Radio Shack®

POWERTOOL
Operations Guide

CUSTOM MANUFACTURED IN U.S.A. BY RADIO SHACK, A DIVISION OF TANDY CORPORATION

PowerTool™
Cat. No. 26-2022

Contents

Introduction

PowerTool is an extremely powerful utility package for use on either the TRS-80 Model I, Model III or Model 4 disk-based computer when used in the Model III mode. Its many routines allow you to:

- Directly examine and modify diskettes written in any of several versions of TRSDOS and LDOS
- Restore many unreadable diskettes to a readable condition
- Format all or part of a diskette
- Backup all or part of a diskette

This manual explains the various utility routines available in PowerTool. Please read it carefully before attempting to use the program for the first time. Due to its power and versatility, PowerTool can do great damage to your diskettes if you use it carelessly.

Throughout this manual, we will assume that you are familiar with the TRSDOS or LDOS Disk Operating Systems. For more information on TRSDOS or LDOS refer to your owner's manuals.

System Requirements

To run PowerTool, you must have the following:

- A Model I or Model III computer, with 48K of RAM
- At least one disk drive (Model I drives may be single or double density)

The PowerTool System diskette contains its own operating system, and hence the same diskette will boot on both the Model I and Model III.

Special Notation Used in this Manual

The PowerTool program requires that you input commands and other pertinent information. Some entries may be made by pressing a certain key. In this manual, when a key is surrounded by a box, simply press the key. For example, [ENTER] means to press the keyboard key labelled "ENTER", and [S] means to press the key labelled "S" (either upper or lower case).

1/ An Overview of PowerTool

PowerTool is a "menu driven" program. This means that to select a particular utility, or "tool", you choose from a set of utilities displayed on the screen. If your selection is itself a subset of utilities, PowerTool will display another menu, made up of more utility options.

In the Main Menu display, each selection is preceded by a single letter which represents a group of tools. By pressing the appropriate letter on your keyboard, you will either be taken into a further menu which will allow you to choose from the tools available in that group, or else to a prompt for more information pertinent to the selected tool. Pressing (ENTER) defaults to the first selection in the Menu.

The Main Menu consists of eight options:

ZAP Tools — Allows you to directly examine, modify and copy the contents of a diskette. The routines in this group also permit you to search a diskette for a particular occurrence of bytes or characters.

PURGE Tools — Allows you to quickly and easily kill or recover files from a TRSDOS, LDOS or data diskette and clean up your directory in the process. You may remove all traces of a file from a diskette, change the diskette name, and view the diskette directory before and after making changes.

DISK FORMAT Tools — Formats a diskette in one of a variety of ways. You may even reformat a diskette without losing any data previously written on it! Also included under the disk formatting tools is a routine which removes all traces of data from a diskette, in effect "bulk-erasing" it.

BACKUP Tool — Performs a standard backup of one diskette onto another, even faster than the standard TRSDOS or LDOS BACKUP utilities.

REPAIR Tools — Lets you restore many damaged diskette directories or boot sectors to a usable condition. You may also recover files which were killed by PowerTool, and check the diskette directory for any errors which may result in the destruction of files later on.

MEMORY Tools — Allows you to examine the contents of memory, move the contents of a segment of memory to another location, search memory for the occurrence of a particular string, read a port, send a byte to a port, and transfer memory to diskette and vice-versa. Additionally, you may also transfer to your own machine-language subroutine in memory for customized needs.

FILE Tools — Allows you to view and modify the contents of a file on a diskette. FILE Tools are similar to ZAP Tools except that they are file-oriented rather than disk-oriented. You may compare two files for differences, copy files from one diskette to another, display the free space on a disk, display the locations of files on a disk, create new files, and check the status of your disk drives.

CONFIGURE System — Allows you to tailor PowerTool to your own system. You can configure PowerTool according to the characteristics of your printer, if you have one, how many disk drives you have, and what kind of disks you expect to be using in each drive. You can also optionally write-protect a particular disk drive, via software, so that no data on it can be inadvertently changed.

Starting-up PowerTool

The PowerTool program comes on a special ''self-booting'' diskette. Insert the PowerTool diskette into Drive 0 and press (RESET). The PowerTool logo will appear on the screen while your disk drive continues to run. After a few seconds the disk drive will stop, and the logo will ''animate.'' At this point, pressing any key will bring you to the main menu.

You will notice that there are six ''alive'' characters wiggling in the corners to show interrupt activity. If you do not desire these ''wigglies'' on the screen or eventually find them annoying, you can turn off the ''alive'' function. Hold down (CLEAR) and press the (A) key to toggle the ALIVE function on and off. If you never want to see ''alive'' again, turn it off using the above method, then refer to the ''Configure'' section and ''save'' the new configuration to disk. From then on the PowerTool will boot minus the alive ''wigglies''.

PowerTool does not require the presence of a TRSDOS or LDOS system disk in Drive 0, nor does it require the presence of its own disk in Drive 0. Once you have successfully brought up the main PowerTool menu, remove the PowerTool disk from the drive and put it in a safe place.

NOTE: The PowerTool diskette uses a special format, and hence cannot be backed up. It may be a good idea to keep a write-protect tab on the diskette (except when setting the default configurations) to prevent accidental over-writing of the PowerTool program. Also remove the diskette from the drive when not using it.

When you first start up, PowerTool displays the main menu, called the ''TOOL Selector'' menu:

Z — ZAP Tools
P — PURGE Tools
D — DISK FORMAT Tools
B — BACKUP Tool
R — REPAIR Tools
M — MEMORY Tools
F — FILE Tools
C — CONFIGURE System
E — EXIT PowerTool

Choice?

By pressing the appropriate letter for the routine you want, you will activate that particular tool. To exit a function at any time, press (BREAK). To return to the Main Menu at any time, hold down (SHIFT) and press (BREAK).

Answering PowerTool Prompts

Many of PowerTool's routines prompt you for additional information. You may enter numeric information in decimal (the default base), hexadecimal (by appending H after the number), octal (by appending the letter O or the letter Q to the number), or binary (by appending B to the number).

Type in any string input directly, using lower case as needed. (SHIFT) (0) acts as a case toggle — pressing it once reverses case. Reverse case means unshifted letters are lower case, shifted letters are upper case (as in normal Model I). When toggled, unshifted keys generate upper case, shifted keys generate lower case. When PowerTool requests several inputs, enter them separated from each other either by commas or by spaces.

Printing Out the Current Screen

Many times, you may need a hard copy of data displayed on the screen. Pressing (SHIFT) (CLEAR) at any time sends whatever is on your screen to the printer. Pressing the (SHIFT) (@) keys will empty the printer buffer and printing stops.

If your printer is capable of producing block graphics, you can configure PowerTool to print the same graphics characters on the printer as are on the screen. Otherwise PowerTool changes all graphics symbols to ''#'' signs before printing. The next chapter supplies details on configuring PowerTool to your printer. ASCII 0-31 are converted to periods on the printer.

Exiting PowerTool

To end the session, from the Main Menu, press (E). PowerTool will ask you to insert a system diskette into Drive 0. It then initiates a system reset and boots the new system.

Error Recovery

Should any I/O errors occur while PowerTool is attempting to read or write to a disk, it displays a message describing the type of error encountered. For example:

Sector NOT FOUND

or

DATA CRC Error

PowerTool then prompts you:

R>etry, S>kip, C>ontinuous, Q>uit ?

You may press (R) or (ENTER) to retry the I/O operation once. If the error occurred as a result of some momentary condition, this is usually sufficient to correct the situation. However, if the error continues to appear, you may want to type (C) for continuous retries. This forces PowerTool to retry the I/O operation until it gets it right, or else you press (CLEAR), which restores the retry prompt. Also, (BREAK) and (SHIFT)(BREAK) abort the operation and return you back to a PowerTool menu.

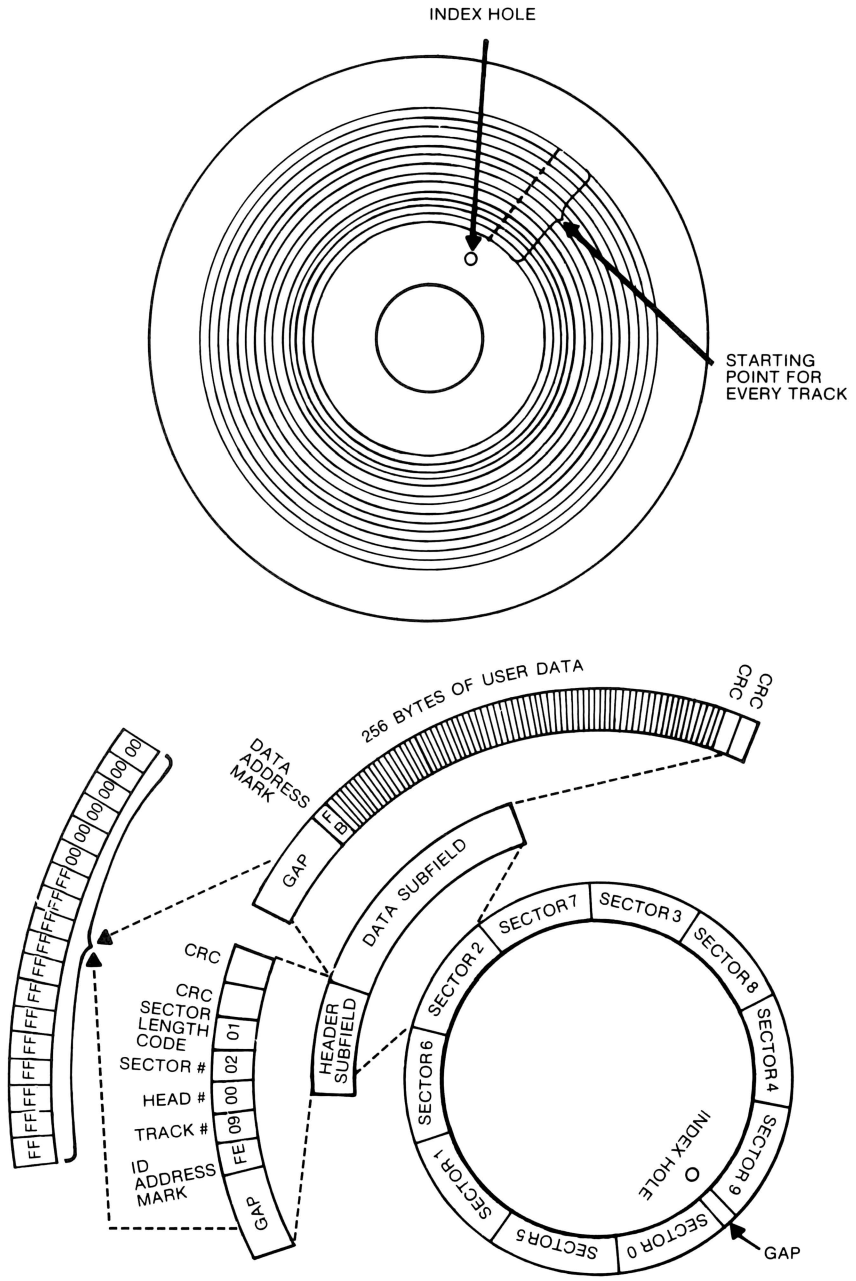
Depending on the operation PowerTool was attempting, pressing (S) may immediately bring up the sector display. Depending on the error encountered, it may or may not contain any data. If the sector could not be read at all, the display will contain all 00 bytes. If PowerTool was successful in partially reading the display, some data will be present in the display. However, you should never assume that what is shown on the display is an accurate representation of the data.

Pressing (Q) aborts the operation in progress and returns you immediately to the previous menu.

> **IMPORTANT NOTE:** The following chapters deal with each group of tools in detail. Please read each chapter carefully and keep this manual handy while using PowerTool. We cannot emphasize enough the dangers of using PowerTool in a careless fashion. If you are not sure of something, check the manual.

2/ Diskette Organization

In order to use PowerTool effectively, you must first have some understanding of how a diskette is organized. When you format a diskette using the FORMAT utility from your TRSDOS or LDOS System diskette, information is written to the diskette in concentric rings called ''tracks'', as shown in the figure below.



On a Model I, each diskette is divided, or ''formatted,'' into 35 tracks, or cylinders (certain newer Model I drives can be formatted to 40 tracks, but this depends on the operating system — Model I Single-Density TRSDOS normally recognizes only 35). On a Model III, a diskette is formatted into 40 tracks.

Each track in turn is divided into ''sectors.'' The number of sectors on a track depends on the diskette's ''density.'' The Model I normally formats diskettes in single-density unless it is equipped with a double-density modification, in which case it can format either single or double-density. The Model III formats diskettes as double-density. A single-density track contains 10 sectors, while a double-density track contains 18 sectors.

Each sector contains 256 bytes of data (unless a special formatting scheme was used) plus some additional bytes of information which identify the sector to the computer in terms of its track and sector location.

The sectors on each track are grouped together into units called granules, or ''grans.'' A granule is the smallest unit the operating system will allocate to a diskette file. Machine-language programs, BASIC programs, and data files are all ''files'' to the system and are assigned granules of space on the diskette as needed.

A single-density diskette, such as that produced by Model I systems without the Radio Shack double-density adapter, will have two granules of 5 sectors on each track. A Model III TRSDOS double-density diskette will have six granules of three sectors each per track. Double-density LDOS diskettes will have three granules of six sectors each per track.

Formatting

When you format a diskette (from TRSDOS, LDOS, or PowerTool), the system writes certain information to the diskette which divides it into sectors. As well as assigning a ''data field'' to the sector, where your data is stored, the system also writes out an ''ID field,'' which stores information regarding the sector number, track number, as well as information about the data field.

Included in the ID field of the sector is the Cyclic Redundancy Check (CRC). Whenever the operating system writes a sector to the disk, the floppy disk controller (FDC) chip performs some mathematical calculations on the ID field data, transforming it into a two-byte number, stored as the CRC. Likewise, the FDC calculates a CRC for the data field and stores a second CRC there.

Whenever data is read from the sector, the FDC recalculates a CRC on the data and compares it to the CRC stored in the data field. If they differ, the FDC signals the operating system that an error has occurred, either in the transmission, or there is a problem with the diskette or in the disk drive itself.

Also stored in the ID field is the Data Address Mark (DAM). The DAM helps differentiate between one type of data field and another. The Model I hardware is capable of writing four types of DAM — standard, deleted data, read-protected, and user-defined. The Model III hardware recognizes two types of DAM — standard and read-protected.

Note: These names are really meaningless, that is, ''read-protected'' does not mean the sector cannot be read at all. These names are just ways to differentiate one type from another.

TRSDOS and LDOS use the DAMs for distinguishing between directory tracks and data tracks. As it turns out, Model I Single-Density TRSDOS uses a DAM which the Model III FDC can't recognize, hence you can't run Model I programs on the Model III without modification of the DAMs. LDOS and Model I Double-Density TRSDOS however, use a DAM which the Model I and Model III FDC's both recognize.

The table below indicates the equivalent types for Model I and Model III as used in PowerTool:

Model I STD & RPT = Model III STD
Model I DDT & UDF = Model III RPT
Model III STD = Model I STD
Model III RPT = Model I DDT

The Directory

When a file is saved to disk, it may occupy several grans of space. It is the job of the disk operating system to keep track of where the file is located on the disk, and how much space it takes up. Whenever possible, the operating system uses consecutive sectors when assigning new granules to a file, so that disk drive's read/write head needn't ''thrash'' around the diskette to find the data in a file.

Sometimes, however, the operating system must break a file into several segments, or ''extents'', in order to make maximum use of the available space on a diskette. The disk operating system keeps track of all this and maintains information on each file in a special place on the disk called the ''directory.'' The directory takes up an entire track, and is often found near the middle track of the diskette.

The directory track consists of a ''hash index table'' (HIT), a granule allocation table (GAT) and a number of directory entries for each file on the disk.

When you ask the operating system to load a file (for instance when you issue the RUN ''filename/ext'' command from Disk BASIC), the operating system first goes to the diskette's directory track and looks for the file you specified. The filename is encoded, or ''hashed'' and compared to the HIT table entries. The location of the HIT table entry, if found, corresponds to the actual location of the rest of the directory entry for that file.

The directory entry contains general information about the file, including the creation date, passwords, and protection levels, as well as information on where on the disk the file can be found. From this information, the operating system can determine what action to take next (for instance loading the file or returning an error message such as ''file not found'' or ''file password protected'').

When the operating system needs to create or extend a file (for example when you copy a file), it looks to the GAT for the location of free granule space. From its findings, it determines if and where it can locate the new data.

