flashbak's Directory Window shows the hierarchy:



The left indentation marks the relative depth of each directory. Here, directories BIN, INVTORY and ORDERS are all at the same level (they're all directly under C:\, the root), while directories INCLUDE and MAKE are at a lower level directly under C.

The Directory Window's right side shows the number of files in each directory, and the number of files tagged the currently highlighted directory.

We will see later how to tag files. Currently, none of the files are tagged. Flashbak therefore displays 0 opposite each directory and under Total Tagged Files and Total Tagged Bytes.

Moving Within the Directory Window

In our example, the directory structure fits in the Directory Window. Yours may not. Flashbak provides cursor commands to move within the Directory Window (these commands apply to moving within the File Window also).

Press ⇩ to move down the file system display. Flashbak highlights each directory in turn. When you reach the bottom of the screen, Flashbak scrolls the display a line at a time.

Press ⇧ to move the cursor back up the Directory Window display. When you reach the top of the display, Flashbak scrolls the display up a line at a time.

Press **[HOME]** to move the cursor to the first entry of your file system. Press **[END]** to place the cursor at the last entry of your file system. Use **[PGUP]** and **[PGDN]** to page the Directory Window. Press **[CTRL]+[PGUP]** to move the cursor to the top line of the display and **[CTRL]+[PGDN]** to move the cursor to the bottom line of the display.

Note that Flashbak updates the Path Window with the path name of the current directory (the current highlighted directory).

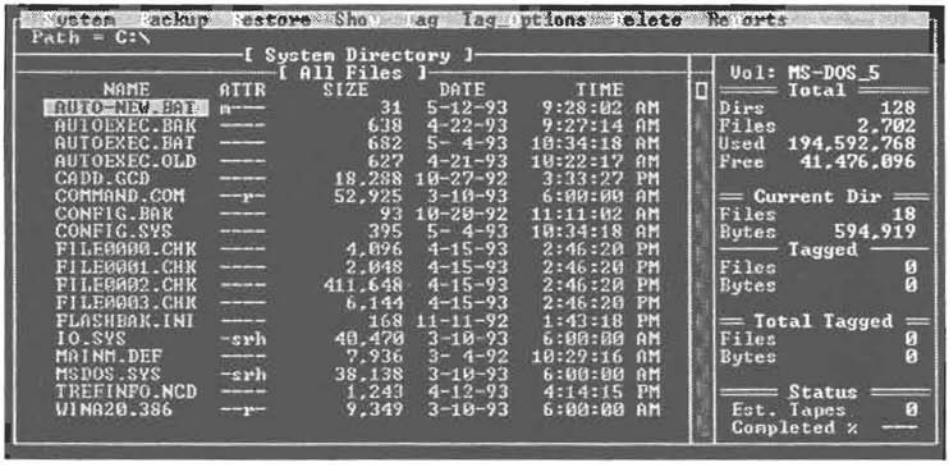
The elevator bar marks the position of the cursor relative to the total number of directories in your file system. Thus, you can easily see how far through the file system you are.

Looking at Your Directories

So far, we have viewed the directory structure of the disk, but have ignored the files within those directories. Now, use Flashbak to examine a directory's contents (in Flashbak, looking at the Directory Window display). Put the Directory Window cursor at the top of the hierarchy (press **[HOME]**). Type **[ALT]**+W to activate the Show menu. We will now use options under Show to change Flashbak's display.

Use the arrow keys to select the menu item Files Only and press **[ENTER]**.

Or use the quick key for Files Only (press **[F8]**). The File Window looks like this:



Flashbak displays the file names of the current directory. This is the "File Window." Each line contains the file name, its attributes, size in bytes, and date and time of last modification.

The attribute field describes up to four characteristics, maintained by DOS: m for modified, s for system, h for hidden, and r for read only status. If configured to do so (see Modify Config in the System Menu), Flashbak clears the modified bit after backing up a file.

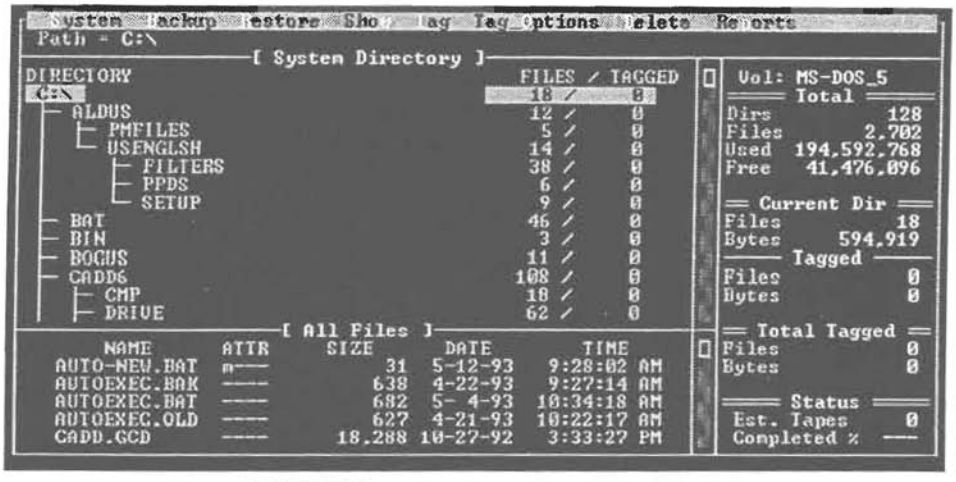
The Path Window shows the path name of the current directory. Move within the File Window using the same cursor control commands. Type ⇦ or ⇨ to move through the list of files. Use **[HOME]**, **[END]**, **[PGUP]**, **[PGDN]**, **[CTRL]+[PGUP]** and **[CTRL]+[PGDN]** to control the display. The elevator bar reflects the relative position of the file within the file list.

Press **[F4]**, taking Flashbak to the Directory Window. Choose another directory ([↓], **[F8]**). You can now view the contents of this directory. Return to the Directory Window **[F4]**, move the cursor to another directory, and call the File Window **[F8]**.

Splitting the Display

Activate the File Window display **[F8]**. Activate the File Window and Directory Window **[F6]**. Flashbak splits the main display window, placing the Directory Window at the top and the File Window at the bottom. Since you entered from the File Window display, Flashbak makes the File Window active (using the cursor commands within the File Window).

In the Directory Window, Flashbak marks the current directory and has an elevator bar for this window *and* the File Window, and highlights the current file in the File Window.



Press **[ENTER]** (or the "-" on the "calculator" number pad on the right of your keyboard) to activate the Directory Window, moving the highlight cursor to the directory window.

Note: Do not use the "-" on the upper row of your keyboard.

Use the cursor control commands to move to different directories. Each directory's file list is displayed in the File Window as you move, and the current directory statistics in the Information Window and Path Window are updated.

Return to the File Window (**[ENTER]** or the "+" on the "calculator" number pad on the right of your keyboard—this toggles between the Directory Window and File Window).

Sizing the Split Window

Press **[ALT]+W** to open the Show menu. Select Size Directory/File Display and press **[ENTER]** (or, just press **[F10]**). This highlights the bar separating the windows. Change the size of the windows by moving the bar up [↑] or down [↓]. Use **[HOME]** and **[END]** to move the bar to the upper or lower bound. When you have sized the screen, press **[ENTER]** (or **[ESC]** to cancel the changes).

This is useful when the number of files is large compared to the directory display, or vice versa. Flashbak prevents you from sizing the window when the window isn't split.

What Is a Tag?

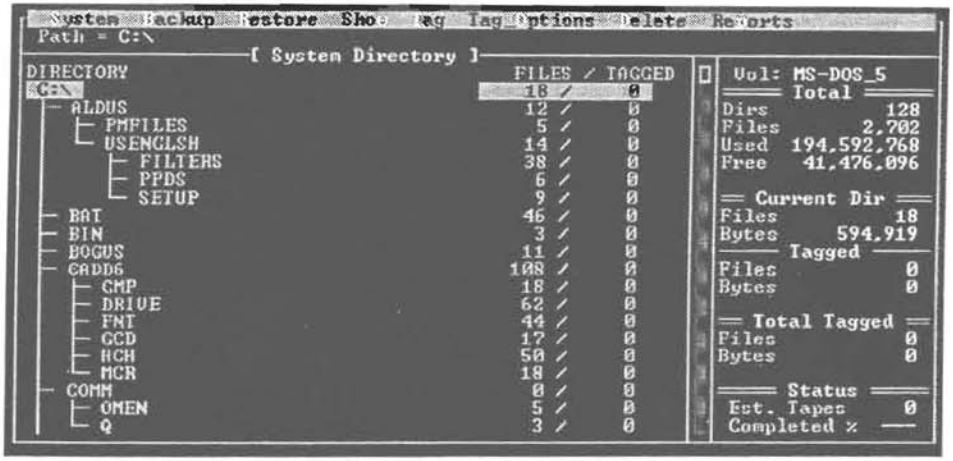
A tag lets you select a file, a group of files, a directory, or all files in a subtree. Flashbak uses tags to mark files—thus, many display options show whether a file (or group of files) is tagged. Files can be tagged manually, or by using the Tag and Tag_Options menu commands you can tag/untag files which meet a set of criteria. Tagging is an important concept. Tags specify which files Flashbak will perform an operation on.

A set of tags are normally only for the current session, but you can save them in a "Specification File" for later or repeated use. Refer to *Using Specification Files*.

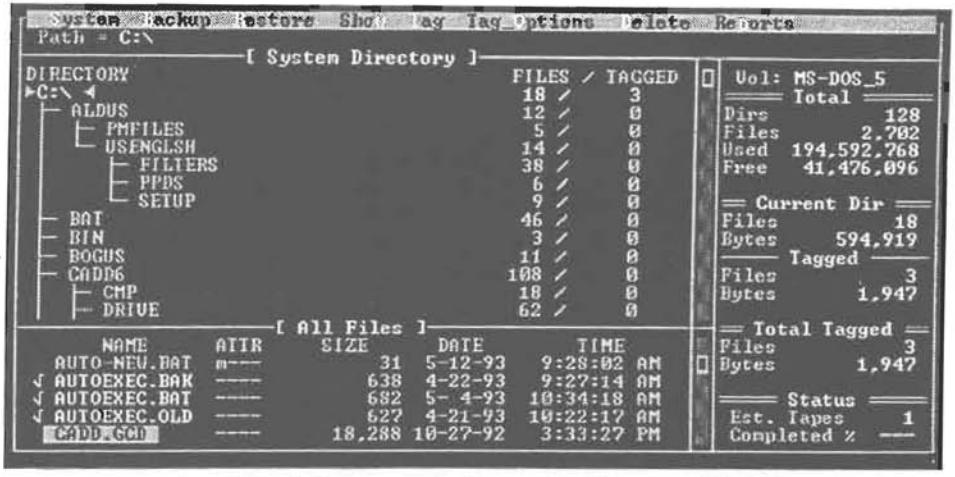
You can view the currently active tag options by pressing **[CTRL]+O**.

You can tell Flashbak to display tagged/untagged files, or use the Reports command to make reports of tagged/untagged files.