The directory track is ''marked'' off from the rest of the tracks on the diskette by the use of a special ''data address mark'', or DAM. This is a piece of information written onto the disk by the operating system during the format process which permits it to locate the directory track quickly.

However, the type of DAM written to the directory is dependent on the hardware in the computer, specifically, the ''floppy disk controller'' chip. The Model I uses a different controller chip than the Model III. This poses certain problems, the major one being the fact that you cannot read a Model I single-density TRSDOS disk's directory on a Model III.

Reading Model III diskettes on a Model I, however, poses no problems if the Model I is equipped with a double-density adapter. LDOS avoids this problem by using a data address mark for the directory track that both Model I and Model III1 controller chips can read and write.

The directory track, then, is one of the most important parts of a diskette. If the directory track is damaged in some manner so that the operating system cannot correctly read the information on it, the disk essentially becomes unusable. The files themselves are still on the disk, but the system no longer has any way of finding them.

The Boot Sector

There is one other place on the disk which is vital to the operation of a system disk. This is the very first sector on the first, or outermost, track of a diskette. This sector is called the ''boot sector''. On a Model I TRSDOS single-density diskette, and LDOS diskettes, this is Sector Ø of Track Ø. On a Model III TRSDOS/LDOS diskette the boot sector is Sector 1 of Track Ø.

DISKETTE ORGANIZATION

When you place a System diskette into Drive 0 and press the (RESET) button, special codes stored in ROM order the disk drive to move its read/write head to track 0 and read the boot sector into memory (starting at 4200H for the Model I, 4300H for the Model III).

Once the boot sector is in memory, the computer jumps to its first byte. This sector contains information which then permits the computer to read the rest of the operating system into memory. If the diskette is not a system diskette, the boot sector code produces the ''NO SYSTEM'' notice on your screen.

If the boot sector on a TRSDOS system disk is damaged, that disk also becomes unusable as a system disk, although it may still be possible to use it as a data disk.

PowerTool has the capability of restoring to a usable condition disks which have sustained damage to the directory track or boot sector, if the damage is not too extensive. Sometimes the damage is so widespread that no recovery is possible, but in many cases, a disk can be restored to working condition at least long enough for you to copy important files over onto another diskette.

3/ Configuring PowerTool For Your System

To get the most out of PowerTool, you first need to ''customize'' the PowerTool Program according to the type of drives you are using, your printer (if one is attached) and the Operating System which created the diskette you are working on.

To configure PowerTool, from the Main Menu, press (C) (for CONFIGURE System). PowerTool displays:

```
#################################################################
# **  PowerTool  **  Ver 01.00.00  **  By: Kim Watt  **         #
# Copyright # 1983 Breeze/QSD, Inc. - Licensed to Tandy Corp.   #
#################################################################
# ##  CONFIGURATION  ##                                         #
=>Dual=N Graphics=N Locase=Y Linefeeds=N Doubler=R               #
# +:0 T3D'    Track Count= 40 Dir= 17 Stp=3 Rdly=4 Wdly=4 WP=N #
#         DO=D Dd=D LSO=1 HSO=18 LSd=1 HSd=18 S/G=3 G/T=6 DD=I #
# +:1 T3D'    Track Count= 40 Dir= 17 Stp=3 Rdly=4 Wdly=4 WP=N #
#         DO=D Dd=D LSO=1 HSO=18 LSd=1 HSd=18 S/G=3 G/T=6 DD=I #
# +:2 T3D'    Track Count= 40 Dir= 17 Stp=3 Rdly=4 Wdly=4 WP=N #
#         DO=D Dd=D LSO=1 HSO=18 LSd=1 HSd=18 S/G=3 G/T=6 DD=I #
# +:3 T3D'    Track Count= 40 Dir= 17 Stp=3 Rdly=4 Wdly=4 WP=N #
#         DO=D Dd=D LSO=1 HSO=18 LSd=1 HSd=18 S/G=3 G/T=6 DD=I #
# ? ___________________________________________________________ #
#################################################################
```

This is the ''configuration table''. Note the little arrow to the left of the top line. The arrow indicates which elements of your system you can currently configure.

Configuring the Printer and Double-Density Adapter

When you first enter the Configuration tool, the arrow points to a line displaying:

Dual = N Graphics = N Locase = N Linefeeds = N Doubler = R

Dual refers to whether or not you want dual output on or off — if you turn it on (Dual = Y), PowerTool sends everything that appears on the screen to the printer as soon as it is displayed on the screen. On printers which can linefeed without a carriage return and have greater than 64 column carriages, the lines may be scattered all over the paper. This is because Powertool uses the linefeed code for both screen and printer. On the screen a linefeed always puts the cursor at the start of the next line. This may or may not be the case with a printer. If you turn it off (Dual = N), PowerTool displays data only on the screen. Configuration display must be screen dumped if you want a hard copy.

Graphics refers to whether or not your printer is capable of printing block graphics symbols. If this is turned on (Graphics = Y), then PowerTool sends the graphics codes out to your line printer. Turned off (Graphics = N), PowerTool replaces all graphics symbols with pound signs (''#'').

Some printers can generate the computer block graphics, but do not use the same ASCII codes as the computer uses. Such printers must be considered as incapable of generating the graphics codes, since PowerTool does not possess any ability to offset the computer's internal graphics codes to the codes required by the printer.

Locase refers to whether or not your printer can print lowercase letters or not. If turned on (Locase = Y), PowerTool sends all lowercase codes as is; if off (Locase = N), lowercase codes are translated to uppercase before they are sent to the printer.

Linefeeds refers to whether your printer requires line feeds (ASCII Code ØAH) following carriage returns (ASCII Code ØDH). If on (Linefeeds = Y), then PowerTool sends a line feed character after every carriage return. If off (Linefeeds = N), then no line feed character is sent. (Radio Shack printers automatically line feed after receiving carriage returns.)

Doubler is relevant to Model I users only. It indicates whether or not you have a ''doubler'' (a double-density adaptor) installed in your machine. Doubler = R indicates the presence of a Radio Shack doubler, Doubler = N indicates that no doubler is present, and Doubler = X indicates that a non-Radio Shack doubler is present. What appears as doubler setting will depend on what the program detects on initialization. Model III users should **NOT** set Doubler = N. Leave it alone!

To set these options, type in values (Y for on, N for off, R for Radio Shack Doubler, and X for non-Radio Shack Doubler) on the prompt line, separating each by commas or spaces. For example, to set the line to:

Dual = Y Graphics = Y Locase = Y Linefeeds = N Doubler = R

type:

Y,Y,Y,N,R and press (ENTER)

Note that you only have to enter the answers, but they must be in the correct relative position. For example, if Doubler was set to R already, then you could have entered:

Y,Y,Y,N

As soon as you press (ENTER), you will see the top line change to reflect your input, and the arrow will move down to the next line. Invalid answers cause PowerTool to prompt for the line; no change takes place in the configuration tables.

If you want the configuration to stay as it is, simply press (ENTER) with no other input.

Configuring Diskette Drives

The remainder of the Configuration Table refers to the disk drives in your system. Two lines describe each drive, however, you may only change the options on the first line — the data on the second line for each drive is implied from the first.

When you first start-up PowerTool, the configuration for Drive Ø looks like this for Model III:

+:Ø T3D' Track Count= 4Ø Dir= 17 Stp=3 Rdly=4 Wdly=4 WP=N
DØ=D Dd=D LSØ=1 HSØ=18 LSd=1 HSd=18 S/G=3 G/T=6 DD=I

The plus sign indicates that this drive, in this case Drive Ø (indicated by :Ø) is active. A minus sign removes the drive from the system and PowerTool refuses to perform I/O with it.

An equals sign here indicates that PowerTool is operating in ''skip'' mode for this drive. Use this mode only when trying to read a 35 or 4Ø track diskette in an 8Ø-track disk drive. (Note: Never write to a 35 or 4Ø track diskette while it is in an 8Ø-track drive. This disk may not be readable in a 35 or 4Ø track drive afterward.)

The next field specifies the operating system that created the diskette in the drive. In the above example, this is T3D. T3D specifies Model III TRSDOS. The operating system specifiers recognized by PowerTool are abbreviated as follows:

T1S	Single-density Model I TRSDOS
T1D	Double-density Model I TRSDOS
T3D	Model III TRSDOS
LIS	Single-density LDOS Mod. I
LxD	Double-density LDOS Mod. I/III
L1D	Double-density Model I LDOS with single-density Track Ø
U	Unknown/unidentifiable DOS type

Table 1

(NOTE: You cannot use the U specifier. PowerTool writes this into the configuration table to indicate that it was unable to correctly identify the DOS — you may not declare an unknown disk type.)

Track Count indicates the number of formatted tracks on the diskette in the specified drive. Dir indicates which track the diskette directory is on.

Stp refers to the ''head stepping'' rate of the drive. The head stepping rate refers to time that the read/write head requires to move from one track to the next. The expression for Step is a coded value, with corresponding step rates as follows:

0	5-6 milliseconds
1	10-12 milliseconds
2	20 milliseconds
3	30-40 milliseconds

A standard Model I disk drive can step its read/write head from one track to the next at a rate of 20 milliseconds. However, some older drives can't step faster than 40 milliseconds. Model III drives are for the most part capable of stepping at 6 milliseconds.

Do not specify a step rate faster than the drive is capable of, or else you may cause I/O errors which can ruin your disk. If you are uncertain of your drive's step rate, consult your drive's Owner's Manual or use 3 for the slowest rate.

Rdly controls the delay from the time a drive's motor comes on to the time when PowerTool first attempts to read from the disk. When set to 1-4 PowerTool waits 1/4 second to 1 second respectively, before reading from the diskette.

Wdly, conversely, controls the delay from the time the disk motor comes on to the time PowerTool first attempts to write to the disk. When set to 1-4 PowerTool delays 1/4 second to 1 second respectively, before attempting a write operation. If set to ''2'', it waits only one-half second. Since writes are more critical than reads, this delay factor is controlled separately to ensure reliable writes to the disk.

WP is a ''switch'' which tells PowerTool whether or not to ''write-protect'' your drive. If you specify ''Y'' to this, PowerTool can read any diskette in that drive, but cannot write to it. This is functionally equivalent to putting a write-protect tab on your diskette.

To alter the settings, you must enter a series of answers to the prompt which describe your disk drive. The settings in the example above would have been given by this string:

+,T3,40,17,3,4,4,N

The + is optional and indicates that the drive should be active in the system; the rest of it indicates that PowerTool should expect a Model III TRSDOS (T3) disk with 40 formatted tracks, a directory on track 17, in a drive capable of stepping at 40 milliseconds (see table on previous page) requiring motor-on delays for both reads and writes, and that the drive should not be write protected.

To change the specifications to a single-density Model I diskette, we might enter:

+,T1S,35

T1S indicates that the diskette in the drive is on single-density Model I TRSDOS with 35 tracks. The rest of the information is the same and does not have to be entered. In most cases, it is also unnecessary to pre-set the directory track; PowerTool can find it automatically.

If you wish to logically remove a drive from the system, you need only enter "–" to disable that drive — the other specifications obviously don't matter.

If the settings on a particular drive are already correct, simply press (ENTER). The arrow moves down to the next drive. Note that PowerTool skipped a line. This line contains information about the formatting of the diskette described in the previous line. This information cannot be changed directly, but rather is implied from the line above it.

D0 indicates the density of track 0. Some disks have track 0 formatted in a different density from the rest of the disk. D0 is either "S" for single-density or "D" for double-density.

Dd indicates the density of the rest of the disk. Dd is either "S" for single-density or "D" for double-density.

LS0 indicates the sector number of the lowest sector on Track 0. Model III TRSDOS numbers sectors starting from 1. Model I TRSDOS and LDOS number their sectors from 0. Model III LDOS numbers sectors from 0.

HS0 tells you the sector number of the highest sector on Track 0. For a single-density/double-density Model I TRSDOS/LDOS disk, HS0 equals 9, for Model III TRSDOS it equals 18, and for Model III LDOS it equals 17.

LSd indicates the sector number of the lowest sector on the rest of the diskette's tracks. This equals 1 for Model III TRSDOS and Model I Double-Density TRSDOS, and 0 for all others.

HSd is the number of the highest sector on the rest of the diskette's tracks. HSd equals 18 for Model I double-density TRSDOS and Model III TRSDOS, 9 for single-density Model I TRSDOS and single-density LDOS, and 17 for double-density LDOS.

S/G indicates the number of sectors per granule. Since the granule is an arbitrary unit, its size can, and indeed does, vary. Single-density disks (TRSDOS and LDOS) have 5 sectors per granule, double-density Model I TRSDOS and Model III TRSDOS have 3 sectors per granule, and Model I/III double-density LDOS disks have 6 sectors per granule.

G/T stands for granules per track. This value obviously varies according to the way a granule's size is defined. For double-density Model I TRSDOS and Model III TRSDOS, G/T equals 6, for single-density Model I disks (TRSDOS and LDOS) it equals 2, and for double-density LDOS it equals 3.

DD indicates the type of data address marks used by the disk. Each disk writes one type of data address mark for the data tracks and another type for the directory track. For all operating systems except Model III TRSDOS, this will read S meaning "standard convention." For Model III TRSDOS, it will read I for "inverted."

Saving the Configuration

After you have configured all of the drive settings to your specifications, press (ENTER) once more. PowerTool asks you if you want to save the configuration onto the PowerTool Diskette. If you reply "Y", PowerTool prompts you to mount the PowerTool Diskette in Drive 0. (Be sure that there is no write-protect tab on the disk and that it is not software write protected!)

Now press (ENTER). Your configuration is written out to the disk and the next time you boot PowerTool up it will use this configuration.

If you reply "N" to the Save Configuration? prompt, then PowerTool stores the configuration in memory but not onto the PowerTool Diskette.

Some Notes About Changing Configuration

In general, the only things that you really need to configure are the printer specifications, and for each drive the DOS type, track count, step rate, motor-on delay, the software write-protect switch, and whether or not a particular drive is to be actively in the system.

The directory location of the disks are determined when PowerTool goes to read a diskette in the specific drive, and will change accordingly. Sometimes, if you swap a disk of a different density into that drive, read it, and then view the configuration table, you will see that the table has changed to reflect the density of that disk.

Remember that anytime you specify a drive number in a command, you may also specify the DOS type and, optionally, the number of tracks on the drive. To do this, follow the drive number immediately with the DOS type (using the abbreviations in Table 1). To specify the track count, follow the DOS type immediately with "=nn" where nn is the number of tracks on the disk. For example:

1T1D=40,1,1

specifies drive 1, track 1, sector 1, using double-density Model I TRSDOS, with 40 tracks per diskette.

4/ ZAP Tools

The ZAP Tools allow you to examine and modify the diskette sector by sector. To use the ZAP tools, from the Main Menu, press (Z), or simply (ENTER). PowerTool displays the ZAP Tools Menu:

```
######################################################################
# **  PowerTool  ##  Ver 01.00.00  ##  By: Kim Watt  ##             #
# Copyright # 1983 Breeze/QSD, Inc. - Licensed to Tandy Corp.       #
######################################################################
# ##  ZAP Tools  ##                                                 #
#                                                                    #
# 1.  Display Sectors          7.  Reverse Sector Data               #
# 2.  Verify Sectors           8.  Exchange Sectors                  #
# 3.  Compare Sectors          9.  String Search                     #
# 4.  Copy Sectors             0.  Sector Search                     #
# 5.  Copy Sector Data         A.  Read ID Address Marks             #
# 6.  Zero Sectors             B.  Alter DATA Address Marks          #
#                                                                    #
# Choice? __                                                         #
#                                                                    #
######################################################################
```

Select the routine you want by entering the letter or number which precedes it. You needn't press (ENTER).

Display Sectors

To display a sector of a diskette, press (2). PowerTool prompts you with:

Drive, Track, Sector?

Enter the number of the disk drive containing the disk you want to examine, then the track number and sector number you want to display. Be sure to separate items with spaces or commas. If you omit the track and sector, PowerTool displays the lowest sector on the lowest formatted track on the disk. If you enter D for track, PowerTool displays the directory track (you don't need to know which track the directory is on). D will use the configured Dir track. This directory will only be correct if configured correctly or if the diskette directory has been read. If you enter an up-arrow ((↑)) for the track, then PowerTool displays the highest configured track on that diskette.

You can also use two special symbols in front of the drive number. The first special symbol is a pound sign (#). This causes PowerTool to identify the diskette's density automatically, in case you are unsure.

The second special symbol is an exclamation point (!). This forces PowerTool to examine the disk and determine which operating system formatted it. This process takes a few seconds. This option is also available at any other routine which reads a diskette's directory.