There are several ways to tag files. Later, we'll create backups of only tagged files. Enter the Directory Window **[F4]** and move to the top (**[HOME]**). The display looks something like this:



Press **[SPACE]** to tag all files in this directory. **[SPACE]** and **[ALT]+T** do not normally tag root directory "hidden" files. After you press **[SPACE]**, a number appears in the Tagged column opposite C:\ in the directory window. The Information Window indicates the number of files (and total bytes) tagged. Press ⇨ or **[HOME]** to move the cursor back to the top directory. Enter the File Window (press **[F6]** and **[ENTER]**):



A marker (✓) appears in front of each tagged file. Note that Flashbak has highlighted the root directory name in the Directory Window (indicating at least one tagged file in the directory). Press **[ENTER]** to return to the Directory Window and untag all files in the current directory (press **[SPACE]**). The markers disappear and the number of tagged files returns to zero (press , to view the top level directory in the Files Window).

Tagging With Subdirectories On

Press **[ALT]+O** to open the Tag_Options menu. Type the command associated with Tag Sub-Directories (**[CTRL]+S**). Flashbak displays a highlighted label, Subdirs On, at the bottom left of your screen. This tells Flashbak to search embedded subdirectories during most tagging or untagging operations (**[SPACE]** limits tagging to the current directory).

Be sure you are still at the top level of your directory hierarchy. Press **[CTRL]+T**. Flashbak tags all (non-hidden) files in this directory, and with Subdirs On selected, it tags all files in each subdirectory (if there are any). Flashbak updates the tagged statistics. Remove all the tags (**[CTRL]+[U]**). Move in the Directory Window and use **[CTRL]+T** and **[CTRL]** +[U] from several places in the file system.

Note that Flashbak tags files within the highlighted directory and all subdirectories below it. If we start from a subdirectory, the root directory and other subdirectories not below the highlighted directory are unaffected.

Tagging By File Name

Press **[ALT]+O**. Highlight Filenames That Match and press **[ENTER]** or press F (the highlighted letter). A dialog box appears.

Use the arrow keys to move the highlight bar. Select between the two options (That Match: or All Files) by pressing the highlighted letter, or by pressing **[SPACE]** when the highlight bar is *on* that option.

If you choose the That Match: option, the text entry box will be highlighted, and the cursor will move there. Enter the appropriate text—you can use standard DOS wild card characters—and press **[ENTER]**.

DOS Wild card Characters

*	Matches any file name part. *.* matches *any* file name (any name with any extension), *.COM matches any file name with extension .COM, FEB.* matches any file name with name FEB. and any extension.
?	Matches any single character. So FEB?.* matches any file name that starts with FEB followed by one character with any extension, and F??.DAT matches any three character file name that starts with F and has extension .DAT.

When Flashbak asks you to enter text, it provides editing functions. For example, press **[INS]** to toggle between Insert Mode (new characters are inserted) and Type Over Mode (new characters replace previous characters). **[HOME]** moves the cursor to the beginning of the string, **[DEL]** deletes the current character, and **[BACKSPACE]** deletes the previous character. Flashbak scrolls the string within the editing window as necessary.

Here, specify all files with a .COM extension by selecting That Match, and entering *.COM; then press **[ENTER]**.

You have created a tag specification. To tag the files that conform to your specification takes one more step.

Return to the top level of the directory hierarchy and tag files using the file matching option. From the top level, press **[CTRL]+T** (Subdirs On should still appear at the bottom of the screen). Flashbak now tags all files with extension .COM.(If Subdirs On still shows in on your screen, then all .COM files in the root directory and all its subdirectories are tagged.)

This shows an important feature of pressing **[CTRL]+T**. Flashbak tags files using options set in the Tag_Options menu. If none are set, **[CTRL]+T** tags all files in the current directory.

While these files are tagged, examine the Show menu again. Press **[ALT]+W**. We are interested in commands **[F3]** (Only Tagged Files in Current Directory) and **[F7]** (Only UnTagged Files). Press **[F3]** so Flashbak displays only tagged files and **[F6]** for a split display. Move around the file system viewing tagged files.

When you enter a directory that has no tagged files in it (its name appears in normal typeface), Flashbak reports:

No Tagged Files In This Directory.

Press **[F7]**; Flashbak displays *untagged* files. Move around the file system and enter a directory with some untagged files.

Enter the File Window and press **[SPACE]** to tag the current file. It should disappear from the display! When displaying untagged files only, tagging a file causes Flashbak to remove the file name from the display. Flashbak reports:

No Un-Tagged Files In This Directory

when you enter a directory with all tagged files.

Tagging By Modification Date

Remove the file matching specification option. Press **[CTRL]+R** and Flashbak resets all Tag Options (except Subdirs).

Type **[ALT]+O** to open the Tag_Options menu. Select Files Dated On or After... A menu appears and Flashbak prompts for a date. Delete the default date by backspacing. Enter the date: month-day-year (the month is from 01 to 12, day is from 01 and 31, and year is between 80 and 99) and press **[ENTER]**.

Specify date: 06-25-91. Return to the top level file system. Press **[CTRL]+S** to enable Subdirs. To tag files with the selected options, press **[CTRL]+T**. Flashbak tags all files modified since 06-25-91.

Tagging by modification date lets you to tell Flashbak to choose only recently modified files. For example, use this tag option to create daily backups of only the files that are changed. This is typically a small percentage of the files in the file system.

Backing Up Your Entire Disk

We will now write to tape. Press **[ALT]+B** to open Backup's menu. To back up the entire disk, press the quick command, **[CTRL]+B**.

A new menu asks you to load the first tape. Press **[ENTER]** when the drive is ready.

Warning! Make sure the tape does not contain valuable data, since Flashbak overwrites any data previously on the tape.

Flashbak displays information about the number of tapes, files and bytes necessary to back up the disk. Flashbak monitors the backup progress by updating the Status box in the Information Window. If it's necessary to change tapes, Flashbak asks for a new tape. Press **[ENTER]** ready.

Press **[ESC]** any time to cancel the backup while it is in progress.

Using Multiple Logical Drives

To back up multiple logical drives, use the DOS command join in your autoexec.bat. Flashbak treats the drives as subdirectories.

Backing Up a Subtree

With the tagging system, you can easily create a tape backup of a directory subtree. Reset all tag options (**[CTRL]+R**). If not already enabled, press **[CTRL]+S** to enable option Subdirs On. Activate the Directory Window **[F4]** and move the cursor to the top level of the subdirectory. Press **[CTRL]+T** to tag all files in the current directory. **[SPACE]** will only tag the current directory.

To create the backup;, enter the Backup menu (press **[ALT]+B**) and select Only Tagged Files. Press **[ENTER]**. Flashbak asks you to load the first tape. Flashbak reports the number of files, bytes and tapes required to create the backup.

Backing Up By the DOS Modified Attribute

DOS maintains a *modified* attribute that is set each time a file changes (m appears under the ATTR heading when set). If you set the Clear Modified Bit option in the Modify Config... screen under the System Menu, Flashbak will clear this modified bit *after* successfully backing up a file. This bit will thus only be set if the file has been modified since it was last backed up. Then, the backup should just back up those files that have the modified bit set. Follow these steps to implement this backup procedure. First, change the default configuration so that Flashbak will clear the modify bit automatically after a backup:

Open the System Menu and select item Modify Config... Move the cursor to Clear Modified Bit and choose Yes.

Open the System menu and select Save Config. Flashbak writes the modified configuration options to FLASHBAK.INI so that each time you load Flashbak, it loads the new configuration .

Back up the file system (clearing the modified bit on each file).

Press **[CTRL]+B** to back up your entire file system to tape (for the tutorial, you might want to simply take this as a fact if your disk as very large, as the backup can take a significant amount of time, or even take multiple tapes).. When Flashbak is finished, it clears the modified bit on every file. Now, for subsequent backups, follow this procedure:

Open the Tag_Options Menu and select Modified Files...

Choose option On.

Press **[CTRL]+S** (if necessary) to enable subdirectory searching.

Press **[F4]** and **[HOME]** to go to the top level directory.

Press **[CTRL]+T** to tag all files according to options (Flashbak will only tag those files with the modified bit set).

Open the Backup Menu and choose Only Tagged Files. Again, Flashbak clears the modified bit after the backup is complete (because we modified the default configuration earlier).

Backing Up Selected Files

You can choose any file for backup by tagging it. Sometimes a group of files cannot be characterized by the tag options. In this case, you need to tag the files "by hand," as follows:

Press **[F6]** to get the split Directory/File Window. Press **[F5]** to show all files in the current directory. Use cursor commands to peruse the directory structure, scanning the File Window for target files. If a file you want to back up appears, press **[ENTER]** to enter the File Window. Use cursor commands to select the file. Tag it by pressing **[SPACE]**. **[SPACE]** toggles file tagging (tag/untag). Continue scanning the file list, tagging files as needed.

Press **[ENTER]** to go to the Directory Window to select other directories. If at any time you want to tag all files in the current directory, simply press **[SPACE]** while the directory is highlighted in the Directory Window, and that window is active.

Pressing **[SPACE]** while in the Directory Window tags all files in the current directory, ignoring any options set (except hidden files in the root directory). It never tags files in subdirectories, even if Subdirs On is enabled. If all files in the current directory are *already* tagged, pressing **[SPACE]** *untags* them. The cursor moves to the next directory. Once you have tagged the necessary files, enter the Backup menu, select the item Only Tagged Files, and press **[ENTER]** to create the backup.

Examining Backup

Flashbak lets you examine the contents of a backup tape using the same commands used to examine a system volume. First we must tell Flashbak to read the file information from the tape.

Press **[CTRL]+L** (Load Backup Directory). Flashbak reads the tape, loads directory information and updates the Information Window (i.e., after loading a tape of our example file system, the Directory Window (labeled Backup Directory) appears).

Note that no files are tagged. Use the options in the Show menu to manipulate the Directory and File Windows as before (e.g., press **[F4]** to view the directory structure, **[F8]** for the file list and **[F6]** for both directory and file information. Use **[ENTER]** to move between the Directory Window and File Window. Use cursor commands to view the file and directory names in each window).

Restoring from Tape

Once Flashbak loads the backup directory, you can restore its contents. To restore all files, open the Restore menu, select Restore All Files and press **[ENTER]**. This restores all files from the backup device to the disk volume. Flashbak creates directories as needed.

Flashbak replaces any file with the same name as a backup file.

Restoring Selected Files

To restore *some* backup files, first tag them. For example, to restore all files with extension .exe, read in the backup directory (press **[CTRL]+L**). Open Tag_Options (**[CTRL]+O**) and select Filenames That Match.... At the prompt, enter the following file matching specification:

*.exe (and then press **[ENTER]**)

Press **[CTRL]+S** (if necessary) to enable subdirectory searching. Return to the top level of the directory tree (press **[F4]** followed by **[HOME]**). Press **[CTRL]+T** to tag all files using options. This selects the target files.

To restore these files, open the Restore menu and select Restore Only Tagged Files. Press **[ENTER]** and Flashbak restores selected files. You may use the tagging mechanisms described earlier to restore by modification date, to restore a subtree or restore files selected "by hand."

Using Specification Files

Since tags disappear when you exit Flashbak , you may want to save a record of files tagged. Flashbak provides a Specification File to do this. Open the Tag menu, select Save Backup Spec to... and press **[ENTER]**. Specify a path name in response to the prompt. Flashbak writes tagged file names to the Specification File.

To use the Specification File, open the Tag menu, select Tag By Spec File... and press **[ENTER]**. Type the Specification File name at the prompt. Flashbak tags all files listed in the Specification File.

Flashbak does not remove any tags from files not listed in the Specification File. Tagging by Specification File simply adds tags to files listed in the Specification File. This is particularly useful when you tag a number of files individually for regular backup. Using a Specification File you tag the files "by hand" just once. After tagging from a Specification File, you can create the backup by selecting Restore Only Tagged Files from the Backup menu.

You may also write specification files using wild cards and other Flashbak commands to tag files. See *Tag Specification File Format* in this chapter.

Changing the Restore Name

What if you want to restore a file or directory subtree under a new name?

Suppose you have a tape that contains all files in the example file system illustrated in *Looking At Your Disk*. You want to restore the subtree labeled C (including all files and subdirectories INCLUDE and MAKE). To avoid replacing the hard disk versions, you rename the subtree C to OLDC as follows.

Load the Backup Directory (press **[ALT]+L**). The Backup Directory must be loaded into system memory before the Change Restore Name menu item is enabled.

Move the cursor to the file or directory you want to restore (in our example we move C). Press **[CTRL]+S** to enable subdirectory tagging (if necessary) and tag the subtree with **[CTRL]+T**.

Open the Restore menu (press **[ALT]+R**) and select item Change Restore Name. Press **[ENTER]**.

A new menu appears with the highlighted directory name in the Edit Box. Press **[HOME]** and type in the new name (in our example, we type OLDC and **[ENTER]**).

Restore the subtree by selecting item Only Tagged Files from the Restore Menu. The subtree renamed OLDC is restored under its new directory name on the disk.

Flashbak supports (using a batch file) unattended backup or restore through a set of command line switches. When used, normal user prompts are bypassed and the operation begins when the program is initialized. The drive *must* be on line, with the appropriate tape mounted, for the automated backup/restore to complete without assistance.

This table lists the valid command line switches for Flashbak:

Switch	Function Description
/B	Back up all files to tape
/R	Restore all files from tape
/BN specfile	backup only files specified in specfile
/RN specfile	Restore only files specified in specfile
/L	Create log of files backed up or restored

In the list above, a "-" may be used to replace the "/".

If the /L option is selected, it *must* precede any of the other listed command line switches.

Note: Automated operation is limited to what will fit on one 9-track tape since operator assistance is needed for tape changes. For 3480s with stackers, the limit is the number of cartridges in the stacker. In essence, if the backup is so large that a new tape, or more cartridges than are in your cartridge magazine are needed, automated backup becomes impossible. Using specification files, selective backups can be made.

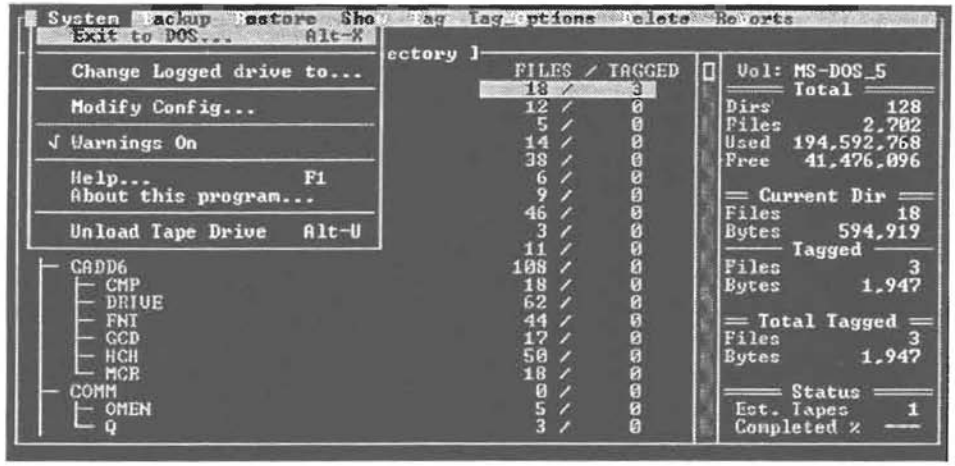
Command Reference

This section is a reference manual of Flashbak's commands. For each command menu we state its function, display its menu, and give a detailed description of its operation.

In addition, a Quick Command Reference is included for easy reference. We also provide separate tables describing the commands to control Flashbak's windows (Display Control Commands), to select menus and execute commands (Menu Selection and Command Execution), to move menus (Moving Menus), and to enter and edit text (Entering Text).

System Menu

Invoke the System menu to perform general system-level tasks such as changing the logged drive, modifying or saving a configuration file, controlling the display of warning prompts, help, or quitting the program.



Exit to DOS ([ALT]+X)

Quit Flashbak. Flashbak does not save tags.

Flashbak displays a menu listing the drive selections. Move the cursor to the appropriate menu item and press **[ENTER]**. Flashbak then loads the directory structure from the selected drive.

Change Logged Drive To..

Flashbak displays a menu listing the drive selections. Move the cursor to the appropriate menu item and press [ENTER]. Flashbak then loads the directory structure from the selected drive.

Modify Config...

Flashbak provides a menu to modify various configuration parameters. For example, changing the hardware address makes it necessary to reconfigure the support software.

Note: **Flashbak does not remote select drive speed or drive density. You *must* set the options in this window to *match* your drive's settings!**

The Flashbak Configuration Menu consists of the following options (the defaults are shown):

(The following are used only for estimating tape length. Because Flashbak can use multiple tapes, the figures will be *estimates*.)

Tape Capacity:

These are for your convenience. They allow you to enter information about your tape, and from this, Flashbak will calculate the *Estimated Tapes*, as shown in the main screen (lower right corner). The purpose is to let you determine whether you are likely to need more than one tape to complete your backup. If you are using 9-track, or a 3480 drive without a cartridge loader, and the estimated tapes figure is more than one, you will not be able to perform an unattended backup (someone will need to change the tape when Flashbak calls for it). If you are using a 3480 drive with a cartridge loader, then the estimated number of tapes simply tells you how many cartridges need to be in the magazine to allow an unattended backup. Flashbak automatically will use all of the 3480 cartridges that are loaded into a stacker's magazine (or as many as are needed).

Standard Size:

Enter the length of your tape(s) and the density you will be using. Flashbak will automatically calculate the number of tapes needed. Note that this is only useful for 9-track tapes.

In Megabytes:

Enter the size of your tape in megabytes (useful particularly for 3480 cartridges—enter 200 MB for standard length cartridges). Flashbak will automatically calculate the number of cartridges needed.

Screen Stuff:

Screen Snows:

If enabled, prevents CGA monitors (only) from snowing.

Don't Use Color:

If you have a color monitor and you don't want Flashbak to display in color, set this option. You must save the configuration changes, exit Flashbak, and re-enter Flashbak for this item to take effect.

Alt Key Menus:

Disabling this option converts Quick Key Commands to the style used by older versions of Flashbak. In general, that means that the use of the **[ALT]** and **[CTRL]** keys is reversed.

I/O Port:

Specify the I/O port address (hexadecimal) of the Overland Data controller board. The default (000) enables autoconfiguration. ***Note:*** You must save configuration changes, exit Flashbak, and re-enter Flashbak for changes to take effect. Any new address here must match the ODI controller board's address (see the Installation Manual to determine how to change the address on your controller card).

Drive Speed:

Drive speed *is not* remotely set. This is used to help calculate the (probable) tape capacity.

Draw Reports With:

When Extended IBM is selected, Flashbak draws directory tree reports using the extended IBM character set. Not all printers use these characters. When Standard ASCII is selected, Flashbak uses standard ASCII characters.

Backup Things:

Write CRCs:

When enabled, Flashbak generates CRCs for each file during backup. This allows data integrity verification during a restore, with a slight performance penalty.

If this option is *not* enabled, the Verify CRCs on Tape option can not be selected.

Clear Modified Bit:

When set, Flashbak clears the modified bit on all tagged files *after* it successfully completes a backup. This allows you to backup files which have been modified since your last backup.

Use Changes

Instructs Flashbak to *use* (but not save) configuration settings you have selected. Flashbak will use them for this session only.

Cancel

Cancel escapes the dialog box without accepting any changes made to it for the current or any future session.

Save Changes

Saves the configuration settings and the current Tag Options, to FLASHBAK.INI, *in the current directory*. Flashbak uses these settings and saves them for future use (this is the same as the Save in the System Menu).

IMPORTANT NOTE: **After changing the configuration, you must select Save Changes to make the changes permanent.**

Warnings On ...

This option provides an extra level of protection against losing files through accidental deletion (see the Delete Menu).

Help [F1]

Help invokes context-sensitive help information. You may access other help screens by selecting NEXT or PREVIOUS. View Help's Table of Contents by selecting CONTENTS.

About this program...

This command displays certain trivia regarding Flashbak's humble beginnings.

Unload Tape Drive ([ALT]+U)

Flashbak sends a rewind and unload command to the drive.

Backup Menu

Backup creates tape backups of the disk file system.

All Files ([CTRL]+B)

Pressing **[CTRL]+B** creates a backup (all files in the file system) to tape. Flashbak displays how many files and bytes will be copied, and the number of tapes needed. Flashbak prompts for a new tape when needed. Type **[ENTER]** when the new tape is ready.

Only Tagged Files

This backs up only tagged files (see Tag, Tag_Options menus). Flashbak shows how many files and bytes will be copied, and the number of tapes needed. Flashbak asks for a new tape when needed. Type **[ENTER]** when a new tape is ready. Abort with **[ESC]**. For 3480s with stackers, the next cartridge is used automatically.

Restore Menu

Use Flashbak's Restore command to load the backup directory from tape and restore files.

Load Backup Directory

Before restoring backup files, Flashbak must load the backup directory into memory. You may then use the Show command to scan the directory tree and the file lists in any of the directories.

Restore All Files

Flashbak restores all files from the backup to the logged disk.

Restore Only Tagged Files

Flashbak restores the tagged files from the backup directory to the logged disk (or alternate restore path).

Verify All Files

This verifies all files, comparing the backup tape to the original files. You are informed of any errors.

Verify Only Tagged Files

This verifies all tagged files, comparing the backup tape to the original files. You will be informed of any errors.

Verify CRCs on Tape

It compares the files on tape to checksums created at the time of backup, and tells you if the files are good (the checksum matches the new checksum). You will be informed of any errors.

The Write CRCs in the Modify Config screen must be enabled for this option to work.

Change Restore Name...

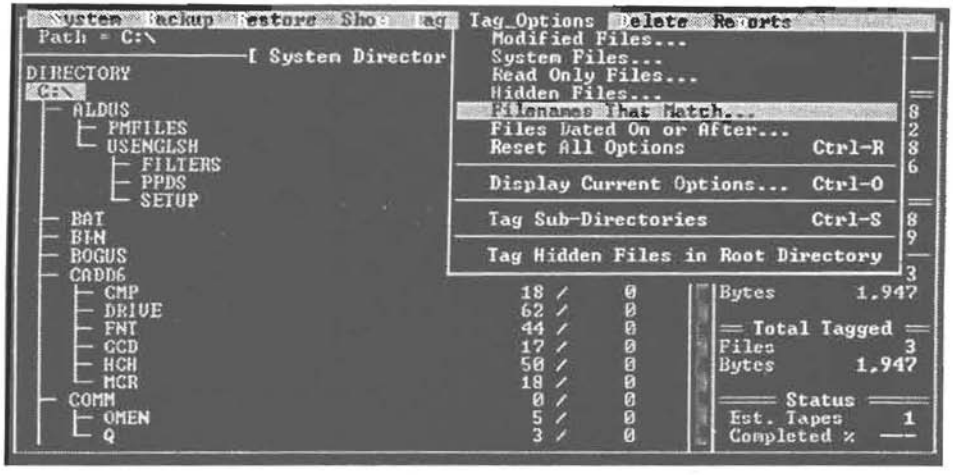
Rename a directory or file from the Backup Directory display. The Backup Directory must be in system memory (press **[CTRL]+L**). At least one file from the Backup Directory must be tagged. Move the cursor to the directory or file to be renamed, open the Restore menu, and select Change Restore Name.