After you press (ENTER), PowerTool displays the requested sector on your screen. It will look something like this:

```
#    00#FE11 3ED0 D3F0 2102 0022 EA43 AF32 EC43##.>###!..”#C#2#C
HEX  10#CD3E 43FE 0128 0CFE 0220 E7CD 3E43 CD35##>C#.(.#. ##>C#5
DRV  20#43E9 FFCD 3E43 D602 47CD 3543 CD3E 4377#C###>C#.G#5C#>Cw
 1   30#2310 F918 DBCD 3E43 6FCD 3E43 67C9 C5E5##.#.##>Co#>Cg###
TRK  40#3AEC 43B7 202E 0609 C5CD 7F43 C1E6 1D28#:#C# ...##C##.(
 00  50#133E 00D3 F010 F13E 17CD 3300 21ED 43CD#.>###.#>.#3.!#C#
TRU  60#1802 18FE 2AEA 432C 7DFE 1338 032E 0124#...#*#C,}#.8...$
 00  70#22EA 43AF 6F26 4D3C 32EC 437E E1C1 C9CD#”#C#a&M<2#C~####
SEC  80#C543 01F3 003E 8103 F457 21B7 4322 4A40##C.#.>###W!#C”J@
 01  90#3EC3 3249 40F3 3EC0 D3E4 1E02 2100 4D3E#>#2I@#>###..!.M>
RPT  A0#84D3 F0CD E043 DBF0 A328 FBED A27A F640######C###(###z#@
UDF  B0#D3F4 EDA2 C380 43E1 AFD3 E43E 81D3 F4CD#######C####>####
     C0#E643 DBF0 C93E 81D3 F42A EA43 7CD3 F33E##C###>####*#C¦##>
     D0#1CD3 F0CD E643 DBF0 CB47 20FA 7DD3 F2C9#.####C###G #}###
     E0#F5F1 F5F1 F5F1 F5F1 00C9 0200 0017 4552#########.#....ER
+00  F0#524F 5200 0000 0000 0000 0000 0000 1328#ROR...........(
```

This display contains a wealth of information. The leftmost column contains information about the display. HEX refers to the current modification mode base (DAM,— see below). DRV and the number directly beneath it refer to the drive number just accessed. TRK and the number beneath it refer to the current physical track number, TRU and the number beneath it refer to the actual track number written on the disk, and SEC and the number beneath it is the sector being displayed on the screen.

Below the sector number is a three-letter code which identifies the particular data address mark (DAM) written on the disk when it was formatted. PowerTool can identify four types of data address marks: STD, for standard, DDT, for deleted data, RPT for ''read-protected'', and UDF, for user-defined.

Below the data address mark identifier is the density of the diskette being examined. This will normally be 0SD for single-density Model I diskettes and 0DD for double-density diskettes. The number preceding the SD or DD will normally be 0 and indicates that you are reading the front side of a diskette. If you were reading a double-sided LDOS diskette, the number will change to 1 when you read the back side. (All of this presupposes you have the proper hardware, of course. You can't read double-sided disks in a single-sided drive!)

At the very bottom of the leftmost column of information you will see +00. This indicator is used by the bit manipulation feature, which we will explain a little later.

The next column gives you the relative byte count (in hexadecimal) of the data immediately to the right of the graphics border. ''Relative byte count'' simply means the position of a particular byte with respect to the first position, designated 00. Each row of the display shows 16 bytes with their values in hexadecimal format. Each group of four hexadecimal digits represents two bytes.

PowerTool displays the sector data between two heavy vertical lines. To the right of each row is the ASCII representation of these bytes. On the Model I display the ASCII codes 0-31 on the ASCII side of the display will be indeterminate. On the Model III display the ASCII codes 0-31 will be displayed as the Model IIIs special character set. Non-displayable ASCII characters are converted to periods on screen print only. Also, if you are using a machine not equipped with lower case, any lower case alphabetic characters are shown in upper case; however the HEX value will always be accurate.

Paging

You can use the arrow keys to move, or "page", across sectors. The right-arrow key pages forward one sector, the left-arrow key pages back one sector. Pressing the up-arrow key pages to the same sector on the next higher-numbered track (unless you are on the highest track already). Pressing the down-arrow key pages to the same sector on the next lower-numbered track (unless you are at the lowest track already). In addition, there are a number of other paging controls, given in the table below.

Key	Action
→	pages one sector higher (or to the lowest sector of the next track if the current one is the last for this track)
←	pages one sector lower (or to the highest sector of the preceding track if the current one is the lowest for this track)
↑	pages one track higher, same sector
↓	pages one track lower, same sector
SHIFT →	pages one sector higher but will not leave current track
SHIFT ←	pages one sector lower but will not leave current track
SHIFT)	displays highest sector on current track
SHIFT (	displays lowest sector on current track
SHIFT ↑	pages to the same sector on the highest track (defined in the configuration tables)
SHIFT ↓	pages to the same sector on the lowest track
R	displays Track 0, Sector 0 (Sector 1 on Model III TRSDOS)
T	displays prompt for new Track, Sector
S	displays prompt for new sector
CLEAR	displays prompt for new Drive, Track, Sector
. or >	pages to the next higher Identifiable sector
, or <	pages to the next lower Identifiable sector
number keys 0-9	displays the correspondingly-numbered sector on the current track
BREAK	returns you to the ZAP Tools menu
SHIFT BREAK	returns you to the main menu

As you can see, the ZAP tool provides you with tremendous flexibility in searching through a disk. You can view any sector on the disk with relative ease.

Modifying the Contents of a Disk Sector

The ZAP tool gives you the powerful ability to directly modify the contents of a disk sector. You should undertake this with great care, since careless modification can make a file or a disk totally useless for other purposes.

To modify sector data, first display the sector you wish to modify using the instructions given above. Then look at the leftmost column on your screen. On the second row is the word HEX. PowerTool's ZAP tool gives you the ability to modify the contents of a disk sector in either Hexadecimal, Decimal, Octal, Binary, or ASCII form.

The default base is hexadecimal (HEX). To change to decimal, press (D), and PowerTool replaces HEX with DEC. To change to ASCII, press (A), and PowerTool displays ASC. To change to octal press (O) or (Q), and PowerTool displays OCT. To change to binary mode press (B), and PowerTool displays BIN.

To begin modification, press (M). PowerTool displays two blinking cursors, one in the sector data portion of the screen, and the other in the corresponding position in the ASCII display to the right. PowerTool displays the current position of the cursor, in relative byte format, at the upper left of the screen.

You may position the cursor anywhere in the data portion using the arrow keys. The (→) moves the cursor one byte to the right, the left arrow moves it one byte to the left. The up and down arrows move the cursor up and down the rows.

By holding down the (SHIFT) key and pressing one of the arrow keys, you can position to the far ends of the display. For example, pressing (SHIFT) and → positions the cursor to the rightmost byte on that row.

Other cursor positioning keys are listed in the table below:

Key	Action
(SHIFT)(ENTER)	reset modification mode (permits reselection of H, D, O, Q, B, or A)
→	moves cursor one byte to the right
←	moves cursor one byte to the left
↑	moves cursor up one row
↓	moves cursor down one row
(SHIFT)→	moves cursor to last byte of row
(SHIFT)←	moves cursor to leftmost byte of row
(SHIFT)↑	moves cursor to top row
(SHIFT)↓	moves cursor to bottom row
(CLEAR) or (S)	returns cursor to relative byte 00 (leftmost byte of top row)
(Q)	moves cursor to the last byte of the bottom row
G relative byte no.	moves cursor to the specified relative byte location (not active if in ASCII modification mode)
L numeric input	moves the cursor to the next occurrence of the specified numeric input, that is, "L3F" will place the cursor on the next occurrence of 3F in that sector (not active in ASCII modification mode). The numeric input MUST be in the current modification base.
+ numeric input	moves the cursor numeric data bytes forward from its current position.
− numeric input	moves the cursor numeric data bytes back from its current position.

When you have positioned the cursor to the byte that you want to change, you may enter the new value. The format of this new value depends on the current modification mode. For example, to modify text you may want to use the ASCII mode, so you can type in the letters or numbers directly. If you are modifying a program, you may want to use the hexadecimal or octal mode. If you are modifying data flags you may want to use the binary mode.

In the ASCII mode, all keys on the keyboard except for the arrow keys, (BREAK), (CLEAR) and (ENTER) are valid for input. In the hexadecimal mode, numbers and the letters A-F are valid input, in the decimal mode, numbers from 0-9 are valid input, in the octal mode, numbers from 0-7 are valid input, and in the binary mode, the numbers 0 and 1 are valid input.

No matter what modification base you are using, the display always appears in hexadecimal, with the corresponding ASCII equivalents on the right.

Once you begin to modify a byte, the old data disappears until you have entered enough data for a complete byte. For example, if you are using the binary mode, this would be eight characters. If you wish to terminate the input early before entering all the digits, simply press (ENTER), and your input is right justified in the byte. To abort input type any invalid digit. For example, in binary mode type 111 (ENTER) (which equals 7). To exit binary mode without any change type 111 and <any invalid digit>.

Special Keys

While modifying sector data, you have access to several special function keys. The keys are valid in any modification mode. These keys are shown in the following table:

Key	Action
>	insert data at the current cursor position and move the rest of the data one byte to the right, losing the final byte of the sector.
<	delete the data beneath the cursor and shift the rest of the data one byte to the left, moving a zero to the final byte of the sector.
(ENTER)	terminate modification mode.
(BREAK)	abort modification mode, return to ZAP Tools menu.
(SHIFT)(BREAK)	abort modification mode, return to main PowerTool menu.

The following keys are valid in any modification mode except ASCII.

P numeric input	copy the byte beneath the cursor the specified number of times to the right. Numeric input is in the current modification base.
(Z)	zeroes out display and holding buffer.

Pressing (ENTER) terminates modification mode. PowerTool now prompts you:

U>pdate, C>ancel, R>eturn to Modify?

Pressing (ENTER) or (U) writes the new sector out to disk, and then redisplays the sector data. Pressing (C) cancels the modification session and displays the old section with the original data intact. Pressing (R) returns you to the sector display without writing the sector to disk, however, any changes you have made are displayed.

Bit-manipulation Operations

While you are in paging mode, you may perform various bit operations on the displayed data. These operations are analogous to those performed by the Z-80 assembly language instructions RLCA, RRCA, SLA, SRL, AND, OR, XOR, ADD, and SUB.

To execute a bit manipulation, you must first be in the paging (not modification) mode. The operation is carried out on ALL the displayed data bytes simultaneously. To begin the operation, press (@). PowerTool displays the prompt:

DCR
_____

(DCR stands for "decryption" — this routine can be used to investigate sectors which may have been encrypted either by bit shift, logical or increment/decrement operations.)

Now to perform the operation, type in one of the commands below, followed by any required numeric input. You must identify the base of the number by appending H, D, O or Q, or B to the number) (hexadecimal is the default).

RR number 0-7	Rotates the bits right the given number of times
RL number 0-7	Rotates the bits left the given number of times
SR number 0-7	Shifts the bits right the given number of times
SL number 0-7	Shifts the bits left the given number of times
A number 0-255	AND the displayed bytes with given input
O number 0-255	OR the displayed bytes with given input
X number 0-255	XOR displayed bytes with given input
+ number 0-255	ADD the given input to each byte and display the result, modulo 256
− number 0-255	SUBTRACT the given input from each byte and display the result modulo 256

You may request to PowerTool that it update the entire screen by preceding the command with a colon (:). Preceding the command with an asterisk, or else nothing at all, tells PowerTool to update only the ASCII side of the display. After updating the screen, PowerTool displays the offset in the lower left hand corner.

For example, to rotate the data in the display 1 bit to the right, updating both sides of the screen, type:

:RR1 (ENTER)

PowerTool immediately updates the screen. As a second example, to logically AND the data on the screenwith the number ØFØH, type:

AFØH (ENTER)

PowerTool ANDs all of the data with the hexadecimal number, and updates the ASCII display (not the hexadecimal display — remember that * is the default value).

In addition, you can command the computer to increment or decrement the display successively. This may be useful if you wish to see if any ASCII words in the displayed sector have been encrypted by adding or subtracting a constant value to its ASCII value.

Typing in an up arrow (↑) (ENTER) at the 'DCR' prompt causes PowerTool to increment the bytes of the display about twice a second. Typing in a down arrow (↓) (ENTER) causes PowerTool to decrement the bytes of the display at the same speed. As PowerTool updates the screen, it displays the current offset in the lower left hand corner of the screen.

You can control the speed that PowerTool increments or decrements by entering a value from 1 to 255 after the down or up arrow. 1 is the fastest speed, 255 is the slowest speed. To pause the operation, press the spacebar — to start it back up, press (ENTER). To cancel the operation, press (@).

After completing an operation, to store the new sector data, first press (@) to get the DCR prompt. Answer the prompt with an exclamation point (!) and (ENTER). This resets the offset display in the lower left hand corner of the screen to +ØØ and changes all the data to the decrypted characters. To write the new sector to disk, press (M), and then press (ENTER) twice.

If you want to return the screen to its original form, simply press (ENTER) in response to the DCR prompt. This updates the screen and resets the offset display.

Verify Sectors

This tool scans all or part of a disk for conditions which would produce I/O errors. This routine does not check the data itself, but rather whether or not sectors are readable.

To use the routine, press (2) from the ZAP Tools Menu. PowerTool prompts you for the drive, track and sector, as well as the total number of sectors, which you want to verify. The program then reads these sectors and report any errors which it encounters.

Compare Sectors

To compare sectors, from the ZAP Tools Menu, press (3). This tool compares the contents of two different sectors and is useful, for instance, if you want to verify that a backup operation made an exact copy. This routine will also check for data address mark mismatches.

PowerTool prompts you for drive, track and starting sector number of the source disk. After you have entered suitable values, the program asks you for the number of sectors to compare. When you have entered this number, your PowerTool prompts you for the drive, track and starting sector location on the destination disk for the comparison operation.

After this, PowerTool asks you whether you want prompts for disk mounts (in case you have only one drive in which to do the comparison or want to check drive hardware by reading the same disk on two drives).

After comparing the sectors, PowerTool reports the number of data and data address mark (DAM) mismatches. The compare halts at the first byte mismatched in each sector.

Copy Sectors

This tool copies sectors from one disk to another, or from one location on the disk to another. To perform this routine, from the ZAP Tools Menu, press (4). PowerTool prompts you for the drive, track and starting sector of the source disk, and the number of sectors to be copied. It then prompts you for the drive, track and starting sector number of the destination disk.

After this, PowerTool asks you whether you want prompts for disk mounts (in case you have only one drive in which to do the comparison or want to check drive hardware by reading the same disk on two drives).

PowerTool copies the data over to the new location, leaving the original locations intact. The track and sector identification data are not copied, but the correct DAMs and the actual contents of the sectors are transferred.

(Note: This operation is not reflected in the diskette's directory.)

Copy Sector Data

This tool copies bytes from one sector to another location on the disk. To perform this routine, from the ZAP Tools Menu press (5). PowerTool prompts you for the drive, track and sector of the source disk, and the relative byte count in that sector that the desired data is located. PowerTool then prompts you for the number of bytes you want to move (1-65535), and then for the drive, track, sector, and relative byte count of the location you want to move the data to.

Finally, PowerTool asks you whether you want prompts for disk mounts (in case you have only one drive in which to do the comparison or want to check drive hardware by reading the same disk on two drives). PowerTool then copies the requested bytes into the new sector(s).

Note: When copying sectors and sector data, the destination sectors must not overlap in the forward direction if you are copying to the same disk and track.

Example:

Source 0,0,0
Destination 0,0,1
Count 1000
will not operate as expected.

Source 0,0,1
Destination 0,0,0
Count 1000
will operate properly.

Zero Sectors

This tool totally removes the data from the specified sectors, setting the entire contents of the sectors to 00 and resetting the data address marks to STD. To use this tool, from the ZAP Tools Menu, press (6). PowerTool prompts you for the drive, track and starting sector number for the operation, along with the number of sectors to zero.

Be careful with this routine!!! After pressing (ENTER) the operation is immediately carried out, and there is absolutely NO chance of recovering the data once a sector has been zeroed out!

Reverse Sector Data

This routine simply takes the data of a specified sector and reverses it, so that the byte that is in relative position 00H is now in relative position FFH and so on. This routine may be useful in creating disk protection schemes for disk files.

To use this tool, from the ZAP Tools Menu, press (7). PowerTool then prompts you for the drive, track, and sector, and then for the number of sectors that you want reversed. After the sector data is reversed, PowerTool reports any errors.

Exchange Sectors

This routine exchanges the data contained in one or more sectors with the data in another set of sectors. To use this tool, from the ZAP Tools Menu, press (8). PowerTool prompts for the source drive, track and starting sector number, along with the number of sectors to exchange. It then prompts for the destination drive, track and starting sector number and whether or not you want to be prompted for disk mounts.

After pressing (ENTER), PowerTool exchanges the sectors, and then reports the success or failure of the operation.

String Search

This tool searches up to an entire disk for a given ASCII string, byte list or word list (a word equals two bytes, a total of 16 bits). You may optionally specify a replacement string which the routine will insert in place of the target string whenever the target string is found.

To use this routine, from the ZAP Tools Menu, press (9). PowerTool prompts for the drive, track and starting sector for the search, and the number of sectors to search. Next, PowerTool prompts you to enter the search string, and lastly, the replacement string (if you do not want to perform any replacement, press (ENTER) in response to this prompt).

Some Special Notes on Strings

If the length of the replacement string is shorter than that of the search string, the replacement string is left justified into the search string, and it only replaces as much of the search string as necessary. If the replacement string is larger than the search string, the replacement string is truncated on the right as needed.

To search for an ASCII string, simply enter within quotes when prompted. To either match upper or lower case only, use double quotes; to match all occurrences without regard to case, use single quotes. For example, if you wanted to search for an occurrence of the name ''Jim'', you would enter:

''Jim'' (ENTER)

If you wanted to find occurrences of JIM and Jim, you would enter:

'jim' (ENTER)

If you wanted to find all occurrences of ''Jim'' and ''Tim'' on the disk, you could enter:

''?im'' (ENTER)

in response to the prompt. The ''?'' is a wild-card character, meaning, in effect, ''I don't care what's in this position, match it with whatever's there and show it to me anyway.''

After searching the disk, PowerTool reports the location of each match and the total number of matches found.

The search string must be contained wholly within a sector; that is, part of it cannot reside in one sector with the rest in the next sector. If this is the case, the routine will not find it. (If you suspect that this may be the case, try performing a search on only a portion of the string.)

To search for byte strings, when you are prompted for the search string, enter a series of values in the range Ø to 255 in any valid numeric base separating each byte by commas or spaces. For example:

10H,13H,CDH

The word search is slightly different. This type of search is normally used to locate sixteen-bit address references in a machine language program. The Z-80 stores such addresses in reverse order, that is, the least significant byte first, followed by the most significant byte. Thus, the address reference 7F42H would be stored in a machine language program as 42 7F.

The word search routine assumes this to be the case, so that when you enter a two-byte value, it automatically reverses them before starting the search. Thus, if you wanted to find the two bytes 7F 42 in that exact order, you must enter them as 427FH or, alternatively, use the byte search mode to look for the byte pair 7FH,42H.

Note, however, that this word search will not work when you are looking for byte pairs that start in ØØ, for example ØØ33H. The reason for this is that when PowerTool evaluates your ASCII input into binary, it will skip over leading zeroes. Thus if you were to enter ØØ33H, it would be evaluated as 33H. For these cases you must use a BYTE search.

The word search is specified by entering 16-bit values on the prompt line in any valid numeric base, although due to the reversal which takes place it is probably easier to use hexadecimal (easier for YOU, not the computer). The values should be separated from each other by commas or spaces. For example:

7F42H,CD30H,402DH

It is also possible to mix different modes in a single search or replacement string. That is, you can use ASCII strings, byte, and words all on the same line. For example:

''Test'',42H,79E0H,'disk'

is a perfectly valid string search specification.

Sector Search

This tool searches out duplicate sectors on one or more disks. To use the tool, from the ZAP Tools Menu, press (0). PowerTool prompts you for the drive, track and sector number of the sector that is to form the search template. You will then be asked for the drive, track, and starting sector number for the search, followed by the number of sectors to search. When the routine completes, PowerTool displays the total number of matches that it found, and their locations on the target disk.

Read ID Address Marks

This routine will examine a disk and identify the track and sector data written on the target disk. This will identify any false or non-standard sectors on the disk and help you determine just how a disk was formatted.

To use this tool, from the ZAP Tools Menu, press (A). PowerTool prompts you for the target drive to examine, and the program will then start reading track 0 and present the information on the display. The information will scroll by very rapidly, but you can stop it at any time by pressing the spacebar. Pressing the spacebar again brings up the next sector, and pressing (ENTER) resumes the fast scrolling.

To advance up one track press (↑) and to move down one track press (↓). You may proceed directly to the highest formatted track on the disk by pressing (SHIFT) (↑) or to the lowest track on the disk by pressing (SHIFT) (↓).

To exit the routine, press (BREAK).

If the routine encounters an unformatted track, it will display the error message "ID Read Error" or "TRACK NOT FORMATTED."

Initially, the routine will display seven columns of information, as follows:

```
######################################################################
# **  PowerTool  **  Ver 01.00.00  ##  By: Kim Watt  ##             #
# Copyright # 1983 Breeze/QSD, Inc. - Licensed to Tandy Corp.  #
##Source  Track Head Sector Length CRC1  CRC2  ckCRC IBM Data###
# :1D'= 02  02    00   09     01    9EH   CDH                    #
# :1D'= 02  02    00   04     01    E8H   91H                    #
# :1D'= 02  02    00   02     01    42H   37H                    #
# :1D'= 02  02    00   03     01    71H   06H                    #
# :1D'= 02  02    00   18     01    41H   44H                    #
# :1D'= 02  02    00   13     01    52H   09H                    #
# :1D'= 02  02    00   14     01    07H   5AH                    #
# :1D'= 02  02    00   12     01    61H   38H    YY   Y   RPT  #
# :1D'= 02  02    00   10     01    CBH   9EH    YY   Y   RPT  #
# :1D'= 02  02    00   11     01    F8H   AFH    YY   Y   RPT  #
# :1D'= 02  02    00   15     01    34H   6BH    YY   Y   RPT  #
######################################################################
```

The leftmost column is labeled Source. This gives you information as to which drive you are reading, and the density of the disk in that drive. For example, :1D' = Ø1 indicates that the disk in drive 1 is double-density, and the read/write head is positioned over physical track 1.

The next column is labeled Track, and this is the track number which is actually recorded on the disk. Some protected disks use non-standard track numbers, and if the disk you are reading is one of these, the track column would not necessarily agree with the actual track number listed under Source.

The third column, Head, is the head number actually recorded on the disk.

The fourth column is Sector. These are the sector numbers as recorded on the disk, and will not appear in any special order. Sectors which are physically located side by side on a disk are not numbered consecutively. This is due to the "skip" factor — under normal I/O conditions, the disk spins a given distance in between reads and writes, and skips over consecutive sectors. Also, some protected disks assign false sector numbers to prevent them from being read by standard operating systems.

The next column, labeled Length, is a coded indication of the amount of data contained in the sector. Under the IBM convention, which TRSDOS and LDOS use, a length of ØØ means 128 bytes per sector, and a length of Ø1 means 256 bytes per sector. If you are scanning a standard disk, this should display Ø1.

If the disk is not formatted with the IBM conventions, then a ØØ length byte indicates 1Ø24 bytes per sector, and a non-zero length byte multiplied by 16 yields the actual number of bytes in the sector. By this convention, a non-IBM length byte of 1ØH (16 decimal) would be equivalent to an IBM length byte of Ø1.

Note that the length code indicates the number of bytes as recorded on the disk's data fields; it does not necessarily mean that there are that many bytes actually present in the sector (again, this may be used by disk protection schemes to lay a false trail).

The next two columns are labeled CRC1 and CRC2. CRC stands for "Cyclic Redundancy Check," and the two bytes (CRC1 and CRC2) are the result of a calculation made by the floppy disk controller (FDC) chip on the ID and on the data fields of the sector, respectively. When a sector is written to disk, the respective CRC calculations are stored at the end of the ID and data fields of the sector.

When a sector is read in by the computer, the disk controller chip recalculates the CRCs and compares them with those recorded on the disk. If the bytes recorded on the disk fail to match those recalculated by the FDC, it signals an error condition, which normally appears on your TRSDOS or LDOS display as "CRC error" or "Parity error during read." Ocassionally these errors may come up even though the track you are reading is good. If this happens only once or twice as you scan a track, don't worry about it. If it happens continuously, then the track is really bad.

Planting false CRC bytes is a common form of disk read protection. Inserting an incorrect CRC on the disk forces standard operating systems to signal an error when attempts are made to read the disks.

Pressing (X) while the display is scrolling produces an additional three columns of information on the screen. The first, immediately to the right of the CRC2 column, is labeled ckCRC. This is the result of the Floppy Disk Controllers own recalculation of the CRC bytes on the basis of the actual data in the sector. The result is indicated by two letters. The letter to the left pertains to the ID field's CRC, and the letter to the right pertains to the data field's CRC.

A "Y" indicates that the CRC byte agrees with the FDC's recalculation, and an "N" means that there is a discrepancy between the recorded CRC byte and the actual byte arrived at by recalculating on the basis of the sector data. If the ID field CRC is bad, then the data field CRC check appears as "*" since it is impossible to check the data.

The next column, labeled "IBM," is an indicator of whether or not the track was formatted using IBM conventions, and will help you in determining the meaning of the LENGTH code.

Finally, the Data column tells you what data address mark was used in formatting. This will display STD for standard, DDT for deleted data, RPT for "read protected," and UDF for user-defined. As we said before, the names do not really mean anything and are used simply to differentiate one type of address mark from another.

> **NOTE:** when X is pressed to bring up the additional three columns of display, the keyboard will tend to become sluggish as it will not be scanned as often as when the normal display is scrolling past. Thus if you wish to use the arrow keys to move to another track, for example, it is necessary to hold these keys down for several seconds before the desired effect is achieved. If you wish to stop scanning with the three columns, press (X) again to return to the normal display.

Alter DATA Address Marks

This routine alters the data address marks on a diskette to something other than the standard marks used by the TRSDOS or LDOS systems, and may be a good way to produce your own "protected" diskettes.

To use this tool, from the ZAP Tools Menu, press (B). PowerTool prompts you for the drive, track and starting sector for the alteration, and the number of sectors to alter. Then PowerTool prompts you for the type of data address mark you wish to use. Answer "S" for standard, "R" for read-protected, "D" for deleted data, and "U" for user-defined. If you are using a Model III, remember that you can only use the standard and read-protected DAMs.

When PowerTool has completed writing the DAMs, it reports any errors that occurred. Again, care should be taken when using this option, as you can wind up with a disk that no operating system can read correctly.

5/ PURGE Tools

The PURGE Tools operate mainly on the directory of your system or data diskettes. They provide a fast and convenient way of killing and recovering files from a disk, facilities for checking your disk directories for errors, changing file names, and even removing all traces of a file from a disk by zeroing out the sectors which it occupied.

To use the PURGE Tools, from the Main Menu, press (P). PowerTool displays the PURGE Tools Menu:

```
#####################################################################
# ##  PowerTool  ##  Ver 01.00.00  ##  By: Kim Watt  ##             #
# Copyright # 1983 Breeze/QSD, Inc. - Licensed to Tandy Corp.       #
#####################################################################
# ##  PURGE Tools  ##                                               #
#                                                                   #
# 1.  Kill Selected Files        6.  Zero Unused Entries            #
# 2.  Kill by Category           7.  Zero Unused Granules           #
# 3.  Remove System Files        8.  Change Disk Name               #
# 4.  Remove All Passwords       9.  Change File Parameters         #
# 5.  Disk Directory             0.  Check Directory                #
#                                                                   #
# Choice? __                                                        #
#                                                                   #
#                                                                   #
#####################################################################
```

To select one of the PURGE Tools, press the number which precedes it in the display. You needn't press (ENTER).

Kill Selected Files

This tool scans the directory of a diskette and ''tags'' certain files for removal. Note here that the KILL commands of single-density Model I TRSDOS and Model III TRSDOS remove all traces of a file from the directory, so that you may not see any inactive file entries on the display.

PowerTool's purge routines, as well as the LDOS and double-density Model I TRSDOS KILL commands, do NOT remove all traces of a killed file from the directory in order to leave open the possibility of recovering killed files if needed. The granule space, however, is returned to the DOS for re-allocation.

To use the tool, from the PURGE Tools Menu, press (1). After prompting for drive number, PowerTool analyzes the directory and reports any irregularities. It then asks you to press any key to continue.

After you press any key, PowerTool displays the files on the screen — active files are shown surrounded by left and right arrows (left and right square brackets on the Model III) and the non-active file entries (killed files) are shown surrounded by heavy vertical lines.

To label a file as inactive, position the cursor to the filename with the arrow keys (or with (N) — for Next), then press (K). Now the file is tagged as inactive (indicated by the heavy vertical lines that surround the filename).

To completely remove a file from the directory, rather than label it as inactive, press (C) instead of (K). This physically removes all traces of the file entry from the directory. Pressing (CLEAR) removes ALL unused directory entries displayed.

To restore an inactive file, move the cursor to the filename and press (R). However, you should exercise caution when recovering killed files, as other files may have overwritten parts of it. There is no guarantee that a recovered file is still intact, unless you recover it before any other write operations to the disk have taken place.

If there are more files than can be shown on one screen display, you may advance to the next "page" by pressing (SHIFT) (↑), or return to a previous "page" by pressing (SHIFT) (↓).

To write the changes out to the disk, press (W). PowerTool then asks you to confirm the write — answering "Y" writes out the new directory track; answering "N" leaves the directory track intact.

You may kill files in multiple directories by inputting more than one drive number when initially asked for the drive. In this case, after writing out the directory of the preceding drive, the next directory is then read in and displayed. If you do NOT want to write the directory to a disk, press (A) (instead of (N)) to advance to the next drive.

Kill by Category

This option kills certain classes of files from a disk. You can kill files based on their extension, part of their name, or by their attribute (invisible, etc.).

To use this tool, from the PURGE Tools Menu, press (2). PowerTool prompts you for the drive, and then for the common category. You may now enter an extension, a wildcard specification, or an attribute.

To kill by extension, simply enter a slash followed by the desired extension. For example, entering /BAS kills all files with the /BAS extension.

> (Warning: On a single-density Model I TRSDOS disk, and on all LDOS disks, if you specify /SYS, you will kill not only all system files, but also the BOOT/SYS and DIR/SYS entries in the directory as well! This is an easy way to make a disk unusable! If you want to remove the system files, use the wild-card characters option, described in the next paragraph, or the "Remove System Files" option.)

To kill by wildcard name, enter a letter or a group of letters. This kills all files which start with that letter or group of letters. For example, if you were to enter B, you would kill the files BASIC/CMD, BACKUP/CMD, BULLDOG/BAS, BEAMIN/TXT, etc. If you entered BA, however, you would only kill the files BASIC/CMD and BACKUP/CMD.

To kill a class of files based on their attributes, you must enter a space as the first character, followed by "I" for invisible files, "V" for visible files, "S" for system files, or "P" for files with a protection level other than 0. You may enter more than one attribute, separated by commas or spaces.

Remove System Files

This tool kills the system files on a diskette. All system files except for BOOT/SYS and DIR/SYS are killed (TRSDOS single-density Model I and LDOS disks, all system files will be killed).

To use this tool, from the PURGE Tools Menu, press (3). PowerTool prompts you for the drive number, and then inactivates the system files, reporting the results when done.

Remember that PowerTool purge routines do not zero out the directory entries of killed files. These files may be reinstated as active files using the Restore command of Kill Selected Files or the disk repair utility. However, this is safe ONLY as long as no normal disk writes have been done to the disk. The exception to this is with double-density Model I TRSDOS and Model III TRSDOS system files which cannot be restored.

Disk Directory

This tool displays information about the directory including its name and date, the number of tracks it was formatted for, the number of free granules remaining, and the number of free file entries available in the directory. In addition, it displays all valid files on the specified diskette(s), including protection level (if any), and the file attributes (whether it is a system file, an invisible file, or a visible file).

To use the routine, from the PURGE Tools Menu, press (5). PowerTool prompts you for the drive number. After entering the drive number, PowerTool displays the disk name, format date, the number of tracks, free granules, free bytes, and free directory slots, and finally a list of the active files in the directory. The information with regard to the file's attributes may be to the right of the file name.

Here, "S" refers to system files, "I" refers to invisible files, and "P=" followed by some number (Ø-7) refers to the protection level. Files without attribute information are simply visible user files with a protection level of zero.

For example:

SYSØ/SYS SIP=7

means that the file SYSØ/SYS is a system file (S), is invisible (I), and has a protection level of 7 (no access).

If the disk directory has more entries than can be displayed on one page, the display automatically stops. Press (ENTER) to view the next screen page. If you specified more than one drive at the initial prompt, the next drive's directory is displayed automatically upon completion of the previous one.