Restoring this file or directory will copy the subtree or file with its new name to disk.

You can abort with **[ESC]**. Flashbak quits the restore at the end of the current file.

Show Menu

Execute Show to choose the display mode: Directory Window, File Window, or split display. Size the split window or specify which files to show (all, tagged, untagged, or by file matching).



Directory Tree Only [F4]

Flashbak displays the Directory Window only. Indented directory names are subdirectories. Highlight indicates that at least one file in the directory is tagged. Use the cursor commands to scan the display, or **[ENTER]** to toggle to and from Files Only **[F8]**.

Both Directory and Files [F6]

Split the screen so that Flashbak displays the Directory Window and the File Window. Press "**+**" (on the numeric keypad on the right of your keyboard) to enter the File Window and "-" (numeric keypad again) to enter the Directory Window (or **[ENTER]** to toggle between the two displays).

Files Only [F8]

Display the File Window only. Markers indicate tagged files.

Size Directory / File Display [F10]

Use ⇦ or ⇨ **[PGUP]** and **[PGDN]** to manipulate the relative size of the split File and Directory Window. End with **[ESC]** or **[ENTER]**.

Move Cursor to File Window / Directory Window

([**ENTER**] or the + and - on the "calculator" pad on your keyboard)

Use [**ENTER**] to toggle the cursor between the File Window and the Directory (Tree) Window. Alternatively, press (on the "calculator" pad on your keyboard) the "+" to move the cursor to the File Window, or "-" to move the cursor to the Directory Window.

Only Tagged Files In Current Directory [F3]

Display (in the File Window) the tagged files only. Flashbak prints No Tagged Files in This Directory if there are no tagged files. This option also applies to the Reports Menu.

All Files In Current Directory [F5]

Display (in the File Window) all files (tagged and untagged) in the current directory. This also applies to the Reports Menu.

Only Un-Tagged Files [F7]

Display (in the File Window) the untagged files only. Flashbak prints No Untagged Files in this Directory if all the files are tagged. This option also applies to the Reports Menu.

Only Filenames That Match... [F9]

Flashbak prompts for a file name specification for files to match. Only files that match the specification are displayed in the *File Window*. Use wild card matching as follows: * matches any file name part and ? matches any single character. This option also applies to the Reports Menu.

Show Backup Directory/Show System Directory [F2]

Toggle between the System and Backup Directory display. You must press **[ALT]+L** to load the backup directory before viewing it by pressing **[F2]**.

Find File Named ([ALT]+F)

Flashbak prompts for a file name (using optional wild card matching), finds the first occurrence within the file system, and highlights it in the File Window.

Find Next File ([CTRL]+T)

Flashbak finds the next occurrence of the target file name and highlights it in the File Window.

Tag Menu

Select files for backup and restore operations. See the Tag_Options menu for options.

Tag/Untag Current File

Press **[SPACE]** to toggle the tag/untag status of the current file. The cursor moves *down* to the next file. Press **[SHIFT]+[SPACE]** to toggle the tag/untag status of the current file. The cursor automatically moves *up* to the preceding file. You must be in the File Window. Hidden files in the root directory are tagged *only* if the Tag Hidden Files in Root Directory option is selected.

Tag/Untag all Files in the Current Directory [SPACE]

You must be in the Directory Window. Press **[SPACE]** to toggle the tag/untag status of all files in the current directory, *regardless* of any tag options. The cursor moves *down* to the next directory. **[SHIFT]+[SPACE]** also toggles the tag/untag status of all files in the current directory, *regardless* of any tag options, but in this case, the cursor automatically moves *up* to the next directory. Pressing **[CTRL]+[SPACE]** toggles the tag/untag status of all files in the current directory *without* moving the cursor (if pressed twice on a directory with mixed tagged and untagged files, it first toggles any untagged files and then untags all the files). Hidden files in the root directory are tagged only if the Tag Hidden Files in Root Directory option is enabled.

All Files in Current Directory

Tag all files in the highlighted directory, *regardless* of Tag options except Tag Hidden Files in Root Directory and Subdirs On. Root directory hidden files will be tagged if the Tag Hidden Files in Root Directory option is enabled. This option tags files in subdirectories of the directory target if Subdirs On option is enabled.

All Files Using Options

Tag files according to the options set in the Tag_Options menu. If option Subdirs On is enabled, search for files to tag in subdirectories under the current directory also.

Tag by Spec File

Tag files listed in a Tag Specification File. Flashbak prompts for the name of the Tag Specification File.

Un-Tag All Files

Remove tags from files in the current directory, regardless of any Tag options. This option also untags files in subdirectories under the current directory if Subdirs On option is enabled.

Un-Tag All Files Using Options ([CTRL]+U)

Remove tags from files according to the options set under Tag_Options menu. If Subdirs On option is enabled, also remove tags from files in subdirectories below the current directory.

Un-Tag by Spec File ...

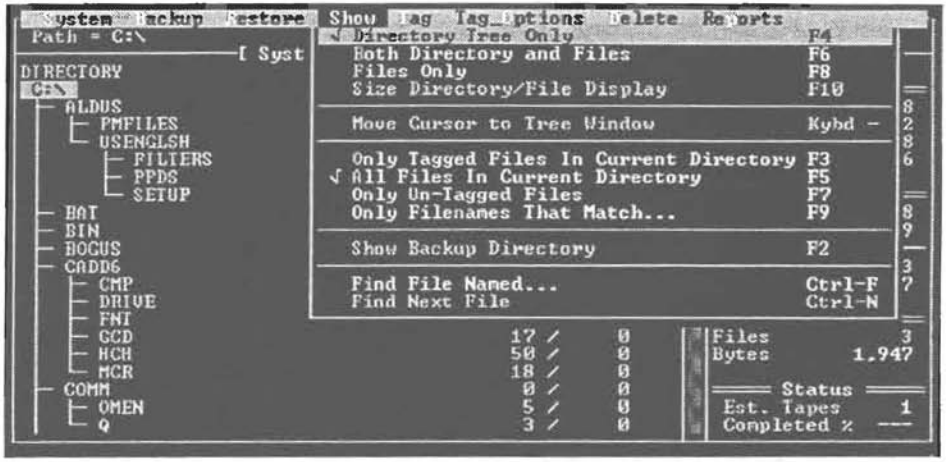
Remove tags from files listed in specified Tag Specification File. Flashbak prompts for the name of the Tag Specification File.

Save Backup Spec to ...

Flashbak writes all tagged file names to a Tag Specification File. Flashbak asks for the name of the Tag Specification File.

Tag_Options Menu

Specify criteria Flashbak will use to tag/untag files. When using the Tag All Files Using Options or Untag All Files Using Options, options include file attributes (system, modified, read only and hidden), file matching, modification date, subdirectory searching and hidden files in the root directory. You can combine options, display the enabled options and easily reset all options.



Modified Files ...

Tags/untags a file only if its modified attribute is set (or cleared).

System Files ...

Tags/untags a file only if its system attribute is set (or cleared).

Read Only Files ...

Tags/untags a file only if its read only attribute is set (or cleared).

Hidden Files ...

Tags/untags a file only if its hidden attribute is set (or cleared).

Filenames That Match ...

Tags/untags a file if its name matches file matching specification: matches any file name part and matches any single character.

Files Dated On or After ...

Sets Flashbak to tag/untag a file only if it was modified on or after the specified date. Specify the date as Month-Day-Year where Month is 01 to 12, Day is 01 to 31, and Year is 80 to 99.

Reset All Options ([CTRL]+R)

Return tagging options (except Subdirs On) to default settings.

Display Current Options ... ([CTRL]+O)

Display the current setting of all options.

Tag Sub-Directories ([CTRL]+S)

Toggle the Subdirs On option. When on, Flashbak searches all subdirectories under the current directory for files to tag or untag.

Tag Hidden Files in Root Directory

Allows **[SPACE]** from the Directory Window and **[CTRL]+T** to tag hidden files in the root directory. Otherwise, the only way to tag these files is to enter the File Window (press **[F8]**) select the hidden file and press **[SPACE]**.

Some copy protected programs put hidden files in the root directory as part of the protection. If these are restored, they will not work!

Delete Menu

Delete individual files, directories, or tagged files. Files may be deleted only from the disk (files on a tape cannot be deleted).

Currently Selected Directory

Delete the directory. It must be empty (it cannot hold any files). If it holds empty subdirectories, they will be deleted also. Flashbak prompts for verification if Warnings On is enabled.

Currently Selected File

Delete the selected file. You must be in the File Window. Flashbak prompts for verification if Warnings On is enabled.

All Tagged Files

Delete tagged files. Flashbak will not remove *any* directories. Flashbak asks for verification if Warnings On is on.

All Tagged Files and Their Directories

Delete all files tagged. If all files are deleted from a directory, Flashbak deletes the directory also. Flashbak prompts for verification before executing this command.

Reports Menu

Make a report (to disk file or printer) of all files, tagged files or untagged files. Use Reports to list the contents of a backup tape.

Print Directory Tree

List all directory names to the default output device. The listing has the same format as the Directory Window, indentation showing directory levels in the tree. This is affected by Draw Reports with |- in the Modify Config Screen (see System Menu).

Print Files Using Display Options

Print files according to options set under the Show Menu: Only Tagged Files **[F3]**, All Files **[F5]**, Only Un-Tagged Files **[F7]**, or Only Filenames That Match... **[F9]**.

Print All Files

Print all files to the default printer device.

Print to File

Change the default output device to the named disk file.

Print to Printer

Change the default output device to the system printer.

Quick Commands

Use the quick commands at any time in a Flashbak session. Certain commands are disabled when their use is inappropriate. For example, it is not possible to size the Directory/File Window **[F10]** when the display window is not split. The full list of Quick Commands is shown in the following tables.

Flashbak Quick Commands:

Description	Command
System	
Exit to DOS	[ALT]+X
Help	[F1]
Unload Tape Drive	[ALT]+U
Backup	
All Files	[CTRL]+B
Restore	
Load Backup Directory	[CTRL]+L
Show	
Directory Tree Only	[F4]
Both Directory and Files	[F6]
Files Only	[F8]
Size Directory/File Display	[F10]
Move Cursor to File Window	*keypad* + or [ENTER]
Move Cursor to Directory (Tree) Window	*keypad* - or [ENTER]
Only Tagged Files in Current Directory	[F3]
All Files in Current Directory	[F5]
Only Un-Tagged Files	[F7]
Only Filenames That Match...	[F9]
Show Backup Directory	[F2]
Show System Directory	[F2]
Find File Named...	[CTRL]+F
Find Next	[CTRL]+N

Flashbak Quick Commands (continued):

Description	Command
Tag Options	
Tag/Untag Current File (then move down)	[SPACE]
Tag/Untag All Files in Current Directory	[SPACE]
Tag All Files Using Options	[CTRL]+T
Un-Tag All Files Using Options	[CTRL]+U
Tag_Options	
Tag/Untag Current File/Directory (move up)	[SHIFT]+[SPACE]
Tag/Untag Current Directory (do not move)	[CTRL]+[SPACE]
Reset all Options	[CTRL]+R
Display Current Options	[CTRL]+O
Tag Sub-Directories	[CTRL]+S

Display Control

These commands move within the Directory or File Window, or to control the display of a file's contents.