If you view a Model I double-density TRSDOS or Model III TRSDOS diskette, you will note that no system file names are displayed. TRSDOS does not log the system files in the directory in the normal manner, so that they aren't part of a normal directory listing. Instead they are displayed as numbers, for example,

ØØ-Ø1-Ø2-Ø3-Ø4-Ø5-Ø6-Ø7-Ø8-Ø9-1Ø-11-12-13-14

These numbers refer to operating system modules stored on the disk. Since they are accessed only by the operating system, they don't need conventional names or entries.

Zero Unused Entries

This tool removes any trace of inactive files from the directory. To use it, from the PURGE Tools Menu, press (6). PowerTool prompts you for the drive(s) to be operated on.

Since single-density Model I TRSDOS and Model III TRSDOS automatically zero out a directory entry when the KILL command is issued, the usefulness of this utility is in cleaning up LDOS and double-density Model I TRSDOS directories, and directories on which PowerTool was used to kill off certain files.

Zero Unused Granules

This tool removes ALL traces of killed files from a disk by writing zeroes to any unassigned granules. To use this tool, from the PURGE Tools Menu, press (7). PowerTool prompts you for the drive(s) to be cleared.

> **NOTE:** Once this utility has been run, no data recovery of any kind is possible for killed or inactive files! This does not effect the directory records.

Change Disk Name

This tool lets you change the disk ID information. To use it, from the PURGE Tools Menu, press (8). PowerTool prompts you for the drive number(s). PowerTool then reads the directory, and prints out the old disk name and format date.

You are then sequentially prompted for a new name, a new format date, and a new AUTO command (if you are reading a system diskette). You may change any of these by typing in an appropriate response.

Pressing (ENTER) alone, in response to Name and Date leaves them as they were. Pressing (ENTER) for the AUTO command disables the AUTO command.

Change File Parameters

This tool lets you change the name and attributes of your disk files. To use the tool, from the PURGE Tools Menu, press (9).

PowerTool prompts you for the filename. It then asks you for the new filename (pressing (ENTER) leaves it unchanged), and the attribute (''V'' for visible, ''I'' for invisible, or ''S'' for system — pressing (ENTER) makes the file visible).

Check Directory

This tool scans the directories on the specified drives for errors. To use the tool, from the PURGE Tools Menu, press (0) and respond appropriately to PowerTool's prompt for drive number(s).

PowerTool reports disk name, format date, and number of formatted tracks, free granules, free bytes, and free directory slots. In addition, PowerTool reports any error conditions in the directory. Such errors may include granules allocated to nonexistent files, HIT table entries which do not have corresponding file entries, or vice versa, or improperly linked extended directory entries. If any such errors occur, you may repair the disk by selecting the REPAIR Tools of PowerTool (See page 51).

6/ DISK FORMAT Tools

The DISK FORMAT Tools let you do a number of styles of formatting to your disk, including standard formatting for any of the accepted operating systems, single track formatting, formatting without erasing the data, and bulk erasing the disk.

To use the DISK FORMAT Tools, from the Main Menu, press (D). PowerTool responds by displaying the following menu:

```
######################################################################
# **  PowerTool  **  Ver 01.00.00  **  By: Kim Watt  **              #
# Copyright # 1983 Breeze/QSD, Inc. - Licensed to Tandy Corp.        #
######################################################################
# ##  FORMAT Tools  ##                                               #
#                                                                    #
# 1.  Standard Format            4.  Write Format Track              #
# 2.  Format Without Erase       5.  Software Bulk Erase             #
# 3.  Build Format Track                                             #
#                                                                    #
# Choice? #_                                                         #
#                                                                    #
#                                                                    #
#                                                                    #
#                                                                    #
######################################################################
```

Standard Format

This tool formats a diskette using the standard format for any of the operating systems which PowerTool recognizes. To use the tool, from the DISK FORMAT Tools Menu, press (1). PowerTool prompts you for the drive number(s), the name, date, and password for the diskette.

If you press (ENTER) in response to the prompt for the name and date, PowerTool automatically supplies the name "* Data *" and the date as "* Disk *". Pressing (ENTER) in response to the password prompt tells PowerTool to use "PASSWORD" as the diskette password. The password encoding algorithm used will be that of the last DOS accessed.

Finally, PowerTool asks you if you want to use the operating system format described in the Configuration Table. If you answer "Y" to this prompt, PowerTool refers to the Configuration Table for the pertinent information. If you answer "N", you get additional prompts asking you to define the format further.

A further prompt will appear as:

:d Type, Tks, Dir, St Tk ?

where :d is the target drive, Type is the operating system type, Tks is the total number of tracks, Dir is the track location of the directory, and St Tk is the track number that you want formatting to begin on. These elements must be separated by a comma or a space.

There is one other command which you can input for the "Use Configuration?" prompt. Enter an exclamation point (!), which forces an immediate format; that is, PowerTool formats the diskette regardless of whether it finds data on it or not.

During the format, PowerTool first scans the disk in the target drive. If it sees something other than a blank disk, it will immediately display the message:

Drive n HAS DATA

where n is the drive you specified for the operation. If possible, it will also display the disk's name and date. PowerTool then asks you whether you want to continue or quit the operation. If you elect to continue, the old data on the disk is overwritten and the new format takes its place. A request to abort returns you to the DISK FORMAT Tools Menu. If ! was specified no prompting takes place.

After formatting is complete, you are asked, "Repeat?". You may now take the disk from the target drive and insert a blank disk. Then, typing "Y" causes PowerTool to format again. Typing "N," however, returns you to the DISK FORMAT menu.

A Note on Configuration

Providing your own configuration provides an easy way to add tracks to a diskette using the standard format. For example, if you wanted to turn a 35 track disk into a 40-track disk, without losing any data that is already on it, just place the disk in a disk drive and select standard format. Go through all the prompts until you arrive at the Use Configuration ? prompt. Now reply (N) and press (ENTER).

Answer the prompts. Make sure you give it the proper DOS type. Give "40" for the track count, "17" for the directory track and "35" for the St Tk (Start track).

PowerTool then proceeds to format the disk, starting at track 35 and moving up to track 39. You are then prompted as to whether you want to write the directory track and boot sector. (This prompt will appear only if the specified starting track number was other than 0.)

Reply "N." DO NOT REPLY "Y" or a blank directory will be written to your disk, rendering previous files on the disk inaccessible!!

To complete the process, check the configuration tables and make sure that the configuration for this drive correctly reflects the number of formatted tracks on the disk. Now go into the REPAIR Tools Menu (see page 51) and select the Repair GAT Sector option, then the repair 'Entire Sector' option. Previously locked out tracks, if any, will be reset to available. This registers the new tracks on your disk, and your 35-track disk is now a functional 40 track disk with 5 extra tracks of space.

This presupposes, of course, that your disk drive was capable of reading and writing 40 tracks in the first place. Also, in order to make use of the added space, your DOS must be capable of recognizing that it is there and is able to make use of it. Adding new tracks to a 35 track with single-density Model I TRSDOS will not help you a bit, since it cannot recognize more than 35 tracks on a disk.

Format Without Erase

This tool reformats a disk without losing any readable data that was already on it. Since all it changes is the ID field of the sector and sector gaps, and not the actual data, this routine is very handy for revitalizing disks which have been lying around for a long time. It is also a good way to repair "Sector NOT FOUND" and "DATA CRC" errors.

To use this tool, from the DISK FORMAT Tools Menu, press (2). PowerTool prompts for the drive number(s). If you prefix the drive number with an exclamation point (!), PowerTool first scans the disk for its current format. Otherwise, PowerTool uses the format as given in the Configuration Tables. As the routine proceeds, PowerTool reads in the sectors of each track and then writes them out using the new format.

If during this process, PowerTool encounters a bad sector, it presents you with the R>etry, S>kip, C>ontinuous, Q>uit ? menu. If you are trying to recover a disk through this method, use (R) or (C) as heavily as possible — don't skip a bad sector without retrying it. If PowerTool can read the sector at all, it will rewrite it back out with the correct format, thereby repairing the error, assuming the media itself is not flawed. Give PowerTool a chance to read the sector in. It may take several minutes, but if it can be done at all, PowerTool will do it — and recover your data.

When using this option, and the actual MEDIA is the error fault, a bad sector may be transposed to a good portion of the disk.

Also, the sector to sector, and track to track skewing effect will be set individually for each operating system to achieve maximum read/write speed. Diskettes formatted, or reformatted by PowerTool will normally be read faster by the operating system than if the formatting was done by the native DOS.

Build Format Track

This routine creates a copy of a new track in memory. This data is the same information used when formatting a diskette, including both ID and data field data. You will generally use this routine prior to the Write Format Track (below).

To use this tool, from the DISK FORMAT Tools Menu, press (3). PowerTool prompts you for the operating system type. Reply with one of the DOS type identifiers T1S, T1D, T3D, LIS, L3D or L1D to create the appropriate track. PowerTool then asks for the track number to assign to it. This should be the track number where you intend to write it out to the disk.

PowerTool creates the track in memory, and displays the message, "(KEY) to display buffer at nnnnH:". Pressing (ENTER) displays the track buffer in memory, and you then have use of the MEMORY Tools (See page 59). Press (BREAK) to return to the DISK FORMAT Tools Menu.

Write Format Track

This tool is the complement of the Build Format Track. It writes the data created by that routine to a given track. This routine is most helpful in repairing diskettes with only one or two bad tracks.

To use the routine, from the DISK FORMAT Tools Menu, press (4). PowerTool prompts you for the drive and track number where you want to write the new track. After receiving the vital information, PowerTool writes the new track to the diskette, replacing the old track.

You can build a track in memory with one track number, but write it out to another location on the disk. This is entirely possible, but not recommended. This option should not be used unless a proper format track image exists in memory.

Software Bulk Erase

This tool completely removes all data from a target disk. You can use it to clean up a disk in place of a bulk eraser or magnet. The tool is useful, for example, when the operating system is rejecting a backup or a format due to ''different pack IDs,''.

To use this routine, from the DISK FORMAT Tools Menu, press (5). PowerTool asks you for the drive number(s) which you want bulk erased. After receiving your input, PowerTool writes all 00's to the disk, thus removing all traces of format information and data.

Take as much care in using this routine as you would in using a bulk eraser, since there is no possible recovery!

7/ BACKUP Tool

PowerTool has a very fast backup routine which can be used in place of the standard BACKUP of TRSDOS or LDOS. It works on a one-drive system as well as on a multi-drive system. Additionally, you may specify more than one destination drive, in which case the source disk is duplicated onto any number of your drives, one at a time.

To use the Backup Tool, from the Main Menu, press (B). PowerTool prompts you for the source drive. It then asks you for the destination drive(s).

PowerTool gives you the option of formatting the destination drives before the backup. You may reply "Y" to format, "N" if the disk is already formatted, "!" to force a format even if the disk has data, or "*" to format and skip the verification cycle. If you type "Y", "!", or "*", each destination drive will be formatted using the configuration of the destination drive. The tool performs any required formatting, then copies the source diskette sector by sector to the destination diskette.

After copying the source disk to all the destination drives, PowerTool asks "Repeat?". Reply "Y" if you want to do more backups, or "N" to return to the menu.

If you backup one disk to another with a higher track count, for example a 35-track disk to a 40-track disk, the backup disk will still reflect the availability of 40 tracks rather than 35 as would result from using other Backup programs.

The BACKUP tool is the easiest way to recover the data from a disk with bad sectors. As each bad sector is encountered, you will be presented with the R>etry, S>kip, C>ontinuous, Q>uit prompt. Use "R" or "C" whenever this happens. If the sector can be read at all, it will be copied over intact to the backup disk, thereby recovering the data.

Even if not all of the sectors can be copied, you will still wind up with a usable (that is, totally readable) disk from which you can copy off as much useful data as you can. In addition, the original disk, bad sectors and all, will still be available for another try.

> **NOTE:** Backup must be used only between comparable diskette types. Although you may backup a T1S=35 to a LID=40, the desired effect will not be achieved; and the destination disk would be useless. Use 'Copy Files' with unlike disk types.

8/ REPAIR Tools

The directory contains much valuable information, hence damage to the directory can make the diskette unreadable. The REPAIR Tools are designed to restore completely or partially unreadable disks to a usable condition, if at all possible, by making certain changes to the directory.

In addition, the REPAIR Tools contain routines which will allow you to check on the condition of a directory, recover any files killed by the PURGE utility of PowerTool, move the directory track from one location to another on the disk, or clear the unused directory entries by setting them to zero.

To select the REPAIR Tools, from the Main Menu, press (R). PowerTool displays the REPAIR Tools Menu:

```
#####################################################################
# ##  PowerTool  ##  Ver 01.00.00  ##  By: Kim Watt  ##             #
# Copyright # 1983 Breeze/QSD, Inc. - Licensed to Tandy Corp.       #
#####################################################################
# ##  REPAIR Tools  ##                                              #
#                                                                   #
# 1.  Repair GAT Sector          6.  Move Directory                 #
# 2.  Repair HIT Sector          7.  Display Directory              #
# 3.  Repair BOOT Sector         8.  Check Directory                #
# 4.  Read-Protect Directory     9.  Clear Unused Entries           #
# 5.  Un-Read Protect Directory                                     #
#                                                                   #
# Choice? #_                                                        #
#                                                                   #
#                                                                   #
#####################################################################
```

Repair GAT Sector

The "GAT" (Granule Allocation Table) sector of a diskette directory is where the operating system keeps track of which granules have been assigned to files, which are available for use, and which have been locked out for one reason or another. An error in the GAT can result in the operating system assigning already used disk space to other files, thus resulting in data mixups. This is a creeping sort of error which can eventually render an entire disk worthless.

This tool repairs a bad GAT table. To use it, from the REPAIR Tools Menu, press (1). PowerTool prompts you for the drive number(s) of the GAT to be repaired. You are then asked whether you want the Allocation Table only or the entire sector repaired.

Also included in the GAT Sector are the disk name and date. So if the disk name and date are unreadable, repair the entire sector, otherwise, select Allocation Table repair only. PowerTool then rebuilds the GAT table of the target disk(s).

If "Allocation Table" only is selected, the current lockout table is copied to the allocation table, and then all files are re-allocated. None of the bytes from 60H up are effected, which includes password, name, date, auto command, and special DOS dependent data.

If ''Entire Sector'' is selected, then all tracks are set to ''available'', name and date are reset, password is set to ''PASSWORD'', NIL AUTO COMMAND, and DOS particulars are initialized. Finally, all files are re-allocated.

Note that repairing the ''entire sector'' may make previously locked out granules available to the operating system. This is, of course, desirable after using the ''Format Without Erase'' utility.

Repair HIT Sector

Each file name in a directory is ''hashed'', or coded by the operating system into a one-byte code. This code is entered into the Hash Index Table sector, in a position which defines the file's actual position in following directory sectors. This permits the operating system to find filespecs in the directory faster than if it had to search the entire directory.

Errors which may occur here are HIT codes assigned to nonexistent files, invalid HIT codes, etc. To check for HIT sector errors, use the Check Directory tool (described later in this chapter), which examines the HIT table and compares it to the actual entries in the remainder of the directory.

If a directory check reveals errors in the HIT sector, you may use this option to repair it. From the REPAIR Tools Menu, press (2). PowerTool prompts you for the drive number(s) to be repaired.

> **NOTE:** Double-density Model I TRSDOS and Model III TRSDOS code the system files in the last 48 and 32 bytes respectively of the HIT table. Since they do not use standard directory entries, all system hash codes are untouched by PowerTool during a HIT repair, because PowerTool requires standard entries for rebuilding the HIT sector. If these bytes have been corrupted, you must use the ZAP tool to effect repairs and follow this by repair GAT. Set all corrupted bytes to FFFF with an even number of bytes to offset one another.

Repair BOOT Sector

If a system disk will not boot when placed in drive 0 and the RESET button has been pushed, the boot sector has probably been damaged. This sector contains code which loads the rest of the operating system into memory, so if this sector has been corrupted the DOS will not load.

PowerTool has the facilities to repair a damaged boot sector for the following operating systems: TRSDOS 2.3 (Model I), TRSDOS 2.7DD (Model I double-density) and TRSDOS 1.3 (Model III). The one which PowerTool writes to your disk will depend on the operating system

specified for that drive in the Configuration table. For example, if drive 1 has been set with T1S then when you execute Repair BOOT Sector for drive 1, the TRSDOS 2.3 boot sector will be written to the disk in drive 1. The repair boot routine will write the boot sector data, but will not log the boot sector in the diskette directory since it will already be there anyway. All the routine does is replace the damaged sector with good data.

> **NOTE:** As with the rest of the routines in the REPAIR Tools section, repair boot assumes that the disk is readable. That is, while the sector data may be damaged, the formatting information is intact (the sector itself can be correctly read and written to). Thus it may be necessary to run Format Without Erase on your disk before executing REPAIR BOOT, to ensure that the boot sector can be properly written to.

Repair BOOT Sector should be used ONLY on the specified operating systems (TRSDOS 2.3, TRSDOS 2.7DD, and TRSDOS 1.3). Do not use this procedure on any other operating system disk (e.g., TRSDOS 6.Ø) or you will compound the problem! To repair the boot sector of a different operating system disk, follow these steps:

1. Place the damaged disk in Drive Ø.
2. Place a GOOD disk in Drive 1. This should be the same type as the disk in Drive Ø, i.e., a SYSTEM disk. Do not use a data disk. Also, do not use a system disk which has been formatted differently (different density, different directory track location) from the damaged disk. They should be as similar as possible.
3. Go to CONFIGURE System.
4. Configure Drive Ø and Drive 1 so that they reflect the same operating system and the same density.
5. Go to the ZAP tools.
6. Use Copy Sectors to copy 5 sectors (6 if double-density) start from Drive 1, Track Ø, Sector Ø, to Drive Ø, Track Ø, Sector Ø.

This procedure will duplicate the information in the boot granule of the good disk on the bad disk, and should allow your bad disk to boot properly. If you still have problems, then the operating system has likely sustained damage to places other than the boot sector (this is also true if, after executing REPAIR BOOT on any of the specified systems above, that system still fails to come up properly). In this case check the system files for damage by comparing them to a good disk.

Read-Protect Directory

Often times, when the disk drive seems to have difficulty in finding a file or if you try to display a directory and the disk drives grind along for several minutes, the chances are that part or all of the directory was re-written using the wrong Data Address Mark (DAM). This can also happen if you try to read a single-density Model I TRSDOS diskette on a Model III machine.

The directory track of a TRSDOS or LDOS-formatted diskette has a DAM which is different from all the other tracks, and is used by the operating system to identify its location. Whenever the operating system tries to read a directory and finds that some or all of the directory sectors do not have the correct data address mark, it will hunt all over the disk for a track that has the proper DAMs. It can't find it, and so your drive sounds like its head keeps moving back and forth (which is exactly what is happening!).

"Read-Protecting" a directory is a misnomer and refers to the process of writing a directory track with the correct data address mark which can be recognized by the operating system. It does NOT mean changing the directory so that it cannot be read at all. Unfortunately, the term stuck.

To rewrite the DAMs of the directory track, from the REPAIR Tools Menu, press (4). PowerTool prompts for the drive number(s) of the disks to be read-protected. After you input the drive number, PowerTool reads the directory track, then writes it back out with the correct data address marks for the computer it is running on. If it cannot find the directory, it asks you to specify its track location and sector count. You may enter the track number where the directory is located, or press (ENTER) to default to the directory track number in the configuration table and the standard length.

Remember that the Model I hardware is capable of writing four different types of DAMs, while the Model III hardware is capable of writing two types of DAMs. As it turns out, Model I TRSDOS uses a directory track DAM is one which the Model III cannot duplicate, hence you cannot read-protect a Model I TRSDOS diskette on your Model III.

LDOS uses DAMs for the directory that both the Model I and Model III hardware can duplicate, so you may read-protect a Model I diskette used with these operating systems.

Normally you would read-protect the diskette using the computer the diskette is intended to be used on. PowerTool will make the closest possible decision on which DAM to use where incompatibility exists. In other words, if you read-protect a Model I single-density TRSDOS disk on a Model III you make it directory readable to the Model III but unreadable to the Model I TRSDOS.

Un-Read Protect Directory

This tool rewrites the DAMs of a directory track as standard data address marks.* You may want to use this routine if you are developing a protected disk with a directory that you do not want any standard operating system to be able to read.

To use this tool, from the REPAIR Tools Menu, press (5). PowerTool prompts you for the drive number(s) to un-read-protect.

*If DAMs are inverted (TRS-80 Model III) the directory will be set to read-protected (opposite of normal).

Recover Killed Files

This routine recovers files previously killed using PowerTool's PURGE utilities or the Model I double-density TRSDOS or LDOS KILL command. You cannot use it to recover files killed with the standard single-density Model I or Model III TRSDOS "KILL" command, as the system zeroes out the entire directory entry when you KILL a file.

To use this tool, from the REPAIR Tools Menu, press (6). PowerTool first prompts you for the drive number(s) of the file(s) you want to recover. The routine returns the disk name, date, number of formatted tracks, and number of free granules and free directory slots on the disk.

After you press (ENTER), PowerTool shows you a list of all active and inactive files. All active files are surrounded by right and left arrows (right and left square brackets on the Model III). Inactive files are surrounded by heavy vertical bars.

To recover a killed file, move the cursor to that file using the arrow keys. (If there are more files than can be displayed on the screen at one time, you can press (SHIFT)(↑) to go the next display page, or (SHIFT)(↓) to move to the previous page.) Then press (R) and the vertical bars are replaced by right and left arrows (or brackets), indicating that the file is now a valid directory entry.

When you have recovered all the files you want to, press (W) to write the updated directory back onto the disk. PowerTool asks you to confirm your decision. Enter "Y" to confirm and write the directory, or "N" to abort the procedure.

Remember that recovering a killed file is a chancy affair if you have performed any disk writes since the time you killed the file. The file you recover may have had parts of it overwritten by subsequent files. See "Kill Selected Files" in PURGE Tools for further information. The routines are identical.

Move Directory

This option will move the directory track to any other track on the target disk as long as that track is not already allocated to another file. To use this tool, from the REPAIR Tools Menu, press (7).

PowerTool prompts you for the drive number(s) and then for the track number of where you want the directory moved. Pressing (ENTER) alone at the track prompt will display a list of free tracks. If the track you selected is already allocated, PowerTool will display a message to that effect and ask you to select another track.

After you enter a valid track number PowerTool moves the directory track, and the boot sector is changed so that it now points to the new updated directory (the position of the directory is coded in the third byte of the boot sector). Also, on TRSDOS 2.3 and all LDOS disks, the DIR/SYS entry in the directory is updated to reflect the directory's new location. Press (ENTER) at the "move to track" prompt and a list of available tracks will be displayed.

Display Directory

This tool displays valuable information about the directory. To use it, from the REPAIR Tools Menu, press (8). PowerTool prompts you for the drive number(s) you want to see.

The routine returns a full screen display of all active files on the disk(s), along with the attributes and protection levels of each file (blank attribute and protection levels indicate a visible, user file with a protection level of 0). In addition, the disk name, date, number of free granules and free space (in kilobytes, units of 1024 bytes), and number of free directory slots are also displayed.

If you view a Model III TRSDOS directory, you won't see any system file names displayed. As was discussed earlier, Model III TRSDOS does not list the system files in the directory in a normal manner. They are displayed as numbers, for example:

00-01-02-03-04-05-06-07-08-09-10-11-12-13-14

> **NOTE:** This routine has been duplicated in PURGE, REPAIR, and FILES simply for convenience. The routines are identical.

Check Directory

This tool performs a very thorough check of the directory on the target disk. To use this tool, from the REPAIR Tools Menu, press (A). PowerTool prompts you for the drive number(s).

The routine first scans the directory, and then displays the disk name, date, the number of free granules and the number of free directory slots. Following this is a listing of any errors encountered in the directory. If the routine reports any GAT or HIT errors, you may use the GAT Repair or HIT Repair options to automatically repair the directory.

Clear Unused Entries

This tool cleans up a diskette directory by completely erasing the entries of any non-active files. Normally the entries of non-active files are left in the directory by single-density Model I TRSDOS, LDOS, and the PowerTool purge routines to leave available the possibility of future recovery. After you use this routine, of course, no future recovery will be possible.

You will normally not need to use this directory to clean up files where TRSDOS 1.3 or 2.7DD (Model I double-density) did the killing, since these TRSDOS systems remove the entries of non-active files automatically.

To use this tool, from the REPAIR Tools Menu, press (0). PowerTool prompts you for the drive number(s) whose directories you want to clear up.

9/ MEMORY Tools

The MEMORY Tools of PowerTool permit you to view and manipulate memory in a variety of ways. Many of these routines are very similar to the ZAP Tools. Other tools let you jump to your own machine language programs stored in memory, and read or write to any of the CPU I/O ports.

To use the MEMORY Tools, from the Main Menu, press (M). PowerTool displays the menu:

```
######################################################################
# **  PowerTool  **  Ver 01.00.00  **  By: Kim Watt  **              #
# Copyright # 1983 Breeze/QSD, Inc. - Licensed to Tandy Corp.        #
######################################################################
# ##  MEMORY Tools  ##                                               #
# 1.  Display Memory            9.  String Search                    #
# 2.  Move Memory               0.  Input Byte from Port             #
# 3.  Exchange Memory           A.  Output Byte to Port              #
# 4.  Compare Memory            B.  Memory to Sectors                #
# 5.  Fill Memory               C.  Sectors to Memory                #
# 6.  Reverse Memory            D.  Memory to Track                  #
# 7.  Test Memory               E.  Track to Memory                  #
# 8.  Jump to Memory                                                 #
#                                                                    #
# Choice? __                                                         #
######################################################################
```

Great care should be taken when modifying memory in any of the ways made available to you from this menu. Unlike the modification routines in the ZAP Tools, changes to memory are not buffered and take effect immediately. Also, don't blindly move data around in memory, as you may overwrite part of the PowerTool Program. In general, pressing (ENTER) at any of the "Address" prompts will always default to safe user memory.

Display Memory

This tool lets you display and modify memory. To use this tool, from the MEMORY Tools Menu, press (1). PowerTool prompts you for starting memory address that you want displayed. You may enter the address in any of the four number bases that PowerTool accepts (hexadecimal, decimal, octal, or binary) — the default base is hexadecimal. If you enter no address, but just press (ENTER), PowerTool displays memory starting at the beginning of free memory.

The display is very similar to the ZAP display, with the exception that the first column of information is empty, and the actual addresses (in hex) are displayed in the second column, adjacent to the actual data. The data is displayed in blocks of 256 bytes each. You can look on this display as a window into the TRS-80's memory which can be moved around so as to present different views of the RAM or ROM "landscape."

Use the arrow keys to scroll through memory. Pressing → or ← moves the display window one byte in either direction. Pressing SHIFT→ or SHIFT← advances or decreases the display 256 bytes at a time.

Pressing ↑ or ↓ also moves the display window 256 bytes at a time in either direction. However, pressing SHIFT↑ has a different effect — it moves the highest 256-byte block of memory available to the display. Similarly, pressing SHIFT↓ displays the lowest 256-byte block of memory.

Pressing CLEAR key tells PowerTool to prompt for a new memory address to display.

As in the ZAP tool's modification mode, you can select your input base by pressing H (hexadecimal), O or Q (octal), D (decimal), B (binary) or A (ASCII) prior to entering the modification mode. Then to begin modifying memory, press M. The control keys are generally the same as with the ZAP Tools — see page 23 and the following for details.

The exception to this is that if you attempt to move the cursor above or below the display window, more data moves from above or below the currently displayed address. If you attempt to move the cursor past the highest address in the computer, FFFFH, the display wraps around to display 0000H.

> **NOTE:** Memory below 3000H on Model I (3BFFH Model III) cannot be modified, as it is Read-Only memory and memory between 3000H Model I – 3C00H Model III and 4000H should be modified only with great care. Use care when modifying below C500H.

While it is possible to display memory starting from address 0000H on up to FFFFH, there are certain regions of memory which cannot be modified. These addresses are the regions occupied by the ROMs or Read-Only Memories. By definition, read-only memories cannot be altered. Although you can enter modification mode while displaying these memory addresses, any attempt to modify these locations will be futile; the original contents will **never** change.

The addresses occupied by the ROMs are from 0000H through 3000H in the Model I, and from 0000H through 3BFFH in the Model III. These regions may be displayed but not changed.

The address region from 3000H through 3BFFH in the Model I is the I/O region. This area should be modified with **GREAT CARE** only. **Do not attempt any modifications in this region unless you know what you are doing!** This region contains the control addresses for the various peripheral devices such as the printer, the disk drives, and the RS-232 board. Indiscriminate modification in this area could have some unwanted results (such as kicking your disk drives into FORMAT TRACK mode, with the resulting destruction of your disks). BE CAREFUL!!!

In both the Model I and the Model III, the address region from 3C00H through 3FFFH is the video memory. Modifications may be made here, but the only result will be to change the appearance of the video. Since PowerTool periodically redraws the video (such as whenever BREAK is pressed) changes here are never permanent. There is little reason to modify video memory.

From 4000H on through about D3FFH is the PowerTool program itself. Modifications here should be avoided, and are done at your own risk. Any attempt to modify the program, unless you know exactly what you are doing, can introduce errors which may result in a "system crash," if not immediately, then at some later point in the session.

Move Memory

This routine enables you to move a block of memory from one location to another. To use it, from the MEMORY Tools Menu, press (2). PowerTool prompts you for the starting and ending address of the block to be moved, and the starting address where it is to be moved to. After the routine executes the memory move it advises you of the number of bytes moved. Blocks may overlap in either direction and be moved correctly.

Exchange Memory

This routine is similar to "Move Memory," except that it actually exchanges the contents of the origin and destination blocks of memory instead of merely copying the contents of the origin block over what was in the destination block.

To use this tool, from the MEMORY Tools Menu, press (3). PowerTool prompts you for the start and end address of the first block, and the start address of the second block. After completing the routine, PowerTool reports the number of bytes exchanged.

Compare Memory

This tool performs a byte-by-byte comparison of two blocks of memory and advises you of any mismatches that it finds. To use the routine, from the MEMORY Tools Menu, press (4). PowerTool prompts you for the start address and end address of the first block, and the start address of the second block. If there are many mismatches (that is, more than a screenful), you may pause the display with the spacebar, and continue it with (ENTER).

Fill Memory

This routine loads a given block in memory with a byte value. To use it, from the MEMORY Tools Menu, press (5). PowerTool prompts you for the starting and ending addresses of the memory block, and the byte value it is to use as the filler. The default value of the starting address is the byte directly above PowerTool, the default value of the ending address is the top of memory, and the default value of the byte value is zero.

Reverse Memory

This tool reverses all the bytes in a block of memory. To use this tool, from the MEMORY Tools Menu, press (6). PowerTool prompts you for the starting and ending addresses of the memory block to be reversed. On completion of the routine you will be advised of how many bytes have been reversed.

To see how reverse memory works, enter for starting address, 3C00H, and for ending address, 3FFFH.

Test Memory

This tool tests a region of memory to make sure it is working properly. The test restores the original byte of each memory location after testing it, so you can even test the area used by PowerTool without destroying the PowerTool program.

To use this tool, from the MEMORY Tools Menu, press (7). PowerTool responds with the message:

DO NOT test between xxxxH and yyyyH

where xxxx and yyyy are memory addresses. The test routine resides in this region, and if you test here, the program may destroy itself! Rest assured, however, that if the Test Memory tool itself functions, then the memory where it is stored is in working condition.

PowerTool then prompts for the starting and ending address of the test block. While executing the program, PowerTool stores the original byte of an address, then writes and reads a new byte to it to verify that it is working. When done, PowerTool replaces the address with the original byte, so as to leave memory undisturbed. You may stop the test at any time by pressing (CLEAR). (BREAK) is disabled for the test.

Note that if you attempt to test the addresses occupied by the ROMs (read only memories), the routine will report that every address is bad. This is because ROMs cannot be written into.

Also, if you test the video RAM on a Model I which has no lower-case capability, those addresses (3C00H through 3FFFH) will also be reported back as being bad. This is normal since the video memory on an upper-case Model I has 7 bits per byte instead of 8. It does not indicate that something is wrong with your video memory.

Jump to Memory

This tool executes a call to a machine language subroutine. This would normally be done if you wanted to jump to your own machine-language subroutine in high memory, for example. To use this tool, from the MEMORY Tools Menu, press (8). PowerTool prompts you for the start address of the routine.

To use your own subroutine, load it into high memory before starting up PowerTool. Remember to end the routine with a RET or EXX and JP(HL) command. After your routine has completed, PowerTool returns at the Main Menu level.

> **NOTE:** DO NOT try jumping into random locations blindly, especially into the PowerTool program!

Search String

This routine searches memory for the occurrence of a particular ASCII string, byte string, or word string, and optionally replaces it with a new string. To use this tool, from the MEMORY Tools Menu, press (9). PowerTool first prompts you for the start and end address of the search. It then prompts for the search string and finally for the replacement string.