Description	Command
Moving the Cursor ...	
Up one line.	[↑]
Down one line.	[↓]
Up one screen.	[PGUP]
Down one screen.	[PGDN]
To the first entry.	[HOME]
To the last entry.	[END]
To the first line on the display.	[CTRL]+[PGUP]
To the last line on the display.	[CTRL]+[PGDN]

Menu Selection and Command Execution

The following cursor commands manipulate Flashbak's menu system:

Description	Command
Activate the menu system	**[INS]**, /
Re-activate main menu from pop-up menu	**[INS]**, /
Open the next menu right	[→]
Open the next menu left	[←]
Open menu starting with highlighted Letter	**[ALT]+Letter**
Select next menu item up	[↑]
Select next menu item down	[↓]
Select next item starting with Letter	Letter (lowercase)
Execute selected menu item	**[ENTER]**
Exit menu without making changes	**[ESC]**

Moving Windows

These commands move a window (if it has move arrows on it).

Description	Command
Enable/Lock menu move commands	**[SCROLL LOCK]**
Moving the Window ...	
Up one line.	[↑]
Down one line.	[↓]
One position right.	[→]
One position left.	[←]
To the top of the screen.	**[PGUP]**
To the bottom of the screen.	**[PGDN]**
To the left of the screen.	**[HOME]**
To the right of the screen.	**[END]**

Entering Text

Use these commands to edit input strings (for example, when typing in filenames, date strings, and so forth).

Description	Command
Toggle insert/typeover (large & small cursor)	[INS]
Delete previous character	[BACKSPACE]
Delete current character	[DEL]
Delete to beginning of string	[CTRL]+[HOME]
Delete to end of string	[CTRL]+[END]
Moving the Cursor ...	
1 character left	[←]
1 character right	[→]
To start of the string	[HOME]
To end of the string	[END]
To left side of window	[CTRL]+[←]
To right side of window	[CTRL]+[→]

Tag Specification File Format

You can create a Tag Specification by executing Save Backup Spec To... from the Tags Menu. Flashbak saves the full pathname of each tagged file, one name per line, as follows:

```
\BIN\FB.EXE
\BIN\BTEST.EXE
\INCLUDE\SYSTEM.H
\INVTORY\JAN.DAT
\INVTORY\FEB.DAT
```

Or you may create your own Specification File by using any text editor that can create plain ASCII files. You may specify files by entering the complete pathname, or you may use the specifiers found under Tag_Options. By including these tag options, you build a command file for Flashbak to tag files. Refer to the Tag_Options menu for details on the individual options. The following table describes Specification File Command Syntax.

Description	Command
Spec File Comment Flashbak ignores the line	#
Ignore the Modified attribute	.MX
Tag files if the Modified attribute is set	.M1
Tag files if the Modified attribute is clear	.M0
Ignore the System attribute	.SX
Tag files if the System attribute is set	.S1
Tag files if the System attribute is clear	.S0
Ignore the Read only attribute	.RX
Tag files if the Read only attribute is set	.R1
Tag files if the Read only attribute is clear	.R0
Ignore the Hidden attribute	.HX
Tag files if the Hidden attribute is set	.H1
Tag files if the Hidden attribute is clear	.H0
Tag files dated on or after	.Ddd-mm-yy
Tag subdirectories	.SD1
Disable subdirectory tagging	.SD0
Enable tagging hidden files in root directory	.HR1
Disable tagging hidden files in root directory	.HR0
Reset options	.RO
Tag matching filenames	<pattern>
Clear modified bit	.CM1
Do not clear modified bit	.CM0

Specification File Command Syntax

<pattern> includes the file matching characters: * (match any file name part) or ? (match any single character).

dd-mm-yy is a date: month-day-year, where month is 01 to 12, date is 01 to 31, and year is 80 to 99.

Case is not significant—RO and ro *both* specify Reset Options.

Controlling Modify Attribute from Tag Specification Files

The modify attribute associated with a file is cleared by including a .CM1 command in a tag specification file. If .CM0 is included in the specification files, the attribute bit will *not* be changed. These commands will only affect operation during Flashbak is active.

Note: If both .CM1 and .CM0 are included in the specification file, the last command found will determine the action taken.

Spec File Example

.SD1

\usr\subdir*.*

Looking at the table on *Specification File Command Syntax* (above), you can see that this spec file will tag all subdirectories (.SD1) and files in \usr\subdir*.*.

Command Line Options

This table lists the valid command line switches for Flashbak:

Switch	Function Description
/B	Back up all files to tape
/R	Restore all files from tape
/BN specfile	Backup only files specified in specfile
/RN specfile	Restore only files specified in specfile
/L	Create log of files backed up or restored

In the list above, a "-" may be used to replace the "/".

If /L is selected, it *must* precede any other command line switch.

XTEST

XTEST tests your integrated tape subsystem. It provides a rapid test of the integrated tape subsystem with Overland Data's controllers. After installing the controller, you should run xtest before attempting to execute any other tape utility. Passing the diagnostics indicates that your system is correctly installed and the hardware is working properly.

Features

XTEST tests the following controller and tape functions:

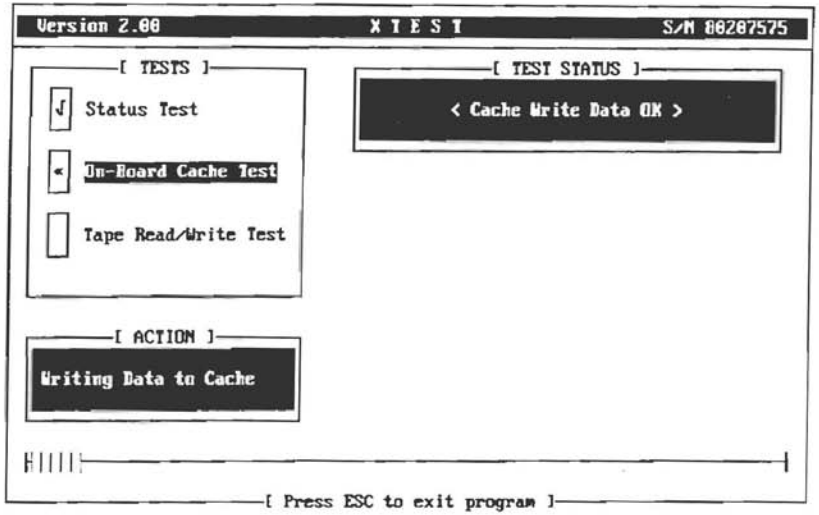
- Status Test
- Onboard Cache Test (TX-8 and XL/2 only)
- Tape Read/Write Test

Use

Find a scratch tape, insert a write enable ring in the back of the tape reel, and load it onto the tape drive. Put the drive ONLINE.

Note: This procedure will write to tape. Make sure that the tape you choose has nothing important written on it.

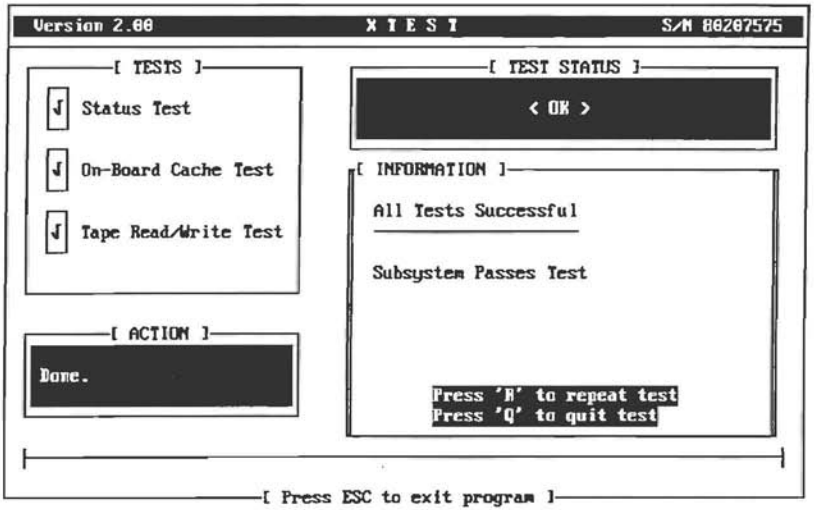
Boot DOS and run XTEST. XTEST displays the following :



XTEST performs the three tests listed in the *Tests Window*. As it completes each test, it places a check mark (√) in the appropriate box. The *Test* Status *Window* displays the status messages of each test. If an error occurs, the *Test Status Window* displays the exact error and the *Information Window* gives you trouble shooting guidelines. Be sure to read this information carefully before calling Overland Data. Sometimes problems with the controller or subsystem are simply incorrect cable installation or improper seating of the board.

The *Action Window* displays the action (for example, reading data, writing data, rewinding tape) of each test.

Upon successful completion of its tests, **XTEST** displays the following screen:


Version 2.00
X T E S T
S/N 80207575
[TESTS]
Status Test
On-Board Cache Test
Tape Read/Write Test
[TEST STATUS]
< OK >
[INFORMATION]
All Tests Successful
Subsystem Passes Test
Press 'R' to repeat test
Press 'Q' to quit test
[ACTION]
Done.
[Press ESC to exit program]

Compress

Overview

This utility provides the capability to activate compression through the interface. It is only available if you have an STK 4220 tape drive equipped with compression. The software activates compression for write operations only. When reading a tape, compression is handled automatically by the tape drive. This utility can be used with any program. *Compress* is also available in the *Depot4* program.

Use

Compression on

Working from your DataTools directory, type **COMPRESS** then **[ENTER]**.

If you have a mouse and your DOS driver is loaded, your mouse pointer will appear. Click the *Use ICRC Data Compression,* check box. An **X** will appear to show that compression has been turned on for all write operations. Choose **OK** to accept this condition.

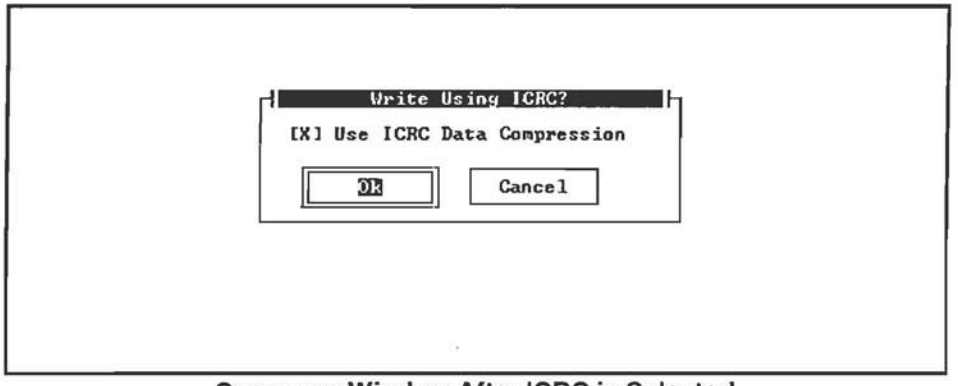
If you do not have a mouse, use your **[TAB]** key to highlight the checkbox then press the **[SPACE]** bar to place an **X** in the box. Highlight **OK**, then press **[ENTER]**.

Compression off

Working from your DataTools directory, type **COMPRESS** then **[ENTER]**.

If you have a mouse and your DOS driver is loaded, your mouse pointer will appear. Make sure the *Use ICRC Data Compression* check box does *not* have an **X** in it. If it does, deselect it then choose **OK** to accept this condition.