Enter an ASCII string by surrounding it with double quotes if you want an exact case match on the search, or with single quotes if you want to match both upper and lower case on the search. To enter a byte string, enter a series of values in the range 00H-FFH, and separate them with spaces or commas. The values may be entered in any of the acceptable numeric bases recognized by PowerTool.

To search for word strings, enter two-byte values separated by spaces or commas. The two-byte values should not exceed the range 0000H-FFFFH. Remember that the routine will reverse the order of two-byte words in order to correctly search out address references, which are kept in lsb-msb (least significant byte — most significant byte) order.

Also, note that since PowerTool ignores leading zeroes in the prompt line, leading zeroes as part of the word may cause the program to evaluate a two-byte value as one byte. For example, 0033H would evaluate to 33H instead of 33, 00 as expected. Use a byte search in such cases.

As with the search string function of the ZAP tools, you may enter a search string of intermixed ASCII, one-byte values, and two-byte values.

As the routine proceeds, if there are many matches, they may scroll by faster than you can read them. If so, press the spacebar to pause the display, and then press (ENTER) to continue.

Input Byte from Port

This tool reads a byte from a hardware port. To use this tool, from the MEMORY Tools Menu, press (0). PowerTool prompts you for the port number.

For a listing of the ports and what functions they serve, consult the Technical Reference Guide for your computer.

Output Byte to Port

This tool sends a byte out to a hardware port. To use this tool, from the MEMORY Tools Menu, press (A). PowerTool prompts you for the port number and byte value.

For a listing of the ports and what functions they serve, consult the Technical Reference Guide for your computer.

Memory to Sectors

This tool writes a block of memory out to disk. To use this tool, from the MEMORY Tools Menu, press (B). PowerTool prompts you for the starting address of the memory block, the drive, track, and sector where you want writing to start, and the number of sectors you want to write. The sectors are written with standard DAMs.

Remember that memory saved to the disk this way is not recorded in the disk's directory, so it will remain invisible to the DOS unless you make an entry in the diskette's directory for it. However, assuming that you remember the locations on the disk where you wrote it out, you may reload the sectors back into memory using the Sectors to Memory Tool (below).

Sectors to Memory

This tool reads sectors from the disk into a region in the computer's memory. To use this tool, from the MEMORY Tools Menu, press (C). PowerTool first prompts you for the drive, track, and sector that reading should start from. It then asks you for the number of sectors you want read (default value is 64 sectors) , and finally, the starting memory address where you want the data loaded (default value is C500H).

After the sectors are loaded, press any key to view the data. At this point, you have the same options to manipulate the data, as you do when using the Display Memory Tool.

> **NOTE:** PowerTool will not allow you to overflow memory. Reading will cease when the buffer is full. Default by sectors is based on default memory address.

Memory to Track

This tool writes a section of memory out to the disk as one track. To use this tool, from the MEMORY Tools Menu, press (D). PowerTool prompts you for the starting address in memory of the track data, and the drive and track number where the data is to be written.

BE CAREFUL WHEN USING THIS OPTION!!! This procedure assumes that you have constructed an image of a formatted disk track in memory. This consists not only of the data to be written, but also the actual formatting information that TRSDOS or LDOS (and PowerTool itself!) needs to write it, including the actual sector data, the track ID fields, the sector ID fields, the CRC bytes and the inter-sector gap bytes. If any of these are not present or are incorrectly positioned, you will construct a totally unreadable track on your disk with NO way to read it back in.

This tool allows you to create tracks using formats other than those used by TRSDOS or LDOS. However, we strongly suggest that if you do not have any experience in constructing format tracks, that you do not attempt to use this option. Use "FORMAT Tools", "Build Format Track" and "Write Format Track" if in doubt of a format track's memory image.

Track to Memory

This tool reads a full track of data into memory, including all formatting information such as sync fields, data address marks, CRCs, and so on. To use this tool, from the MEMORY Tools Menu, press (E). PowerTool prompts you for the drive and track number that you want read, the memory address where you want the data stored, and whether or not you want the floppy disk controller to synchronize on the ID address marks during the read or not. (This means that the controller chip will start its accumulation of data at the data address marks rather than elsewhere — if you want this option, reply ''Y'' otherwise reply ''N'', or simply press (ENTER).) This is significant only in single-density operations.

After PowerTool reads in the track, press any key to view the data. At this point you have access to the same memory manipulation tools as when using the Display Memory Tool.

Experienced programmers can use this option to create protected tracks on a disk, for example, by changing the CRC bytes on each sector so that a standard DOS signals an error on sector read operations, or changing the actual track numbers or sector numbers so that the DOS will not be able to read the track at all.

The altered track can be written back out to disk with the Memory to Track option. However, this exercise should not be undertaken lightly, since certain codes in the ID field are required or are interpreted by the FDC as commands. You could create an unreadable disk!

The Track to Memory Tool used with a double-density disk, may or may not, successfully read a track without an error. During double-density track reads, the SYNCHRONIZE is always on, and the controller may be ''fooled'' by certain bit streams, which it interprets as address marks. This consequence will effectively de-synchronize the controller and the ''DATA LOST'' error will be generated. Which tracks may be read successfully is not easily predictable. However, single density normally has no problems.

Some people are fooled by the idea that one may ''read a track'' and then ''write a track''. This seemingly innocent operation will, in fact, make a track totally useless and unreadable. During a ''write track'' operation all bytes above F5H are actually ''commands'' to the FDC. Bytes in this range may be modified by the controller. Also, these bytes will be written with different clock pulses so the FDC may easily detect their locations. These bytes will normally be written using ''write sector'' operations (no change). Since normal data files usually contain many bytes above F5H, the ''read track'' will load these into memory as they exist on the disk, but the ''write track'' will convert them as decoded. Only familiar users should execute the ''write track'' option!

1Ø/ FILE Tools

The FILE Tools let you manipulate the data concerning a file, similar to the way the ZAP Tools manipulate data. The FILE Tools, however, are file-oriented rather than sector oriented. This means that if you display a file using the FILE Tools, you can view all of that file even though it may be in two or more extents on the disk.

The FILE Tools use the file information maintained in the TRSDOS/LDOS diskette directory to know where to find the file. Of course, this also means that you cannot use the FILE Tools to examine a disk with a non-standard or missing directory.

To use the FILE Tools, from the Main Menu, press (F). PowerTool returns the FILE Tools Menu:

```
#####################################################################
# **  PowerTool  **  Ver 01.00.00  **  By: Kim Watt  **             #
# Copyright # 1983 Breeze/QSD, Inc. - Licensed to Tandy Corp.       #
#####################################################################
# ##  FILE Tools  ##                                                #
#                                                                   #
# 1.  Display File Sectors      7.  File Locations                  #
# 2.  Compare Files             8.  Drive Status                    #
# 3.  Copy Files                9.  Sector Allocation               #
# 4.  Disk Directory            0.  Build File                      #
# 5.  Free Space                A.  Clear File                      #
# 6.  Offset File               B.  Disk Allocation                 #
#                                                                   #
# Choice? #_                                                        #
#                                                                   #
#####################################################################
```

Select the desired tool by pressing the number or letter to the left of the tool name.

Display File Sectors

This tool operates much like the Display Sectors Tool of the ZAP Tools. It lets you view and alter the data stored in a file.

To use this tool, from the FILE Tools Menu, press (1). PowerTool prompts you for the filename (you must include the extension). It then searches the disk directories of all active drives, starting with Drive Ø (unless you specified a drive with the filename).

Double-density Model I TRSDOS and Model III TRSDOS system files (that is, operating system files) need to be handled slightly differently in order to be viewed. This is due to the lack of a standard directory entry for the system files. To view a system file, precede the module number with an asterisk, for example, to see SYS5 enter *Ø5.

If the file is inactive in the directory (that is, it has been killed), a "File NOT FOUND" message will be issued. This format will work only for Display File Sectors and only for these versions of TRSDOS.

If PowerTool finds the file, it displays the following:

End of allocation sector number (EOA) — Since TRSDOS and LDOS allocates disk space in granules, it often happens that a file does not completely fill up the last granule allocated, so that one or two sectors are allocated to it but are actually empty. The EOA refers to the last sector of the last granule allocated to that file by the operating system.

End of file sector number (EOF) — This is the LAST sector of the file which actually contains file data. It may or may not agree with the EOA number, but will normally be within the last allocated granule.

PowerTool will then prompt you for the sector of the file which you want displayed. Remember that this is a relative value, that is, the first sector of the file is sector zero, and so on. You may enter a sector number, or else press (ENTER) to see the first sector (sector 0), enter "E" for the EOF sector, or "A" for the EOA sector.

The desired sector will be displayed on the screen, in a format similar to the ZAP Tools Display Sector Tool, the only difference being the columns to the left of the sector data, as shown below:

```
    I00#4B49 4C4C 2053 5045 4449 543A 310D 4B49#KILL SPEDIT:1.KI
HEX N10#4C4C 2043 4845 434B 2F43 4D44 3A31 0D4B#LL CHECK/CMD:1.K
DRV S20#494C 4C20 574F 5244 532F 5553 523A 310D#ILL WORDS/USR:1.
 1  T30#434F 5059 2050 524F 4F46 2F43 544C 2054#COPY PROOF/CTL T
TRK A40#4F20 3A31 0000 FF00 0000 0000 0000 0000#O :1..#........
 04 L50#0000 0000 0000 0000 0000 0000 0000 0000#................
TRU L60#0000 0000 0000 0000 0000 0000 0000 0000#................
 04 /70#0000 0000 0000 0000 0000 0000 0000 0000#................
SEC B80#0000 0000 0000 0000 0000 0000 0000 0000#................
 16 L90#0000 0000 0000 0000 0000 0000 0000 0000#................
RPT DA0#0000 0000 0000 0000 0000 0000 0000 0000#................
DDD  B0#0000 0000 0000 0000 0000 0000 0000 0000#................
FPDE C0#0000 0000 0000 0000 0000 0000 0000 0000#................
RSEC D0#0000 0000 0000 0000 0000 0000 0000 0000#................
0000 E0#0000 0000 0000 0000 0000 0000 0000 0000#................
+00 -F0#0000 0000 0000 0000 0000 0000 0000 0000#................
```

The leftmost column of information first gives you the current modification mode base. Below this is the drive number, the physical track number, the track number recorded on the disk, and the sector number. Below this is the data address mark that it found on the disk (see page 10 for details) along with the density of the disk — 0SD for single-density, 0DD for double-density.

FPDE ("File Primary Directory Entry") means that the sector you are viewing is allocated in the file's primary directory entry. FXDE ("File Extended Directory Entry") means that the sector you are viewing is allocated in the one of the file's extended directory entries. Double-density TRSDOS doesn't use FXDE's, so this message should never appear if you are scanning a disk formatted by this system.

RSEC and the number below it refer to the relative sector number. The +00 is the current encryption value, as described later in this section.

Next to this column is a single vertical column which displays the NAME of the file being viewed, for example:

B
A
S
I
C
/
C
M
D

At the bottom of this column is one of three symbols: −, +, or @. The minus sign ("−") indicates that you have not yet reached the EOF sector of the file. The plus sign ("+") indicates that you have already passed the EOF sector and are viewing a sector that was allocated to the file but not used by it. The @ symbol indicates that you are viewing the EOF sector itself. When this symbol appears, there will be a hex number above it, viewed vertically. This is the first available byte after the end of the file itself. The last byte of the file would be the one preceding this byte. If this number is ØØ, it means that the last byte of the file was at relative byte FFH of the preceding sector.

To page through a file, you may enter special commands by pressing one of the following keys:

Key	Action
→ or ↑	Pages to the next higher sector of the file.
← or ↓	Pages to the next lower sector of the file.
SHIFT → or ↑	Displays last allocated sector of the file.
SHIFT ← or SHIFT ↓	Displays initial sector of the file.
CLEAR	Requests new relative sector to display.
E	Displays end-of-file sector.
BREAK	Returns to FILE Tools menu.
SHIFT BREAK	Returns to main menu.

If you attempt to page beyond the limits of the file, you will be given an error message.

Modifying a File Sector

Once PowerTool displays a sector on the screen, you may alter the data in a file sector with this tool. To do so, first select a ''modification base'', by pressing (H) for hexadecimal, (D) for decimal, (O) or (Q) for octal, (B) for binary, or (A) for ASCII. The routine will update the base display to the specified base.

To begin modifying, press (M). At this point, the routine is identical to its ZAP counterpart. Complete directions for modification mode are given on page 19 under Display Sector.

Bit Manipulation

You may also use any of the sector bit manipulation operations as with the Display Sector routine. To use these, you must be in the Paging mode, rather than the Modification mode. For further details, refer to the Display Sector description on page 19.

Compare Files

This routine compares two files byte-by-byte to check for mismatches. You may find this routine useful if, for example, you have doubts about the integrity of a particular file and wish to check it against another copy.

To use this tool, from the FILE Tools Menu, press (2). PowerTool prompts you for the source filename and the compare filename. PowerTool compares the files, sector-by-sector. If it finds any mismatches, it reports them as:

Mismatch, Relative Sector xxxx, Byte nnH

and jumps to the next sector. When PowerTool has completed the comparison, it informs you of any disk errors (due to CRC errors, for example) it encountered, and the total number of sectors in which mismatches were found between the two files. Compare halts at first byte mismatch in each sector. When comparing two files, one longer than the other, if the shorter file is used as the source, the compare will stop when the source end-of-file is reached. If two files are identical up to this point, no errors will be reported. It is always a good idea to use the longer file as the source file.

Copy Files

This routine is a fast multiple-file copy utility which will let you move any number of files between two disks, the only restraint being the available space on the destination disk.

To use this tool, from the FILE Tools Menu, press (3). PowerTool asks you for the drive number of the source drive, then reads the directory of the diskette in that drive and displays a directory listing on the screen. At the bottom of the screen PowerTool displays a filename, followed by its length in granules and the query, COPY ?

If you press (Y), the filename scrolls upward and you will be presented with another filename. If you press (N) then the filename is replaced with a new one.

When the last file has been "tagged," PowerTool asks for the destination drive. It displays the total number of sectors to be copied, and then copies each file. If a file already exists on the destination disk, PowerTool will display the message "*****" beside the filename as it copies. If there is not enough space on the destination disk to hold all the files, the routine terminates before starting the copy, and notifies you of the problem.

After the last file has been copied over, the directory of the destination disk is updated and written back out to the disk. You may specify the same disk drive for destination as for source (for example, if you have only one drive), and PowerTool will prompt you for disk swaps. Single drive copy to unlike DOSs is not allowed.

This routine writes the files on the lowest available track and sector on the destination disk, building upward from there, without leaving any gaps between files. It is a very efficient routine to use if you wish to "pack" a disk. Wherever possible, the file copy routine keeps the file in just one extent rather than splitting it up.

If you are copying a file from a Model III or from a double-density Model I formatted disk to a Model I formatted disk or vice versa, PowerTool will deliberately strip any passwords which may have been set for the source file. The reason for this is that Model III TRSDOS computes passwords differently from Model I TRSDOS (single or double density) and LDOS and if the encoded passwords were copied over, even if you gave the correct password, it would not be recognized.

Hence, if you have any password-protected files on your source disk, you must reassign the passwords using the ATTRIB command of the operating system on the destination disk after the copy routine is completed. However, if you are copying files to like systems (disks formatted with the same system) passwords are transferred intact. When copying files from one DOS type to another, the 'granule' count may be misleading. For example, let's assume that you have a 5 granule file on a L3D disk. Copying this file to a T3D disk will actually require 6 granules. However, an error message such as:

> 5 granules to copy , 6 granules available on destination disk. Out of Disk Space.

may appear.

This message may seem erroneous, but in fact is correct because:

> L3D = 5 granules = 30 sectors
> T3D = 6 granules = 18 sectors

Disk Directory

This tool displays the directory of a diskette. To use this tool, from the FILE Tools Menu, press (4). PowerTool prompts you for the drive number(s). It then displays the disk name, date, number of formatted tracks, number of free granules, and number of free directory slots.

Following this, PowerTool displays all of the valid files on the disk, along with their file attributes and protection levels, using the abbreviations S for system file, I for invisible file, and P=n for protection level. If a file has no listed attributes, then it is a visible file with a protection level of zero.

Free Space

This tool scans all mounted disks and displays the disk name, date, number of formatted tracks, number of free granules, free space in kilobytes and number of free directory slots for each disk.

To use this tool, from the FILE Tools Menu, press (5). PowerTool returns the information for each diskette currently in the system.

> **NOTE:** The mounted disks must all be readable, that is, they must have been formatted by a TRSDOS or LDOS System (Model I or Model III) and must contain recognizable directory tracks. If a non-standard disk is on any drive, you will be presented with an error message.

Offset File

This routine rewrites the load and entry information of a machine language routine stored on disk. This is useful to use on programs written for tape systems, since many of these programs load on top of the DOS, and hence make the complete load impossible. Use this routine after converting the program from tape to disk, via the tape-to-disk utilities of your operating system.

To use this tool, from the FILE Tools Menu, press (6). PowerTool prompts you for the filename, and then goes to the disk and gets the load and entry information from the file. It then asks for the new load address, and whether or not you want to add an ''appendage'' to the program.

The appendage consists of a short routine which relocates the program back to its original load position. This is necessary since many machine language commands require absolute addresses, and unless the program is relocated, the computer will probably get lost. Certain programs use relative addressing, exclusively, and hence won't need the appendage routine.

If you select the appendage addition, the file must have a spare 20 bytes at the end of the file, as PowerTool won't allocate more space. If there is not enough room, PowerTool will respond with a message to that effect, and abort the operation.

PowerTool then asks if you want the interrupts disabled or not. Some programs will not execute properly unless the interrupts are disabled. You will have to determine whether the program you are offsetting will work correctly with the interrupts enabled or disabled, and set this option accordingly.

After using this routine, whenever you load the program, it will load to the new address. If you added the relocation appendage, then before the program is executed, it will move to its original load position. The appendage saves the contents of the Z-80 registers at load time so that if your program requires them, they will be passed unchanged.

File Locations

This tool displays complete directory information about each diskette file. To use this tool, from the FILE Tools Menu, press (7). PowerTool asks you for the drive number(s) to be scanned. Each drive specified must contain a standard formatted disk with a readable directory track.

PowerTool reads the disks one-by-one, and displays information about each file. The display stops between screenfuls to give you a chance to examine the information — press (ENTER) to scroll to the next screen.

For each active file on each disk, five lines of information are displayed. The format is:

```
filename      attributes
FPDE - :x, Track = xx, Sector = xx, Byte = xxH, DEC = xxH
EOFS = xxxxx, EOFB = xxH, LRL = xx
EACC = xxxxH, EUPD = xxxxH, Grans = xx
Extents / xx,xx, xx / EOF
```

The first line gives the file's name and attributes. Attributes can be "S" for system file, "I" for invisible file, and "P=n" for protection level (n is number from 0 to 7). No listed attributes implies a visible user file with a protection level of zero.

The second line gives the disk location of the file's directory entry, or "FPDE" ("File Primary Directory Entry"). DEC stands for "Directory Entry Code", and represents the directory slot location of the file (and hence the relative byte location of the file's HIT byte).

The third line gives information about the EOFS (End Of File Sector), which is the last sector used by the file, (not necessarily the last sector allocated to the file), the EOFB (End Of File Byte), which is the last byte of the file within the EOFS. Also on this line is the logical record length (LRL) of the file.

The fourth display line gives the encoded values of the access password (EACC), update password (EUPD), as well as the number of granules used by the file.

The final line gives the disk address(es) of the file. Each extent of the file is separated by a slash (/). The first number in the extent address is the track number, the second number is the starting sector number, and the third number is the number of sectors used in that extent.

On Model I TRSDOS or any LDOS directory, an exceptionally large file may require an extra directory entry, known as the "File Extended Directory Entry" (FXDE). The FXDE is very similar to the FPDE except that it doesn't use the filename and it is not displayed when a directory is requested from TRSDOS or PowerTool. If PowerTool finds such a file, it also displays the FXDE in a similar fashion to the FPDE.

Double density Model III TRSDOS does not allow FXDE's, so this display should never appear when scanning directories formatted by this operating system.

Drive Status

This tool checks the status of all active drives in your system and reports back on each. To use this tool, from the FILE Tools Menu, press (8). You needn't supply any other information to PowerTool.

Drives which are disabled from the configuration tables, or not powered, will be reported as NOT IN SYSTEM. Drives with no disks but otherwise powered up will be reported as such.

If drives are found to be NOT IN SYSTEM, PowerTool automatically changes their Configuration Table settings to reflect this fact. Thus if you had a drive turned off when you executed this procedure and later turn it on in order to use it, you must return to the configuration table and restore it to an active status by changing the active/inactive indicator for that drive.

Sector Allocation

This tool reports if a given sector is allocated to a file, and if so, to which file. To use this tool, from the FILE Tools Menu, press (9). PowerTool prompts you for the drive, track, and sector. The disk to be scanned must contain a readable directory track, since this routine will use the directory information to determine whether the specified sector is assigned to any active file.

Build File

This routine creates and pre-allocates space on a disk for a file. To use this tool, from the FILE Tools Menu, press (0). PowerTool prompts you for the drive number and filename. It then scans the directory, reports how much free space is available, and asks you how many granules to allocate to the file.

PowerTool allocates the space in as contiguous a manner as possible. The pre-allocated space is recorded in the file's primary directory entry, so that when the system writes to this file later on, it can do so faster, since the DOS won't need to allocate space as often.

Clear File

This routine clears all of the data from a file without actually killing the file itself. To use this tool, from the FILE Tools Menu, press (A). PowerTool requests the filename, and then asks you to verify your choice. It then writes zeroes into all of the sectors allocated to that file.

Be careful in using this routine, as there is absolutely NO way of recovering data from a file that has been cleared.

Disk Allocation

This routine displays a granule allocation map of a disk, track by track. To use this tool, from the FILE Tools Menu, press (B). PowerTool prompts you for the drive number(s), and then displays the map.

The map lists each track by number, with the status of each granule on that track to its immediate right. An "x" means that the granule has been allocated to a file, a "." means that the granule is free, a "D" means that the granules are used by the directory, an "L" means that the granule has been locked out by the system, and a "_" means that the granule was not formatted. Note that different DOSs use different granules per track which will be reflected in the display.

Appendix/ Error Messages

Bad extents
A file has been found to have incorrect extents in the directory entry.

Bad backward link
One or more of a file's Extended Directory Entries (FXDE) has been found to not point back to the preceding FXDE (or FPDE).

nnnnn Bad Sectors
This message appears at the end of the VERIFY SECTORS operation. When a disk (or part of a disk) is verified, it is actually being tested for readability. If PowerTool is unable to read a particular sector, it usually means that there is something wrong with that sector — either incorrect CRC bytes, or damaged formatting information. A count of how many such sectors are encountered during a VERIFY SECTORS procedure is maintained, and is reported at the end of the operation as nnnnn Bad Sectors where nnnnn is the total number encountered.

Cannot establish disk type
Indicates that PowerTool is not able to recognize a disk as being formatted by one of the acceptable DOS types.

Cannot locate directory
Track Sector Count?
The program could not locate the directory and is requesting the user to specify its location.

Cannot locate directory Drive n
PowerTool could not read the directory on the specified drive. Usually indicates that the directory track has been written with incorrect or incompatible data address marks.

Cannot repair Non-TRSDOS boot sector
The BOOT REPAIR function of PowerTool will repair the boot sector ONLY on standard Model I and Model III TRSDOS disks (TRSDOS 2.3, TRSDOS 1.3, TRSDOS 2.7DD). If you attempt to execute this operation on a disk which is not formatted by any one of these three systems, PowerTool will not carry out the operation and will inform you that it ''Cannot repair non-TRSDOS boot.'' In this case to carry out a boot repair, you should copy the boot sectors from a good disk onto the bad one. Note: The Boot Repair function will also NOT work on TRSDOS 6.0 disks.

Cannot update directory
An attempt to update a diskette directory failed for some reason.

Cannot write directory
During any I/O operation which requires the updating of the diskette's directory, PowerTool was unable to do the update. This may occur, for example, if you attempt to write to a Model I single-density diskette's directory on a Model III computer, or if the target disk is write-protected.

DATA CRC Error
The CRC bytes for the sector data were wrong. This may have several causes: the sector is a ''protected'' sector, or the disk drive may be at fault. If it is intermittent, suspect a drive fault (e.g., a worn head pressure pad).

DATA LOST
Data was lost during a read/write operation. Usually due to software problems, (the disk transfer code of the program was not fast enough to keep up with the floppy disk controller). Track reads of a double density disk will normally produce these errors.

Data lost on track write
While writing data to a track, the timing was off by a sufficient amount so that data was lost before it could be written. This may happen if the TRS-80 CPU has been slowed down.

nnnnn Data Mismatches
nnnnn DAM Mismatches
This message reports the status at the end of a COMPARE FILES operation. When two files are compared, both the DATA and DATA ADDRESS MARKS (DAMs) on the disks are compared for an exact match. The number of times the data does not match on a certain spot is counted by the program. Also, if the data address marks on the disk should differ at any point, this is noted by the program and counted. At the end of the operation, the number of data mismatches is reported, followed by the number of Data Address Mark (DAM) mismatches. In both cases, nnnnn is the actual count of mismatches of each type.

Directory There
An attempt was made to move a track to the place occupied by the diskette directory.

Directory Unreadable
The directory is non-standard, or damaged.

nnnnn Disk Errors
During a COMPARE FILES operation, there are two possible errors which can give rise to a ''mismatch'': (1) A disk I/O error occurred on either one of the disks being compared preventing that sector from being read, for example, a DATA CRC error; (2) the data in the sectors being compared actually differed. The first is referred to as a DISK ERROR; the second is referred to as a SECTOR MISMATCH. A count of the number of times each error occurs during a comparison operation is maintained by PowerTool. At the end of the operation the number of times each type of error occurred is reported.

Disk READ ERROR
An attempt to read a disk failed.

Disk WRITE ERROR
An attempt to write to a disk failed.

Drive n DEACTIVATED
The specified drive has been disabled in the PowerTool Configuration Table. Change the +/− setting for this drive to bring it into the system. A possible cause, if the drive was automatically deactivated, is that the step rate exceeds the drive's capability.

Drive dropped ready
A selected drive dropped its ''ready'' status bit before the I/O operation could be completed.

Drive TIME OUT
The drive shut down before the I/O operation could be completed.

EOF Dest file reached
During a file compare operation, the end of the destination file was encountered unexpectedly.

Error on track write
An error occurred during an attempt to write memory to a disk track. May indicate a hardware problem.

File exists
An attempt was made to build a file with a name that already is in the diskette directory.

File NOT FOUND
The specified filespec was not on the disk directory.

GAT Table is BAD
The Granule Allocation Table (GAT) has been found to be damaged or incorrect by the Check Directory routine.

nnnnn Granules LOCKED OUT
Reports the total number of unusable granules on a disk following a format operation.

Hardware write fault
An attempt to write to a disk failed due to a defect in the drive or controller hardware.

HIT Table is BAD
The Hash Index Table (HIT) sector has been found to be damaged or contain errors by the Directory Check routine.

ID CRC Error
The CRC byte for the track and sector ID fields was wrong. Usually indicates a flawed format.

Interrupt on Pending Command
A disk I/O operation was interrupted while a command to the floppy disk controller was still pending.

nnnnn I/O Errors on Exchange
This message appears at the end of an EXCHANGE SECTORS operation. If during an operation a disk I/O error occurs, preventing the exchange of two or more sectors, a count will be kept by the program. At the end of the operation, the number of disk I/O errors which occurred will be reported using the message, where nnnnn is the actual count of errors.

nnnnn I/O Errors on Transfer
This message appears at the end of a COPY SECTOR DATA operation. If during the operation a disk I/O error occurs, preventing the exchange of one or more sectors of data, a count will be kept by the program. At the end of the operation, the number of disk I/O errors which occurred will be reported using this message, where nnnnn is the actual count of errors.

Invalid Filespec
The user entered a filespec in invalid or non-TRSDOS format. The correct format is ''FILENAME/EXT.PASSWORD:D.'' See your TRSDOS or LDOS manual for more details.

Invalid Name/Date
In reading the directory of a disk, if PowerTool encounters a disk date which contains non-alphanumeric characters, or graphic characters, or both, it will report the disk as having an ''Invalid Name/Date.''

Load File Format Error
This error message occurs when PowerTool is asked to OFFSET a FILE which is not a machine-language program. Machine language programs are kept in a certain format, known as ''executable load modules.'' This format contains information as to where in memory the module is to be successfully OFFSET from its original location in memory. If PowerTool is unable to recognize the ''executable load module'' format, it will return the message ''Load File Format Error.'' If this error occurs, the file cannot be offset.

No Destination Drives
No destination drives were specified for a file or sector copy or backup operation.

NO DISKETTE in Drive n
Self-explanatory.

Non-Standard Format
An attempt was made to read a disk that does not have a TRSDOS-compatible format.

OPEN DOOR on Drive x
Self-explanatory.

Out of Disk Space
An attempt was made to BUILD a file larger than the available space on a diskette, or to copy files onto a diskette with no free space remaining.

R>etry, S>kip, C>ontinuous, Q>uit ?
When a disk I/O error is encountered, in most cases, you will be presented a chance to retry the operation. This prompt gives you several options. Pressing (R) will retry the operation once; (C) will retry the operation over and over until it succeeds; (S) will skip the portion of the disk where the I/O error occurred; (Q) will abort the operation completely and return you to the menu.

nnnnn Reversal Errors
This message is returned by the REVERSE SECTOR DATA operation. If any error occurs, such as a disk I/O error, which prevents the data in one or more sectors from being reversed as requested, PowerTool will keep a count of it and report the number at the end of the operation, using this message. Note: If no errors occur, PowerTool will report "0 Reversal Errors." Read this message carefully. It means that ALL the requested sectors were sucessfully exchanged, because there were NO (zero) errors.

Sector NOT ASSIGNED
This is a message returned by the SECTOR ALLOCATION procedure. If the specified sector is not assigned to any file, or is not locked out for any reason, then PowerTool will inform you of the fact with this message.

Sector NOT FOUND
PowerTool attempted to read a sector that either was not there (unformatted) or had a non-standard and unreadable format.

nnnnn Sector Mismatches
During a COMPARE FILES operation, there are two possible errors which can give rise to a "mismatch": (1) A disk I/O error occurred on either one of the disks being compared preventing that sector from being read, for example, a DATA CRC error; (2) the data in the sectors being compared actually differed. The first is referred to as a DISK ERROR; the second is referred to as a SECTOR MISMATCH. A count of the number of times each error occurs during a comparison operation is maintained by PowerTool. At the end of the operation the number of times each type of error occurred is reported.

Sector OUT OF RANGE
Positioned to Relative Sector xxxxx
During a Display File Sectors operation, the user attempted to page beyond the boundaries of the file being viewed.

nnnnn sectors could NOT be verified
Reports the total number of sectors that could not be verified during a Format or Verify Disk Sectors operation.

nnnnn sectors LOST
During a copy sectors or move sectors operation, if some reason prevents one or more sectors from being successfully written to, this message will appear at the end of the operation.

nnnnn Sectors NOT Altered
This message appears at the end of an ALTER DATA ADDRESS MARKS operation. If any error which prevents the requested alteration from being carried out occurs, it will be counted and reported at the end of the operation. Note: if the operation is completely successful, PowerTool will report "0 Sectors NOT Altered." Read this carefully. It means ALL sectors involved were successfully altered.

nnnnn Sectors NOT copied
During a copy sectors operation, several sectors could not be read. This message will usually appear in conjunction with some other error message pinpointing the cause of the failure.

nnnnn Sectors NOT read
Several sectors could not be read. Possible causes are the same as above.

nnnnn Sectors NOT zeroed
During a "Zero sectors" operation, several sectors could not be written to. Possible causes are a hardware drive fault, or incompatible ("protected") formats on the target sectors.

nn sectors not written
Sectors could not be written during an I/O operation. This message will usually be accompanied by other error messages pinpointing the cause.

Track Allocated
An attempt was made to move a directory track to a track already occupied.

Track Read Error
An error occurred during an attempt to read a track into memory.

Write fault
Usually signals a hardware problem with the disk drive's write circuitry.

WRITE PROTECTED Disk
The diskette has a write-protect tab on it, or the drive has been declared write-protected in the configuration tables.

RADIO SHACK, A DIVISION OF TANDY CORPORATION

U.S.A.: FORT WORTH, TEXAS 76102
CANADA: BARRIE, ONTARIO L4M 4W5

TANDY CORPORATION

AUSTRALIA	BELGIUM	U. K.
91 KURRAJONG AVENUE MOUNT DRUITT, N.S.W. 2770	PARC INDUSTRIEL DE NANINNE 5140 NANINNE	BILSTON ROAD WEDNESBURY WEST MIDLANDS WS10 7JN

8749510

PRINTED IN U.S.A.