If you do not have a mouse, make sure there is *no* **X** in the checkbox. If there is, use your **[TAB]** key to highlight the checkbox then press the **[SPACE]** bar to remove the **X**. Highlight **OK**, then press **[ENTER]**.



Compress Window After ICRC is Selected

Batch File

Compress can also be used in a batch file. The command has three options:

1. **Compress On** - When executed from a batch file or command line will report the current status then turn compression on.

2. **Compress Off** - When executed from a batch file or command line will report the current status then turn compression off .

3. **Compress Report** - When executed from a batch file or the command line will report the current status of the compression mode.

Appendix A

NovaBack: Overland Data Version Differences

This document describes differences between the standard NovaBack product as described in the NovaBack manual (Second Edition April 1993) and the Overland Data NovaBack product you have purchased.

General Information

This release of NovaBack supports the full line of Overland Data controller cards and 9-track and 3480 subsystems. It will not work with SCSI or Pertec host adapters from other manufacturers.

Controllers Supported	Tape Drives Supported
▪ TX-8 ▪ XL/2 ▪ TXi-16 ▪ TMC-ISA ▪ TMC-MCA ▪ Parallel Port	▪ ODI 3201 ▪ ODI 3601 ▪ ODI 3210 ▪ ODI 3610 ▪ ODI 5212 ▪ ODI 5612 ▪ HP 88780 ▪ Cipher 995 ▪ Most other 9-track tape drives ▪ Cipher T480 ▪ STK 4220

The NovaBack manual makes references to QIC (Quarter-Inch Cartridge), 4mm DAT (Digital Audio Tape), and QFA (Quick File Access) tape devices. This version of NovaBack works only with half-inch 9-track and 3480 tape drives attached to the Overland Data controllers listed above.

The NovaBack manual makes references to OS/2 and HPFS. This version of NovaBack does not support OS/2 backup or restore of either FAT or HPFS filesystems.

Installing NovaBack

This Overland Data version of NovaBack does not include a separate install program. Disregard the installation instructions included in Chapter One of the NovaBack manual.

To install the Overland Data version of NovaBack do one of the following:

Note: To perform automatic installation you must have an Overland Data *DataTools* diskette version 1.06 or later. If you do not have this disk, contact Overland Data, or install NovaBack manually as described below.

- **Automatic Installation**

  Insert your Overland Data *DataTools* diskette into your computer's floppy disk drive, make the floppy disk the default drive, and type `install`. (Overland Data DataTools disks This invokes the Overland Data automatic installation program. When prompted to choose the packages you wish to install, choose Network Backup. (You may also choose other listed packages if you wish to install them at this time.) All NovaBack files will be copied onto your hard disk for you automatically.

  -or-

- **Manual Installation** Insert the NovaBack release disk into your computer's floppy drive and type the following:

```
xcopy a:\*.* c:\odi\novaback
```

(This assumes that you are copying files from floppy drive a:) This copies all NovaBack files to the default directory. You may substitute another directory for C:\ODI\NOVABACK if you wish, but you must specify the new directory in your PATH if you intend to launch NovaBack from the DEPOT4 program. (The PATH environment variable is set in your AUTOEXEC.BAT file. See your DOS documentation for more information.)

Testing Your Tape System

The NovaBack manual recommends that you run a program called SCSITEST to verify that your tape drive is working. This program is not included with the Overland Data version of NovaBack. The XTEST program included with the Overland Data software provides a superset of the SCSITEST functions and should be used instead.

Using NovaBack

Cipher T480 Automatic Cartridge Loader

An automatic cartridge loader (stacker) is configured in the Depot4 Interface Options dialog box.

NovaBack Data Compression

Data can only be effectively compressed once, therefore use only one dose of data compression when reading or writing data.

Important: Do not use NovaBack's built-in data compression simultaneously with the ICRC compression included with STK 4220 drives!

If you have the STK 4220 drive with ICRC[1], use ICRC when performing compressed backups. Because ICRC is implemented in hardware, it executes much faster than the NovaStor software-based data compression.

Continuing Over Hard Errors During Restore

By default, NovaBack will halt a restore operation if it encounters a hard (unrecoverable) error. When this happens no data beyond the error can be recovered.

You can change this behavior by setting an environment variable named TAPEHER to the value "SKIP". This is most conveniently done by adding the following line to your AUTOEXEC.BAT file:

SET TAPEHER=SKIP

When configured this way, NovaBack will mark any unrecoverable files as such in the logfile and continue restoring any files it finds beyond the error. We recommend that you set the environment variable to change NovaBack's default behavior.

Manual Errata

The following details places in the NovaBack manual that do not match the Overland Data version of the program.

Chapter One

Page 1 This Overland Data version of NovaBack does not include a separate install program. Disregard the installation instructions included. Instead, see the installation instructions above.

---

[1]The STK 4220 drive is included with the TL3481 and TL3490 subsystems. ICRC is an optional feature of this drive.

Pages 3 - 4 This Overland Data version of NovaBack does not support OS/2. Disregard the OS/2 information presented in the manual.

Chapter Four

Pages 89 - 96 The SCSITEST program is not included with this version of NovaBack. The XTEST program included with the Overland Data software provides a superset of the SCSITEST functions and should be used instead.

Appendix A: Supported Devices

See the table above for devices supported in this release of NovaBack.

Appendix B: DOS Environment Variables

Changing the environment variables listed, TAPEUNIT and TAPEMT will have no effect. In this version of NovaBack there is no reason to change the Tape Unit Address, and the Overland Data controllers will always multitask if possible.

Appendix C: Adaptec Installation (ISA/EISA Bus)

The information in this appendix does not apply to this version of NovaBack.

Appendix D: Adaptec Installation (Micro Channel)

The information in this appendix does not apply to this version of NovaBack.

Appendix B

Controller Specifications

TXi-16

Specifications for the TXi-16 are provided in a separate document. Call Overland Data Technical Support to obtain this information.

TX-8 and XL/2

Input Output Map

Default base port address as shipped (hex):

TX-8 and XL/2 Input Port Map	
Offset From Base Port	**Port Function**
BASE+0	Tape Status Byte A
BASE+1	Tape Status Byte B
BASE+2	Tape/TX Error Status
BASE+3	Tape Read Data
BASE+4	TX Cache Count Read (Low Byte)
BASE+5	TX Cache Count Read (High Byte)
BASE+6	Reserved
BASE+7	Reserved

TX-8 and XL/2 Output Port Map	
Offset From Base Port	**Port Function**
BASE+0	Tape Command Byte
BASE+1	Tape Control Byte
BASE+2	TX Control Byte
BASE+3	TX Extended Control Byte
BASE+4	TX Cache Count Set (Low Byte)
BASE+5	Tape Write Data
BASE+6	TX Cache Count Set (High Byte)
BASE+7	TX Adder Offset Set

TX-8 and XL/2 Status Input Bit Map

Bit	Signal	Latched?
PORT 0—TAPE STATUS A		
0	FBY	no
1	DBY	no
2	RDY	no
3	FPT	no
4	ONL	no
5	SPD	no
6	BOT	no
7	EOT	yes
PORT 1—TAPE STATUS B		
0	RESERVED	
1	IRQ	yes
2	INRZ (low true)	no
3	IRWD (low true)	no
4	IDENT	yes
5	ERROR	yes
6	DBYT	yes
7	FMK	yes
PORT 2—TAPE/TX ERROR STATUS		
0	TOGGLE-Signature	n/a
1	RESERVED	
2	RESERVED	
3	OVERFLOW	yes
4	TAPE parity	yes
5	MEMORY parity	yes
6	CER	yes
7	HER	yes
PORT 3—TAPE READ DATA		
0-7	READ DATA	yes
PORT 4—TX CACHE COUNT LOW		
0-7	CACHE COUNT (Low Byte)	yes
PORT 5—TX CACHE COUNT HIGH		
0-7	CACHE COUNT (High Byte)	yes
PORT 6—RESERVED		
PORT 7—RESERVED		

TX-8 and XL/2 Output Bit Map

BASE+0 (Tape Command Byte):

This port is used to send commands to the tape drive as follows:

BASE+0 : TAPE COMMAND BYTE						
IHSP (D5)	ERASE (D4)	EDIT (D3)	WFM (D2)	WRT (D1)	REV (D0)	Function Initiated
x	0	0	0	0	0	Read 1 block fwd
x	0	0	0	0	1	Read 1 block back
x	0	1	0	0	1	Read reverse edit
x	0	0	0	1	0	Write 1 block fwd
x	0	1	0	1	0	Overwrite
x	0	0	1	1	0	Write file mark
x	1	0	0	1	0	Erase variable block
x	1	0	1	1	0	Erase fixed block
x	1	1	1	1	0	Security erase
x	1	0	0	0	0	Skip fwd 1 block
x	1	0	0	0	1	Skip back 1 block
x	1	1	0	0	0	No operation
x	1	1	1	0	0	Select 1600 bpi
x	1	1	1	0	1	Select 3200 bpi*
0	x	x	x	x	x	Select 25 IPS *
1	x	x	x	x	x	Select 100 IPS *

The data written to the tape command port is latched and a pulse approximately 750 nanoseconds wide is produced and presented to the IGO interface line concurrently with the data. Data bits D6 and D7 are not used.

* These operations are drive dependent. Check the tape drive manufacturer specifications.

BASE+1 (Tape Control Byte):

BASE+1 : TAPE CONTROL BYTE		
0	TAD1	Drive Select LSB.
1	TAD0	Drive Select.
2	FAD	Drive Select MSB.
3	UNLD	Drive Unload.
4	FEN	Clear formatter.
5	RWD	Rewind to BOT .
6	RTH1	Read Threshold 1 (Reserved)
7	RTH2	Read Threshold 2 (Reserved)

BASE+2 (Tape Control Byte):

BASE+2 : BOARD CONTROL BYTE
Proprietary

BASE+3 (Pulse Decodes):

BASE+3 : PULSE DECODES		
	Proprietary	
Value of 2	STATRST	Status Reset

BASE+4 (Load Cache Count, Low Byte):

BASE+4 : LOAD CACHE COUNT (LB)

BASE+5 (Tape Write Data):

BASE+5 : TAPE WRITE DATA

BASE+6 (Load Cache Count, High Byte):

BASE+6 : LOAD CACHE COUNT (HB)

BASE+7 (Adder Input):

BASE+7 : ADDER INPUT
STATRST then LOW byte, HIGH byte

Appendix C

Translation Tables

Control Characters

Control character representations used in these translation tables:

ACK	Acknowledge
BEL	Bell
BS	Backspace
CAN	Cancel
CC	Cursor Control
CR	Carriage Return
CU1	Customer Use 1
CU2	Customer Use 2
CU3	Customer Use 3
DC1	Device Control 1
DC2	Device Control 2
DC4	Device Control 4
DEl	Delete
DLE	Data Link Escape
DS	Digit Select
EM	End of Medium
ENQ	Enquiry
EOT	End of Transmission
ESC	Escape
ETB	End of Transmission Block
ETX	End of Text
FF	Form Feed
FS	Field Separator
HT	Horizontal Tab
IFS	Interchange File Separator
VT	Vertical Tab
IGS	Interchange Group Separator
IL	Idle
IRS	Interchange Record Separator
IUS	InterchangeUnit Separator
LC	Lower Case
LF	Line Feed
NAK	Negative Acknowledge
NL	New Line
NUL	Null
PF	Punch Off
PN	Punch On
RES	Restore
RS	Reader Stop
SI	Shift In
SM	Set Mode
SMM	Start of Manual Message
SO	Shift Out
SOH	Start of Heading
SOS	Start of Significance
SP	Space
STX	Start of Text
SUB	Substitute
SYN	Synchronous Idle
TM	Tape Mark
UC	Upper Case

EBCDIC-TO-ASCII

In	Disp. Char	Out	Disp. Char	In	Disp. Char	Out	Disp. Char
0x00	NUL	0x20	SP	0x20	DS	0x20	SP
0x01	SOH	0x01	SOH	0x21	SOS	0x1b	ESC
0x02	STX	0x02	STX	0x22	FS	0x1c	FS
0x03	ETX	0x03	ETX	0x23		0x20	SP
0x04	PF	0x20	SP	0x24	BYP	0x20	SP
0x05	HT	0x09	HT	0x25	LF	0x0a	LF
0x06	LC	0x20	SP	0x26	ETB	0x17	ETB
0x07	DEL	0x7f	DEL	0x27	ESC	0x1b	ESC
0x08		0x20	SP	0x28		0x20	SP
0x09		0x20	SP	0x29		0x20	SP
0x0a	SMM	0x20	SP	0x2a	SM	0x20	SP
0x0b	VT	0x0b	VT	0x2b	CU2	0x20	SP
0x0c	FF	0x0c	FF	0x2c		0x20	SP
0x0d	CR	0x0d	CR	0x2d	ENQ	0x05	ENQ
0x0e	SO	0x0e	SO	0x2e	ACK	0x06	ACK
0x0f	SI	0x0f	SI	0x2f	BEL	0x07	BEL
0x10	DLE	0x10	DLE	0x30		0x20	SP
0x11	DC1	0x11	DC1	0x31		0x20	SP
0x12	DC2	0x12	DC2	0x32	SYN	0x16	SYN
0x13	TM	0x13	DC3	0x33		0x20	SP
0x14	RES	0x20	SP	0x34	PN	0x20	SP
0x15	NL	0x20	SP	0x35	RS	0x1e	RS
0x16	BS	0x08	BS	0x36	UC	0x20	SP
0x17	IL	0x20	SP	0x37	EOT	0x04	EOT
0x18	CAN	0x18	CAN	0x38		0x20	SP
0x19	EM	0x19	EM	0x39		0x20	SP
0x1a	CC	0x20	SP	0x3a		0x20	SP
0x1b	CU1	0x20	SP	0x3b	CU3	0x20	SP
0x1c	IFS	0x1c	FS	0x3c	DC4	0x14	DC4
0x1d	IGS	0x1d	GS	0x3d	NAK	0x15	NAK
0x1e	IRS	0x1e	RS	0x3e		0x20	SP
0x1f	IUS	0x1f	US	0x3f	SUB	0x1a	SUB

EBCDIC-to-ASCII

In	Disp. Char	Out	Disp. Char	In	Disp. Char	Out	Disp. Char
0x40	SP	0x20	SP	0x60	-	0x2d	-
0x41		0x20	SP	0x61	/	0x2f	/
0x42		0x20	SP	0x62		0x20	SP
0x43		0x20	SP	0x63		0x20	SP
0x44		0x20	SP	0x64		0x20	SP
0x45		0x20	SP	0x65		0x20	SP
0x46		0x20	SP	0x66		0x20	SP
0x47		0x20	SP	0x67		0x20	SP
0x48		0x20	SP	0x68		0x20	SP
0x49		0x20	SP	0x69		0x20	SP
0x4a	¢	0x7e	~	0x6a		0x20	SP
0x4b		0x2e	.	0x6b	,	0x2c	,
0x4c	<	0x3c	<	0x6c	%	0x25	%
0x4d	(	0x28	(	0x6d	_	0x5f	_
0x4e	+	0x2b	+	0x6e	>	0x3e	>
0x4f	|	0x7c	|	0x6f	?	0x3f	?
0x50	&	0x26	&	0x70		0x20	SP
0x51		0x20	SP	0x71		0x20	SP
0x52		0x20	SP	0x72		0x20	SP
0x53		0x20	SP	0x73		0x20	SP
0x54		0x20	SP	0x74		0x20	SP
0x55		0x20	SP	0x75		0x20	SP
0x56		0x20	SP	0x76		0x20	SP
0x57		0x20	SP	0x77		0x20	SP
0x58		0x20	SP	0x78		0x20	SP
0x59		0x20	SP	0x79		0x60	`
0x5a	!	0x21	!	0x7a	:	0x3a	:
0x5b	$	0x24	$	0x7b	#	0x23	#
0x5c	*	0x2a	*	0x7c	@	0x40	@
0x5d	)	0x29	)	0x7d	`	0x27	'
0x5e	;	0x3b	;	0x7e	=	0x3d	=
0x5f	¬	0x7e	~	0x7f	"	0x22	"

EBCDIC-to-ASCII

In	Disp. Char	Out	Disp. Char	In	Disp. Char	Out	Disp. Char
0x80		0x20	SP	0xa0		0x20	SP
0x81	a	0x61	a	0xa1		0x20	SP
0x82	b	0x62	b	0xa2	s	0x73	s
0x83	c	0x63	c	0xa3	t	0x74	t
0x84	d	0x64	d	0xa4	u	0x75	u
0x85	e	0x65	e	0xa5	v	0x76	v
0x86	f	0x66	f	0xa6	w	0x77	w
0x87	g	0x67	g	0xa7	x	0x78	x
0x88	h	0x68	h	0xa8	y	0x79	y
0x89	i	0x69	i	0xa9	z	0x7a	z
0x8a		0x20	SP	0xaa		0x20	SP
0x8b		0x20	SP	0xab		0x20	SP
0x8c		0x20	SP	0xac		0x20	SP
0x8d		0x20	SP	0xad		0x5b	[
0x8e		0x20	SP	0xae		0x20	SP
0x8f		0x20	SP	0xaf		0x20	SP
0x90		0x20	SP	0xb0		0x20	SP
0x91	j	0x6a	j	0xb1		0x20	SP
0x92	k	0x6b	k	0xb2		0x20	SP
0x93	l	0x6c	l	0xb3		0x20	SP
0x94	m	0x6d	m	0xb4		0x20	SP
0x95	n	0x6e	n	0xb5		0x20	SP
0x96	o	0x6f	o	0xb6		0x20	SP
0x97	p	0x70	p	0xb7		0x20	SP
0x98	q	0x71	q	0xb8		0x20	SP
0x99	r	0x72	r	0xb9		0x20	SP
0x9a		0x5e	^	0xba		0x20	SP
0x9b		0x20	SP	0xbb		0x20	SP
0x9c		0x20	SP	0xbc		0x20	SP
0x9d		0x20	SP	0xbd		0x5d	]
0x9e		0x20	SP	0xbe		0x20	SP
0x9f		0x20	SP	0xbf		0x20	SP

EBCDIC-to-ASCII

In	Disp. Char	Out	Disp. Char	In	Disp. Char	Out	Disp. Char
0xc0		0x7b	{	0xe0		0x5c	\
0xc1	A	0x41	A	0xe1		0x20	SP
0xc2	B	0x42	B	0xe2	S	0x53	S
0xc3	C	0x43	C	0xe3	T	0x54	T
0xc4	D	0x44	D	0xe4	U	0x55	U
0xc5	E	0x45	E	0xe5	V	0x56	V
0xc6	F	0x46	F	0xe6	W	0x57	W
0xc7	G	0x47	G	0xe7	X	0x58	X
0xc8	H	0x48	H	0xe8	Y	0x59	Y
0xc9	I	0x49	I	0xe9	Z	0x5a	Z
0xca		0x20	SP	0xea		0x20	SP
0xcb		0x20	SP	0xeb		0x20	SP
0xcc		0x20	SP	0xec		0x20	SP
0xcd		0x20	SP	0xed		0x20	SP
0xce		0x20	SP	0xee		0x20	SP
0xcf		0x20	SP	0xef		0x20	SP
0xd0		0x7d	}	0xf0	0	0x30	0
0xd1	J	0x4a	J	0xf1	1	0x31	1
0xd2	K	0x4b	K	0xf2	2	0x32	2
0xd3	L	0x4c	L	0xf3	3	0x33	3
0xd4	M	0x4d	M	0xf4	4	0x34	4
0xd5	N	0x4e	N	0xf5	5	0x35	5
0xd6	O	0x4f	O	0xf6	6	0x36	6
0xd7	P	0x50	P	0xf7	7	0x37	7
0xd8	Q	0x51	Q	0xf8	8	0x38	8
0xd9	R	0x52	R	0xf9	9	0x39	9
0xda		0x20	SP	0xfa		0x20	SP
0xdb		0x20	SP	0xfb		0x20	SP
0xdc		0x20	SP	0xfc		0x20	SP
0xdd		0x20	SP	0xfd		0x20	SP
0xde		0x20	SP	0xfe		0x20	SP
0xdf		0x20	SP	0xff		0x20	SP

ASCII-TO-EBCDIC

In	Disp. Char	Out	Disp. Char	In	Disp. Char	Out	Disp. Char
0x00	NUL	0x00	NUL	0x20	SP	0x40	SP
0x01	SOH	0x01	•SOH	0x21	!	0x5a	!
0x02	STX	0x02	•STX	0x22	"	0x7f	"
0x03	ETX	0x03	•ETX	0x23	#	0x7b	#
0x04	EOT	0x37	EOT	0x24	$	0x5b	$
0x05	ENQ	0x2d	ENQ	0x25	%	0x6c	%
0x06	ACK	0x2e	ACK	0x26	&	0x50	&
0x07	BEL	0x2f	BEL	0x27	'	0x7d	`
0x08	BS	0x16	BS	0x28	(	0x4d	(
0x09	HT	0x05	HT	0x29	)	0x5d	)
0x0a	LF	0x25	LF	0x2a	*	0x5c	*
0x0b	VT	0x0b	VT	0x2b	+	0x4e	+
0x0c	FF	0x0c	FF	0x2c	,	0x6b	,
0x0d	CR	0x0d	CR	0x2d	-	0x60	-
0x0e	SO	0x0e	SO	0x2e	.	0x4b	
0x0f	SI	0x0f	SI	0x2f	/	0x61	/
0x10	DLE	0x10	DLE	0x30	0	0xf0	0
0x11	DC1	0x11	DC1	0x31	1	0xf1	1
0x12	DC2	0x12	DC2	0x32	2	0xf2	2
0x13	DC3	0x13	TM	0x33	3	0xf3	3
0x14	DC4	0x3c	DC4	0x34	4	0xf4	4
0x15	NAK	0x3d	NAK	0x35	5	0xf5	5
0x16	SYN	0x32	SYN	0x36	6	0xf6	6
0x17	ETB	0x26	ETB	0x37	7	0xf7	7
0x18	CAN	0x18	CAN	0x38	8	0xf8	8
0x19	EM	0x19	EM	0x39	9	0xf9	9
0x1a	SUB	0x3f	SUB	0x3a	:	0x7a	:
0x1b	ESC	0x27	ESC	0x3b	;	0x5e	;
0x1c	FS	0x22	FS	0x3c	<	0x4c	<
0x1d	GS	0x1d	IGS	0x3d	=	0x7e	=
0x1e	RS	0x1e	IRS	0x3e	>	0x6e	>
0x1f	US	0x1f	IUS	0x3f	?	0x6f	?

ASCII-to-EBCDIC

In	Disp. Char	Out	Disp. Char	In	Disp. Char	Out	Disp. Char
0x40	@	0x7c	@	0x60	`	0x7d	`
0x41	A	0xc1	A	0x61	a	0x81	a
0x42	B	0xc2	B	0x62	b	0x82	b
0x43	C	0xc3	C	0x63	c	0x83	c
0x44	D	0xc4	D	0x64	d	0x84	d
0x45	E	0xc5	E	0x65	e	0x85	e
0x46	F	0xc6	F	0x66	f	0x86	f
0x47	G	0xc7	G	0x67	g	0x87	g
0x48	H	0xc8	H	0x68	h	0x88	h
0x49	I	0xc9	I	0x69	i	0x89	i
0x4a	J	0xd1	J	0x6a	j	0x91	j
0x4b	K	0xd2	K	0x6b	k	0x92	k
0x4c	L	0xd3	L	0x6c	l	0x93	l
0x4d	M	0xd4	M	0x6d	m	0x94	m
0x4e	N	0xd5	N	0x6e	n	0x95	n
0x4f	O	0xd6	O	0x6f	o	0x96	o
0x50	P	0xd7	P	0x70	p	0x97	p
0x51	Q	0xd8	Q	0x71	q	0x98	q
0x52	R	0xd9	R	0x72	r	0x99	r
0x53	S	0xe2	S	0x73	s	0xa2	s
0x54	T	0xe3	T	0x74	t	0xa3	t
0x55	U	0xe4	U	0x75	u	0xa4	u
0x56	V	0xe5	V	0x76	v	0xa5	v
0x57	W	0xe6	W	0x77	w	0xa6	w
0x58	X	0xe7	X	0x78	x	0xa7	x
0x59	Y	0xe8	Y	0x79	y	0xa8	y
0x5a	Z	0xe9	Z	0x7a	z	0xa9	z
0x5b	[	0xad		0x7b	{	0xc0	
0x5c	\	0xe0		0x7c	|	0x4f	|
0x5d	]	0xbd		0x7d	}	0xd0	
0x5e	^	0x5f	¬	0x7e	~	0xa1	
0x5f	_	0x6d	_	0x7f	DEL	0x07	DEL

ASCII-to-EBCDIC

In	Disp. Char	Out	Disp. Char	In	Disp. Char	Out	Disp. Char
0x80		0x00	NUL	0xa0		0x00	NUL
0x81		0x00	NUL	0xa1		0x00	NUL
0x82		0x00	NUL	0xa2		0x00	NUL
0x83		0x00	NUL	0xa3		0x00	NUL
0x84		0x00	NUL	0xa4		0x00	NUL
0x85		0x00	NUL	0xa5		0x00	NUL
0x86		0x00	NUL	0xa6		0x00	NUL
0x87		0x00	NUL	0xa7		0x00	NUL
0x88		0x00	NUL	0xa8		0x00	NUL
0x89		0x00	NUL	0xa9		0x00	NUL
0x8a		0x00	NUL	0xaa		0x00	NUL
0x8b		0x00	NUL	0xab		0x00	NUL
0x8c		0x00	NUL	0xac		0x00	NUL
0x8d		0x00	NUL	0xad		0x00	NUL
0x8e		0x00	NUL	0xae		0x00	NUL
0x8f		0x00	NUL	0xaf		0x00	NUL
0x90		0x00	NUL	0xb0		0x00	NUL
0x91		0x00	NUL	0xb1		0x00	NUL
0x92		0x00	NUL	0xb2		0x00	NUL
0x93		0x00	NUL	0xb3		0x00	NUL
0x94		0x00	NUL	0xb4		0x00	NUL
0x95		0x00	NUL	0xb5		0x00	NUL
0x96		0x00	NUL	0xb6		0x00	NUL
0x97		0x00	NUL	0xb7		0x00	NUL
0x98		0x00	NUL	0xb8		0x00	NUL
0x99		0x00	NUL	0xb9		0x00	NUL
0x9a		0x00	NUL	0xba		0x00	NUL
0x9b		0x00	NUL	0xbb		0x00	NUL
0x9c		0x00	NUL	0xbc		0x00	NUL
0x9d		0x00	NUL	0xbd		0x00	NUL
0x9e		0x00	NUL	0xbe		0x00	NUL
0x9f		0x00	NUL	0xbf		0x00	NUL

ASCII-to-EBCDIC

In	Disp. Char	Out	Disp. Char	In	Disp. Char	Out	Disp. Char
0xc0		0x00	NUL	0xe0		0x00	NUL
0xc1		0x00	NUL	0xe1		0x00	NUL
0xc2		0x00	NUL	0xe2		0x00	NUL
0xc3		0x00	NUL	0xe3		0x00	NUL
0xc4		0x00	NUL	0xe4		0x00	NUL
0xc5		0x00	NUL	0xe5		0x00	NUL
0xc6		0x00	NUL	0xe6		0x00	NUL
0xc7		0x00	NUL	0xe7		0x00	NUL
0xc8		0x00	NUL	0xe8		0x00	NUL
0xc9		0x00	NUL	0xe9		0x00	NUL
0xca		0x00	NUL	0xea		0x00	NUL
0xcb		0x00	NUL	0xeb		0x00	NUL
0xcc		0x00	NUL	0xec		0x00	NUL
0xcd		0x00	NUL	0xed		0x00	NUL
0xce		0x00	NUL	0xee		0x00	NUL
0xcf		0x00	NUL	0xef		0x00	NUL
0xd0		0x00	NUL	0xf0		0x00	NUL
0xd1		0x00	NUL	0xf1		0x00	NUL
0xd2		0x00	NUL	0xf2		0x00	NUL
0xd3		0x00	NUL	0xf3		0x00	NUL
0xd4		0x00	NUL	0xf4		0x00	NUL
0xd5		0x00	NUL	0xf5		0x00	NUL
0xd6		0x00	NUL	0xf6		0x00	NUL
0xd7		0x00	NUL	0xf7		0x00	NUL
0xd8		0x00	NUL	0xf8		0x00	NUL
0xd9		0x00	NUL	0xf9		0x00	NUL
0xda		0x00	NUL	0xfa		0x00	NUL
0xdb		0x00	NUL	0xfb		0x00	NUL
0xdc		0x00	NUL	0xfc		0x00	NUL
0xdd		0x00	NUL	0xfd		0x00	NUL
0xde		0x00	NUL	0xfe		0x00	NUL
0xdf		0x00	NUL	0xff		0x00	NUL

Index

A-C

Accessibility 131
Address Conflicts 56
Address (tape) 270
Amount To Read 89
ANSI Labels 120, 122
Append tape 111
Append To Existing File 90
ASCII to EBCDIC 97, 114, 134, 266, 276
Automatically Generate Filenames 91
Automatic Operations 278
Backup:
    Network 136
    DOS 136
BCD 20
Before Reading, Skip 96
Block Size 14, 138
Blocks/Pass: [n] 139
Blocks to Read [n] 89
Bulletin Board System 5
Changing I/O Port Address 58
Changing ROM BIOS Address 57
Choosing an Appropriate Block Size 14
Command Line 99, 116
Command Reference 313
Compress (Menu) 340
Compression 171
Control Character 133
Conventions 2, 181-184
Copy Disk to Tape 215
Copy Tape to Disk 222, 243
Create Or Overwrite A File 90
Creation Date 130

D-F

Data Blocks 14
Data Bytes 7

Data Compression 115, 134
Data Fields 16
Data Set ID 130
DataTools 80
Density (set) 270
Depot2
Depot4 81
    Help 84
    Menus 82
    Overview 81
    Setup 82
    Shell program 81
Display 173
Display (commands) 98, 115
DOS Backup 136
DOS Batch File 99, 116
DOS Filter 271
EBCDIC to ASCII 97
Edit 108
Error retry count 270
Examples
    Read Unlabeled Tape 101
    Write Unlabeled Tape 118
    Write Labeled Tape 134
    Test the Tape Drive 140
    Heads-Up Display 174
Exit 86
Expiration Date 130
FDUMP 202
File Menu 85
Filename:[ ] 93, 108
Fixed Block Size 138
Fixed-Length Data 217, 236, 247
Flashbak 286

G-L

Gap (set inter record) 269
Header 1 130
Header 2 132

Heads-Up Display....173
IBM Labels....120, 122
ICRC Data Compression....115, 134
ID (Data Set)....130
ID (Job Step)....132
ID (owner)....129
IEEE Floating Point....19
Ignore Errors....98, 115
Incrementing....139
Insert CR/LF At:[n]....89
Installation, Software....69-70
INSUTIL.EXE....283
Interface:
ASPI....144
Parallel Port....149
TMC SCSI....150
TX-8 Cipher/Pertec....156
XL/2....160
Interface Controller:
Drive Type....159
I/O Address....158
Keyboard....182
Keyboard Mapping....290
Labeled Tapes....11
Label Type....126
Line Length....113
Lines per Block....113

M-R

Many Files....91
Menu selections....85
Microsoft Floating Point....18
Mouse....181
MS DOS Shell....85
Network Backup....136
Number Formats....16
Opening/Closing the PC....54-55, 59-60
Options Menu....142

Configuration 143
Output Processing 112
Overwrite tape 111
Packed BCD 20
Parallel Port Interface 51
Setup and Configuration 51-52
Place Data In 90
Positioning Command 257-266
Random Block Size 138
Read 1 File 89
Read Only 139
Read Unlabeled Tape 88
Record Formatting 88
Record Length 112, 127
Records per Block 113, 128
Record Processing 112
Records 15
Records to Read [n] 89
Repeat Test 138
Rewind 262-263

S-T

Serial Number 128
Set Bit 97, 115
Set tape speed 268-269
Shell Program 81
Signed Integers 17
Skip
Files[ ], Blocks[ ] 97
Spacing:
by files 259
by blocks 260
Logical End 262
Speed (tape) 268-269
Stacker 140
System Code 132
Tape Capacity 7
Tape Data 14

Tape Density 7
Tape Format 7
Tape Layout 9
Tape Operation 111
Tapes To Read [n] 90
TAPEUTL 211
Technical Support 3
Test (tape drive) 137-140
Text 113
    Line Length 113, 128
    Lines per Block 113, 128
TMC-MCA
    Address Conflicts 59
    Installation/Configuration 59-60
TMC-ISA 54
    Address Conflicts 56
    Changing I/O Port Address 58
    Changing ROM BIOS Address 57
    Configuration and Testing 55
    Factory (Default) Configuration 54
    Host Resources Required 54
    Installation 55
TMS Controller Card 53
    TMC-ISA 54
    TMC-MCA 59
Transfer (data) 215-254
Translation 97, 114, 134
    Default 97
Translation Tables 163
    Default 163
    Deleting 169
    Editing 168
    Existing 167
    New 165
TX-8 Controller Card 22
    Installation 23-27
    Inventory Checklist 22
    Standard Features 22
    System Requirements 23
TXi-16 Controller Card 31

Installation 32-36
Inventory Checklist 32
Setting configuration mode 37
Standard Features 31

U-Z

Unlabeled Tapes 9
Unpacked BCD 20
Unsigned Integers 17
Utilities Menu 87
Variable-Length Data 219, 238, 250
Variable-Length Record 252
View Tape 136
Volume (options) 128
When to configure 37
Write a Labeled Tape 120
Write Tapemark 263
Write Unlabeled Tape 107
XL/2 Controller Card 42
Configuration 46-50
Installation 43-46
Inventory Checklist 42
Standard Features 42
System Requirements 43
Testing 50
XTEST 337

Overland Data, Inc.

8975 Balboa Avenue
San Diego, CA 92123-1599
TEL: (619) 571-5555
FAX: (619) 571-0982